SESSION

BIOMETRICS, FORENSICS, AND SECURITY EDUCATION + INFO ASSURANCE

Chair(s)

Prof. Kathy Liszka

Analysis of Evidence in Cloud Storage Client Applications on the Windows Platform

R. Malik[1], N. Shashidhar[1], and L. Chen[2]
[1]Department of Computer Science, Sam Houston State University, Huntsville, Texas, USA
[2]Department of Information Technology, Georgia Southern University, Statesboro, Georgia, USA

Abstract - *In the research proposed in this paper, we present an approach to conduct a simple forensic analysis of cloud client storage applications on a Windows 8.1 virtual machine, in order to find possible traces left on the system that indicate the use of the cloud storage client applications even after the application is deleted. Our analysis focuses on the locations where evidence can be gathered and on the different types of files that can constitute possible evidence. The aim of this work is to collect data remnants from different cloud client applications once the applications is installed; remove the application and look for data remnants. Finally, we try to recover files that may have been deleted from the storage space.*

Keywords: cloud storage forensics, cloud application artifacts, data remnants, data carving, Windows forensics, digital forensic investigations.

1 Introduction

As we all noticed, in the last decade there was an exponential increase in the use of cloud computing [1]. Unsurprisingly, this also led to an increase of cybercrime that involves the cloud infrastructure, and therefore arose the need of cloud forensics [2]. This increase brought to light many issues and challenges for digital forensics experts. In fact, the National Institute of Standards and Technology (NIST) identified 65 of these challenges that need to be addressed [3]. There are two means to connect to one's cloud storage account. The first is through the use of a web browser and the second through a client application installed on the user's device. This research will analyze some of these client applications and provide some examples of how to gather data. Since the previous work in this field analyzed applications such as Microsoft OneDrive, Google Drive and Dropbox on a Windows 7 [4] [5] [6] [7], this research will show an example of client application analysis on a Windows 8.1 operating system virtual machine created with VMware Workstation 10. In addition, to diversify and provide different examples, the cloud storage services that we selected are different from the one chosen by the authors in previous work. Nevertheless, after testing these applications, we found that they represent valid alternatives to the most known cloud storage services. These alternatives are Copy [8] and ownCloud [9]. Dropbox was also analyzed, since unlike Copy and ownCloud it encrypts the files that contain evidence, and it was interesting to compare the two different scenarios. To decrypt some of the files, Dropbox Decryptor by Magnet Forensics was used [10]. The sections in this paper are organized as follow: the first section regards a web browser analysis, such as Google Chrome, due to the fact that in most cases users access their accounts through a web browser instead of a client application. Hence, it is very important to analyze Google Chrome files to see if any valuable evidence is present. The next section analyzes the client applications, and where these applications store the most important files that may contain evidence. Once these files were gathered, the application was uninstalled to see if any trace of the evidence was left. As it is possible to see in the appropriate section, evidence of use of the client applications was possible to be found inside the registry. Then, the possibility of recovering deleted files (a process named data carving) is explored. To test this possibility we used TheSleuth Kit (TSK) [11]. Finally, the last section will show that useful data can be found inside the memory (.vmem files were used).

2 Prior Work

The following literature review, explores the procedures and approaches used by other researchers in this particular field. Three main prior and related researches were analyzed to discover the approach taken in order to collect artifacts. Artifacts collected can be files either accessed or modified by the cloud storage applications on the client devices, or artifacts related to web-based cloud storage services (which are accessed through a web browser). Two main approaches were identified. The first approach represents a presumption of where artifacts should be located on a device, and then perform a search in those specifics locations, based on the examiner's knowledge. Meanwhile, the second approach is based on the use of programs and tools, such as Process Monitor from the Sysinternals Suite [6] to determine the location, in a dynamic manner, of the artifacts and data remnants. All the prior work done in this field was performed on a Windows 7 system using virtual machines. The following is a brief discussion of the prior work.

The paper by H. Chung, J. Park, S. Lee and C. Kang [7] provides a procedure to investigate devices such as PCs and smartphones. According to this procedure, the investigator collects and analyzes data from all devices that a user has used to access a cloud storage service. Based on the type of the device that is being analyzed, the procedure can take a different approach. Simply put, if the device is a PC then it is very important to collect volatile data from physical memory (if live forensic analysis is possible) and nonvolatile data such as files,

directories, internet history, and log files. Physical memory contains useful information about users and their activities. For example, physical memory can contain login attempts and login credentials used to access cloud storage accounts through a web browser. If the device is instead a smartphone, and if the system is running Android OS, after rooting, it is possible to collect data from the main system folders. In the case of an iPhone, after connecting the device to a PC, important data of user activity related to the cloud can be found in backup files or in data synchronized with iTunes. Once all data is collected an analysis in order to find useful artifacts is performed. According to the paper, cloud-storage services can be web-based services accessed through a web browser or client applications installed on the device. In the first case, it is essential for an expert to analyze data such as web browser log and database files (cache, history, cookies, and downloaded files) that are stored in the user profile directory on a Windows system. Cache files include downloaded image files, text files, icons, HTML files, XML files, download times, and data sizes. History files contain visited URLs, web pages titles, visit times, and the number of visits. Cookie files store information about hosts, paths, cookies modification times, cookies expiration times, names, and values. Download lists include local paths of downloaded files, URLs, file size, download times, and whether downloaded files were successful. Through such web browser files an expert can identify a user's activity, including access or logins to a cloud storage service. However, when a client application is installed on a Windows system, traces of it are left in the registry, log files and database files. Mac systems have similar traces except registry files. These files are essential during a digital forensic analysis, since they provide proof of the use of a cloud storage service. These log files contain information such as logins attempts, if and when services were used, and times of synchronized files. Database files contain information about synchronized folders and files (creation times, last modified times, and whether files were deleted) on a PC. All this information can be used to create a timeline of the user activity. In a smartphone device traces are left in database files, XML files, and plist files (which contain information about a user account). Finally, the rest of the paper provides examples of forensic analysis and shows where data is found on a PC or a smartphone. The cloud services that were used in this work, are Amazon S3, Dropbox, Google Docs, and Evernote.

In the research work done by M. Katz and R. Montelbano [8], to obtain the locations of artifacts, Process Monitor is used. The result is filtered to show the file system activity, and changes to the registry and files. The cloud storage applications used in this research are SkyDrive (now OneDrive), Dropbox, and Google Drive. When SkyDrive was installed 4959 artifacts were either created or modified. Presence of the files modified using the client, was found in unallocated space, $Recycle.Bin CSV files, pagefile.sys, and inside the AppData folder. In the case of Dropbox installation, 4163 artifacts were either created or modified. Evidence of deleted files was found in unallocated space and in pagefile.sys. During Google Drive installation, 9438 artifacts where either created or modified. Evidence of files modified or deleted was found in unallocated space, $Recycle.Bin CSV files, pagefile.sys, and configuration files. The result of this research proved that a large number of files are affected during the installation of the application, and a large number of files are left behind, once the uninstallation process is completed. Evidence of file manipulation was mainly found in the unallocated space, $Recycle.Bin CSV files, and pagefile.sys. The type and number of artifacts varied depending on the application, but evidence of the use of the cloud application was still present after uninstallation of the client in all the cases.

The following research, performed by D. Quick, B. Martini, and R. Choo [9], provides a well formatted methodology and a very exhaustive analysis of data remnants left by cloud storage applications. This research is performed on a Windows 7 machine and the cloud storage services analyzed are Microsoft SkyDrive (again, now OneDrive), Dropbox, and Google Drive. The objective of the research was to solve questions, such as, which data remains on the hard disk after a user used the client software? Which data is left once the user has had access to the cloud storage through a web browser? What is the location of the data remnants on the operating system and in the memory? Other questions that were attempted to answer relate to network traffic data and smartphones. Based on the work of this research, artifacts of files either access or modified, and data remnants left behind by the applications are found inside prefetch files (which are used to analyze the software activity, such as the number of times the software has run or the associated files used by the application), registry files (they can contain references, activities, settings or other information), link files (files' shortcuts), thumbnails pictures within the thumbcache, event logs (which contain information relating to system, software, and other events recorded by the operating system), and finally, directory lists file ($MFT files). Forensic analysis has been also performed on the memory, $Recycle.Bin (in order to find deleted files), client applications (analysis of installation path, sample files, synchronized files and folders), on the account accessed through a web browser (can contain information about the number and the type of devices used to access the storage space), and finally, on the files related to the browser. Network traffic was also captured and analyzed to find activity related to login sessions. To conclude, data carving was performed through allocated and unallocated data. Thumbnails icons and large size pictures were recovered. This research, used a dynamic approach: tools to dynamically find evidence that were used were Process Monitor, Wireshark, among others.

Another research was performed by M. Epifani et al [10], on Microsoft SkyDrive (OneDrive), Google Drive, Dropbox, and iCloud. Again in this case, the collection of artifacts left behind by the applications was performed on a Windows 7 system. To track the disk usage DiskPulse was used (to determine information related to created, modified and deleted files), Regshot, and RegFromApp were used to track registry changes. By monitoring the registry changes, researchers were able to obtain installation locations and installed client applications versions. Other useful data was collected from configuration

files present in the installation folder (inside the user profile), from online accounts (information about deleted files, devices connected to the account, version history for every file, and last browser sessions), from the memory (it can contain user email, display name, filecache.dbx path, server time, file list, deleted files, username and passwords in the case of a web-based storage access), from Hiberfil.sys and pagefile.sys, link files, browser history and cache, registry point, and volume shadow copies. As we can see, this research collected evidence from the same locations as the previous researches.
To conclude this literature review, we can assert that the procedures and approaches taken in these prior research works are in concordance with each other. Even if the approaches were of two different types, the locations analyzed and the data remnants found were similar.

3 Methodology

To perform this research and to gather data, both a static analysis and a dynamic analysis are used. For example, the directory where all the information related to Google Chrome can be found (databases, history files, cookies and so on) are analyzed in a static way. The database files were accessed through SQLite Browser [13]. Then, the client application is installed. During the installation, Process Monitor and Process Explorer are executed to find the locations on the system that stores the main files and directories. It is possible to use the Process ID (PID) of the process during the installation in Process Explorer, and use it as a filter parameter in Process Monitor. Once the useful data is gathered during the installation, the application is removed and the folders that were created during the installation process are controlled. To see if there is any data remnant left, the registry is scanned with RegScanner [14] searching for strings that may indicate the presence of evidence. In the section called "Recovering Deleted Files", we will see that data carving from the client applications is possible. To confirm this possibility, the TSK toolkit will be used.

4 Main Research

4.1 Research

As pointed out in the previous work section, there are two ways to connect to the cloud storage account. The first way is through a Web browser, while the second one is through a client application. It is important, therefore, to gather information in both cases. This sections contains many different subsections that explain where evidence can be found on both the web browser (Google Chrome), and in the main directories of client applications. Finally, the physical memory contains a great source of useful data. Therefore, a subsection is dedicated to the volatile information found in the memory.

4.2 Web Browser Analysis

C:\Users\<User_Name>\AppData\Local\Google\Chrome\User Data\Default is the main directory that contains Google Chrome files where evidence of accessing the cloud storage space through Chrome can be found and contains the following described files. The table "*cookies*" inside the database file *Cookies*, stores the creation times of the cookies, the host names, the names of the cookies, the expiration times, the times of last access, and the encrypted value. The host name field of the table contains the name of the website accessed, and here it is possible to find evidence: in fact, some values are *.dropbox.com*, *.google.com*, *.copy.com*, *onedrive.live.com*, and *.owncloud.com*. The creation times shows when the website was first accessed by the user, and the time of last visit. The database file *Favicons* stores icons associated to websites, along with URLs of the *favicons*. Among these URLs it is possible to find the cloud storage server address plus the *favicon* path. One of the most interesting files in the directory above mentioned, is the *history* database file. The table *urls* in this file stores the URL of the web pages visited, the titles of the pages, the number of visits and the last visit times. The table *downloads* contains the names of the downloaded files, the target paths on the local system, the start and end times of download, the received bytes and total bytes counts, the servers addresses, the last modified times, among many other useful information. The *keyword_search_terms* table contains the keywords typed inside the browser search bar. The *Login Data* database file stores the logins attempts and the server logins file paths, the types of the usernames (for example, a string username or a *login_email*), the passwords types, the encrypted passwords value, the timestamps and other information that can be useful during a digital forensic analysis. The *Network Action Predictor* database file contains the URLs visited by the user. In order to load web pages faster, based on the input text in the search bar typed by the user, the browser tries to predict which webpage should be opened [15]. The *QuotaManager* database file can contain reference to the same URLs. The *Web Data* database file stores the *autofill* table, which logs usernames and other credential values and personal information, such as the location, address or phone number of the user. The files *Current Session*, *Current Tabs*, *Last Session*, and *Last Tabs* could store a great source of valuable information since they collect all the sessions started and the tabs opened by the user. This is in order to restore the *sessions* and *tabs* in case of an unexpected crash of the browser. Other files that may contain evidence are *History Provider Cache*, which appears to contain random strings and references to the user activity. Once useful artifacts have been looked through the web browser files, it is important to analyze the memory to see if it contains references to the user activities on the cloud storage accounts by using the web browser. It was possible to find evidence inside the memory by simply taking a snapshot of the state of the virtual machine, and then performing a keyword search (after importing the *.vmem* file in a Linux machine, Ubuntu 14.04 64-bit) using the commands *strings* and *grep* to filter the output. Inside the captured memory there is many useful information. In fact, it is

possible to locate in plain text user's credential such as email addresses, usernames, user IDs, first names and last names of the user, logins attempts, timestamps of logins, paths on the server, server names or server addresses, lists of files accessed, creations times, times of last synchronization, uploaded files and sizes, modification times, deletion times, messages and actions taken by the server. For example using the keyword *login*, it is possible to see that a login attempt has been made for both Dropbox and OneDrive.

4.3 Client Application Analysis: Copy

By launching *Process Explorer* when installing the Copy client application and when running it, we can easily find where Copy stores the main files. In our case, the files are located in *C:\Users\<User_Name>\AppData\Roaming\Copy*. The same result can be confirmed by using the *PID* found in *Process Explorer* and using it as a filter parameter in *Process Monitor*. The following are the important files found during the analysis:

- *config.db*: this file stores settings such as the user email, first name, last name, and user ID. It also contains the path of Copy's root directory and the root cache, among other settings.

- *trace.txt*: this log file's entries contain information regarding the hosting machine (operating system, host information, etc.), the client application, and the server.

- *synclog.txt*: this is another log file and it stores the operations executed by the application along with timestamps and other information relative to the operations types. Some operations are authentication attempts and file manipulation (such as upload, download, modification, and deletion).

- *copy <User_Email>.db* (in our case the name of the file is *cloudstorage.test.mail@gmail.com.db*): this file is very interesting from a forensic point of view since it contains the list of files and metadata stored in the root directory of Copy.

After uninstalling Copy, the main root folder is still present on the hard disk, as well as all the files contained in the folder. The folder *C:/Users/<User_Name>/AppData/Roaming/Copy* is still present, however *config.db*, *trace.txt*, *synclog.txt* and *copy cloudstorage.test.mail@gmail.com.db* have been deleted. This means that after the uninstallation of Copy, evidence of use is still present on the system and some user activity can be determined. Finally, by using RegScanner we performed a search based on the string *Copy*, and it was possible to find some registry keys left once is the application is removed and uninstalled. This strengthens the possibility for a forensics expert to find evidence related to the use of the client application.

4.4 Client Application Analysis: ownCloud

After installing and executing ownCloud for the first time, the output of *Process Explorer* is checked. The main directory of ownCloud is stored in *C:\Users\<User_Name>\ownCloud*. In this directory the most interesting files are hidden, and are:

- *.csync_journal.db*: this database file contains the metadata for the synchronized files. It also stores in the table *downloadinfo* and *uploadinfo*, information regarding downloaded and uploaded files.

- *.owncloudsync*: this log file keeps track of the user and the application activity. Some fields of the log entries are timestamps, duration of the actions, files involved, types of instruction, working directories, size of files, file IDs, modification times, and so on.

During the uninstallation of the ownCloud client application, the uninstaller asks the user whether the ownCloud root folder should be deleted or not. If it is not deleted, then the files, database files, and log files will still be available for analysis. Otherwise, they will be deleted. However, if the hard disk has not been wiped, or the unallocated space overwritten, recovering these files is still possible. This will be shown in one of the subsections of the Main Research section. Inside the registry, after the client application is removed, there are not many references to ownCloud. However, some traces are still left. In fact, the string ownCloud is still present in the *AUTORUN* key.

4.5 Client Application Analysis: Dropbox

During Dropbox installation and execution, we started *Process Explorer*, which shows in the lower pane, the files that Dropbox opens, reads and writes to. Unlike Copy, Dropbox encrypts the configuration and database files, and does not release the decryption keys to the users. Due to this reason, it is difficult to open and analyze these files. However, during a digital investigation, and with the use of a proper warrant, investigators might be able to obtain the encryption keys from the cloud service providers. Nevertheless, as the time of this writing, there is one particular tool that allows to decrypt some of the files of Dropbox. The tool name is *Magnet Forensics Dropbox Decryptor* (see the references for more information). To decrypt .dbx files (the encrypted files) with *Magnet Forensics Dropbox Decryptor* the Dropbox's .dbx file, the output folder where the decrypted files will be stored, the Windows protection folder, the value of the registry key

HKEY_CURRENT_USER/NTUSER.DAT/SOFTWARE/DROPBOX/KS/CLIENT, and the user Windows's account password should be specified within the fields of the tool.

The most important files that can be analyzed during a forensic investigations are stored in the folder

C:\Users\<User_Name>\AppData\Roaming\Dropbox and are contained in sub-directory *instance1* and are:

- *config.dbx*: this file is one of the encrypted files, and after it has been decrypted with *Magnet Forensics Dropbox Decryptor* it is possible to see that it stores the host IDs, the user's email addresses, the Dropbox root folder paths, among other settings.

- *filecache.dbx*: the table *file_journal* stored in this file contains the server paths, the files lists and the files names, the sizes of the files, the modification and creation times.

When Dropbox is uninstalled, the Dropbox root folder is still present on the disk, as well as all the files contained in it. The folder, *C:\Users\<User_Name>\AppData\Roaming\Dropbox* is still present, however, the encrypted files have been deleted. There are also many registry keys and traces left once the uninstallation of Dropbox is completed, that can be used as evidence of the use of client application.

4.6 Deleted Files Analysis

One important challenge that arises during a digital investigation, is the recovery of deleted files. The unallocated space can store valuable information for an investigator and cannot be ignored during an analysis. There are two methods to recover files deleted from the cloud storage account: the first consists of recovering a deleted file from the server-side, while the second from the client-side. In fact, both Copy and ownCloud client applications have a feature known as *undelete*. This feature restores a deleted file both locally and on the server. The Dropbox client does not have a similar feature, however, it allows to restore a deleted file or a modified file, once the account has been accessed with a web browser and not through the application like Copy and ownCloud. Dropbox also allows to permanently delete a file, in a way that it cannot be restored. In order to recover deleted files on the client, some basic understanding of the *NTFS* structure is necessary. The *NTFS* file system maintains a table known as *MFT* (*Master File Table*), in which each folder and file stored on the file system have an entry. The entries in the *MFT* table describe the files metadata information and contain pointers to the clusters that contain the file's data content. When the files are saved onto the hard drive, both the entries inside the *MFT* and the clusters that store the data are allocated [16].

To attempt a recovery of deleted files, we decided to use TheSleuth Kit, which is installed by default on a Kali Linux operating system. To attempt the deleted files recovery, we can either image the Windows 8.1 operating system, or use VMWare Workstation to mount the hard drive as read-only on the Kali Linux virtual machine. We chose to use the second option. Once the hard drive is mounted, the command *ls –ai*, allows to list all files on the terminal, even the hidden ones, along with their *inode* address. The *inode* address is an EXT file system concept, but it basically has the same function as the *MFT* entry. Using *ls –ai* we were able find the entry number for a file inside the *MFT*. The *istat* tool from the TheSleuth Kit prints in output the metadata information, described in the *MFT*. In this output, the the clusters numbers are specified. This clusters numbers are the pointers contained in *MFT* that point to the data content. If we open the logical hard drive with a *hex editor* tool, such as *HxD* [17], we can prove that those clusters really contains the file's data. The hard drive is organized in sectors, so a conversion from cluster to sectors is needed. Since in this version of Windows 8.1, there are 8 sectors per cluster, we can use the following equation to find the sector address:

$$\text{Sector} = \text{Cluster} * \text{Sectors\_per\_cluster} \quad (1)$$

By inserting the cluster number and the number 8 (sectors per cluster) in (1) we obtain the sector number, or the sector address. When opening the logical hard drive in *HxD*, this editor allows to jump directly to the specified sector. This will confirm that the sector do in fact contain the data content and it is allocated. Going back to Kali Linux, now it is possible to use the *blkstat* tool from TheSleuth Kit, which allows to see the allocation status of a specified data unit (in this case a cluster). As expected, the clusters are allocated. The next step is to un-mount the hard disk, delete the file that was contained in those clusters, and then mounting the drive again in Kali Linux, as read-only. We run again *istat* on the same entry address and this time the output is different, since the entry header says that the file is not allocated. Even if the file is deleted, the entry in the table is not deleted and it still contains the metadata information of the deleted files. When a file is deleted, a Boolean value is changed so the file system knows that the file is deleted and it is now unallocated. However, the *MFT* entry still points to the clusters that contain the file's data content. By running the *blkstat* tool from the TheSleuth Kit toolkit, the output is also in this case different since now the clusters are not allocated. Therefore, it is possible to deduce that the file is now in the unallocated space. In addition, one can also deduce that when a file is deleted, both the entry and the clusters still contain data (the file system can thought to be "lazy", since it does not wipe the content of the cluster). Unless the clusters that contain the file's data content are wiped by a user, or the clusters are over-written by the operating system, when a different file is stored on the drive, the data content of the deleted file is still recoverable. A simple tool used to recover the content inside the clusters is *icat* from the TheSleuth Kit. By using *icat*, the content of the clusters is printed on output on the terminal. It is possible to redirect the output to a file and analyze the file. The *icat* tool from the TheSleuth Kit is not the only tool an expert can use. There are plenty of recovery tools, such as Foremost and Scalpel and they both represent excellent tools. Finally, to prove that our deductions were correct, we open the drive again with a *hex editor* and then access the first sector of the file, and confirm that the content is still there. This deleted file recovery demonstration was executed on a file deleted from the Copy root folder, however, this method works for every cloud storage client application installed and tested in this research.

4.7 Physical Memory Analysis

To analyze the memory, a snapshot of the Windows 8.1 virtual machine was taken and a keyword search on the *.vmem* file was performed. Inside the memory there is a great deal of useful information and evidence. However, performing a live analysis is not always possible, so there is a possibility that memory cannot be analyzed when performing a digital forensic investigation. Nevertheless, if a live acquisition of the memory is possible, then memory should be acquired. Evidence that can be found inside the memory can include emails and user credentials, personal details, passwords, file lists and file's information, accessed files and folders, processes information, processes instructions, host names, loaded libraries and modules, libraries imported through the server, temporary files, accessed database files and log files, log files entries, authentication attempts, and so on.

5 Conclusions and Future Work

This research showed that it is possible to find plenty of evidence that relates to the use of a cloud storage client application on Windows 8.1, and evidence that relates to the activity of the user. These evidence is found mainly in databases and log files created by the application, but it also can be found inside the browser if a desktop client application version is not used, inside the memory, and inside the registry. If the application is then uninstalled, traces of the use and evidence can be found inside the memory, system log files, and registry keys and so on. If files are removed by the user there are two approaches that can help a forensic analyst to recover them. Possible future work can consist in a forensic analysis on the server side of the cloud, however, this is complicate issue since many problems arise, such as geography impossibility (servers can be spanned all over the world) and the servers can be outside jurisdiction of the investigators as well as a not transparent collaboration from the cloud service providers.

6 References

[1] Columbus, L. "Predicting Enterprise Cloud Computing Growth", September 4, 2013 Available: http://www.forbes.com/sites/louiscolumbus/2013/09/04/predicting-enterprise-cloud-computing-growth/

[2] Millis, E. "Cybercrime moves to the cloud", June 30, 2012, available: http://www.cnet.com/news/cybercrime-moves-to-the-cloud/

[3] NIST. "Cloud Computing Forensic Science Challenges". Draft NISTIR 8006, NIST Cloud Computing Forensic Science Working Group Information Technology Laboratory, NIST. June, 2014, Available: http://csrc.nist.gov/publications/drafts/nistir8006/draft_nistir_8006.pdf

[4] Chung, H., Park, J., Lee, S., & Kang, C. "Digital forensic investigation of cloud storage services". *Elsevier*, May 4, 2012.

[5] Katz M., Montelbano R. "Cloud Forensics", The Senator Patrick Leahy Center for Digital Investigation, Champlain College, 4 November 2013.

[6] Quick, D., Martini, B., & Choo, R. "Cloud Storage Forensics", 1st ed., p. 208, Syngress, 2013.

[7] Epifani, M. "Cloud Storage Forensics", SANS European Digital Forensics Summit, Prague, 2013, Available: https://digital-forensics.sans.org/summitarchives/Prague_Summit/Cloud_Storage_Forensics_Mattia_Eppifani.pdf

[8] Copy.com. "Copy: Store, protect, and share amazing things." from https://www.copy.com/home/

[9] OwnCloud.com. "OwnCloud 7 Community Edition is here!" from http://owncloud.org/

[10] Magnetforensics.com. "Dropbox Decryptor: A Free Digital Forensics Tool." June, 2014, from http://www.magnetforensics.com/dropbox-decryptor-a-freedigital-forensics-tool/

[11] Carrier, B. "The Sleuth Kit: Download" from http://www.sleuthkit.org/sleuthkit/download.php

[12] Technet.Microsoft.com. "Windows Sysinternals: Documentation, downloads and additional resources." from http://technet.microsoft.com/en-us/sysinternals/bb545021.aspx

[13] Sqlitebrowser.com. "DB Browser for SQLite', from http://sqlitebrowser.org/

[14] Nirsoft.net. "RegScanner: Alternative to RegEdit find/search/scan of Windows", 2014, from http://www.nirsoft.net/utils/regscanner.html

[15] Carrier, B. "*File system forensic analysis*." Boston, Mass.: Addison-Wesley, 2005.

[16] Support.google.com. "Make webpages load faster." from https://support.google.com/chrome/answer/1385029?hl=en

[17] Mh-nexus.de. "HxD - Freeware Hex Editor and Disk Editor", from http://mh-nexus.de/en/hxd/

Biometric Authentication and Data Security in Cloud Computing

G.L.Masala[1], P. Ruiu[2], A. Brunetti[1], O.Terzo[2], E. Grosso[1],
[1]Department of Political Science, Communication, Engineering and Information Technologies,
Computer Vision Laboratory, University of Sassari, Sassari, ITALY.
[2] Istituto Superiore Mario Boella (ISMB), Turin, ITALY.

Abstract - *The paper presents a new Cloud platform designed to support basic web applications shared by small and medium companies. The platform guarantees secure access of multiple users and complete logical separation of computational and data resources related to different companies. A peculiar data fragmentation approach ensures a high-level of protection of the data stored in the Cloud.*

The platform is built using the OpenStack architecture, while the user authentication is based on an original biometric approach that easily integrates finger and face modalities. Details of the authentication process and of the service modules involved in the biometric authentication are given. The platform proved to be effective and it is currently under beta testing phase for a limited set of candidate companies.

Keywords: *Security of data residing ,data fragmentation, multimodal biometric authentication, Openstack.*

1 Introduction

The migration from local to web applications, sharing critical data and resources and giving support to multi-user/multi-tenancy scenarios, is probably one of the most significant advances of the recent years in the arena of the application software.

The development of service-oriented architectures (SOA) and WEB services are key issues in this framework. SOAs support designing and developing in terms of services with distributed capabilities, which can be under the control of different ownership domains. These architectures are essentially a collection of services or, in different terms, repeatable activities that perform single or a few specialized operations and communicate with each other by simple data passing. Service consumers view a service provider as a communication endpoint supporting a particular request format or contract; this request format (or interface) is always separated from the service implementation.

As a matter of course, security breaches on web applications are a major concern because they can involve both enterprise and private customer data: protecting these assets is then an important part of any web application development. This process usually includes authentication and authorization steps, asset handling, activity logging, auditing. Traditional protection mechanisms, like password management, encryption, intrusion prevention and vulnerability analysis have been developed for this purpose.

The extension of the web application paradigm to the cloud computing model is denoted as software as a service (SaaS).

The adoption of Cloud computing, in particular leveraging on the public and hybrid models [1], involves many advantages in terms of flexibility, scalability and reliability, but also implies new challenges on security, data privacy and protection of personal data.

The security specific risks of the cloud are primarily derived from the complexity of the architecture (which includes different models of services and distribution) and its characteristics of multi-tenancy and resource sharing, allowing to allocate the same resources in different times to different users [2].

A first element of risk is related to the failure of the isolation systems for storage and computational resources. When data of individuals and organizations, who may have different interests and requirements or even conflicting/competing objectives, reside on the same physical infrastructure a failure of the isolation systems can compromise machines hosted through guest-hopping, SQL injection and side channel attacks [4]. To this concern, it is necessary to protect data and systems using methods that guarantee the physical and logical separation of resources and data flows [3].

Moreover, being the Cloud a distributed architecture, this implies an increased use of networks and data communication flows compared to traditional architectures. For example, data must be transferred for the synchronization of images of the same virtual machine among various and distributed hardware infrastructures. Or else, simple storage operations can involve communication between central systems and cloud remote

clients. Risks are, therefore, those of incurring on sniffing, spoofing, man-in-the-middle and side channel attacks.

An additional element of risk is related to the cloud model adopted. In fact, some cloud models require the user to transfer part of the control over his own data to the service provider. In this case, not only the data are allocated on the provider's servers, but also the user cannot apply specific protection mechanisms like encryption or access control, as the service provider is the sole subject having total control of the cloud resources.

Finally, some key roles for managing the cloud infrastructure, such as system administrators and managers of security systems, must be considered. These actors usually have the power to perform all types of activities within the system and this would potentially break safety requirements imposed by corporate policies. Yet, the assessment of this kind of fraudulent actions is very complex and there is a lack of certification agencies internationally recognized for the independent evaluation of cloud security .

This paper deals with “remote user authentication” or “logical access control”, one of the fundamental steps in protecting data and IT infrastructures. Authentication protocols allow to verify that each of the participants in the electronic communication is really who he claims to be. This task is commonly demanded to a specialized architecture denoted as the Authentication Server (AS). The AS preserves and manages the access keys to the various subsystems. In order to access private services or data, every authorized person must first establish a connection with the AS, declare and prove his own identity and obtain a session key useful to require further services.

Currently, the most common authentication mechanisms of the ASs make use of passwords and private tokens. Passwords are subject to various security threats; for example, they can be easily stolen or intercepted and used fraudulently. Tokens are more difficult to be reproduced and for this reason they are often used in banking services. However, being more expensive and difficult to manage, they are far to be an optimal solution. Moreover, they are usually based on the possession of a physical card or device that can be easily shared with different people.

As reported in the scientific literature [5-6], the efficient use of multiple biometric features for identity verification is still an open and attracting scientific problem; biometric physical access systems are perceived as reliable [5], then minimizing the typical risks of traditional authentication systems, in applications that require a high level of security like border control. On the other hand, the use of biometric data for the logical access to IT services is a more challenging and still unsolved problem. Certainly, the use of biometric techniques can be considered as one way to ensure a significant increase of security in the authentication protocols managed by modern authentication servers.

In this paper, we present a Cloud system that uses biometric authentication based on fingerprints [13]. This advanced access control is combined with a very peculiar fragmentation technique guaranteeing the security of the data residing on the cloud architecture. In chapter II we introduce some preliminary considerations concerning the cloud platform while in chapter III the realized Cloud system is described in detail and the main results on the cloud security are discussed. Section IV draws some conclusions, pointing out issues and problems that will be faced in the near future

2 Preliminaries

2.1 Cloud platform

OpenStack [7] is an open source project that many identify as the first true Cloud Operating System. OpenStack has to be considered as a basic technology rather than a solution; by analogy is often associated with the Linux kernel.

The project described in this paper has the primary goal of supporting basic web applications shared by small and medium companies; candidate platforms for Cloud computing should be, therefore, oriented to scalability, to be implemented according to the public or private Cloud models. In this respect, OpenStack has many interesting features; it allows a prompt and elastic control of computing resources such as CPUs, storage and networks, and includes many features for general system management, process automation and security.

OpenStack consists of several individual sub-components. This modular design facilitates great flexibility because each component may be used alone or in combination with others. Some of these modules, marked as cores (such as compute, storage, networking) represent the essential parts of the platform. Other modules are initially placed in an incubator from which they come only if needed. The main modules of OpenStack, fully distributable and replicable, are the following: computing (Nova), networking (Neutron), image templates (Glance), block (Cinder) and object storage (Swift), authentication, and accounting (Keystone). The architecture is based on the concept of "sharing nothing" that make components independent and self-sufficient, avoiding the sharing of memory or storage. Communications between the different modules are

asynchronous and are managed by queue managers (message brokers) that implement the Advanced Message Queuing Protocol (AMQP). The various services communicate with each other through specific Application Programming Interfaces (APIs) that implement the REST model. All these features make OpenStack an ideal tool to be deployed on commodity hardware, with consequent economic benefits and flexibility.

Virtualization is an important element of cloud computing because it guarantees the required elasticity in resource allocation. Virtualization is a technique that allows to run multiple virtual machines on a single physical server and to optimize the available resources. It is possible to provide different levels of abstraction that make the operating system do not see the physical hardware but the virtual hardware. This abstraction is achieved by a software layer, called *hypervisor,* which is usually integrated into the operating system kernel and it is loaded at system startup. The *hypervisor* does not offer any management capabilities to virtual machines. Like many of the cloud computing platforms also OpenStack is not released with a specific *hypervisor*, but it is the system administrator who chooses one of the supported: VMware, Hyper-V, Xen and KVM. In our project we use the Kernel-based Virtual Machine (KVM); it is one of the most supported and popular among scientific developers.

KVM is a Linux kernel module that allows a user program to use hardware virtualization capabilities of various processors. It supports in particular processors from AMD® and Intel® (x86 and x86_64) having these features (Intel VT or AMD-V). From the point of view of the operating system each virtual machine is seen as a regular Linux process that can use the hardware resources according to what established by the scheduler. A normal Linux process has two execution modes: kernel and user. KVM adds a third mode: guest mode that allows to isolate processes that grain inside. The main benefit of KVM is that being integrated into the kernel improves performance and reduce the impact on existing Linux systems.

2.2 Security of data residing

A possible solution to guarantee the security of data residing on distributed cloud infrastructure is the use of systems for the fragmentation and distribution of data, which allow to split the data into fragments and disperse them on all machines available to the cloud. In this way the recovery and the use of the data is very complex for an unauthorized user. By using fragmentation techniques, it is possible to distribute data on platforms of different providers, and to problems arising from the lack of trust in the service provider. However, in order to achieve a proper fragmentation and distribution of the data in the network, it is necessary to develop support tools to ensure the prompt availability and integrity of these data, without increasing the complexity of the system. In fact, an excessive consumption of resources or performance degradation related to procedures of information retrieval would compromise this approach.

3 Main results

3.1 General implementation of the Cloud System

OpenStack has a distributed nature, therefore, during installation it was necessary to take account of the number of nodes required for the installation of the platform. The meaning of the term “node” usually relates to individual machines running the functions of the cloud. In some cases a node corresponds to a physical machine, in other cases it corresponds to an instance of a virtual machine (VM). From the official documentation of OpenStack, the minimum number of nodes to be used in a stable installation is 5, at least one for each of the following functions: Horizon, Keystone, Neutron, Compute, Swift.

In particular:

- Neutron is the system that allows to manage the network connectivity and to address the VMs in the cloud. It includes some advanced features of type "networking as a service" that support the use of advanced networking.
- Swift is a distributed storage system that can accommodate data of users of the platform or VMs. It allows to manage the policies of replication and consistency ensuring the integrity, safety and protection of distributed data in the cloud.
- Keystone manages all security policies and authorization for user access to the platform, with different privileges. Moreover, it is the system that functions use to request services through a specific *Application Programming Interface (API).*
- Horizon is a graphical interface platform accessible via the web, for easy and intuitive management of the cloud.
- Glance is the Virtual Machine Image Repository, or a catalog of images of the operating system that users can use to instantiate VMs.
- Cinder allows to provide storage that can be used by nova to serve the VMs. Storage is provided in the form of block storage device and may be required as a service without reference to the real physical allocation.

In our system two modules of OpenStack were not installed: Ceilometer, which allows monitoring and billing use of cloud resources, and Heat, which manages the orchestration of processes of the cloud.
Figure 1 highlights the distribution of modules in the nodes

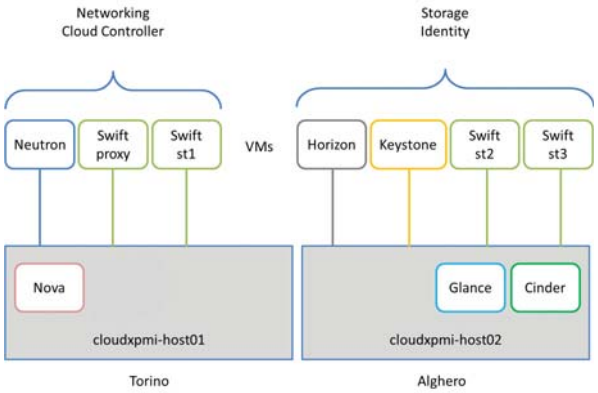


Fig.1 Subdivision of Open Stack functions between our two Italian data centers of Alghero and Turin. Services Nova and Heat have a physical machine on the server of Turin and all other services are arranged on virtual nodes

The network configuration of the platform is illustrated in Figure 2. We split the architecture in two different Italian data centers located in Alghero and Turin. Each server stands on a virtual private LAN: we have a server in Turin that uses the *em1* interface while another server in Alghero uses the interface *em4*.
The other network adapters are used to configure the three networks necessary for the operation of OpenStack. The public network is used to allow the connection of the virtual machines to the outside (Internet). For this network it is necessary to configure a virtual interface for the Neutron node with a public IP address. This interface will then be used to configure the bridge virtual audience (*br-pub*) managed by Neutron.
The management network is used for the communication between hosts and virtual machines on which is installed the entire platform OpenStack. This network has been configured as VPN, so as to enable secure communication between the nodes. The server of Turin has been configured as the VPN server, the bridge *tap0*, using the interface *em2*. The host of Alghero and the nodes hosted in the same server connect to the VPN server through another bridge *tap0*, always on the respective interface *em2*.
The data network is used to allow communication between virtual machines. For this network OpenStack defines another VPN, through three tunnels type *Generic Routing Encapsulation (GRE).* A tunnel is established between the two hosts and the other two tunnels between the same host and the Neutron node.

The service Keystone provides authentication and accounting for the entire platform and it is installed on a dedicated virtual machine on the physical server of Alghero (hostname: cloudxpmi-host02). This is necessary to facilitate its interface with a dedicated biometric authentication, via private network connection; the service is hosted in the authentication server (AS) of the data center of Alghero but externally with respect to the platform OpenStack.

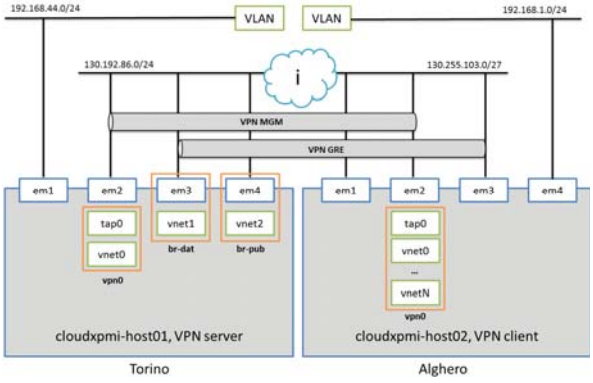


Fig.2 General network configuration of the Cloud platform

3.2 Integration of biometric recognition with cloud computing platform

Biometric authentication is proposed in order to access to the Cloud platform; a Client desktop application has then been implemented on the user side and a dedicated authentication server (AS) has been connected to the Keystone module.

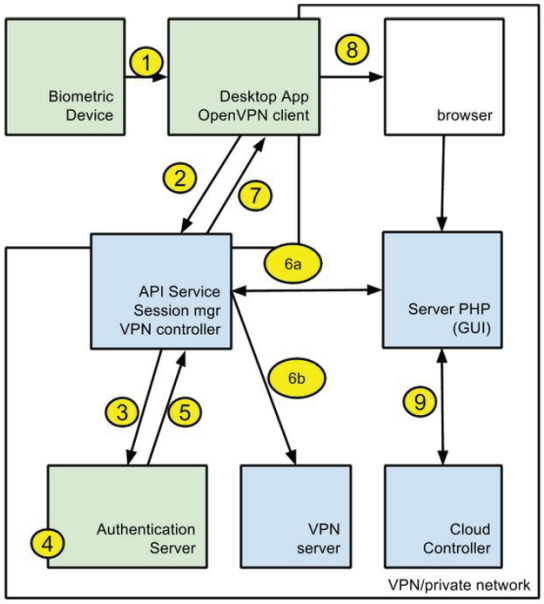


Fig.3 Authentication to the Cloud platform.

The authentication to the Cloud platform procedure, as shown in Figure 3, is based on the following steps :

1. The user interacts with the fingerprint scanner through a desktop client application. Such application produces a *model file* from the original fingerprints
2. The desktop client application contacts the API service, through a REST call (POST) and sends the *model file*. The call is asynchronous, so no one is waiting for a response. At the same instant a series of calls (GET) require the identifier of the user session to be created.

3. The service REST API connects to the Authentication Server (AS) asking for authentication and sending to this purpose the *model file* through the REST API (POST).
4. The AS performs the comparison of the *model file* with the content of its database. Once recognized, the user can retrieve the username and password associated with it.
5. If authentication is successful, the AS sends to the service API username and password (in response to the call POST)
6. Having username and password the API server :
   a)creates a new session on the web server;
   b)creates a new route on the VPN server that enables the user to access its subnet;
7. The API server responds to GET requests from the Client desktop application by sending the session ID.
8. The Client desktop application opens the browser with the address of the web graphical user interface (GUI) and the session ID.
9. The web server, knowing the username and password related to the session ID, can contact the cloud controller to manage cloud services.

Currently, a VPN is placed between the system and the user. This VPN selectively enables the services that can be accessed by the user: at the start of the process the user only sees the API server while, if authenticated, the system creates a route to the GUI. In this way, communications between the client and the API are always protected and the session ID is never transmitted in clear. A detailed overview of communication services is also given in Figure 4.

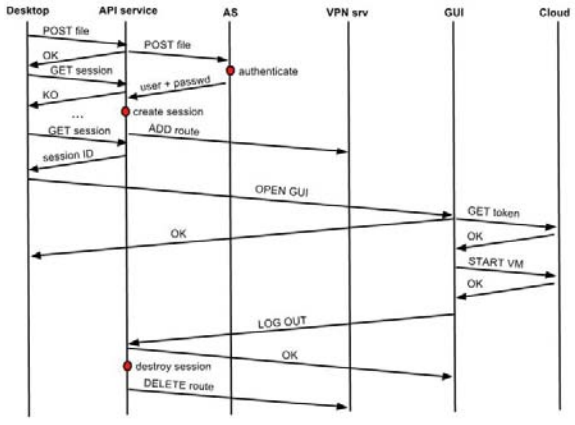


Fig.4 Communications between the authentication services of Cloud

With regard to the figures 3 and 4 it is worth to highlight some important aspects of the implemented security procedure:

- User and password to access the Cloud are never transmitted out of the cloud itself.
- Web GUI, AS and private cloud controller are not accessible outside the cloud.
- Sensitive data residing on the Cloud (fingerprint model file) are compared inside the cloud.
- The data transfer is not related to the user (nobody outside the cloud can associate the model file with some user information).

3.3 Multimodal biometric recognition

The Client desktop application is composed by a software for the enrollment of new users and an authentication application.

During enrollment, the new user's fingerprint is converted into a compact representation, called *model*; this *model* will be used to recognize the user. It is not necessary to store the fingerprints in the AS database; only the *models* are recorded.

The features to produce the model are obtained by using the Scale Invariant Feature Transform (SIFT) representation [9-11]. Recently SIFT has emerged as a cutting-edge methodology in general object recognition as well as for other machine vision applications [8-12]. One of the interesting features of the SIFT approach is the capability to capture the main local patterns working on a scale-space decomposition of the image. In this respect, the SIFT approach is similar to the *Local Binary Patterns* method [14-15], with the difference of producing a more robust view-invariant representation of the extracted 2D patterns.
The matching for the authentication application is performed considering the SIFT features located along a regular grid and matching overlapping patches; in particular the approach subdivides the images in different sub-images, using a regular grid with a light overlap. The matching between two images is then performed by computing distances between all pairs of corresponding sub-images, and therefore averaging them [12]. A fusion module takes the final decision.

3.4 Security of data residing

Cloud computing services and applications are faced with many challenges, including latency, unreliability, malicious behavior, mostly related to the public shared environment in which are hosted. In particular, security of outsourced data is still one of the main obstacles to cloud computing adoption in public bodies and enterprises. The main reason is impossibility to trust the cloud provider due to the lack of control that the user has over the infrastructure, an issue intrinsic of the public cloud model. To cope with these challenges innovative architectures and algorithms have to be developed.

In this project a secure and high availability data chunking solution based on innovative distributed cloud storage

architecture is proposed. The basic idea is to shard data in small chunks and spread them on different VMs hosted on cloud computing. The complete control of the distribute storage system is delegated to the user who hosts the master node of the system, as shown in Fig.6. The master node maintains the namespace tree and the mapping of blocks to the slaves nodes. Thus, only the user knows the location of the chunks needed to recompose the data. Even if a malicious user can access to one of the nodes which possess the chunks he cannot use it as the information is incomplete. This solution is a viable countermeasure also for malicious behaviour of the cloud provider.

Some of the features of the proposed solution are:

1. distributed storage system implemented in cloud, with client-server architecture and partially trusted environment;

2. security granted by chunking data and spreading it on different nodes (virtual machines) possibly hosted by different cloud providers;

3. availability and resiliency ensured by the redundancy of nodes and replica of chunks;

4. the possibility to use different cloud providers prevent also the so-called vendor "lock-in".

3.4.1 Distributed Storage Systems

There are two main categories of distributed storage systems architectures: Peer-to-Peer and client-server [16]. The latter architecture has been chosen for the implementation because best fit the objectives of the proposed solution. A client-server based architecture revolves around the server providing a service to requesting clients. The server is the central point, responsible for authentication, sharing, consistency, replication, backup and servicing requesting clients. In our implementation the master node embraces the server's role and slave nodes the client's role. As slaves nodes are hosted on the cloud, the system operates in a partially trusted environment; users are exposed to a combination of trusted and untrusted nodes [16].

In Distributed Storage Systems data can be replicated across multiple geographical sites to improve redundancy, scalability, and data availability, as shown in figure 5.

Although these solutions provide the scalability and redundancy that many cloud applications require, they sometimes do not meet the concurrency and performance needs because of the latency due to the network [17]. Some examples of the most known distributed storage systems are HDFS[18], Ceph [19], MooseFS [20], mongoDB [21].

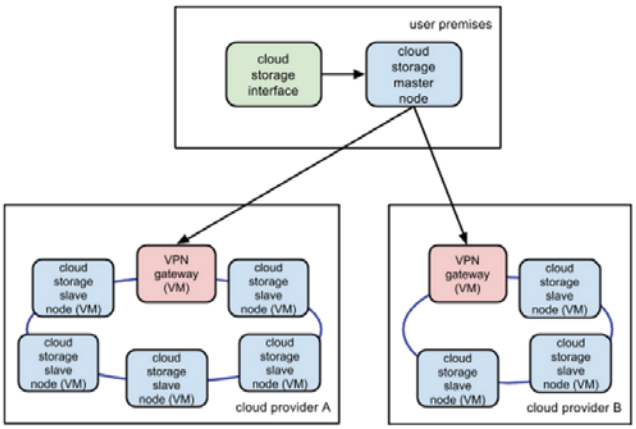


Fig. 5 – Architecture of the distributed storage system

3.4.2 Architecture of the system

The architecture of the solution is comprised of interconnected nodes where files and directories reside. There are two types of nodes the master node, that manages the file system namespace and regulates client access to files and the slave node which stores data as blocks within files. All nodes communicate with each other using TCP-based protocols. The mechanism of data protection does not rely on RAID approaches, but the file content is replicated on multiple slaves for reliability. Master and slave nodes can run in a decoupled manner across heterogeneous operating systems and on different cloud providers.

The complete control of the system is delegated to the master node, which maintains the namespace tree and the mapping of blocks to slave nodes. Slave nodes have little intelligence and not know the location of other slaves or chunks of data.

User applications access the system using a specific client, a library that exports the filesystem interface. When a user wants to perform a reading action on filesystem, the client first asks the master node for the list of *namenodes* that host the chunks of the file. After that, the client contacts a slave node directly and requests the transfer of the desired block. Instead, when a user wants to write on the filesystem, it first asks the master to choose slaves to host chunks of the file. All decisions concerning replication of the chunks are taken by the master node. This ensures the reliability of the data and the fault tolerance of the system.

3.4.3 Hardware and software

The server used for the Cloud Service is a Dell PowerEdge R620 with processor 2x Intel Xeon E5-2650 with high performances : 2GHz, 8C, cache 20MB, 8GT/s QPI, 95W, 256 RAM and 16 physical cores.

The AS is a Dell PowerEdge R210II with processor Intel Xeon E3-1220v2 Processor :3.1GHz, 4C/4T, 8M, Cache and 32GB RAM.

The fingerprints are acquired using the HI-SCAN PRO BIOMETRIKA scanner (FTIR, 500 dpi, 1" x 1").

The installed version of OpenStack is Icehouse 01.02.2014 issued on August 8, 2014. The installed operating system is a Linux distribution, free and open source: Ubuntu version 14:04 x86_64 server Long Term Support (LTS) with KVM (Kernel integrated) and included in the official repositories.

4 Conclusion

A complete system for web applications and data management over the Cloud, coupled with strong biometric authentication, is presented. The system guarantees the identity of the users and makes easy and secure the access to data and services. Moreover, the adoption of a data chunking solution based on a distributed cloud storage architecture is proposed. This provides protection of data residing also from provider's administrators and hardware supervisors. A further improvement of the system will extend biometric access to multimodal techniques, thus including face and face+fingerprint authentication. The development of a web server application for the user side, aimed to avoid the installation of local software, will be also pursued.

5 Acknowledgment

This work was supported in part by Regione Autonoma Sardegna LR 7 2007, N.7: "Promozione della ricerca scientifica e dell'innovazione tecnologica in Sardegna", *Piattaforme di Cloud computing per le PMI*, Codice: CRP-61647.

6 References

[1] M.K Srinavasin at "'State of the art cloud computing security taxonomies: a classification of security challenges in the present cloud computing environment, ICACCI 2012 *Proceedings of the International Conference on Advances in Computing, Communications and Informatics* Pages 470-476, 2012.

[2] European Commission, "Exploiting the potential of cloud computing in Europe," 27 September 2012. [Online]. Available: http://europa.eu/rapid/press-release_MEMO-12-713_it.htm.

[3] NIST, «NIST Cloud Computing Standards Roadmap,» 2013.

[4] M. K. Yinqian Zhang, «Cross-VM Side Channels and Their Use to Extract Private Keys,» in CCS'12, Raleigh, North Carolina, USA, 2012.

[5] Ross, Arun A., Karthik Nandakumar, and Anil K. Jain. *Handbook of multibiometrics*. Vol. 6. Springer, 2006.

[6] Vielhauer, Claus. "Biometric user authentication for IT security: From fundamentals to handwriting (Advances in information security, Vol. 18)." 2005.

[7] OpenStack, «OpenStack Cloud Administrator Guide,» [Online]. Available: http://docs.openstack.org/admin-guide-cloud/content/.

[8] Y. Ke and R. Sukthankar. PCA-SIFT: A more distinctive representation for local image descriptors. In IEEE Conf. on Computer Vision and Pattern Recognition, 2004.

[9] D. Lowe. Object recognition from local scale-invariant features. In Int. Conf. on Computer Vision, pages 1150–1157, 1999.

[10] D. Lowe. Local feature view clustering for 3d object recognition. In IEEE Conf. on Computer Vision and Pattern Recognition, pages 682–688, 2001.

[11] D. Lowe. Distinctive image features from scale-invariant keypoints. Int. Journal of Computer Vision, 60(2):91–110, 2004.

[12] Bicego, M., Lagorio, A., Grosso, E., & Tistarelli, M. (2006, June). On the use of SIFT features for face authentication. In Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on (pp. 35-35). IEEE.

[13] Jain, A. K., Bolle, R., & Pankanti, S. (Eds.). (1999). Biometrics: personal identification in networked society. Springer Science & Business Media.

[14] G. Heusch, Y. Rodriguez, and S. Marcel. Local binary patterns as an image preprocessing for face authentication. IDIAP-RR 76, IDIAP, 2005.

[15] G. Zhang, X. Huang, S. Li, Y. Wang, and X. Wu. Boosting local binary pattern (lbp)-based face recognition. In L. 3338, editor, SINOBIOMETRICS 2004, pages 179–186. Springer Verlag, 2004

[16] M. Placek and R. Buyya, The University of Melbourne, A Taxonomy of Distributed Storage Systems, Reporte técnico, Universidad de Melbourne, Laboratorio de sistemas distribuidos y cómputo grid (2006).

[17] M. Assuncao, R. Calheirosb, S. Bianchia, M. Nettoa, R. Buyya, "Big Data Computing and Clouds: Challenges, Solutions, and Future Directions", Journal of Parallel and Distributed Computing, 2014

[18] https://hadoop.apache.org

[19] http://ceph.com/

[20] http://www.moosefs.org/

[21] https://www.mongodb.org/

Using third party cookies for forensic identification

T. Eggendorfer
Hochschule Ravensburg-Weingarten, Weingarten, Germany

Abstract – *In network forensics it is common practice to rely on IP addresses to identify offenders. Considering most providers only delegate IP addresses to their customers for a short period, usually 24 hours, there is no fixed user to IP relation resulting in more complex investigations. Furthermore due to NAT IPs are possibly shared between multiple users, thus only allowing to identify a "line", but not a specific computer. Due to privacy laws in some legislations it might be illegal for the Internet provider to log which account used which IP when. Therefore this paper evaluates a new methods to identify perpetrators using third party HTTP cookies. Although this method is only useful for www-related offences and crimes, it solves most of the IP related issues in these cases and might in combination with IP addresses offer better and clearer evidence pointing to a specific person.*

Keywords: Forensics, Network Forensics, User Identification, User Tracking, Cookies, Privacy

1. Introduction

Currently upon investigating www-related crime investigators rely heavily on IP addresses transmitted. Since all Internet communication requires an IP this is the most ubiquitous piece of evidence. Because most systems automatically log the IP addresses of their communication partners, they are easily accessible for investigators. Plus all that is needed to identify the perpetrator is to call his provider, ask for the owner of the device this IP pointed to at a specific time and – ideally – arrest him.

In reality however IP addresses are far from being so useful. Chapter 2 will explain in detail why. Also IP address logging is being closely observed by privacy advocates, for good reasons. Thus for example the German constitutional court ruled "Vorratsdatenspeicherung" to be illegal after the German government has passed a bill forcing Internet access providers to store connection data for six months, allowing law enforcement to identify users of a certain IP address within this time period. According to the court this is in contradiction to EU privacy laws. [1]

Although the EU first tried to sue Germany to reinstate their law, in the meantime even EU politicians realised it might be a human rights issue and are reworking EU privacy laws.

The case was won by a German Internet access provider who claimed not having any operational reason to log IP addresses: Since all they offer would be flat rates, there would be no need for IP logs for operational reasons.

Based on that decision it might be a wiser concept to look for data that needs to be stored and saved for legal and / or operational reasons rather than trying to rename "data-retention" to "securing evidence" as some German politicians recently suggested – explaining this might help to "break the resistance" [2]: Doing so would still be illegal without at least suspecting an offence under the German constitution.

For most web service providers cookies are essential to provide their services to their customers. Especially in advertising but also for social networks and other Web 2.0 applications, user tracking using cookies is widely established. However this data is hardly ever evaluated during criminal investigations.

This paper will deliver a concept of how and when to use cookie data to be able to identify a perpetrator during criminal investigations. It will also show advantages over IPs as well as its limitations, possible legal implications and point to other concepts to identify offenders.

Previously cookies were only used to analyse a suspect's web usage after having seized his computer. There cookies might reveal pages looked at, support behaviour analysis and might actually provide login data through session data. This is a well established technique [3][4][5], however using cookies to identify perpetrators as suggested in this paper is a new approach.

This first section gives a motivational overview of the research conducted. Section 2 explains why IP addresses are very limited with a view to identify wrong-doers in the Internet. The following section 3 offers a short overview of cookies and their usage, from which section 4 develops a test case to determine whether the authors assumption of being able to use cookies for identification purposes is valid. The result of the test is used to show forensic usage of cookies in chapter 5, where its advantages and limitations are discussed as well. The last section summarises and offers an outlook on related and future research.

2. IP addresses as evidence

All relevant, i.e. higher level Internet communication relies on IP. Thus IP addresses are needed for all data transmissions. Since IP addresses are considered to be unique they are useful in identifying communication partners. Also most servers automatically log IP addresses, as well as network intrusion detection systems (NIDS) and firewalls do.

Due to their ubiquitous nature and them being readily available, IP addresses are heavily relied on during

investigations. From the author's experience many German police officers for example ended their investigation into offences and smaller crime as soon as they realised the IP address is registered in a foreign country, since this would mean tedious cross-border work which would rather unlikely yield any useful results due to the long time needed to manage the needed organisational overhead.

2.1. Dynamic vs. static IP addresses

This holds especially true for dynamic IP addresses, even though even the difference to static IPs is hardly ever noticed by most detectives. Since dynamic IPs are usually only assigned for a limited time only to a device, investigation is even more complex than for static IPs, since it requires a precise time line. Whereas for most static IPs, the owner is easily identified by looking at the respective whois record. Even if whois does not reveal the person, most providers could easily identify the owner of a specific static IP.

Considering privacy laws, static IPs are often considered to be contractual data which is a lot easier to be handed over to a third party than "movement" data, to which dynamic IPs would count [6], since they are only assigned while the contract is fulfilled and only for a short time. This is especially an issue in countries with very restrictive privacy laws such as Germany, where privacy is considered a constitutional right directly derived from human rights [7] and a requirement for a functional democracy, since it would protect citizens from a spying government.

In fact the German constitutional court ruled in favour of an ISP who denied logging whom a specific IP was assigned to, since it would not need the data for billing or other contractual purposes. The ISP therefore claimed it would be unable to provide this data to law enforcement agencies as required by German law [1].

As a result most German ISP now delete who has been assigned which IP after seven days. This again creates a massive requirement for quick investigations, which is hard to realise: For most offences the claimants would need a certain amount of time to realise they were victims to some kind of crime. Then they would need to file charges with the police, where usually non specialised officers are on duty. They forward the claim to their respective criminal investigation units, usually by in-house post, which might easily take another day. There again, for example a fraud specialist would realise support by the respective cyber crime unit is required and would forward the case. Then a court order would be obtained to require the ISP to provide the required data.

Adding up all those delays, realising that more often than not police is understaffed and overworked, seven days are a very tight time frame, thus in a lot of cases data has expired – ending investigation before it even really began.

2.2. Multiple Users / NAT

Another requirement in criminal investigations rather than civil law is to precisely identify the perpetrator. It is not sufficient to name and prosecute the contractor of e.g. the DSL line, the person who used it for the incriminated deed needs to be known. Since in most cases multiple users share a line with a common public IP address using network address translation (NAT) the IP address is not pointing at the offender. This would require a court to issue a search warrant for the premises to which the DSL line belongs.

However the inviolability of the home, a constitutional and fundamental right requires a search warrant to be as specific as possible as to on whose premises need to be searched as well as what is to be searched for. E.g. in a shared flat, the perpetrator's flat mates' rooms might not be searched, since they are no suspects. In court a search without proper warrant would invalidate evidence. Therefore it is somewhat difficult to obtain a search warrant under these circumstances.

It would be a lot easier if a specific person instead of only the address of e.g. a DSL subscriber could be named. Because of NAT this is often impossible.

2.3. (Prepaid) Mobile Internet

Among criminals it is very common to use prepaid mobile phone SIM cards to avoid leaving useful traces. This is especially common with drug dealers. They usually register their mobile contracts under false identities and use a multitude of different accounts for a short time.

More and more accessing Internet over mobile networks is becoming feasible, with sometime 4G networks being faster than WiFi networks. Over the last years mobile providers are offering very convenient prepaid SIMs with data coverage, usually with flat rates and fast 4G-Internet. This enables criminals to easily hide behind false identities when using mobile Internet. Any IP logged on the victim's side therefore points to a mobile device misidentifying the contractor.

Again, in most legislations triangulation using GSM base stations does require severe crime and thus is unavailable for "simple" fraud, thus this option would not be feasible for most www-related offences e.g. under German law. In a recent decision by a local court the fire brigade Dortmund was actually denied to triangulate a phone during an emergency call where the caller was unable to talk – it would infringe his privacy [13].

Additionally, some mobile providers use NAT by randomly bundling multiple mobile devices to one IP, coping with rare IPv4 addresses. In these cases not even the SIM card could be easily identified.

2.4. Public / Hotel WiFi

The same holds true for most hotels and public WiFi networks, e.g. at train stations, restaurants or airports. Even

though most might require some kind of pseudo-identification, they usually accept bogus data. Again, with that the perpetrator is hard to identify. Even though MAC addresses might be logged as well, they are rather useless, since they might easily be faked and would only point to a certain device – most computers are sold anonymously, i.e. there is no record of who owns which MAC address. Even if there was a record derived evidence might be challenged in legislations with restrictive privacy laws.

And again, most public WiFi hot spots would use NAT, therefore identification is only down to a certain access point – if at all. If the provider did not install additional logging devices, which again might be illegal due to privacy laws and thus render evidence useless, there is no way to identify the perpetrator's device.

Additionally cyber offenders also try to use random unsecured or not properly secured WiFi networks which would hardly ever provide useful evidence. The German high court in one decision actually suggested that if someone had turned on extensive logging on his not-public WiFi, this would actually mean he knew about the risk of his network being abused, therefore has not taken enough security measures and is liable for – in this case – pirate copies downloaded over his WiFi.

2.5. IP addresses and privacy

As pointed out above, some legislations are very strict on when logging of IP addresses would be in accordance with their privacy laws, therefore such logs might be invalid evidence – if available at all. Even though the author suspects quite a few providers to log a lot more than they would reveal to investigators, they would never offer this evidence since they might be charged for illegal logging.

2.6. Results of IP addresses as evidence

IP addresses are widely available, easy to obtain and might offer a first pointer to criminals, however, due to the internationality of the Internet requiring cross-border-investigations, ubiquitous usage of NAT, mobile Internet and public WiFi, they are often useless to identify a perpetrator.

Also, since there is usually no reason for a provider to log IP-user-correlating data, since e.g. with flat rates it is hardly ever needed for billing purposes, privacy legislation would disallow logging.

Therefore IP addresses alone are less useful than hoped for.

3. Cookies

Since HTTP is a stateless protocol [8], implementing features such as a shopping cart or Amazon's well known “One-Click”-order, recognising returning users or just maintaining session information for logged in users, requires another way to track state. This is what cookies were designed for [9], and are widely used for.

3.1. Cookies – a conceptual overview

A cookie is a little piece of information initially transmitted in the HTTP header from the server to the client containing a maximum of 4 KByte data which the web client will then store and re-transmit with each HTTP request sent to the server [9]. The server could thus either store a shopping cart in the cookie, which might not be a very good idea for security reasons [14], or use it to retrieve a unique ID identifying a specific user against the server's database and fetching associated login, cart etc. data.

Generally cookies will only be sent back to the server they originated from. Since the browser is in charge of deciding which cookies to store and to send, it is after all the user's decision on whether cookie data is available to the server. However on most modern web sites, disabling cookies would break the web page's functionality and render it unusable. Therefore most users will leave their browser's cookie functionality switched on.

Not allowing the server to poll cookie data but requiring the client to decide which cookies to send is considered to be a privacy feature [9].

3.2. Cookie usage for banner ads

For ad providers serving banners and the like of more or less annoying advertising, it is however important to track users across servers to identify their interests and thereby offering individualised advertising, which overall is considered to be less annoying and – by most users – actually seen as somewhat useful compared to non-personalised ads [10],[11],[12].

Since cookies may only be sent to the server they originated from, ad servers set their own cookies. This is possible, since – in a very basic setup – a banner is nothing more than an image fetched from the ad server via HTTP. Since requesting and sending the image back is a regular HTTP transaction with the ad server, cookies are transmitted between ad server and the user's web client.

For tracking and payment purposes, the original web site provider would include additional information into the URL needed to fetch the respective banner. Usually, these links include some kind of a unique ID to identify the web site the banner was requested from and might also provide some additional context information, such as where on the page or which items are currently being looked at in a web shop.

Banners have since become more complex, using iframes to load their content in to, Javascript to construct their URLs, Flash to display content or complex HTML5 constructs. Even though their presentation has become a lot more complex, adding in evasion techniques to work around banner filters, the basic principle described above remains unchanged: the entire exchange with their server still relies on HTTP and mostly cookies.

3.3. Similar usages of cookies

Similar to ad networks, Facebook, Twitter and other social networks use cookies to track users across web pages: On these buttons like Facebook's "I like" are integrated by downloading the button graphics from the respective network's web servers, thus enabling them to send and receive cookies in the context of different web pages.

Social networks then correlate these pieces of information to their user data base, as soon as the user logs in, and thus may identify their interests a lot better. Even though this is unacceptable from a privacy point of view, for these providers it is a very useful tool.

3.4. Privacy and Cookies

Cookies might well be acceptable under most countries privacy laws. In most legislation it is legal to store, retrieve and use data that is needed to conduct someone's business. Since ad providers need personalised data to offer personalised advertising, user tracking is legal under these legislations. The same holds true – to some extent – for social networks.

Additionally most legislation allow to store, retrieve and use data if the user has voluntarily agreed, some require a (somewhat) informed consent. Even though the author's definition of "voluntarily agreeing" differs from having to consent to the usage of cookies by clicking a full screen overlay away to access a web page, most legislations do assume that using a web site is voluntarily, thus the user could always opt out of cookie usage by not looking at that page.

Social networks add a bit of complexity to the privacy discussion: Registered users usually agreed to some conditions of usage that basically wave all their privacy rights, thus they might legally be allowed to trace them across the Internet. However judging by German law the author would assume that such a massive breach of privacy might be a surprising term in their usage conditions that therefore might not withhold being challenged in court. Obviously most social network providers seem to fear the same since they try to have their headquarters in countries with more convenient privacy laws, claiming data is not being collected within e.g. Germany.

Still assuming the legal position these social networks assume was in accordance with privacy laws they would be entitled to collect a plethora of information about their users' Internet usage which could be useful in identifying offenders.

3.5. Conclusion on Cookie Usage

Cookies allow user tracking across multiple web pages through social networks and ad banner providers. Depending on the legislation and interpretation of privacy laws these social networks and ad providers might be entitled to store this data, thus allowing to identify users based on cookies.

By contrast, most ISPs were not allowed to log IP data and extensively correlate it. Therefore cookies might be a meaningful tool to offer a lead to a suspect in web related crime.

4. Testing Cookies

To verify whether this assumption holds true some analysis needs to be done: The first assumption is there is only a small amount of ad providers and social networks serving a big community of web pages.

The second assumption is cookies are being used on all these pages and are thus available to track users.

The first assumption is easily tested for social networks: With Facebook, Twitter, Instagram and Google+ being linked on virtually any web page, a few adding in professional social networks such as LinkedIn and Xing, and with billions of web pages out there, these few networks would be able to track users across a relevant portion of all webpages.

The more social networks are being analysed the higher the probability of being able to trace someone around the entire web.

With ad providers this is a bit more complex, since their requirement is not so much to collect as many users as possible to offer most users a chance to find a multitude of their contacts, but to provide ads relevant to a certain audience and thus to certain web page providers.

A quick google search already offers a few million banner providers. But also listings of recommended big ad providers compiled by several web service agencies are hardly shorter than a hundred each and usually only have a few recommendations in common.

Another way to estimate the amount of banner providers is to go by "do-not-store"-cookie lists provided by privacy activists, some random sampling shows these lists will list between 10000 and 15000 entries each, with an overlap of less than 50%. This supports the idea of at least a few ten thousands of ad providers being on the market.

On the other hand some reports suggest among the biggest ad providers are Google AdSense, Tradedoubler, Doubleclick etc. If only a few ad providers would be used this would be of big advantage for this paper's idea to use their tracking cookies to provide a trace to online criminals.

4.1. Test setup

In order to test the hypothesis of only a few ad servers being used among a majority of websites a test setup was created. A web spider capable of storing and sending cookies back to web servers was written in Perl, which started of a list of programmed web pages, scraped all links from these pages and added them to a database of links to be followed.

Since banner cookies would be set using images and iframes rather than "href" links, all HTML "src="-tags were

also to be followed.

A database maintained also which page linked to where using href or src to allow for an analysis which banner networks were used on which site.

Since scraping the entire Internet would not be feasible, the test setup would stop after 1'000, 10'000, 30'000 and 100'000 different server-to-server relations were collected and analysed.

Some consideration needed to be paid to the pages the spidering started from since ideally a lot of different sites should be reached. For this reason, two pages offering a lot of links were chosen for the "big" searches of 10'000, 30'000 and 100'000 relations, that took multiple days to complete: A google search for "Cookie" as well as the Wikipedia entry for "Cookies". Both pages yielded very different results, because the word itself is ambiguous.

The smaller set of 1'000 pages started of a SOHO's homepage and went through all the links found there. This test set was chosen to identify whether starting on major or smaller pages would have an effect.

4.2. Findings

To test 10'000 different relations between webserver-names 121'623 different, normalised links were followed. This already indicates that quite a few pages link to each other. A factor of approximately ten new links to find a new server-to-server relation remained quite constant over the test.

This might partially be because the Google search and Wikipedia lead to huge amount of links to themselves and might thus affect this ratio, since roughly 50% of the links collected either pointed to Google or Wikipedia for the test sets with 10'000, 30'000 and 100'000 server-to-server relations .

One might argue this starting point for a search might have yielded sub-optimal results, since most tested links leaded back to some subpages of the respective starting points, but since it were the server-to-server-relations what was looked at, it probably would only increase the amount of links to be looked at. One would assume other starting pages might have lead to fewer links needed to be followed, thus increasing the probability to find new server-to-server-relations, however smaller test sets with private and SOHO home pages did double this server-to-server-relations links ratio, i.e. more links had to be followed to identify new server relations, which is most likely because smaller pages and business web pages tend to link less to external pages to protect their own business.

4.2.1. Testing cookie usage

With 1'000 different server-to-server relations, starting from a SOHO web page, 240 different cookies were detected, of which 137 (57%) were set from different domains, with some domains posting multiple cookies. Of these, German computer news service "heise.de" posted a total of 40 cookies, Twitter's URL shortener "t.co" 10 and Twitter itself 16.

With 10'000 different server-to-server relations, starting from the Google and Wikipedia pages, a total of 832 cookies were stored, only 475 – 60% of them pointed to different domains, with some domains again setting multiple cookies. The Twitter URL shortener t.co set a total of 72 Cookies, followed by Wikipedia, needing a total of 56 cookies and Microsoft with 24, Google and Amazon using a total of 16 each.

With 30'000 and 100'000 different server-to-server relations, again starting from Google and Wikipedia, the ratios did not change: still 60% of the cookies pointed to different domains.

4.2.2. Link-Relations between websites

As anticipated Wikipedia and Google were the pages with the most outgoing links – in all but the smallest test set they accounted for roughly 30% of all links to other pages. In the smallest set, due to the different starting point, 23% of all links pointed to "heise.de", 8,1% to "munich-business-school.de", 5% to "plus.google.com".

On the other hand, looking at which pages were linked from where, on the smaller 1000 pages set, 3% of all pages had links to facebook.com, 2% to twitter.com, followed by heise.de, plus.google.com, youtube.com and LinkedIn. After them, the first online marketing company appeared, the German ioam.de.

For the larger test sets, links to Wikipedia's imperium of pages accounted for a majority, followed by Facebook and Twitter, both of which had approx 2% of all links. A little less pages pointed to Google+, only 1,2 %, and 1% to Youtube.

Removing "href" links and only counting "src" links, which would be used in "img", "iframe" or "script"-tags, all of which are also used by online advertisers, on the 1'000 server-to-server-relation test set, approx 4,5% pointed to each ioam.de, heise.de, Facebook, Google Ads, DoubleClick, addthis and yieldlab. For the larger sets, approx 10% of the src-links pointed to Google Ad's and related Google pages, other services were at approx 2% each: youtube.com, ioam.de, facebook, cloudflare, jsdelivr, twitter and Amazon.

4.2.3. Cookie usage

All of the domains that appeared to be highly linked using src-tags used cookies – with the exception of ioam. However quite possibly ioam as well would use some kind of tracking service, that was not discovered using this test.

4.2.4. Test results and cookie usage

All test sets show that even though a large set of pages were visited, some receive a lot more links than others. These heavily linked pages also use cookies which – as the

test also showed – would also be set on items linked via HTML "src"-tags.

However this test only showed a surprisingly small subset of ad providers, considering on how many web pages ads appear, this is less than expected: One of the main reasons is very likely that the HTML needed to display the ad is being generated using JavaScript in the browser. The test bed did not evaluate JavaScript though, thus would neither identify these links nor follow them, ergo not loading the related banners and not receiving their respective tracking cookies.

Another issue are the more and more common Flash ads, with the test setup not being able to evaluate Flash movies and trace their cookies. However, Flash offers several technologies to track users.

Therefore it is safe to assume this test bed underestimated the usage of cookies in the Internet.

On the other hand, as a side result, it gave a quite interesting look on the importance of social networks and would allow a ranking of these based on web pages linking to them: Facebook and Twitter are almost as important, being linked from 2% of the tested pages, followed by Google+ with only 1,2%.

5. Cookies as evidence

Since most web services rely heavily on cookies to deliver their service, cookies might well be used to track users. Especially through content syndication, sharing, social media buttons and advertising cookies are able to offer cross web page tracking.

Cookies are stored on an individual computer in an individual account. They would therefore, if the computer could be seized for another reasons, uniquely identify this specific user account, but would also help in identifying an unknown perpetrator: If he would not have deleted his e.g. Facebook cookies and the page he was on used facebook, all action could be linked to his Facebook account.

5.1. Concept of using cookies as evidence

Analysing cookie usage in conjunction with the IP address used to access a specific web page would offer new ways to track online offenders down: If a crime was committed at a specific time on a web page, supported by ads from an ad server, the provider of the ad server would be able to identify this user's cookies. They would belong to the presumed perpetrator.

These cookie's data would point to an entry in his users' database, showing this users interests and pointing to other web pages the user identified by this cookie has also visited, and when. These pages again could reveal other information on the offender, such as pointing to other tracking services, e.g. by social media services' cookies and thus potentially related social media profiles. They could also point to his email provider's webpage, which might again be able to identify a specific account based on time and IP when e.g. an ad this mail provider's page was display related to the specific banner cookie.

Other constellations might as well be possible, e.g. if the offender has an ebay account or used online banking, forums etc.

All these information could be gathered, since for ad providers and social media sites user tracking is crucial and usually done to a massive extend.

Adding these pieces of information might well lead to the perpetrator or provide data to at least narrow down a search. Depending on the legislation the initial ad provider could then also be requested to further track the user and inform authorities as soon as he accesses specific pages etc.

5.2. Tools to support cookie forensics

A first tool to identify cookie usage on webpages and its relations with other pages would be a small programme analysing this web pages HTML. Another supportive tool might be "lightbeam" [15], whose primary intention was to educate users about privacy and the dangers of cookies.

Another option would be a publicly available knowledge base, providing information on which site cooperates with which one based on content-inclusion and thus cookie-sharing.

5.3. Advantages of cookies

Where IP addresses might or might not – as elaborated above – point to a specific Internet usage contract, be it via mobile or land line, cookies directly relate to a specific person, might – if used properly – relay information about his habits, other Internet usages, other network accounts and might even find evidence of other, previously unknown offences, since third party cookies offer a lot of tracing options.

Especially since IP addresses and visits to web sites are known, tracing along these used sites might help finding further evidence, finally pointing to the offender.

By contrast to IP addresses which might not be logged or stored for a longer time due to privacy reasons, third party web sites usually have a business requirement helping to bypass privacy regulations, allowing for longer storage. This might allow investigators to identify perpetrators in cases older than a few days.

5.4. Legal aspects of using cookies as evidence

Cookies are argued to be legal by banner networks and social media sites, however there is discussion on this being in accordance with privacy laws. While they are presumed to be acceptable in this context, in most legislations they are thus considered to be court admissible evidence.

They would also allow to formulate for much more precise search warrants, as requested by most legislations, since pointing to a specific person.

On the other hand, acquiring cookie data is a multi-step

process, since it first requires to identify third party operators on a certain webpage, sub-poena this third party to provide all known data about this user and his virtual whereabouts, identify these pages, sub-poena them to rely all usage patterns they could provide about a IP / time combination derived from the first set of cookies, identify further third-party cookies on these sites, and repeat the previous steps until the user has been tracked down.

This might – from a legal perspective – require a lot of interaction between courts and providers, imply cross-border requests and thus be a bit cumbersome.

5.5. Real world issues

This cross-border-approach might actually be the most difficult part of this research. This holds especially true for non-US investigators, since they would always need to rely on cooperation of US agencies when trying to access Facebook's, Twitter's and other social networks' as well as most ad networks' data, since most of these companies try to hide behind their US headquarters in other countries when it comes to criminal investigations against their users.

However these issues might easier be solved than with IP addresses: While IP addresses are understood to be data protected by privacy laws cookies are less so. Their data is also being collected for business purposes thus their existence cannot be denied, they are accessible through sub-poenas in some legislations, and there are only a few big players in that market that are usually fairly interested in successful criminal investigations and the related positive publicity, compared to some hundred Internet access providers alone in Germany.

Other issues could be users deleting their cookies regularly or using anonymiser services such as CookieCooker, where different users' cookies are mixed to protect their privacy.

In some cases ad services or social networks might also try to play down how much data they would have collected of a certain user, as the case “Europe vs. Facebook” shows. This could hinder investigations. However some legislations have powerful means, such as sub-poenas and search warrants against third-parties to overcome their reluctance. In most legislations court filings are somewhat public, thus after a while, data available at each third-party cookie-provider would be well known, thus resolving this potential issue.

6. Conclusion and further Research

Cookies provide a powerful mean of identifying perpetrators, although they have hardly ever been used yet. They offer a lot more information on the virtual whereabouts of an offender, easier tracking of his Internet behaviour than IPs would and might even reveal his email, social network or through banner ads even point to bank accounts.

Cookies, especially third-party cookies should therefore become a standard mean of web related crime investigations, instead of only relying on IP addresses.

Cookies are especially useful in legislations with strong privacy laws, protecting IP addresses. Since there is a multitude of third party cookie providers, it also seems probable to find more of them willing to cooperate with investigators and operating of countries with less strict privacy laws, thus providing more information.

The analysis of cookie usage and the relevance of third party cookies done during this research points out that on a relevant part of available web pages ad- and social-network cookies are being used.

Considering quite a few pages in the mean time require a Facebook login, a lot of online fraud is being conducted at eBay, and many forums etc. are financially supported by ads and often abused for stalking, insulting or other abuse, an even huger portion of potential relevant pages are infiltrated by third party cookies, thus allowing for more cases to be resolved this way.

Further research needs to be conducted onto the usage of cookies, especially on banners displayed using JavaScript, to further document the relevance of this new approach.

There is also a need for real cases in some legislations to add empirical knowledge to this. Other relevant tracking features include flash-tracking, browser fingerprinting or abusing the browser cache using E-Tagging, that might be worthwhile investigated.

7. References

1: BVerfG 1 BvR 256/08, http://www.bundesverfassungsgericht.de/SharedDocs/Entscheidungen/DE/2010/03/rs20100302_1bvr025608.html

2: http://www.csu-landtag.de/index.php?ka=1&ska=1&idn=603#.VSOqPhfGvwc

3: Volonino et al, Computer Forensics, Principles and Practices, Prentice Hall, New Jersey, 2007

4: Versalone (Editor), Mac OS X, iPod, and iPhone Forensics Analysis DVD Toolkit, Syngress, 2009

5: http://www.cclgroupltd.com/product/dunk-web-cookie-tool/

6: Laue, Vorgangsbearbeitungssysteme in der öffentlichen Verwaltung, Kassel University Press, Kassel, 2010

7: BVerfGE 65,1

8: Fielding et al, RFC7230, https://tools.ietf.org/html/rfc7230, 2014

9: Barth, RFC 6265, http://tools.ietf.org/html/rfc6265, 2011

10: Roger, Thorson, Publishing models and article dates explained, Routledge, Abingdon, 2013

11: Tucker, Social Networks, Personalized Advertising, and Privacy Controls. Journal of Marketing Research, 2014

12: Hugh, Cude, ‘Hello, Mrs. Sarah Jones! We recommend this product!’ Consumers' perceptions about personalized advertising, International Journal of Consumer Studies, 2009

13: http://www.ruhrnachrichten.de/staedte/dortmund/44137-Dortmund~/Datenschutz-Feuerwehr-darf-bei-Notfaellen-keine-Handys-mehr-orten;art930,2711053

14: https://www.owasp.org/index.php/Web_Parameter_Tampering

15: https://www.mozilla.org/en-US/lightbeam/

Emptying the Malicious Suitcase: Unpacking Malware in a Lab Setting

Tyler Hudak† and Kathy J. Liszka ‡

†Korelogic Security
Annapolis, MD
thudak@korelogic.com

‡ The University of Akron
Dept. of Computer Science
liszka@uakron.edu

Abstract

Packers are frequently used to obfuscate or encrypt the internal components of malware making static analysis difficult to perform. This paper presents information on how packers work for a security classroom lecture. Tools and techniques for both discovery and unpacking are given. A hands-on lab is described that provides essential skills to students.

Keywords: Malware analysis, packers, reverse engineering, static analysis, security labs

1. Introduction

Attackers use custom malware to target organizations, compromise their systems and steal information. Anti-virus (AV) software can no longer be relied upon as the sole defense against malware. It has become a necessity that organizations utilize in-house manpower to analyze malware. Having someone with the ability to perform malware analysis greatly benefits the incident response process. Primarily, malware removal will be faster as an organization will no longer be as tightly tethered waiting for detection and protection from an AV. The malware will be able to be analyzed in-house, allowing a fast determination on how best to remove it. Faster clean up means lower costs associated with the compromise. For this reason, computer security programs and classes have become more popular in academia. At the University of Akron, we are continually evolving a course in software security that teaches issues with software vulnerabilities and basics of malware analysis. Discussions on the need for malware labs including packers can be found in Kendall and McMillan's Black Hat talk [1] and in the SANS Reading Room [2, 3].

This paper describes what a packer is and how they are used with enough detail to construct an academic lecture. Tools and techniques for detecting if a file is packed are presented with a lab description. Three carefully crafted scenarios give students hands-on experience with different malware using packers. The paper is organized as follows. Section two describes an overview of static analysis. We present the mechanics of packers, why they are used by malware developers, tools for discovery, and techniques for unpacking malware for further investigation. In the third section, we discuss basics on setting up a security lab then give three very specific scenarios that uses carefully selected malware to demonstrate some of the core issues. Conclusions are drawn in the final section.

2. Overview

Malware analysis is the process of analyzing a potentially malicious file in order to determine as much information about that file as possible. This includes discovering if the file is malicious, what risk it poses to your organization, how it behaves, whom it contacts, how it can be removed and how it compromised the system in the first place. Malware comes in many forms and on many operating systems. The focus of

this course, however, will be Windows malware; specifically malicious Windows executables.

2.1 Static Analysis

Static analysis is the process of examining a malware sample without executing it. By examining specific information contained within an executable, we can determine much about what it does, who made it and how well known it is. Dynamic analysis, on the other hand, is the process of executing the malware in some type of sandbox environment and observing the behavior.

When performing static analysis, the goal should be to get as much information out of the malware as possible. The more thorough you are in the static analysis phase, the easier the rest of the analysis will be. To assist students, ask them to consider the following questions:

- What kind of file is this?
- Is it packed? If so, what packer was used?
- Do the cryptographic hashes of the file reveal anything?
- Is this already known by any anti-malware programs or listed on anti-malware sites?
- Are there any embedded strings in the executable? Are they revealing? If there aren't any, what does that mean?

The tools and techniques described in this paper address the questions on packing and provide a meaningful experience to student on the subject through a carefully crafted lab.

2.2 Packers

Files are often compressed to reduce their size. There are two common ways to approach this. The first is to use a file archiver to zip the file, such as WinZip[1] or 7-zip[2]. While virtually anything can be compressed, in this context, we are only concerned with executables. When the file is needed, the compressed file must be explicitly uncompressed with a program before accessing the executable and running it. The second approach is to compress the file by packing it. In this case, once compressed, no tool is necessary in order to uncompress and run the executable; it can be run directly. When run, the packed executable is uncompressed in memory and executed.

The mechanics of a typical packer are relatively simple. The original executable is compressed with a specific algorithm and placed inside wrapper code which contains unpacking functionality. When the new, packed executable runs, the unpacking code decompresses the original executable in memory and restores the original entry point (OEP). The original program then continues normally.

There are many packing programs available, many of them open source. Probably the most commonly used packer is UPX[3] because it is free, very efficient, and can pack a number of different executable formats. When unpacking, UPX restores a program to its original, uncompressed state. This is useful in static analysis, providing the analyst with the original code. When the original packing program is not available, or worse, unidentifiable, analysts need to manually unpack the program in a debugger which is extremely difficult and time intensive.

2.3 A Tool for Malware Developers

There are definite advantages to packing. The time to uncompress the executable in memory is typically not noticeable. It also isn't obvious that the file is packed to begin with as the file extension remains the same. One would need to be interested in knowing if a file was packed and also need the skills and tools to look for evidence of packing. Most notable, however, is the unintended side effect of obfuscating the original executable code, unless of course, the

[1] http://www.winzip.com
[2] http://www.7-zip.org/
[3] http://upx.sourceforge.net/

packer software is written by hackers for this express purpose. Not only does the malware become smaller and thus, easier to transfer, it encrypts the internal components making static analysis more difficult to perform [4] [5].

For example, there are numerous tools available to look at contiguous bytes of readable characters that can reveal useful information such as file names, email addresses, registry keys, URLs, function names, IRC commands, and so forth. This helps the analyst determine some of the malware functionality. In the process of packing an executable, however, the internal strings are compressed, hiding them from view. One needs to unpack the malware to view these strings or alternately, run the malware in a sandbox and dump the malware process from memory to analyze.

Some packers actually go on the offense and include anti-debugging components that prevent the packed executable from running if it detects analysis software running. This makes it much harder to analyze the malware.

2.4 Detecting Packed Malware

Students in our classroom start by determining if a malware sample is packed or not, and if so, what tool was likely used [6] [7]. This can be done with a number of different tools and observations. Multiple tools and techniques are used because one method cannot be relied upon to provide consistently accurate information. Following are four items students are asked to exam in the malware samples: entropy, embedded strings, section names, and common packer signatures.

- Entropy, in this context, is a measure of randomness in the code. During the process of compressing the original, the code tends to look more random [8]. The higher the entropy, the higher the probability that the malware is packed. We use pecheck.py[4] and PEiD[5] to calculate entropy. These tools are used to detect common packers for Windows Portable Executable (PE) files.

- Our students are exposed to embedded strings analysis in a previous lab. The presence or absence of "interesting" strings can serve as an indicator whether malware is packed or not. The absence of strings other than common library calls should lead one to suspect that the binary is packed. We use the Windows SysInternals program called Strings[6] and McAfee's BinText[7] programs for this purpose.

- When a packer is creating the wrapper, it must use different section names than the original executable uses. This can lead to unusual names that stand out and flag packed software. Some packers use a very specific naming scheme, as in the case of UPX. Others appear to generate section names randomly.

- Many packers leave signatures which tools like PEiD, TriD[8], and others use to identify the packer. In fact, the PEiD packer detection database has become the base standard for detecting many types of packers.

In many cases, the tools students use in the lab make probabilistic suggestions on which packer was used. If the tool is available, the malware is unpacked and further investigated. Dynamic analysis, as well, can be performed on the unpacked malware.

In some cases we've found that the signature of a particular packer is bogus. We suspect that the real packer used does this to intentionally

[4] http://blog.didierstevens.com/2013/04/19/3462/

[5] http://www.woodmann.com/BobSoft/Files/Other/PEiD-0.95-20081103.zip

[6] http://technet.microsoft.com/en-us/sysinternals/bb897439.aspx

[7] http://www.mcafee.com/us/downloads/free-tools/bintext.aspx

[8] http://mark0.net/soft-trid-e.html

thwart analysts. In those cases, we use a generic unpacker. This has two important implications. On the positive side, embedded string analysis can proceed without much trouble. On the negative side, the malware is no longer executable, as the generic unpacker does not completely reassemble the original executable with the proper sections and OEP.

It's possible that malware can be packed more than once and with more than one packing tool. We have not come across a sample like this but students are encouraged to do another packing analysis on successfully unpacked malware that do not provide more information and insight..

3. The Lab Session

3.1 Lab Setup

While malware analysis can be an extremely rewarding experience for students there are steps that should be taken to protect both students and the university. Any time malware analysis is performed, caution needs to be exercised to ensure accidental infection or compromise of other, non-lab systems does not occur. It is because of this, malware should never be analyzed on a system that is not designated and set up for malware analysis.

We use a dedicated closed facility that is physically accessible only to students currently enrolled in a security course or conducting faculty-led research. Computer systems in the lab are dual-boot Linux/Windows XP systems. VMPlayer is used on the Linux image for static and dynamic analysis of the malware. The machines are networked locally but there is no external network connection to a university network or the Internet.

3.2 Instructions

Students need to develop skills to determine if malware is packed, identify the packer used, unpack it and exam internal strings of the unpacked file. They are provided with the following instructions and three specific malware to work with in the first part of the lab.

The executables you will be analyzing are malware taken from compromised machines. This code is malicious and will compromise any computer it runs on! Take every necessary precaution to ensure that you only examine the malware on your analysis system and that the program is not accidentally executed. The MD5 hashes for the three zip files containing the malware are as follows:

- malware-01.zip
  8cab8f403e3f177297ca4b02bcb1c809
- malware-02.zip
  080172a72a5bb60ee6fe7b86767e2d2b
- malware-03.zip
  ef82555f52b0c472e4a95496a51bd1c0

3.3 Malware-01 Pedagogy

The first malware we provide for study is packed with UPX. This can be determined by running packer detection tools (TrID, PEiD, pecheck.py) on the file. When section headers of the packed malware are examined, section names ".UPX0", ".UPX1" and ".UPX2" appear in the packed executable while ".rdata", ".data" and ".recol" appear in the unpacked executable. A simple string analysis reveals further information that cannot be seen in the packed executable. As they do this, they are led to answer the following questions.

1. Is the malware packed? What do PEiD and TrID say about this malware? (Include the entropy from PEiD.)
2. Open the file in the HxD editor. Search for UPX. What did you find?
3. Use BinText to do a string analysis. What things did you find?
4. Move a copy of the malware into the UPX folder. This makes it easier to work with on the command line. Unpack it. What was the original

size, packed size, and ratio? What format is this file?
5. Use BinText to do a string analysis on the unpacked file. What interesting things did you find?
6. What registry keys are present? (These typically start with SOFTWARE\...) What URLs are present?

3.4 Malware-02 Pedagogy

The second malware is more interesting because it appears to be packed with WinUpack, a tool not available on our systems. After identifying that the file is packed and the packer used, they must use a generic unpacking tool, Gunpacker[9,10]. After unpacking, it cannot be executed because Gunpacker dumps the malware from memory. It does not rebuild the instruction address table (IAT) to produce a functioning executable. As they are working through this, they are led to answer the following questions.

1. Is the malware packed? What do PeID and Trid say about this malware? (Include the entropy from PeID.) You should notice a distinct difference of opinion. Document this.
2. Were there any modifications done to the packed executable to make unpacking more difficult?
3. Use Gunpacker to unpack it. What is the name of the new file you created?
4. Use BinText to do a string analysis on the unpacked file. What things did you find?

3.5 Malware-03 Pedagogy

The third malware is actually packed with UPX but upon inspection, has had the section names modified from UPX0 and UPX1 to XYZ0 and XYZ1. When they attempt to unpack it with UPX it fails. They are then led to use the HxD[11] editor to change the section names and attempt to unpack it again. It succeeds and they can proceed with embedded strings analysis. This has proven to be a simple, yet powerful part of the lab. As they are working through this, they are led to answer the following questions.

1. Is the malware packed? What do PEiD and TrID say about this malware? (Include the entropy from PEiD.) You should notice a distinct difference of opinion.
2. Were there any modifications done to the packed executable to make unpacking more difficult?
3. Move a copy of the malware to the UPX folder and unpack it. What happens?
4. If we use gunpacker to unpack this, we won't be able to execute it. Let's do something different. Use the HxD editor and do a search and replace of XYZ with UPX on all strings. Save the file. Delete the copy of it in the UPX folder and then copy this edited one into there. Now try to unpack it with UPX. What are the results and statistics?
5. Do a strings analysis on the unpacked file. What is likely a license number for Internet Explorer? What URLs do you see?

3.6 Malware in the Wild

A second component of the packer lab requires students to study malware not pre-analyzed for specific content and lab use. A large collection of malware retrieved from various sources is available in the lab. Students are instructed to take their knowledge and classify a subset of the malware as being packed or unpacked, identify the packers, and attempt to unpack them to further investigate the malware for interesting URLs or other useful strings. If an "in the wild" approach is taken where the instructors have not looked at all of the malware samples, we strongly advise that students be warned that offensive language may be found in code during malware analysis, whether it is in

[9] http://www.woodmann.com/collaborative/tools/index.php/GUnPacker

[10] Care should be taken with handling Gunpacker in a non-analysis system as it is identified as malware by popular AV programs. This nicely illustrates to students why malware should be analyzed in a closed environment --- sometimes you can't even trust your own tools!

[11] http://hxd.en.softonic.com/

packed or unpacked form. Our experience is that in most cases, it is not seen in packed form but some unusual strings can be found in unpacked malware.

4. Conclusions

It has become a necessity that organizations have the ability to analyze malware and improve the incident response process. For this reason, it is important that students joining the workforce have the ability to perform basic malware analysis to provide a fast determination on how best to remove it. A faster clean up means lower costs associated with the compromise. Training students with theory and practical hands-on labs is key to this process.

This paper discussed packed malware as a threat, provided material for instruction on the methodology for identifying packed software and provided techniques for unpacking to facilitate further analysis. A lab scenario that has been successfully used in the classroom was described. The authors may be contacted for the three pedagogical malware and written procedures used for the lab.

5. References

[1] K. Kendall, C. McMillan, "Practical Malware Analysis", Black Hat Conference, USA, 2007.

[2] C. Wright, "Packer Analysis Report-Debugging and Unpacking the NsPack 3.4 and 3.7 Packer", SANS Reading Room, Aug. 24, 2010, http://www.sans.org/reading-room/whitepapers/malicious/packer-analysis-report-debugging-unpacking-nspack-34-37-packer-33428.

[3] D. Distler, "Malware Analysis: An Introduction", SANS Reading Room, Feb. 12, 2008, http://www.sans.org/reading-room/whitepapers/malicious/malware-analysis-introduction-2103.

[4] S. Debray and C. Linn, "Obfuscation of Executable Code to Improve Resistance to Static Disassembly", *Proc. 10th. ACM Conference on Computer and Communications Security* (CCS 2003), Oct. 2003, pp. 290--299.

[5] Baig, M.; Zavarsky, P.; Ruhl, R.; Lindskog, D. "The study of evasion of packed PE from static detection", *2012 World Congress on Internet Security* (WorldCIS), pp. 99 - 104, June 2012.

[6] W. Yan, Z. Zhang, and N. Ansari, "Revealing Packed Malware", *IEEE Security & Privacy*, Vol. 6, No. 5, pp. 65-69, 2008.

[7] S. Han, K. Lee, and S. Lee, "Packed PE File Detection for Malware Forensics", 2nd International Conference on Computer Science and its Applications (CSA '09), pp. 1-7, December 2009.

[8] R. Lyda and J. Hamrock, "Using Entropy Analysis to Find Encrypted and Packed Malware", *IEEE Security & Privacy*, Vol. 5, No. 2, pp. 40-45, 2007.

Analysis of Feasibility and Security Measures on Dynamic Authentication

Jing-Chiou Liou
Deaprtment of Computer Science,
Kean University
Union, NJ 07083, USA

Abstract - *Authentication is the process of verifying users' credentials when accessing secure IT systems. Today, most of the computer users rely on a single-factor static authentication. However, many severe security breaches prompt us that we need to develop a better authentication mechanism. One solution is to adopt the multi-factor dynamic authentication. Nevertheless, it is financially not possible to implement multi-factor authentication for all the computer users. In this paper, we will study currently available dynamic authentication schemes and evaluate their performances with both feasibility and security measures. We propose eight feasibility measures that can be categorized into two groups: cost and deployment. In addition, we also use five security measures to analyze their capability against security attacks. The comparison indicates there is no perfect solution to the authentication. However, SofToken, SMS virtual tokens, GUS and SiFaDA score high in most categories*.

Keywords: Encryption, Multi–Factor Authentication, Dynamic Authentication, One-Time Password

1. Introduction

With advance in mobile technology and cloud computing, computers today have emerged and changed everything around the world. To most of us, it is becoming absolutely necessary to use technology in our daily lives. Through technological advancement, information is currently shared and accessed over millions of servers without boundaries. Even though computers are augmenting our daily lives, they require certain measures on access control and user authentication to assure the security. Authentication is the process of verifying a user's credentials when they are requesting services from any secure system.

A simple single-factor authentication only involves a username and password and this can be easily deciphered. Adding an extra strong factor will greatly reduce the chances of the user's identification from being hacked. For the second factor, there are many techniques available today. However, among billions of computers and the Internet users, due to deployment complexity, multi-factor authentication is only utilized in some close, controllable environment setting. These setting include those used by the employer in a company, suppliers in a supply chain, member with paid/profitable membership (e.g., Amazon Web Services AWS. Dropbox, PayPal, etc.).

Therefore, by considering the human psychological factor and deployment complexity, we study all currently available dynamic authentication techniques and evaluate their strength and weakness against on both feasible and secure measures. In this paper, we will propose, in section 2, evaluation measures for authentication techniques. Then we will study in section 3 the dynamic authentication methods that are available today. In section 4, we will review dynamic authentication schemes that use the virtual One-Time-Password (VOTP). All of the authentication techniques will be assessed using the feasibility and security measures. Finally, in section 5, we conclude our discussion and project on possible future works.

2. Authentication Evaluation Measures

Authentication is the process of verifying users' identities when they are requesting services from any secure system. During the authentication process, several validation factors may be needed for verification of the client's identity. An authentication factor is a portion of information that is given by the client and used to verify identity the client who is applying for access under certain security constraints. The authentication factor is usually one of three techniques: "proof by knowledge" (e.g., username/password), "proof by possession" (smartcard or token), or "proof by property" (fingerprint scan).

In this section, we propose two sets of measures to evaluate the authentication techniques. The first set exams the feasibility with eight measures. The second set assesses the security performance against five types of attacks

2.1 Feasibility Measures

There exist eight feasibility measures that can be categorized into two sub-groups: cost and deployment. Each of these eight measures may appear in both categories based on their specific requirements.

- *Hardware requirement:* This measure identifies the hardware cost for both the server and the users.

- *Deployment Complexity:* This measure specifies how difficult it is to deploy the technique.
- *Portability:* This is the measure that indicates how easy for users to use the particular scheme in different devices, either private or public.
- *Identity backup:* This measure shows how difficult to get the identity recovered if stolen or lost.
- *Lost Recovery:* This measure indicates the efficiency of recovering the credential, once lost. Single-factor has the best lost recovery, so this measure is primary concerning about the loss of second authentication form.
- *Replacement cost:* This measures the cost of replacing damaged or lost device that is used for authentication process.
- *Multi-Tenant:* This measures the capability of providing a universal mean to be used for multiple online services.
- *Human-factor impact:* This is to measure the human psychological behavior on using the authentication technique. Obviously, the single factor is the most popular with its easy to use and, traditionally, quite get used to it.

2.2 Security Measures

We will compare the five security measures for different authentication techniques. These will demonstrate why we should not use the single-factor authentication as it performs the worst in each of these measures.

- *MitM prevention:* This measure exams how well the authentication scheme preventing man-in-the-middle (MitM) attack. Single factor techniques are more vulnerable to this type of attack.
- *Phishing Prevention:* This measures how easy an attacker can acquire the account/password by masquerading as a trustworthy online service. Most of the OTP techniques will perform strong in this measure.
- *Spoofing Prevention:* This measure designates if the authentication scheme can withstand spoofing attack. The single factor does not achieve high in this measure due to it being incapable of protecting the user's identity from unauthorized parties.
- *Password Cracking Prevention:* This measure indicates the impacts of the password cracking on the password file stored on the server when it is suffered from data breach.
- *Shoulder Surfing Prevention:* This is the measure for how the user can avoid losing the credentials caused by shoulder surfing. The single factor performs weakly in this measure due to it being incapable of protecting the user's identity from direct observation.

3. Dynamic Authentication Schemes

Authentication schemes can be categorized into two different classes based on their attributes. Most commonly, they are grouped based on the number of authentication factor. Also, they can be classified by the frequency of passcode change.

When they are grouped by the number of authentication factor, they can be classified as either in a group of Single-Factor Authentication (S-FA) or Multi-Factor Authentication (M-FA). In M-FA, the authentication system requires the use of two or more different authentication factors.

If they are defined by the frequency of passcode change, they are referred as either a static authentication or a dynamic authentication. In the static authentication, the passcode is usually not changed until it is required by the security policy or per user's will. For the Dynamic authentication, the passcode changes in very short period of time, usually in minutes. Sometime the passcode changes every time it is generated which is called One-Time-Password (OTP).

S-FA focuses on only one factor: username/password, and is mostly widely accepted technique which is proved to be a weak method especially when it comes to protecting data.

In a study by a data security firm [7] that analyzed 32 million passwords exposed in the Rockyou.com breach in December 2009, the top five most common passwords among those 32 million users are: 123456, 12345, 123456789, Password, and iloveyou. In 2013, 4 years later, SplashData's annual "Worst Passwords" list shows again that 123456, password, and 12345678 top all other passwords among 38 million users [21].

Even using secure passwords, phishing and spoofing attacks may use an email or a site that looks like a legitimate one to tricks the user into supplying the password. As a matter of fact, news on October 8, 2009 reported that phishing scheme almost catches FBI Chief [15].

In addition, people usually don't change their passwords frequently. It was reported, in some cases, that less than 25 % of people change their password monthly and some 34% in a survey said they never change their passwords [22]. Therefore, a keystroke logger can be installed physically [9]or in the form of software to catch passwords entered manually on a login screen.

One improvement in S-FA is to utilized password management utility. Password management is achieved by using various password valet applications, such as RoboForm [17] and KeePass [8], which store user passwords and can automatically enter the required fields in a web form.

Nevertheless, the data is still kept on the host computer or device and can potentially be stolen through browser exploits, Trojan horses, etc.

Multi-factor dynamic authentication requires extra factor(s) other than the username/password. Using MFA will increase security, but also will increase difficulty in deployment complexity, hardware requirement and other aspect such as portability, lost recovery, identity backup and replacement cost.

The FFIEC issued supplemental guidance on this subject in August 2006 [6], "By definition true multifactor authentication requires the use of solutions from two or more of the three categories of factors. Using multiple solutions from the same category ... would not constitute multifactor authentication." To balance the tradeoff between security and deployment issues, the most popular MFA is two-factor authentication. Using two factors as opposed to one factor generally achieves a higher level of authentication assurance.

However, among billions of the Internet users, due to deployment complexity, TFA is only utilized in some close, controllable environment setting.

3.1 Security Token

Security tokens, also called OTP tokens, have an LCD screen that displays fixed number of alphanumeric characters. The OTP tokens are mainly based on two types of algorithms: Time synchronized and event-based.

- Time synchronized algorithm produces a pseudo-random number with a built in pseudo-random number generator. Pseudo-random number changes at pre-determined intervals, usually every 60 seconds.
- Event-based algorithm such as that proposed by the Open Authentication (OATH) consortium [14] uses a user event, such as the user pushing a button on the token.

Some devices, such as RSA SecurityID [18] and VeriSign (now part of Symantec) [24] shown in Figure 1, display 6 digits pseudo-random number and require periodically resynchronize the server with the token.

Figure 1 Security tokens

Taking portability into account, these security tokens must use materials that are small and consume less power. Still, these tokens need to be replaced every few years when the battery is dead. In addition, once the token is lost, the time and cost to replace can frustrate the user due to not being able to access their data. Finally, the security tokens do not prevent Man-in-the-Middle (MitM) based attacks against online transaction and is unable to defend against malicious users who could use the legitimate user's credentials for authorizing an illegitimate operation as explained in [20].

3.2 Virtual Token

Virtual tokens were first introduced in 2005 by a security company, Sestus [25]. Virtual token enables any portable storage devices to work as an authenticate token, where a protected file is stored on the device for authentication. Figure 2 displays a Safenet hybrid token [19] that use a USB drive as the security token.

Figure 2 Safenet hybrid token

Figure 3 Duo Push: One Tap Authentication

SMS-based T-FA uses user's cellphone as a virtual token. In this type of scheme, when a user enters the first set of credential, a text message is sent to the user's cellphone. This text message contains either a message for approval or an OTP for user to enter into the 2nd factor field of the login screen. A recent development, called "Duo Push: One Tap Authentication," that uses smart phone as the security token by [4] is shown in Figure 3.Virtual tokens reduce the costs normally associated with implementation and maintenance of multi-factor solutions by utilizing the user's existing portable storage device. Since the user's portable storage device is communicating directly with the authenticating website, the solution claims to not suffer from man-in-the-middle attacks and other forms of online fraud. However, if not implemented with OTP, can be attacked by phishing and spoofing. The virtual token can also be the weak link of defense.

3.2.1 Contactless Token

Contactless tokens form a wireless connection to the client computer that makes them more convenient than both connected and disconnected tokens. Examples of popular contactless tokens are RFID tokens and Bluetooth tokens.

RFID tokens uses RFID tags that store an agent (a small application program) and a pre-defined code (second factor code). The client computer should equip with RFID reader. The need for RFID reader on the client computer significantly increases the hardware requirement.

One example of the RFID token is the RFAA that was firstly introduced in 2011 by Liou, Egan, Patel, and Bhashyam [12]. RFAA token utilizes any RFID devices to

work as an authenticate token, where a protected file is stored on the RFID tag for authentication. RFAA is an enhancement process of SofToken mentioned in next subsection. RFAA will required a hardware specification that will be used as Second –factor authentication.

With the advance in smartphones and tablets, a new development in contactless tokens is to utilize NFC (Near Field Communication) that is already implemented in mobile devices.

Bluetooth tokens can be used in contact or contactless connection. When the client computer does not equip with Bluetooth, a USB input device is required to plug into the client computer. Thus, this causes uncertainty in hardware requirement.

3.2.2 Software Token

There are two primary architectures for software tokens: Shared secret and public-key cryptography. Shared secret architecture is considered more vulnerable than the hardware token. For both types, the configuration file can be compromised if it is stolen and the token is copied.

Figure 4 Examples of software token

As an example shown in Figure 4, RSA SecurID software tokens [18] basically support the same algorithms as their RSA SecurID hardware authenticators. Therefore, like its hardware token, a RSA software token produces either 6 or 8 digits number, called tokencode, and displays next tokencode, every 30 or 60 seconds. For online transaction service, in addition to a web server, it requires RSA Authentication Manager for token provisioning.

The generation of the tokencode is not triggered by the server, but is on client's device(s). A user enters the PIN to the installed application, and the client software generates the tokencode. One major concern with such time-based software tokens is that it is possible to borrow an individual's cell phone or laptop, setting the clock forward, and to generate tokencodes that will be valid in the future. In addition, anyone who provides the PIN correctly can retrieve the tokencode and use it for two-factor authentication on a web server from any cloned devices, such as an SIM card in a cell phone, or a USB installed with such application.

SofToken is an improved software token technique. SofToken was firstly introduced in 2010 by Liou and Bhashyam [10]. SofToken, rooted on software token, sends not just a pseudo-random number (an OTP), but also the encrypted key to the server for authentication. The technique significantly improves on feasibility and deployment cost of the two-factor authentication.

During the registration, client software installs two components onto user's computer with user's consensus: A logon application and a pseudo-random number generator.

During the initialization process, an encrypted public key will be created and issued to the user's computer as the seed of pseudo-random number generation. The key can be produced based on either a user's favored challenge-response or by the server. This encrypted key will be stored at the user's computer as part of the pseudo-random number generator.

Figure 5 Login scheme for SofToken

Shown in Figure 5, the logon application is directly communicating between the server and user's computer. The logon application requires filling in users credentials that are set up with the server. The user provides the first-factor to the server, username/password. When the server verifies the first-factor, the server sends a request to the pseudo-random number generator installed on the user's computer to trigger the generation of a random number, called code word. The logon application will provide the user the code word. The user is now able to enter the code word as the second-factor authentication. The code word will be verified again by the server. Depending on the code word, if it is correct the server will grant access to the database otherwise server will close the connection. SofToken acts as second-factor authentication.

4. Virtual OTP in Dynamic Authentication

Dynamic authentication uses cryptography or other techniques to create per-session authentication that changes with each authentication session between the claimant and verifier [13].

Dynamic authentication may have different types of protocols, such as challenge-response and virtual OTP. In a challenge-response protocol, the server which runs the authentication submits a *challenge*, e.g. a random sequence of bytes, to which the client computer responds by computing a cryptographic function which uses both the challenge and a secret data contained in the device.

Protocols using virtual OTP can be viewed as a type of challenge-response protocol in which the challenge is not sent by the server, but is a publicly known ever changing value. In this case, both the client computer and the server

compute the next random credential information based on the first factor. And the two computed random credential information is authenticated at the server side.

4.1 Graphical User Authentication

Graphical password scheme was firstly proposed by G. Blonder [1]. Researchers have latterly developed into several different fashions and now are all referred as Graphical User Authentication (GUA). As the name stands, a graphical user authentication uses a graph, instead of text, as the password.

Based on how they are being used, there are two types of Graphical User Authentication techniques:

- Recall based: The user needs to reproduce something that is created or selected earlier during registration phase.
- Recognition based: A set of images is presented to the user to recognize and identify the images selected earlier during registration phase.

4.1.1 Recall based GUA

This is somewhat very similar to handwriting scheme in the biometrics which may heavily rely on the touch screen on the device. A user will draw something in a way similar to what was drawn during registration. This technique can be further separated into two sub-groups.

- Signature scheme [2]: User will hand write on a 2-D lattice the signature shown in Figure 6, or some pre-entered words at certain area of the lattice.
- Draw-A-Secret [23]: As shown in Figure 7, user will draw a pattern on a 2-D lattice that correctly matches to the coordination pre-entered.

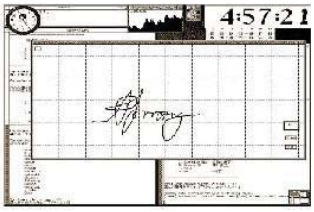


Figure 6 Signature technique by Syukri

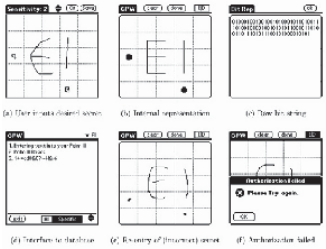


Figure 7 DAS technique by Jermyn

One major weakness for this type of GUA is the shoulder surfing security attack. Since neither technique uses an OTP, one can learn other's password by direct observation while the user is entering the password.

Another issue is the user has to correctly re-produce not just the word but also the coordination which may be a challenge to many people. Operate with a mouse to redraw can be also difficult to reproduce correct result. Moreover, this technique requires more storage and process time for authentication.

4.1.2 Recognition based GUA

There are quite a few techniques proposed in this type of GUA. One of the disadvantages for recall based technique is the shoulder surfing attack, therefore, many researchers proposed this type of technique to minimize the attack.

Some typical techniques in this category are:

- Dhamija and Perrig [3]: A user will select a number of images out of set of random pictures during registration. Later, during login the user has to identify the pre-selected images for authentication. An example of random images is depicted in Figure 8(a).
- Passface [16]: shown in Figure 8(b), a user sees a grid of nine faces, and selects one face previously chosen by the user. Here, the user may choose four images of human faces as their password. Since there are four user selected images, the authentication is done for four times.

Figure 8 (a) Random images used by Dhamija and Perrig (b) An example of Passfaces

- Shoulder Surfing Resistant: Quite a few recent researches focus on shoulder surfing resistant GUA. Shown in Figure 9, Haichang et al [5] proposed a new shoulder-surfing resistant scheme where the user is required to draw a curve across their password images orderly rather than clicking on them directly. This graphical scheme combines DAS and Story schemes to provide authenticity to the user.

The GUA has medium or better strength in all security measures but shoulder surfing. Although some recognition based GUAs claim to reduce the chances for shoulder surfing attack, they significantly increase the time for registration and later to enter correct password. The same drawbacks, as what the recall based GUA has, is the storage and process time for authentication. GUA is strong for

feasibility measure, except for lost recovery and human factor.

Figure 9 Haichang's shoulder-surfing technique

To maintain the security level high, the images arrange cannot be too simple. And that complicates the process of memorizing many different passwords. For many people, especially senior citizen, it is convenient for them to use simple password that they can remember. This is the psychological effect that keeps the single factor authentication alive. Hence, graphical user authentication is best used for private device authentication, but not online services.

4.2 SiFaDA VOTP Technique

The single-factor dynamic authentication (*SiFaDA*) was firstly proposed by Liou and Conway [11]. The idea centers in a virtual one-time password process. As the name stands, user just needs to keep hold one factor of credential. However, when the single factor is entered into the client computer, the client computer will generate, according to a binding code chosen during the registration process, a one-time password and transmit it through the Internet to the server. The server will compute, based on the same binding code, its own one-time password and match it with the one it received. The binding code is actually the virtual second factor for the dynamic authentication.

For a private device, user must firstly enter the username and the agent program will send it to server to request for a random code. Once the client computer receives the encrypted random code, the agent program will use it, along with the binding code, to produce a random pattern keypad.

If the password is 1234, the encrypted and transmitted password will be a random number, for example, 7208. Since the encrypted random code sent by the server lives one session only, the random pattern keypad will change every time the user log on to the server. Hence, SiFaDA will create a one-time password that is actually transmitted to server; even the entered password is actually the same.

When the server receives the username and VOTP from client device, it will retrieve both the encrypted binding code based on the username and the random number just sent to the client device, and produce its own VOTP. The server then matches its own VOTP with the received VOTP.

For a public device, the authentication process is very similar to the one for a private device, except that the user needs to enter the binding code first to temporally bind the device for service. The server will generate a random code as it would be sent to the private device. Thus the correct random pattern keypad will be produced, as it would appear on the private device.

In this case, the server will receive the username, VOTP, and the private binding code at the same time from client device. The server will retrieve user's binding code and match with the one sent by the client device. If they match, the server then matches its own VOTP with the received VOTP. Since the server knows the user is now using a public device, it will force the user to change the private binding code.

5. Conclusion

In this paper, we propose the performance measures for evaluating the dynamic authentication schemes on feasibility and security. We then conduct a study on currently available dynamic authentication schemes, and use the two sets of measures to assess these authentication techniques. The comparison, shown in Table 1, indicates there is no perfect solution to the authentication. However, SofToken, SMS virtual tokens, GUA and SiFaDA score better in most categories. This is primary due to the use of OTP or VOTP that attribute not only to its characteristics, but also its ability to maintain a higher level of security for the users. GUA is better for private devices. The SiFaDA is the only single factor authentication that stands out from all schemes and provides a better human-factor effect, multi-tenant on user friendly and security measures.

6. References

[1]. G. E. Blonder, Patents: Graphical Password. Last retrieved on 3/3/2015. WWW.google.com/patents/US5559961

[2]. The design and analysis of graphical passwords https://www.usenix.org/legacy/events/sec99/full_papers/jermyn/jermyn.pdf

[3]. R. Dhamija, and A. Perrig. "Déjà Vu: A User Study Using Images for Authentication". In 9th USENIX Security Symposium, 2000.

[4]. Due Push: one authentication. https://www.duosecurity.com/

[5]. H. Gao, Z. Ren, X. Chang, X. Liu, and U. Aickelin, "A New Graphical Password Scheme Resistant to Shoulder-Surfing," Proceedings of the International Conference on CyberWorlds, 20-22 October 2010, Singapore, 194-199, 2010.

[6]. FFIEC press release. Last retrieved on 3/3/2015. http://www.ffiec.gov/press/pr081506.htm

[7]. Imperva Releases Detailed Analysis of 32 Million Breached Consumer Passwords http://www.imperva.com/ld/password_report.asp

[8]. Keepass official site. Last retrieved on 3/3/2015 http://keepass.info/

[9]. Keysweeper. Last retrieved on 3/3/2015 http://samy.pl/keysweeper/

[10]. J-C Liou and S. Bhashyam, A Feasible and Cost Effective Two-Factor Authentication, Proc. 2nd International Conference on Software Engineering and Data Mining (SEDM '10), pp. 47 – 51, Chengdu, China, June 2010.

[11]. J.-C. Liou and J. Conway, A Single-Factor Dynamic Authentication for Computer Systems with Touch Screens, Proc. 23rd International Conference on Software Engineering and Data Engineering (SEDE '14), Paper ID. 29 in CD-ROM, New Orleans, LA, October 2014.

[12]. J-C Liou, G. Egan, J. K. Patel and S. Bhashyam, A Sophisticated RFID Application on Multi-Factor Authentication, in Proc. 8th International Conference on Information Technology: New generation, pp. 180-185, Vol. 1, pp. 6-10, Las Vegas, NV, April 2011.

[13]. NIST Guide to Selecting Information Technology Security Products (NIST Special Publication 800-36, Oct. 2003)

[14]. Open Authentication Consortium supports event based, and even time based OTP algorithms, http://www.openauthentication.org

[15]. Phishing Scam Spooked FBI Director Off E-Banking. Last retrieved on 3/3/2015. http://voices.washingtonpost.com/securityfix/2009/10/fbi_director_on_internet_banki.html

[16]. Real User Corporation: Passfaces. www.passfaces.com

[17]. Roboform official site. Last retrieved on 3/3/2015 http://www.roboform.com/index.html

[18]. RSA security http://www.emc.com/security/rsa-securid.htm

[19]. Safenet. Last retrieved on 3/3/2015. http://www.safenet-inc.com/multi-factor-authentication/

[20]. SC Magazine, Web Application Security in Un-trusted Client Scenarios, Last retrieved on 3/3/2015. http://www.scmagazineuk.com/web-application-security-in-un-trusted-client-scenarios/article/110448/

[21]. *SplashData's annual "Worst Passwords" list* http://splashdata.com/press/worstpasswords2013.htm

[22]. Steven Furnell. "Computer Insecurity: Risking the System, "pp. 54 – pp.56, Springer, London, UK, 2005.

[23]. A. F. Syukri, E. Okamoto, and M. Mambo, "A User Identification System Using Signature Written with Mouse," in Third Australasian Conference on Information Security and Privacy (ACISP): Springer-Verlag Lecture Notes in Computer Science (1438), 1998, pp. 403-441.

[24]. VeriSign. Last retrieved on 3/3/2015. http://www.verisign.com/static/043732.pdf

[25]. Virtual Tag™ multi-factor authentication. Last retrieved on 3/3/2015. https://sestus.international/

Table 1 Performance Comparison on Dynamic Authentication Schemes

Performance	**Security Toekn**	**Virtual Toekn**	**Software Toekn**	**SofToekn**	**RFAA**	**GUA**	**SiFaDA**
Hardware requirement	Medium	Medium	**Low**	**Low**	Medium	**Low**	**Low**
Deployment complexity	High	Medium	**Low**	**Low**	**Low**	Medium	**Low**
Portability	Medium	**Medium/High**	Medium	Medium	Medium	**High**	**High**
Identity backup	Low	Medium	**High**	**High**	**High**	**High**	**High**
Lost recovery	Low	Medium	**High**	**High**	**High**	Medium	**High**
Replace cost	High	Medium	**Low**	**Low**	Medium	**Low**	**Low**
Multi-tenant	Low	**Medium/High**	Medium	Medium	Medium	Low	**High**
Human factor impact	High	High	**Low**	**Low**	**Low**	Medium	**Low**
MitM prevention	Medium	**Strong**	Medium	**Strong**	**Strong**	**Strong**	**Strong**
Phishing prevention	**Strong**	**Strong**	Medium	**Strong**	**Strong**	Medium	**Strong**
Spoofing prevention	**Strong**	**Strong**	Medium	**Strong**	**Strong**	Medium	**Strong**
Password cracking	**Strong**	**Strong**	**Strong**	**Strong**	**Strong**	Medium	**Strong**
Shoulder surfing	Medium	**Medium/High**	Medium	Medium	**Strong**	Weak/medium	**Strong**

SESSION

SPECIAL TRACK: CYBERSECURITY EDUCATION

Chair(s)

Prof. George Markowsky
Dr. Linda Markowsky

Cyber Defense Training and Human Systems Integration

An HSI Based Method for Conducting a Cyber Defense Job Task Analysis and Evaluation Process

W. Quintana

UMaine Cybersecurity Lab, University of Maine, Orono, Maine, USA

Abstract

The selection of a pedagogical approach to cyber-defense team education and learning has been a difficult decomposition problem when considered in the context of traditional learning systems. This problem has been exacerbated by the social stove-piping associated with the "hacking culture" and the insular social nature of "cyber-warfare" professionals.

This in turn creates a lack of progress by professional teaching organizations at the collegiate level, to define the requirements for and build training path systems which would support the development of cyber-defense professionals, the accreditation of the systems which educate them, and the ability to validate their level of knowledge and expertise through measured and qualified examination methods.

What this paper will explore, is the promotion of a Criterion Performance Based approach to the characterization of the cyber defense learning process, and some early anecdotal evidence that that process can be functional and can be evaluated, in an iterative manner.

Introduction

Many cyber-warfare and cyber-defense professionals exist at the individual level and are self-educated from a need based perspective. In an operational environment in which individual prowess and acclaim are critical to survival, then this model works and is self-promoting. Hackers base their existence on two fundamental rules which are being very capable and the self-protection or isolation of ones skill set [1]. Popular culture further promotes this paradigm as many hackers are seen as modern day Robin Hoods who take advantage of the global dependence on the unlimited information enterprise, while balancing their self-serving need to keep that continuum in existence. Politics aside, the reality is that great economic and civil rights crimes are often created in the name of cyber freedom, and often at the loss of those whose primary utility for the internet and its digital extensions are of necessity in the conduct of their daily lives. Many recent thefts of the digital and economic identities of mass numbers of individuals, and hacker internet intrusions into business operating systems do not require amplification here.....they number in the tens of millions. Digital blackmail by internet terrorist organizations operating under the guise of social restructuring, as well as by terrorist organizations using it for the dissemination of their particular agendas, have made the internet a dangerous place to exist. While promoting the destruction of the same societies which are responsible for its creation and maintenance, many of these entities ignore their need for a sense of social contract, and impose their "will" without due regard for impacts or cost to others. For all of these entities, their focus is disruption and economic gain using nefarious tools and methods, which need to be countered to maintain some sense of social "law" within the net itself.

The need to maintain the net as a viable working entity then falls upon cyber-defense individuals and teams who are in a constant "response to another new threat mode", as prolific and previously unseen viruses, bots, invasive macros, malware and intrusive programs are introduced. The cyber-defense teams are in all respects the internet police who act at many levels within the internet, and can either be privately employed or an instrument of government systems

and programs. Ironically, some of these government organizations are themselves in the business of net interruption and complicate the matter even further. For cyber-defense professionals, their jobs become even more difficult as they must also contend with the problems associated with both the intentional and unintentional vagaries of inside users who expose the systems over which they have responsibility to intrusion and harm....most times out of ignorance or inattentiveness.

The US Department of Defense has traditionally used *Criterion Based Education* (CBE) as the base process for the creation of training development within its education programs because by using this construct they are able to clearly define the performance requirements of hard skills. From a traditional collegiate pedagogical perspective, this is counter to method and form in many non-STEM (Science, Technology, Engineering and Math) learning progressions. Paradoxically, and mostly by default, many STEM assessment measures are truly objective based. Writing code, developing data tables, and engineering IP routing structures are examples of a few of these "hard" skills.

Where traditional education excels is in understanding and defining methods for the measurement of soft skills. Soft skills being those behaviors which are outcome performance based, but often evaluated by measuring exhibited nuance and outcomes. Communications, organizing people, evaluating data, linking clues and leadership are examples of a few of these soft skills. Many of these skills are intangible from a hard objective criterion based measuring perspective, and require a more subtle, experiential observation and evaluation process. At a high level, cyber defense, and more importantly cyber defense team effectiveness is developmentally the result of individual knowledge, skills and abilities, (KSAs) combined with team aggregate performance.

Where traditional education is struggling, is finding ways to develop a path to that higher team performance level and doing so in ways which can be measured with a test set that understands the need for both soft and hard skills. Traditional education is also individually focused where military "training" is more optimized to a team outcome. Military success is often measured in less than optimum socially positive outcomes driven by aggressive competitive behaviors, while traditional educational measures are less destructive and sometimes less of a hard target tangible.

The Human Systems Integration (HSI) Model

The recent emphasis on the use of ergonomics as a discipline, and more importantly Human Systems Integration on a system level by organizations within the United States Department of Defense has resulted in the creation of the DoD HSI concept (Figure 1). This model provides insight into how defense organizations and combat teams can be decomposed, engineered and organized so that their net effectiveness can become a realization of the man-machine synthesis and deterministic performance that is so often sought by DoD organizations using a comprehensive engineering approach [2].

Figure (1): US Air Force Conceptual HSI Model.

One of the core tenets of this approach has been one in which the optimization of education and training, and the development of the Human

Computer and Human Machine interface tasks have been decomposed to a significant level of detail. It is an engineered component of system and device design in which the integration of the human component carries the same commoditization approach as fuel or power.... people are an allocated resource; a resource which needs to be managed and optimized to its greatest potential as a component part of the larger system structure. From an engineering perspective, the human component is analyzed to a detailed series of tasks which they (the operator or team) must perform, in concert with the system which they are a part of. This detailed task analysis effort is linked to specific criterion based objectives and outcomes which not only measures operator and team performance, but provides feedback on the usability of the systems themselves. This closed cycle then provides feedback on the design of the system, resulting in team performance improvements.

Within some DoD HSI programs, the use of usability testing, which is uniquely akin to the Collegiate Cyber-Defense Competition structure, has demonstrated that properly structured evaluation environments can be used to evaluate the use of both soft and hard skills within defined limits. This process, in some context, can help program developers create educational programs which are derived by using cognitive analysis and team dynamic skills performance evaluations to help reverse engineer what skills would be required by cyber-defense teams, and creating learning systems to support those skills.

However, in order to properly define or characterize outcomes for hard skills, the cyber-defense team actions need further decomposition to a skill level [3]. Socially, this will be somewhat difficult because traditional task based analysis will require the cooperation of experienced users who would have to work to define user levels skills and tasks, something which they (the hacker community) have not shared well. Within the cyber-defense community, this is probably achievable, but will most certainly never be fully achieved as the basis for the skill set is constantly evolving – experience in the field is required. So just as a true infantry soldier isn't really qualified until they have "seen the elephant", most cyber-defense operators are not really "qualified" until they have been in the field.

For soft skills, the use of observer based evaluations would still be required, but as has been preliminarily evaluated, may be easier to conduct using traditional HSI observation evaluation tools, methods and devices. Though some of these observation devices, like the NASA-TLX subjective evaluation tool may not be scholastically appropriate. In addition, the evaluation of cyber-defense teams, as measured against a traditional Tuckman Team Development Pyramid [4] (Figure 2) has strong application. Observers can be trained to use the human performance clues that support the team progressions suggested by Tuckman's model, assuming they maintain their objectivity. Educational and behavioral psychologists will differ on the appropriate triggers, but a council of both disciplines should be able to create a series of appropriate observation protocols.

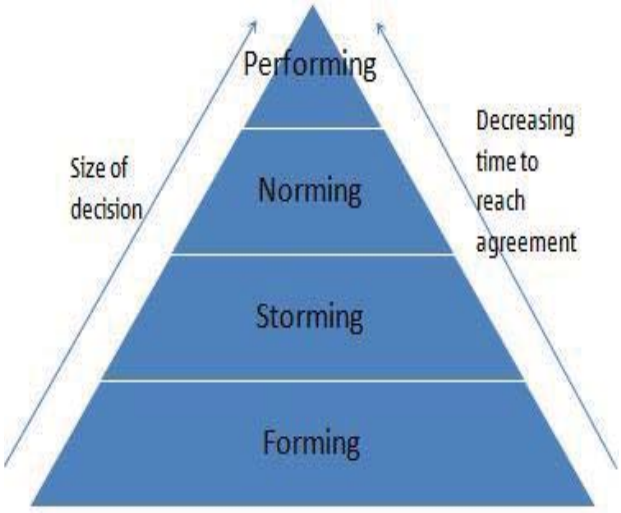


Figure (2): Tuckman Team Development Pyramid.

NECCDC Soft Skills Observations

During the last two NECCDC (North-Eastern Collegiate Cyber-Defense Competition) competitions, trained observers with human systems integration backgrounds were tasked to evaluate the performance

of competing teams using soft and hard skill observations usually indicative of overall team performance and abilities. In both collection efforts, the same subjective evaluation areas were used with emphasis on observable behaviors, dynamic and environmental conditions, and operator/team actions. They were:

- Use of administrative tools
- Interpersonal conflicts
- Noise levels
- Maintenance of Situational Awareness (SA)
- Organization of facility and operators
- Inter-operator CLAF
- Team Construct
- Defined Team Objectives
- Dealing with unplanned events
- Re-norming (loss of an operator)
- Group-Think induced errors

Using the items above, each team was given a dashboard color grade by the observers for each of the following:

- Physical Organization
- Personnel Organization
- HCI Optimization
- Communications – Internal
- Communications External
- Work Assignments and Assignment compliance
- Administrative Organization
- Logistics

All observers specifically kept their observation comments outside of the normal grading and scoring process and then when the competition was complete, compared their subjective dashboard to the final competition results (Figure 3). Additionally, the observers attempted to create a template for measuring team time results based on injects and problem resolutions, or problem identification, to try to make some connection on Team Development.

The dashboard scoring does show that the observed measures, loosely quantified, did give some initial indication of how teams could be measured predictively. For both competitions that were observed the teams who measured well in the dashboard dynamics did perform well in the overall competition. Not all teams performed well across the full spectrum of observed dimensions, noting that the team with all green dimensions placed well but did not "win"). However teams with a preponderance of less than green evaluations did in fact not perform well.

This work produced some preliminary results, but there is much follow on work that needs to be done here before any precise connections can be made. However, team behavior observations noted the reduction in reaction times could be connected to overall team performance and behavior within the construct of the competition. Simply put, the better teams defined and reacted to competition injects faster and with better clarity and diagnosis (performance optimization).

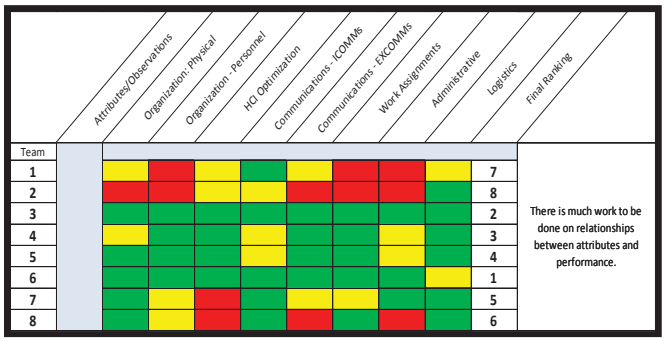


Figure (3): NECCDC 2014 Competition Dashboard.

Preliminary Evaluation

At a very preliminary level there is some indication that the ability to evaluate soft skills as they apply to the conduct of cyber-defense team success is achievable, but it must be emphasized again that much follow on work will be required. A clearer set of measures and measuring tools will have to be used. Also, the sample rate, given the number of teams and competitions observed was low, though the overall number of operators exceeded one hundred in total. Any future efforts must also include the ability to evaluate individual operator performance as each teams' results are the compilation of operator tasks and abilities leading to an aggregate outcome. For

each operator "watchstation", the number and decomposition of operator tasks are currently assessed to be on a hundreds per operator basis, and the education of each operator (user) would involve the creation of operator to sub-team to team continuums. More discrete measures would be required and more akin to an HSI style KSA cognitive decomposition, which attaches a Verification Item (VI) based evaluation of each skill and operator with trace-ability to total team outcomes.

For overall evaluations, a set of conditions, behaviors and standards would have to be assigned and assessed to each task and driven by the test scenarios so the appropriate stimulus is provided. Automation taps and triggers (like button or keystroke actions) would help as well, though this would complicate and drive the cost of conducting the competitions, as well as evaluation reconstruction.

There may also be some benefit to examining a skills based progression which places value on skills mastering first at the individual operator level, and then as a gradual melded team member. Mastery before progression would be the primary objective with mastery being the *key measure metric*. Just passing would not be sufficient – this will be a hard sell in a tough high cost education market. And it will be even tougher for some students to accept the fact that they haven't made the cut.

The concern by educators on the use of this process to drive operator classification is not without merit, and could lead to pigeon holing or pre-disposition of individuals. Review board checks and balances would have to be used to protect the students.

Next Steps

From the very preliminary information achieved, some measure of follow on work would be required to take this exploration of alternatives to a definitive conclusion. However, the author believes that a defined training path can be developed, using a multi-phase approach; to wit a Training Path System that can be audited and measured, and adjusted in process. The proposed development path would be as follows:

The first phase would be a high order definition of an education continuum or degree program which defines the requirements for leading a cyber-defense team as its final T0 (tee-zero) objective (military expression) or terminal objective. Many soft skills would be associated with the plan and would have to include the use of traditional base courses, learning paths and course curriculums, and include a focus on communications, writing, math and computer skills. Basic computer coding and decomposing should be included early as there is some measure of operator aptitude which would need to be assessed soonest in the progression of the student. There is still some truth to the concept of knack and inherency which has to be considered. Natural ability needs to be identified and quantified so as to be optimized.

The second phase would be the decomposition of the higher order individual operator and team tasks and their required KSAs, and the optimization of traditional devices to organize them into trainable or learning objectives. Some will be hard skills and others will be soft skills. For example, teaching students methods for obtaining user information by soft skills (social engineering) or by using brute force code intrusions. This is probably the most difficult of the development phases as it would require participation by field experts (experienced hackers) and educators experienced in the execution of cognitive task analysis collection efforts.

The third phase would be to involve the individual students in a rolling set of team experiences which require changes in rolls, increasing levels of conflict and complexity and evolving team dynamics (i.e. loss of an operator or team member).

The final phase would be one in which each student is taught to evaluate other students and teams which would improve their ability to manage teams in the field, as well as assess the current state and abilities to any organization to which they become attached after their education is complete.

As the DoD model is often based on a traditional labor model which divides skill competency into apprentice, journeyman and master levels, equating them into a traditional collegiate structure may be difficult. When one adds the dimension of field experience this is even harder to quantify or measure. A strong field experience or intern program would be required. A key question being how does one get apprenticed to an approved experienced hacker?

It is important to note that none of the phases listed above can be completed without first conducting a thorough series of both Job Task and Cognitive Task Analysis events. The Job Task Analysis, (JTA) which is a process device usually associated with the creation of Criterion Based Education (CBE) systems, would focus on high order learning objectives and goals. The Cognitive Task Analysis (CTA) would focus on defining the more discrete HCI and HMI centric tasks usually associated with a traditional computer-based work flow. The initial task assessment conducted during the NECCDC competitions discussed above, attempted to use some of these historic task items, but at a high level, and without due regard for the details of those tasks. They also focused on observations of human behaviors which were more interactive and less introspective.

The actual execution of the task analyses described above would have to be an interesting derivation of the traditional BIL (Business, Industry and Labor) approach to creating task based education. For starters, the cyber community does not dovetail well with the Business, Industry and Labor categories associated with the traditional BIL makeup. In addition, how does the authoring authority of this effort validate the technical competency of the participants, and their sense of fair play in light of the "hacker culture" discussed above. This would most certainly not be a survey based data gathering effort.

Conclusion

The process for the creation of Cyber-Defense education exists if one is willing to adapt to a hybrid system that incorporates the best of what traditional education can structure, with a systems development approach used by defense professionals.

Previous experience, such as that reported in the *Career Technical Education Pathways Initiative Annual Report (2014)* [5], provides some examples of how objective based education, using criterion based analysis can be accomplished. This was done using an established BIL process and relationship, combined with an understanding of the nature of skills based education.

A thorough decomposition of the skills required by cyber-defense teams, both soft and hard, would require a cooperative BIL effort. The cooperation of experienced cyber professionals and the hacker talent pool is the long pole in the tent. More importantly, the follow on evaluation and certification process could be problematic if a carefully defined set of testable objectives is not agreed upon. Further exploration of this initial work is recommended and deterministic statistical analysis would have to be attached to any additional efforts.

References:

[1] Tim Jordan and Paul Taylor. "The Sociology of Hackers". The Sociological Review: Kings College of London, Vol 46 pp 757-780, Nov 1998

[2] Harold Booher. "Handbook of Human Systems Integration": John Wiley and Sons, Inc., 2003

[3] Beth Crandall, Gary Klein and Robert Hoffman. "Working Minds, A Practitioners Guide to Cognitive Task Analysis": The MIT Press (2006)

[4] Bruce Tuckman. "Developmental Sequence in Small Groups". Psychological Bulletin: Naval Medical Research Institute and American Psychological Association, Vol 63, pp 384-399, Jun 1965

[5] California Community College. "Career Technical Education Pathways Initiative Annual Report": California Community College's Chancellors Office, CACommColleges, 2014

Experiences with Establishment of a Multi-University Center of Academic Excellence in Information Assurance/Cyber Defense

R. T. Albert[1], C. Bennett[2], D. Briggs[3], M. Ebben[4], H. Felch[5], D. Kokoska[5], L. Lovewell[6], C. MacDonald[6], G. Markowsky[7], L. Markowsky[7], J. Murphy[8], E. Sihler[6], G. Wilson[9]
[1]Arts & Sciences Division, University of Maine at Fort Kent, Fort Kent, ME, USA
[2]Division of Mathematics and Computer Science, University of Maine at Farmington, Farmington, ME, USA
[3]Department of Computer Science, University of Southern Maine, Portland, ME, USA
[4]Communication and Media Studies Department, University of Southern Maine, Portland, ME, USA
[5]Computer Information Systems Department, University of Maine at Augusta, Augusta, ME, USA
[6]Maine Cyber Security Cluster, University of Southern Maine, Portland, ME, USA
[7]School of Computing and Information Science, University of Maine, Orono, ME, USA
[8]Philosophy Department, University of Southern Maine, Portland, ME, USA
[9]Department of Technology, University of Southern Maine, Portland, ME, USA

Abstract – *The National Security Agency (NSA) and Department of Homeland Security (DHS), in response to an unmet workforce need for cybersecurity program graduates, jointly sponsor a program by which a post-secondary education institution may achieve recognition as a Center of Academic Excellence in Information Assurance/Cyber Defense (CAE IA/CD). The program identifies standards, criteria, and an evaluation process. Many individual institutions have achieved recognition. The University of Maine System, composed of seven universities, is the first multi-university entity to achieve the CAE IA/CD recognition. The purpose of this paper is to share the key challenges, opportunities, and experiences that contributed to this achievement, and offer recommendations.*

Keywords: Cybersecurity, Information Assurance, Education, Collaboration, NSA

1 Introduction

Concerns over the unmet workforce need for cybersecurity program graduates continue as they have for the past few decades. Much effort has been expended to raise student awareness and interest in cybersecurity education. For example, progress has been made in the establishment of cybersecurity competitions that have stimulated interest.

Similarly, much progress has been made to improve the quality and consistency of post-secondary education through efforts such as the National Security Agency (NSA)/Department of Homeland Security (DHS) jointly sponsored National Centers of Academic Excellence in Information Assurance/Cyber Defense (CAE IA/CD) programs. One can argue that the NSA/DHS CAE IA/CD recognition program is serving as a de facto accreditation standard for the fledgling cybersecurity discipline.

The public University of Maine System (UMS) is comprised of seven, highly geographically dispersed universities, not one of which could generate adequate evidence to garner NSA/DHS CAE IA/CD recognition. Many factors, most of which are present in other public university systems in the nation, contributed to establishing a context ripe for change that ultimately led to the UMS achieving NSA/DHS recognition as the first *multi-university shared/distributed* CAE IA/CD.

The aim of this paper is to share the key challenges, opportunities, and experiences that contributed to this achievement, and offer recommendations for harnessing the potential synergy resulting from direct engagement in mutually rewarding collaboration and cooperation to achieve significant gains in preparing the future cybersecurity workforce.

2 Constituents

2.1 NSA/DHS

The NSA/DHS, in response to an unmet workforce need for cybersecurity program graduates, jointly sponsor a program by which a post-secondary education institution may achieve recognition as a CAE IA/CD. The goal of the NSA/DHS CAE IA/CD education programs is to "reduce vulnerability in our national information infrastructure by promoting higher education and research in IA/CD and producing a growing number of professionals with IA/CD expertise in various disciplines. " [1]

Recognized education program categories include two-year programs (CAE2Y), four-year programs (CAE IA/CD), and research focused programs (CAE-R). Designation by the NSA/DHS is valid for five academic years, after which the institution must successfully reapply in order to retain its designation.

"Applicants must clearly demonstrate, in sufficient detail, how they meet each of the program criteria and include supporting documentation where required and appropriate. Minimum requirements must be met for each of the ten criteria." [2]

The review and selection procedures indicate, "Qualified IA professionals from the National Security Agency, the Department of Homeland Security, and other government and academic partners will assess applications. Other qualified individuals will be invited to assess applications as needed." [2]

"…Applications for CAE IA/CD will be rated independently, using system identified in the criteria. Two reviewers will assess each application. If the two reviews are not in agreement, a third reviewer will be assigned. The program office will make a final determination based on all three assessments. The burden is on the Institution to clearly demonstrate their qualifications for the CAE IA/CD program." [2]

"The National IA Education and Training Program (NIETP) operates under national authority as the national manager for IA education and training relating to national security systems. Its programs, including the National Centers of Academic Excellence (CAE), assure the very finest preparation of professionals entrusted with securing our critical information."[3]

NIETP provides specific guidance for preparing an application and an online application submission website. Applicants are required to map their institution's curriculum to an established standard consisting of a framework of knowledge units and student learning outcomes.

CAE IA/CD requirements and evaluation criteria specifically related to inter-institution cooperation and collaboration include:

- *Outreach/Collaboration.* The institution must demonstrate how IA/CD is extended beyond the normal boundaries of the Institution.
    - Shared Curriculum or shared faculty (NSA CAE IA/CD Criterion 1a)
    - CAE Collaboration (NSA CAE IA/CD Criterion 1d) Partner in research/shared classes or shared events with other institutions. Institutions are encouraged to partner with other CAEs on cyber or IA/CD research/instruction.

- *Number of IA/CD/Cybersecurity faculty and course load* (NSA CAE IA/CD Criterion 7c).

2.2 University of Maine System

The UMS is "the state's largest educational enterprise with nearly 40,000 students of all ages enrolled in our seven universities, law school, and eight outreach centers located across the state." [4] Figure 1 illustrates the significant geographic dispersion of the universities comprising the UMS.

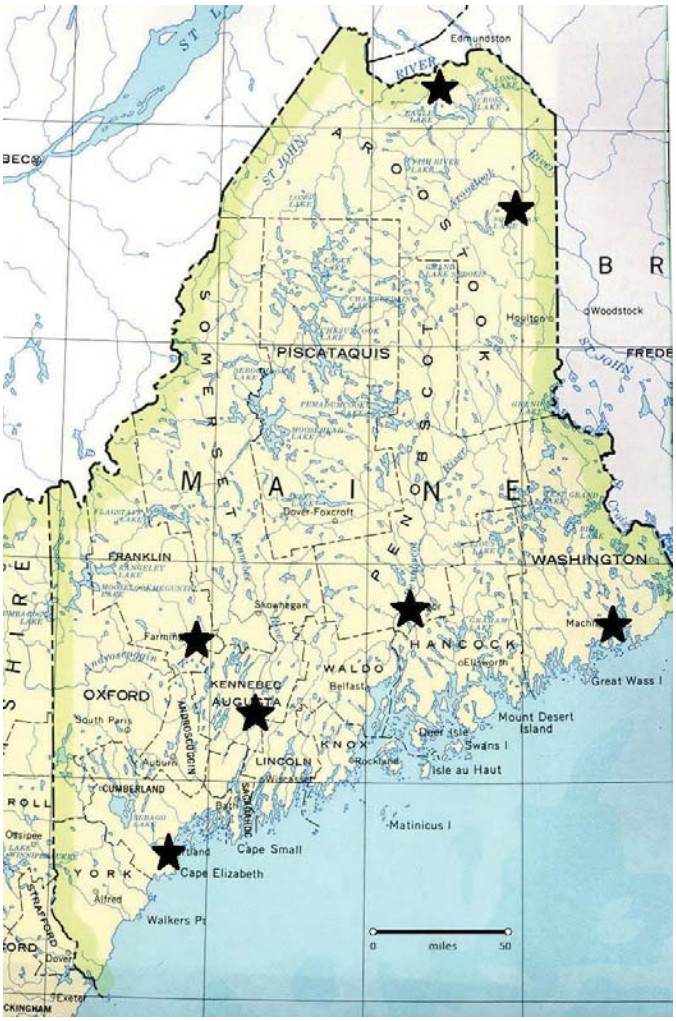


Figure 1. University of Maine System

A president who is accountable to the UMS Chancellor who, in turn, is accountable to a 16 member Board of Trustees (BoT) leads each university of the UMS. BoT members are appointed by the Governor and approved by the Maine Legislature for a five-year term.

In 2012-13, the Maine Cyber Security Cluster (MCSC) was funded by the Maine Technology Institute and the Maine Economic Improvement Fund to provide workforce and economic development initiatives across the state of Maine. Both the Executive Board and the Technical Advisory Board members were asked to fill roles on mission-critical subcommittees. These committees were comprised of business and industry, government, and military leaders in the cybersecurity, computer science, and information technology

sectors across the state of Maine, and selected faculty and staff of the UMS. An essential component of MCSC's mission is to create public-private partnerships and to build and support a collaborative environment among all interested and qualified parties.

Among the first MCSC subcommittees constituted was the Curriculum Subcommittee. This subcommittee was, agreed by all, to be extremely important to the educational and curricular mission of MCSC and to the Research and Development and Commercialization initiatives. Its efforts focus primarily on the design of a rigorous, appropriate, and topical student-oriented and knowledge unit based curriculum. The goals of Subcommittee were broadly stated: (1) attract strong students to the field of cyber security, (2) provide excellent training to students associated with MCSC, (3) develop programs that span the needs of the traditional university student and the wider community.

Curriculum development occurred in concert with the submission of the application for NSA recognition of the UMS as a CAE IA/CD. These two initiatives were pursued by a highly qualified and dedicated team of UMS faculty and staff.

2.3 ACM/IEEE

The Association of Computing Machinery (ACM) and Institute of Electrical and Electronic Engineers (IEEE) [5], though not yet playing a direct role in cybersecurity program accreditation, acknowledge the role of collaboration among faculty, educational institutions, business and government entities to build an "improved pipeline for a cybersecurity workforce" (p.4). Additionally, acknowledgement of barriers to such collaboration is exemplified through recognition that "academic departments are notoriously self-contained and reluctant to share resources, impeding collaboration and integration" (p.7). These are but two factors that influence inter-institution collaboration and cooperation.

3 Key Challenges and Opportunities

The challenges and opportunities that most significantly influenced this achievement arose from a geographic, governmental, and administrative context ripe for change. Collaboration and cooperation are key factors from which these challenges and opportunities arose. These factors have been widely cited in efforts to address cybersecurity from a multi-disciplinary approach [6], overcome barriers associated with academic silos [6], develop a strategy for fostering a shared understanding of concepts, guiding principles, and messages [7], and to improve the effectiveness of instructional methods [8].

Maine is often characterized as being a predominantly rural state with a widely geographically distributed public university system. The significant distances that exist between UMS universities presents challenges to inter-university communication and in turn inter-university collaboration and cooperation. Poor driving conditions, often due to winter weather, have also contributed to impeding efforts to engage in face-to-face meetings. Technology has provided a means to attenuate such communication impediments however, progress on inter-university efforts is often retarded by such physical separation

The governmental landscape of Maine, specifically relating to post-secondary education, has been significantly influenced by Governor-led efforts and tension resulting from increased fiscal pressure. Among the casualties of such tensions has been the resignation of the Maine Community College President, demanded by the Governor, because he "… has not acted on some of his requests…" [9] The Governor's proposed budgets during his tenure have called for flat funding or very modest acknowledgement of requested funding. This has led to public cries and legislative action for much greater investment in higher education as a means to spur the state's economy toward greater prosperity through appropriate preparation of a highly skilled workforce. As with most other states in the nation, such fiscal pressures at the governmental level have steadily increased during the period of recovery from the most recent economic recession.

Regarding the administrative context, there have been several events, which in totality, presented key opportunities and an environment receptive to change. Among these are the UMS Mission Excellence campaign that was established in January 2012 by the UMS Board of Trustees (BoT).

> *"Mission excellence* is our term for a comprehensive process to sharpen our focus on our mission and bring increased value to our constituents. It means focusing our scarce resources and ensuring that our systems, structures, processes, and employees are as efficient and effective as they can be. Mission excellence is about creating an environment where faculty and staff are engaged in serving our constituents to the best of our abilities, where decision-making is done as close to the clients as competencies allow. In short, it is about being "best in class."[10]

One of the key aims of the UMS BoT during its development of the *Mission Excellence* goals and actions was to engage in "cost control" efforts to help avert a significant structural budget gap identified by the UMS BoT relating to increased cooperation and collaboration include forming additional business partnerships and collaborations (Directive II.a) and aligning academic and certificate program development with workforce needs (Directive II.b) [11].

The UMS BoT approved the UMS Strategic Outcomes statement in July 2014. The UMS was identified as being in "… a period demanding transformative change …" and would achieve these outcomes" …through an intensely collaborative approach …" (UMS 2014 Strategic Outcomes, p 1). Collaboration was specific addressed through Strategic

Integration Target 2 that calls for the UMS to "Develop and implement a comprehensive financial management structure for the entire System that enhances transparency, enables appropriate fiscal control, and advances comprehensive intra-system collaboration."(p.1)

Finally, intra-institution cooperation is being fostered through Strategic Integration Target 4 that calls for the UMS to "Develop a model of academic program and portfolio review and integration that leverages academic resources to enhance program quality, expand access, and meets appropriate financial benchmarks, with at least three pilot projects underway in FY15"(p.2)

The UMS established the Academic Portfolio Review and Integration Process (APRIP) process during fall 2014 to bring about academic program integration. The process utilizes two approaches (program integration, portfolio review) to address Strategic Integration Target 4 [12].

Nine discipline-based teams led by faculty members and involving over 100 participants system-wide to develop recommendations for new multi-institutional collaborations that can increase quality, access, and fiscal sustainability are addressing academic program integration.

Academic portfolio review is being addressed by a multi-year process to make program inventory decisions systemically rather than in the silos of individual universities. Members of the Presidents Council and the Chief Academic Officers are to identify academic programs that are less able to meet two or more of three criteria: (1) centrality to institutional mission as differentiated through Strategic Integration Target 1, (2) fiscal sustainability, and (3) meeting the needs of the state.

The calls for increased collaboration and cooperation, especially in light of increasing financial pressure and the concomitant need to accomplish more with less, combined to provide the catalyst for academic program reformation within the UMS.

Key cybersecurity faculty, instructional resources and related academic programs have slowly evolved at four of the UMS universities within the UMS. For example, the University of Maine at Fort Kent was the first post-secondary education institution in Maine to offer an Associate of Science in Information Security and subsequent academic concentration and certificate option. The University of Maine has regularly prepared and fielded a student team at the annual Northeast Collegiate Cyber Defense Competition. The University of Maine at Augusta has excelled in providing cybersecurity instruction, an academic concentration and certificate to the adult non-traditional population. The University of Southern Maine has very successfully garnered grants and industry support and involvement in its establishment of the Maine Cyber Security Cluster (MSCS) [13] that aims to combat and mitigate risk by 1) training cyber security personnel, 2) conducting assessments in lab and at sites, and 3) providing space, resources, and expertise for cybersecurity research and development. This center also houses as a virtual cybersecurity collaborative learning environment that promotes inter-institutional, innovative, hands-on collaborative learning experiences aimed at preventing and mitigating cyber-attacks in real time [14].

These cybersecurity-related faculties, curricula, instructional laboratories resources, student engagements, research contributions, and community engagement activities have slowly evolved. With respect to the NSA CAE IA/CD program application process, no single UMS university could generate sufficient evidence to achieve recognition. Key faculty members at each of the four UMS universities, upon realizing the synergy that existed, initiated an effort to combine their respective accomplishments and jointly apply for NSA/DHS recognition of the UMS as a shared/distributed CAE IA/CD.

The effort and successful application process received significant support from academic, governmental and industry leadership and in October 2014, the NSA/DHS designated the University of Maine System as a CAE IA/CD through academic year 2019.

The Chancellor of the UMS cited the success of this faculty initiative in his biennial State of Higher Education in Maine presentation to the Joint Session of the 127th Maine Legislature:

> "The University of Maine System benefits from hundreds of able faculty dedicated to their profession and their students, but whose efforts are also constrained by university and bureaucratic silos. Teams of faculty and staff from across Maine are breaking down those barriers and re-imagining their programs, drawing resources from every corner of every university in order to develop student-focused scholarship, with priorities given to academic programs that meet demonstrated community needs.
>
> Here is an early and resoundingly successful example of where we are going. This past fall the University of Maine System was designated a National Center of Excellence in Cybersecurity Education by the NSA/Dept. of Homeland Security. No individual Maine institution had the resources or expertise to achieve this designation, but by working collaboratively, faculty from UMFK, UMA, UM, and USM created a unified program which earned Maine the nation's first ever multi-university, System designation... It is my honor to recognize [the] Professors whose vision, creativity, and expertise has secured Maine this opportunity, and whose work is an outstanding example of what we can accomplish when we work together."

Getting a Bachelor of Science in Cybersecurity program stood up in the UMS can be considered an example of the "which came first, the chicken or the egg?" There is an urgency to getting the degree program going, but the University of Maine System has no mechanism for offering a system-wide degree. Degrees are offered by the individual universities. The universities have different procedures for approving degree programs. Consequently, the program approval process has proceeded along somewhat different approval lines at each university.

In parallel with the process of establishing the degree program is an effort by the administrations at each of the universities to adopt a Memorandum of Understanding (MOU) defining how the universities will handle some of the inter-university issues. The work on the MOU is being handled by the administrations of the participating universities and is outside the scope of this paper.

Cybersecurity Requirements (36 hours)

CYB 100 Introduction to Computer Science	4 hours
CYB 200 Introduction to Information Security	3 hours
CYB 250 Introduction to Programming	3 hours
CYB 300 Computer Programming	3 hours
CYB 330 Networking	3 hours
CYB 340 Cyber Ethics	3 hours
CYB 350 Databases	4 hours
CYB 360 Network Security	4 hours
CYB 370 Operating Systems Security	3 hours
CYB 390 Cybersecurity Internship (or 2-course alternate)	3 hours
CYB 400 Cyber Defense (Capstone)	3 hours
Cybersecurity Portfolio (see *Techniques for Assessment*)	
	36 hours

Figure 2. Proposed Curriculum

An Academic Governance Board consisting of key faculty and staff from participating universities has driven the process. This group adopted the model curriculum shown in Figure 2 that derives from the curriculum specified in the NSA CAE IA/CD application.

Students are required to complete a Cybersecurity Portfolio in addition to those courses defined within the program. Students construct a portfolio from the projects completed as part of the major course requirements portion of the program. The portfolio is intended to enable assessment of those learning outcomes that are best assessed in an integrative fashion, spanning all of the student's course work and therefore reflects overall academic growth.

4 Recommendations

Given increased pressure upon public post-secondary education institutions by local, regional and national entities to accomplish more with less and continuously improve service to their respective constituencies, the case for increased intra-university collaboration and cooperation is clear. The time to embrace change, consider new approaches, realize synergy through collegial engagement is, or likely soon will be, upon all such institutions.

4.1 Maintain a willingness to listen and a desire to advance the cause through change. Do not become ensnared in callous recalcitrance. Seek out those who are similarly open to change. Explore initiatives that achieve mutually rewarding benefits/outcomes.

4.2 Identify the goals and explore new and/or untried approaches to achieve them. Avoid working in a vacuum or falling back to being a singular source of ideas. Reach out to those with whom you have not previously engaged (students and staff included) to share perspectives and engage in brainstorming approaches. Recall the words of John Donne, "No man is an island, entire of itself." [15]

4.3 Capture the potential of synergies that exist only when two or more collaborate, cooperate, or share and view the "whole is *other* than the sum of the parts" as suggested by Gestalt theory. Such synergistic potential can be substantial, as exemplified in the NSA/DHS recognition of the UMS as a shared/distributed CAE IA/CD.

The establishment of a shared/distributed CAE IA/CD is expected to avail new opportunities to the UMS. One example is the potential to establish new shared/distributed academic degree programs. The establishment of such programs is similarly fostered and supported by the same contextual challenges and opportunities presented.

5 Conclusion

The UMS is the first multi-university entity to achieve NSA/DHS recognition as a *shared/distributed* CAE IA/CD. The challenges and opportunities that most significantly influenced this achievement arose from a geographic, governmental, and administrative context ripe for change.

This context strongly favored vastly increased collaboration and cooperation among the universities within the UMS that formerly evolved their functioning as a loosely coupled federation. As exemplified by the experiences presented, it is possible to better harness the potential synergy resulting from direct engagement in mutually rewarding collaboration and cooperation to achieve significant gains in preparing the future cybersecurity workforce.

6 References

[1] National IA Education & Training Programs (2015). "CAE Requirements and Resources". Retrieved from https://www.iad.gov/NIETP/CAERequirements.cfm

[2] National IA Education & Training Programs (2015). "About CAE Program". Retrieved from https://www.iad.gov/NIETP/aboutCAE.cfm

[3] National IA Education & Training Programs (2015). "Welcome". Retrieved from https://www.iad.gov/iad/index.cfm

[4] University of Maine System (2015). "Chancellor's Office". Retrieved from http://www.maine.edu/about-the-system/chancellors-office/

[5] A. McGettrick (2013). "Toward Curricular Guidelines for Cybersecurity: Report of a Workshop on Cybersecurity Education and Training". Association of Computing Machinery (ACM). Retrieved from https://www.acm.org/education/TowardCurricularGuidelinesCybersec.pdf

[6] L. Hoffman, D. Burley & C. Toregas (2012). "Thinking Across Stovepipes: Using a Holistic Development Strategy to Build the Cybersecurity Workforce". IEEE Security & Privacy Magazine. DOI: 10.1109/MSP.2011.181.

[7] E. McDaniel (2013). "Securing the Information and Communications Technology Global Supply Chain from Exploitation: Developing a Strategy for Education, Training, and Awareness". Issues in Information Science and Information Technology. Retrieved from http://iisit.org/Vol10/IISITv10p313-324McDaniel0083.pdf

[8] D. Rowe, B. Lunt & J. Ekstrom (2011). "The Role of Cyber-Security in Information Technology Education", Proceedings of the Special Interest Group on Information Technology Education (SIGITE), October 2011, pp. 113-121.

[9] C. Cousins (2015). "LePage's willingness to broaden sales tax base surprises some lawmakers". *Bangor Daily News.* Retrieved from http://bangordailynews.com/2015/01/09/politics/state-house/lepages-willingness-to-drastically-broaden-sales-tax-base-surprises-some-lawmakers/

[10] University of Maine System (2015). "About Mission Excellence at Maine's Public Universities". Retrieved from http://thinkmissionexcellence.maine.edu/about-thinkme/

[11] University of Maine System (2012). "Board of Trustees Goals and Actions". Retrieved from http://thinkmissionexcellence.maine.edu/messages-from-the-chancellor-2/goals-and-actions/

[12] University of Maine System (2013). "Academic Portfolio Review & Integration Process". Retrieved from http://thinkmissionexcellence.maine.edu/priority-initiatives/academic-review/

[13] Maine Cyber Security Cluster (2014). "About Us". Retrieved from http://www.mcsc.usm.maine.edu/aboutus.php

[14] J. Murphy, E. Sihler, M. Ebben, L. Lovewell, & G. Wilson, (2014). "Building a Virtual Cybersecurity Collaborative Learning Laboratory (VCCLL)", *Conference Proceedings, SAM '14 Conference.* Las Vegas, NV 2014 World Congress in Computer Science, 2014. Retrieved from http://worldcomp-proceedings.com/proc/p2014/SAM4153.pdf

[15] Donne, John (1623). "Devotions upon Emergent Occasions". Retrieved from Project Gutenberg's Devotions Upon Emergent Occasions, by John Donne (p. 108). Retrieved from http://www.gutenberg.org/files/23772/23772-h/23772-h.htm

Covert Channel over Apple iBeacon

Joseph Priest
Rochester Institute of Technology
Rochester NY, USA
joseph@josephjpriest.com

Daryl Johnson
Rochester Institute of Technology
Rochester NY, USA
daryl.johnson@rit.edu

ABSTRACT

Opportunities for covert channels exist using Apple iBeacon technology. Apple iBeacons are an emerging technology designed to provide additional proximity based information to iOS devices. iBeacons are implemented using Bluetooth Low Energy advertisements. As such, a manipulated iBeacon advertisement can be issued with Bluetooth 4 compatible hardware. There are fields within this iBeacon advertisement that can be modified without adversely affecting the transmission of the iBeacon, and as such, provides an opportunity for covert messages. Despite reliability concerns, the technology is continuing to be adopted, and as such, the opportunity for using covert channels over iBeacon is growing as well.

Keywords

Covert channels, iBeacons, Bluetooth Low Energy, Proximity dependent

1. INTRODUCTION

For most concepts in computing, as technology evolves to take advantage of hardware and algorithmic advances, the concept also is adapted to keep pace; however, the core principles of the concept remain. Consider, for example, the concept of outputting information through a computer monitor. Perhaps, originally the monitor displayed information via a monochrome screen, but as technology has advanced, the concept has adapted to utilize color, higher resolutions, or even 3-dimensional illusions; yet through these adaptations, the core principle of displaying information to the user remained.

This idea can apply to the concept of covert channels as well. Butler Lampson first coined the term,"covert channels" in 1973, defining them as "those not intended for information transfer at all..." [1]. Despite the fact that this was written more than 15 years before the invention of the World Wide Web, the original idea has been preserved through the various adaptations and implementations of covert channels for newer technologies. As covert channels have matured and been researched, different characteristics can be defined, regardless of whether the channel was created thirty years ago or yesterday. Some of the important characteristics to understand when discussing a covert channel include mechanism for hiding data, type, throughput, robustness, detection, and prevention [2].

It can be argued that covert channels can actually be created over almost any protocol, examples include TCP/IP, ICMP, ptunnel, DNS, and so on [3]. With some exploration and creativity, opportunities for covert channels can also be unearthed in the many new emerging technologies of mobile and "cloud computing". The aim of this paper is to delve into and describe an opportunity for a covert channel using Apple iBeacons, which is a Bluetooth Low Energy based, proximity dependent technology.

2. UTILITY OF PROXIMITY DEPENDENT COVERT CHANNELS

Most covert channels have been created over networked technologies. Whether the goal is remote command-and-control, exfiltration of data, or other operations, it is likely to be beneficial to the attacker to be able to hide that communication within normal network traffic. However, there are comprehensible circumstances where the network is not a viable medium for covert communication[1]:

- A non-networked device
- A device on a local network with no gateway
- A device on a firewalled network, such that all outbound traffic is denied save for some type that an attacker cannot utilize
- A device on a network under scrutiny for anomalies, such that risk of detection of the attacker's channel is significantly increased

[1]While irrelevant to the focus of this paper, it is interesting to consider how a non-networked or similar device is compromised. It is assumed that in these cases, a malicious insider can gain physical access to the device, or a legitimate user can be phished into compromising the system.

In such instances, it may be useful to utilize a covert channel which relies not on a network medium but on some other method of transmission, most likely between devices located within a close physical proximity between each other. Such technologies may include Bluetooth, near field communication (NFC), wireless networks, sounds, lights (visible or infrared), or similar physical channels. The experiments presented in this paper utilize Apple iBeacons, which rely on Bluetooth technology.

The receiver in a proximity dependent covert channel must also be within the attacker's control. This device can either be the destination for the covert message, or forward it along via legitimate or other covert channels, given it is networked. The following figure shows the simplicity of the concept of a proximity dependent covert channel:

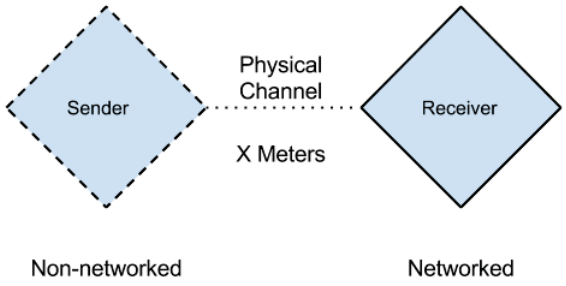


Figure 1: Model - proximity dependent covert channel

To give a real world scenario, let there be a supermarket *XYZ*. *XYZ* processes credit cards during checkout, and to remain PCI-compliant, also segments and firewalls its network such that the outbound traffic from Point-of-Sale (POS) machines can only travel on the secure PCI network. *XYZ* also provides open guest wireless Internet for its customers. *XYZ* has also begun an initiative with its mobile app and in-store iBeacons to help customers find products faster. During off hours, a rogue insider installs credit card sniffing malware onto a POS machine. This rogue insider also slips a nondescript Bluetooth USB dongle into the machine. The following day, as numerous credit card transactions occur, the POS sends iBeacons, disguised as ordinary store iBeacons. Really, these iBeacons covertly transmit credit card numbers. An associate of the rogue insider can then sit in the cafe with his laptop, receive the covert iBeacons and save them, or immediately repackage them and send them out the unfiltered guest wireless, completely undetected.

3. TECHNICAL SPECIFICATIONS OF IBEACON PROTOCOL

The purpose of iBeacons, as described by Apple's iBeacon documentation[4]:

"Introduced in iOS 7, iBeacon is an exciting technology enabling new location awareness possibilities for apps. Leveraging Bluetooth Low Energy (BLE), a device with iBeacon technology can be used to establish a region around an object. This allows an iOS device to determine when it has entered or left the region, along with an estimation of proximity to a beacon"

Apple frameworks automatically handle the low-level specifics of Bluetooth Low Energy and the iBeacon *Protocol*, allowing the developers to incorporate iBeacon functionality simply by configuring a UUID, an optional Major Field, and an optional Minor Field.

Apple's guidance for using these iBeacon fields[4]:

UUID *(16 bytes):* Application developers should define a UUID specific to their app and deployment use case.

Major *(2 bytes):* Further specifies a specific iBeacon and use case. For example, this could define a sub-region within a larger region defined by the UUID.

Minor *(2 bytes):* Allows further subdivision of region or use case, specified by the application developer.

Beyond these 3 fields, developers are given no other means of making modifications to the iBeacon advertisement packet. However, a core feature of iBeacon is the ability to determine an approximation of distance between iBeacon transmitter and the iOS receiver. This is accomplished through the comparison of RSSI (Received Signal Strength Indicator) and *measured transmit power*. The beacon transmits the advertisement packet with a measured transmit power value attached[5]. The receiver then uses both the included measured transmit power and the RSSI of the collected advertisement to determine a distance. Note that this measured transmit power is not set by the developer, but by the device sending the beacon.

However, just like other protocols (e.g. HTTP), there are other pieces to an iBeacon packet that are not explicitly set by the developer/application. There is, unfortunately, no iBeacon public specification. By using packet captures and the Bluetooth Core Spec[6], enough data can be gathered regarding the additional parts of the packet, at least for the purposes of manipulating them into forming a covert channel.[2]

The packet capture partitions an iBeacon advertisement into the following (slightly modified for readability):

Parameter Length: 42 (0x2A)
LE Advertising Report
NumReports: 0X01
EventType: Nonconnectable unidirectional advertising
AddressType: Random Device Address
PeerAddress: 0D:EF:97:32:B8:A5
LengthData: 0X1E
Flags: 0x06
Manufacturer Specific Data: –
Data: 02 01 06 1A FF 4C 00 02 15 F1 EB BC 09 A3 13 7F CD 81 DF 67 C7 79 76 38 88 18 8C 02 43 C0
RSSI: -80 dBm

[2]**Open source tool used for sending iBeacons using iOS APIs:** https://github.com/Intermark/Buoy

For the scope of this paper, the *Data* field will be what is analyzed. Upon sending the above iBeacon, the following is known (and configured in code):

UUID: F1 EB BC 09 A3 13 7F CD 81 DF 67 C7 79 76 38 88
Major: 18 8C
Minor: 02 43

Applying this knowledge to the *Data* field, it becomes:

Data: 02 01 06 1A FF 4C 00 02 15 [UUID] [Major] [Minor] C0

From this *Data* field, it is also known that **4C 00** is the Bluetooth company identifier for Apple[7].

As will be discussed later in the paper, the remaining octets prior to the UUID appear to be part of the iBeacon Prefix, BLE Flags, or BLE advertisement packet requirements; in summary, these are what identifies an iBeacon as an iBeacon (manipulations to these octets causes the transmission to no longer be detected by an iBeacon listener).

It was also determined that the last octet in the data field is the transmitted power, as set by the sender. Note that this is not the *RSSI*, which is the actual received signal strength.

With this information, the summary of the iBeacon *Data* field is (**size in bytes within parenthesis**):

Data:
Prefix/Flags (**5**)
Company ID (**2**)
Prefix/Flags (**2**)
UUID (**16**)
Major (**2**)
Minor (**2**)
Transmitted Power (**1**)

To summarize for the purposes of describing this covert channel, it is important to understand an *iBeacon* is:

- designed to provide additional location-based information to an iOS device
- a unidirectional Bluetooth Low Energy advertisement (i.e broadcast)[3]
- without a true data payload–the way it conveys information is through identifiers and signal strength (however, it is important to note the various components of the *Data* field)

[3]For in-depth information about Bluetooth Low Energy, refer to the core specification document[6]

4. REQUIREMENTS AND CHARACTERISTICS OF IBEACON COVERT CHANNELS

An iBeacon covert channel can be simplified by the following 2 requirements:

- It must be capable of transmitting a covert message using iBeacon packet(s) as a vehicle
- Upon reception by a legitimate iBeacon receiver (i.e. an Apple iOS device), the packet must be decipherable and interpreted as an iBeacon.

These requirements, however, sit on top of the traditional requirements and characteristics of covert channels:[4]

- **Mechanism for hiding data**: The hidden message must be somehow encoded within the iBeacon advertisement packet. The idea is that not only is there a secret message, but to outside observers, it appears that no data-exchange is even taking place. Because iBeacons are not designed to transmit data payloads, they inherently do not appear to be transmitting extra data.
- **Type**: iBeacon advertisements could potentially be used for either *storage based* or *timing based* types of covert channels. However, the lack of reliability of delivery of advertisements makes timing channel opportunities much more challenging.
- **Throughput**: Throughput of an iBeacon covert channel is dependent on the amount of data sent per advertisement, the interval at which advertisements are broadcasted (which, based upon the BLE spec requirements, is between 20ms and 10.24s[8]), and any necessary retransmissions, due to the volatile nature of BLE.
- **Robustness**: A proximity based covert channel requires a certain distance be maintained between sender and receiver. For BLE-based iBeacons, other physical factors could also influence the successful delivery of packets, such as walls, humans, or other objects impeding the path of the signal.
- **Detection**: Because iBeacons are broadcast on physical radio waves, packets are not traversing networks or the detection tools that would be placed on them, such as intrusion dectection systems, firewalls, or SIEMS. As such, the most obvious method of detection would be Bluetooth sniffers that analyze BLE packets. Given the existence of such a device, it then must detect that a given iBeacon is anomalous.
- **Prevention**: Prevention is difficult without sacrificing the availability of Bluetooth or emerging technologies they provide, such as iBeacons.

The iBeacon covert channel discussion in the next section builds upon these requirements and characteristics.

[4]This list of characteristics is based upon the research completed by Johnson et al.[2]

5. IBEACON COVERT CHANNEL

An analysis of each of the iBeacon *Data* fields can show where opportunities for covert channels exist.

5.1 Prefix/Flags

Modifications to the elements of either of the *Prefix/Flags* fields results in the Bluetooth advertisement no longer being recognized as an iBeacon by an iOS device, which breaks one of the core requirements for this covert channel. As such, none of these bytes provide opportunity for an iBeacon covert channel.[5]

5.2 Company ID

Apple's company identifier for Bluetooth transmissions is **0x004C**, as registered with the Bluetooth SIG[7]. Note that as this is transmitted as a BLE advertising packet, it follows the *Little Endian* format, with the least significant bit first.[6] As shown in the example in the previous section, this company identifier is physically sent as the bytes **4C 00**. It was discovered that modifying the **00** byte to some other value than *00* resulted in the advertisement continuing to be recognized as an iBeacon. Modifying the **4C** byte causes the iBeacon receiver to ignore the packet, which indicates that some sort of validation of the company identifier field does occur. However, it appears to only check the first byte. Perhaps this is an oversight on Apple's implementation, or perhaps it is for optimization. Regardless this 2nd company identifier byte can be modified and included in an iBeacon transmission without adversely affecting the successful delivery of the iBeacon. The location of this byte, in relation to the earlier packet example (where *XX* indicates the modifiable company identifier byte):

Data: 02 01 06 1A FF 4C **XX** 02 15 F1 EB BC 09 A3 13 7F CD 81 DF 67 C7 79 76 38 88 18 8C 02 43 C0

5.3 UUID, Major, Minor

Modifications to the UUID, Major, or Minor bits do not affect the validity of an iBeacon transmission. In fact, these fields are promoted to developers as the way iBeacons work. Using these three fields as a channel would yield 20 bytes of data per transmission, 16 for the UUID, 2 for the Major, and 2 for the Minor. At most however, this would be considered an obscured channel, simply because of the unlikelihood of the channel being monitored. Given a suitable device, such as a BLE packet sniffer, detection of this anomalous traffic is a certainty, as the UUID, Major, and Minor bytes would not match any of the expected *production* iBeacons. As such, modifying any of these fields is out of scope for a true covert channel.

5.4 Transmitted Power

Typically, the transmitted power represents a measurement by the Bluetooth device manufacturer at one meter away[5]. This calibrated transmitted power is then sent with the transmission of the advertisement. The receiver can then compare the calibrated transmitted power with the RSSI (Received Signal Strength Indicator). The general concept for determining distance is if two advertisements with identical calibrated transmitted power values are received, the packet with a stronger RSSI has a closer sender.

Apple API provides developers 4 descriptors when calculating range from an iBeacon: *immediate, far, near, unknown*[4]. These descriptors are not very precise, and for valid reasons as RSSI has been shown to not be a very reliable method for determining the distance between objects[10].

The transmitted power byte can be modified to any valid hex character without breaking the iBeacon protocol. If the receiver cannot process the difference between calibrated transferred power, it will return the unknown descriptor; otherwise it will compute the range as immediate, far or near. All of these result in a valid iBeacon. As such, this byte provides a clear opportunity of another option for a covert channel. The location of this byte, in relation to the earlier packet example (where *YY* indicates the modifiable transmitted power byte):

Data: 02 01 06 1A FF 4C XX 02 15 F1 EB BC 09 A3 13 7F CD 81 DF 67 C7 79 76 38 88 18 8C 02 43 **YY**

5.5 Theoretical Covert Channel Throughput

In a theoretical context, the following assumptions are made:

- Every iBeacon sent is always received
- iBeacons are able to be sent every 20ms (the shortest technically possible interval for BLE advertisements)

This leads to a throughput of 2 bytes (one in the company identifier and one in the transmitted power) every 20ms.

In more common terms, the maximum possible **throughput** is: **100 bytes per second**.

5.6 Practical Covert Channel Throughput

The actual throughput of an iBeacon covert channel is difficult to estimate. Because of the volatility of BLE advertisements and objects in the physical environment, it is not likely that all iBeacons transmitted by the sender will be received. Further, because of expected iBeacon loss, the covert channel implementation will probably have to sacrifice a byte as some kind of *sequence number*, similar to the sequence number used in TCP/IP. This drops the throughput down to 1 byte per transmission. In the proof of concept, it will be shown that the actual throughput is more on the scale of 1 byte per 3 seconds.

6. PROOF OF CONCEPT

6.1 Tools

Sender

Kali Linux machine equipped with the BlueZ stack.[6]
Satechi USB 4.0 Bluetooth Adapter.

[5]This does not conclusively determine that *Prefix/Flags* fields do not contain opportunity for BLE covert channels, simply not for an iBeacon specific channel. For more information on the BLE advertising packet, see Bluetooth Core Spec, Volume 3, Part C, Section 11 and Section 18.[9]

[6]Official Linux Bluetooth protocol stack. www.bluez.org

Receiver
Macbook Pro (late 2013) equipped with internal Broadcom Bluetooth device and Bluetooth sniffing software.

Normal iBeacon Advertisement Listener
iPad Air equipped with the Locate Beacon app[7]

6.2 Methods

Both the company identification byte and the power byte are utilized in the following proof of concept. However, to compensate for the expected volatility of Bluetooth Low Energy advertisements, the power byte is used only as a sequence number, not as a data payload.

Sending iBeacons

The following command will instruct the Bluetooth interface to configure the advertising payload (length of payload specified as *1e*)[8]:

hcitool -i hci0 cmd 0x08 0x0008 1e ***$payload***

where an example **$payload** is (XX representing covert channel data, YY representing a sequence number):

02 01 06 1A FF 4C XX 02 15 F1 EB BC 09 A3 13 7F CD 81 DF 67 C7 79 76 38 88 18 8C 02 43 YY

To instruct the Bluetooth interface to begin advertising the configured packet:

hciconfig hci0 leadv

Changing the Advertising Interval

The default advertising interval is 1.28 seconds[9]. The following commands were issued to change the interval to 100ms (derived using Bluetooth Core Spec, Volume 2, section 7.8.5) and then begin advertising at that rate:

hcitool -i hci0 cmd 0x08 0x0006 A0 00 A0 00 03 00 00 00 00 00 00 00 00 07 00
hcitool -i hci0 cmd 0x08 0x000a 01

Receiving iBeacon Advertisements

For a real-world deployment of this covert channel, the tool *hcidump* will probably be the most useful for scripts and parsing iBeacons. An example command for sniffing Bluetooth packets and outputing hex:

hcidump -i hci0 -x

For the sake of this proof of concept and human-readability, a GUI OSX Bluetooth packet sniffer was used.

Distinguishing Covert Channel iBeacons

An important attribute of a covert channel is the method the receiver uses to tell the difference between legitimate traffic and covert traffic. In this iBeacon covert channel, it is quite easy to distinguish the covert packets; if the company identifier is set to any other value than *4C 00*, it is known to be an iBeacon covert channel. It is particularly important for those using this channel to recognize that *00* is not a valid value for the covert channel.

Other Implementation Considerations

There are other details that may be desirable to consider while creating this covert channel:

- Length of message, including how to handle sequence number (e.g. should *FF* roll over to *00*?)
- Duration of time to advertise each payload
- How to specify the end of the message
- Retransmitting missed advertisements

However, many of these are dependent on use-case, payload type, and other details that are beyond the scope of this proof of concept.

7. RESULTS AND DISCUSSIONS

7.1 Sending a Message

Covert message transmission was successful for the message *123456*. A 3 second duration of advertising was chosen for each byte of the message, which proved necessary, as in one of the experiments, only one advertisement was received containing the *12* byte of the message. For the sake of simplicity, *00* was chosen as the first sequence number and then incremented for each following value advertised (sequence number - value, *00 - 12, 01 - 34, 02 - 56*). It should be noted that this choice of sequence number results in iBeacons with an *unknown* distance.

7.2 Advertising Speed

With the default advertising interval, an experiment was conducted simply measuring how many packets were received by the receiver. In 1475 seconds, only 105 advertisements were received, which corresponds to approximately 1 received advertisement every 14 seconds.

After the advertising interval was modified to send every 100ms, the experiment was run again. This time, after 401 seconds, 629 advertisements had already been received. This corresponds to 1 advertisement received every .64 seconds.

For these experiments, the sender and receiver were placed approximately one meter apart. Different results would be expected with different hardware or environments. However,

[7]Created by Radius Networks
[8]The commands sent to the interface: 0x08 - OGF Code for LE Controller Commands; and 0x0008 - LE Set Advertising Data, see Bluetooth Core Spec, Volume 2, section 7.8 and 7.8.7[9]

the concept of faster advertising intervals correlating with more quickly received packets should still apply.

7.3 Importance of sequence numbers

Because there is a duration of time where the same value is advertised over and over, it is crucial to specify when an advertisement contains a new value. This is accomplished through the use of a sequence number in the transmitted power field. If the sequence number has changed from the previous packet, the data is new, otherwise, it should not be appended to the secret message. Using a sequence number also assists with keeping the packets ordered, or determining when an advertisement is completely unreceived (e.g. when the sequence numbers received go from 04 to 06, skipping 05).

7.4 Reliability concerns

As has been apparent throughout much of this paper, advertisement loss is very prevalent with iBeacons, or at least the implementation used for this proof of concept. It is unknown whether the problem resided with the sender not sending as fast as it was configured to (due to limited resources/hardware), the receiver dropping packets (with similar resource or driver limitations), elements in the environment, or a combination of all of the above. However, the message tests and the speed tests make it very apparent that at some point in the system, advertisements were lost. While this issue is mitigated to a certain extent by the sequence numbers solution, this cuts into potential throughput of the channel.

8. CHALLENGES AND CONCLUDING REMARKS

8.1 Challenges of Covert Channels in Emerging Technologies

Bluetooth Low Energy is a relatively new field, and with technology advancements in hardware, is susceptible to volatility. As an example, mere software updates to the operating system of one of the experiment machines caused the Bluetooth interface to work differently. Similarly, iBeacon, as a proprietary advertisement format on top of BLE, could change the way it operates. These technologies might not even exist in years to come. All of these caveats factor into the development and research into covert channels over these mediums. Furthermore, this covert channel, as proposed, relies upon the fact that the company identifier field is not verified in full. In the event this is rectified, covert channels would have to be created using other techniques (perhaps timing channels monitoring the transmitted power field, or similar).

8.2 Concluding Remarks

After delving into the packet structure and configuration of Bluetooth Low Energy advertisements, it is clear that opportunities are present for covert communication over iBeacon advertisements. It is also clear that there are many challenges surrounding using this technology, especially with overcoming reliability issues; however, these are the same challenges that have to be solved for the normal usage of this technology. As the adoption of iBeacon technology grows, the opportunities for this covert channel will expand as well. As further research is conducted in emerging technologies utilizing proximity dependent channels, such as the discussed iBeacon, it will be important to consider the possibility of covert channels; traditional network security equipment might not cover this new communication space.

9. REFERENCES

[1] B. W. Lampson, "A note on the confinement problem," *Commun. ACM*, vol. 16, pp. 613–615, Oct. 1973.

[2] D. Johnson, B. Yuan, P. Lutz, and E. Brown, ""covert channels in the http network protocol: Channel characterization and detecting man-in-the-middle attacks"," `http://scholarworks.rit.edu/other/781/`.

[3] E. Couture, "Covert channels," *SANS Institute*.

[4] Apple, Inc., *Getting Started with iBeacon*, 1.0 ed., June 2014. `https://developer.apple.com/ibeacon/Getting-Started-with-iBeacon.pdf`.

[5] T. Andersson, "Bluetooth low energy and smartphones for proximity-based automatic door locks," 2014.

[6] Bluetooth SIG, *Bluetooth Core Specification 4.1*. `https://www.bluetooth.org/DocMan/handlers/DownloadDoc.ashx?doc_id=282159`.

[7] Bluetooth SIG, *Company Identifiers*. `https://www.bluetooth.org/en-us/specification/assigned-numbers/company-identifiers`.

[8] J. Liu and C. Cheng, "Energy analysis of neighbor discovery in bluetooth low energy networks," *Technical Report, Nokia Research Center/Radio Systems Lab*.

[9] Bluetooth SIG, *Bluetooth Core Specification 4.0*. `https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=229737`.

[10] A. T. Parameswaran, M. I. Husain, and S. Upadhyaya, "Is rssi a reliable parameter in sensor localization algorithms - an experimental study," *Field Failure Data Analysis Workshop*, 2009.

The Security Implications of IMSI Catchers

Bryan Harmat, Jared Stroud, Daryl Johnson, Bill Stackpole,
Sylvia Perez-Hardy, Rick Mislan, PhD, Tae Oh, PhD
Department of Computing Security
Rochester Institute of Technology
Rochester, NY 14623

Abstract

According to various news sources, rogue cellular towers referred to as "IMSI catchers" have been deployed across the nation. An academic interest has been taken in determining what information can be acquired when a mobile device associates with one of these towers. These towers focus on manipulating authentication methods to pose as a legitimate GSM tower. Through the use of software defined radios, and open source software an inexpensive GSM protocol-based cell tower was deployed to determine what, if any, security vulnerabilities exist in the current mobile network infrastructure.

1 Introduction

IMSI (International Mobile Subscriber Identity) catchers are devices constructed to execute a man in the middle attack of mobile phone network traffic. These towers can be used to intercept voice calls, texts, and data (such as web browsing)[8]. Federrath notes, "IMSI Catcher(s) [are] capable of signaling to the mobile phone that it should discontinue using encryption on the radio link." [7]. Due to this security flaw in the Global System for Mobile communication (GSM), phones that associate with an IMSI catcher may not be using encryption to secure data in transit if the tower did not tell the device to use an encryption method. This vulnerability results in the attacker possessing the capability to see all data to and from the device. This information allows for an attacker to associate an individual based on their unique individual mobile subscriber identity (IMSI) stored on the mobile device's SIM (Subscriber Identity Module) with a mobile device at a specific location (if telecommunication providers were subpoenaed for such information). A SIM is used to uniquely identify a user for subscription purposes and includes the user's IMSI [9].

2 GSM

GSM is a standard developed to describe cellular network protocols that a large market share of cell phones. The original specifications were developed by the European Telecommunication Standards Institute (ETSI)[23]. As of February 2015, there are approximately 3.6 billion mobile subscribers [11]. Historically, parts of the GSM protocol have been kept proprietary [24]. These "secret items" include encryption and authentication methods. Welte notes, "The specifications of the GSM proprietary On-air encryption A5/1 and A5/2 are only made available to GSM baseband chip makers who declare their confidentiality."[24]. In an effort to make GSM obtainable for academia, open source movements such as OpenBTS[17] have been developing freely available tools to deploy a personal GSM tower. OpenBTS is a solution that allows for a low cost GSM tower deployment. This implementation involved Software Defined Radios (SDRs) as well as OpenBTS.

2.1 GSM Architecture

Bettstetter et al. explain that, "GSM networks are structured hierarchically. They consist of at least one administrative region, which is assigned to a MSC."[2]. GSM architecture can be broken up into two large parts: the Base Station Subsystem (BSS) which consists of a base transceiver station (BTS), and a base station controller (BSC). The other critical section is the Networking Switching Subsystem (NSS).

The BSS consists of a base transceiver station, and the base station controller. These two elements of GSM architecture are responsible for controlling which radio frequency bands and channels are transmitted on, as well as functions that affect data in transport. The NSS is responsible for all information related to the user such as activity status, location information, and the handover process (the process by which devices are able to move

through coverage areas of various base transceiver stations).

There are four types of handover processes that may occur[18]:

- Intra-tower handover - when the mobile must change frequencies due to some type of interference. The device still remains connected to the same tower.
- Inter-BTS Intra BSC Handover - when the device moves out of the coverage of one BTS but into the coverage area of another BTS controlled by the same BSC.
- Inter-BSC Handover - when the mobile device moves out of the range of base station transceivers controlled by a single BSC. This handover is controlled by the MSC.
- Inter-MSC handover - the two MSCs involved between the handover negotiate in order to complete the successful handover.

Base transceiver stations are a core component to radio communication in all wireless communication (for example, GSM, CDMA, 802.11). The BTS portion of the GSM architecture is responsible for handling all radio activities to and from the tower. Each BTS is additionally responsible for encoding, decoding, encrypting, decrypting, and a plethora of other functions that occur during data communication from the mobile equipment to the physical tower.

The base transceiver station lies under the base station controller. The BSC defines which channel and spectrum radio frequency signals are broadcasted on. It is important to note the GSM standard for broadcast channels varies depending on geographic location. The deployment for this project made use of 850, 900, and 1800 MHz bands.

The Networking Switching Subsystem (NSS) is an integral part of the GSM architecture. Key components include the Home Location Register (HLR), Visitor Location Register (VLR), Mobile Services Switching Center (MSC), Equipment Identity Register (EIR) and Authentication Center (AUC).

The Mobile Services Switching Center (MSC) is the "brain" of the Networking Switching Subsystem. Authentication, handover, location updating, new user registration, and call routing all occur via actions taken by the MSC.

The HLR is a database used solely for user subscription management. All information stored here is considered permanent. A user's profile that contains location information and activity status is also stored here. Information from a SIM card such as a user's data and calling plan is additionally stored here.

The Visitor Location Register (VLR) is a database that contains temporary information about a subscriber. When users roam into a new GSM cell, the VLR may request data about the mobile equipment from the Mobile Services Switching center so that it will have information required for forwarding calls without querying the HLR. This is a sort of caching mechanism.

The Equipment Identity Register (EIR) contains a information about all currently valid mobile devices allowed on a GSM network. The EIR correlates the unique number provided to each mobile device known as the International Mobile Equipment Identity (IMEI). The FCC also uses the EIR and the IMEI to identify lost or stolen devices.[6]

3 OpenBTS Implementation

In the implementation for this project, OpenBTS was used for the software portion of the base station. A laptop running Ubuntu 12.04.4 LTS was the base operating system for the software. The software defined radios used for building this tower were Ettus N210[20] and B210[19] radios. The deployment was running OpenBTS 5.0, which was when the software was in its alpha stages of development, and had contained a few bugs. Utilizing developers' comments and mailing lists, all initial issues were solved. After resolving these issues, it was possible to observe transmitted traffic via Wireshark, and examine appropriate log files for sent text messages.

3.1 OpenBTS Architecture

OpenBTS aims to replicate a modern GSM tower through software implementations. Asterisk, an open source VoIP service, is used to handle all voice traffic to and from the OpenBTS cell tower [1]. While other documented implementations have also used Asterisk, other VoIP solutions could be substituted in Asterisk's place. Part of the the OpenBTS software stack is sipauthserve. Sipauthserve is used for all authentication in OpenBTS, and was specifically developed to handle cellular authentication by the OpenBTS developers.

3.2 Cost Overview

The implementation discussed in this paper is affordable at a cost of approximately $1,250, not including the price of computer hardware. The implementation discussed in the paper requires a Ettus B210 ($1,100USD at the time of this writing)[19] and four Vert 900 antennas ($35USD each at the time of this writing)[21]. While this implementation does not have direct telecommunication access as industrial deployments would, it can successfully

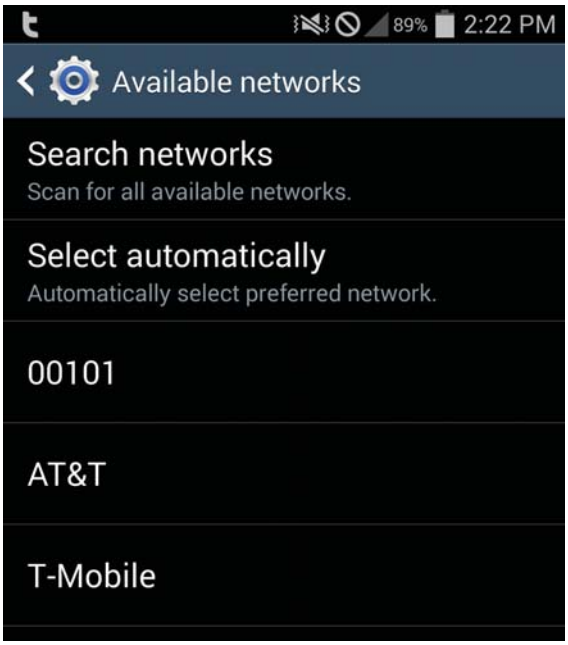


Figure 1: OpenBTS broadcasting as 00101.

act as an IMSI catcher. However, in order to successfully route calls, it would be necessary to configure call routing via a SIP provider. Below is a screenshot of the default network that OpenBTS will advertise (as seen from a Samsung Galaxy S4), 00101.

4 Encryption

As Meyer and Wetzel noted,[14], all encryption and authentication parameters are controlled by the tower. Due to backwards compatibility, GSM phones support a wide band of different cellular technologies. 2G networks originally used A5/1 and A5/2 encryption. A5/2 was developed as "weaker" encryption to conform to US export laws. However, both algorithms have suffered from an onslaught of cryptanalysis, especially with rainbow tables that exist today to crack A5/1 encrypted traffic. Lo and Chen note, "the A5 algorithm, uses a proprietary algorithm for message encryption/decryption."[13]. Kerckhoff's Principle contradicts this paradigm, which as Simmons explains that according to Kerckhoff's Principle, "the opponent knows the system "[22]. The central argument for this principle is that the keys for encryptions should be kept secret, but it should be assumed that the adversary can determine how the cryptosystem works.

4.1 Encryption Attacks

Due to this flaw in the logic behind keeping the algorithm secure for the A5 encryption, there have been successful attacks against the A5 cryptosystem that have been developed [3][14]. The cellular base station can depict which encryption algorithm to use for data communication as long as both the phone and the tower support it. Similarly to TLS downgrade attacks, base stations can lower the level of encryption used in an effort to capture weakened encrypted traffic that will later be cracked via brute force or rainbow tables. Additionally the tower could just attempt to use A5/0 (absence of encryption) and collect all free flowing plain text data. Meyer and Wetzel explain that the tower can send false information regarding its encryption capabilities to the mobile device and due to this, encryption can be completely disabled [14]. Since the initial creation of the algorithms, some have been made public, and patented [16].

5 Authentication

Mobile devices must authenticate with the network before being able to utilize the network resources. There are some differences between GSM authentication versus CDMA authentication, however both are still vulnerable to the IMSI Catcher threat. The following sections discuss the authentication methods for the respective architectures.

5.1 GSM Authentication

In order for a cellular phone to perform any normal mobile function such as SMS, MMS, calling, it must first authenticate to a mobile cell. GSM authentication occurs through the A3 authentication algorithm. A high level overview of the algorithm is as follows, the GSM tower will send a random number that has been incorporated with a shared key (Ki) as well the A8 ciphering algorithm to the mobile equipment. The mobile device will then process the random number, and return a signed response known as SRES to the tower [25] .This response will have been signed via the mobile equipments shared private key known as "Ki." Once the tower receives the signed response it will compare the mobile devices signed response to the towers signed response with the users Ki. If they match, authentication was successful. It is important to note that Ki itself is never sent over the network.

5.2 CDMA Authentication

CDMA utilizes a similar challenge response authentication mechanism. The difference lies in the encryption

algorithms in place by CDMA. The Cellular Authentication and Voice Encryption (CAVE) algorithm utilizes a 128-bit key referred to as the "Shared Secret Data" (SSD), an "A-Key", and the electronic serial number (ESN) of the mobile device.

The Shared Secret Data (SSD) is composed of two parts, SSD_A and SSD_B. SSD_A is utilized for creating an authentication signature by the mobile device to return to the tower. This signature proves to the tower that the mobile device is whom it claims to be. SSD_B is used to generate keys for voice and messaging encryption.

The A-Key is programmed into the mobile device (similar to an IMSI on a SIM) and is also stored in a CDMA tower's authentication center. This A key is utilized to generate separate sub-keys for voice and message encryption. A major difference that lies between CDMA and GSM is that the A-Key can be re-programmed where the IMSI cannot. However, after reprogramming the A-Key, the authentication center must be updated with the new number as well.

Upon receiving a random number challenge (RAND) from a tower, the mobile device will use the SSD along with the RAND number as parameters for the CAVE algorithm. The return value of the CAVE algorithm is an 18 bit authentication signature. The authentication signature is then sent to the base station, which compares the authentication signature for validity.

5.3 OpenBTS Authentication

OpenBTS supports the standard mechanisms of GSM encryption as discussed previously in this paper. However, failing to have a copy of Ki also on an OpenBTS deployment will prevent a phone from ever successfully authenticating. If an attacker is attempting to have a mobile device join their GSM tower they will most likely not know the Ki of the mobile device's SIM either. However, successful authentication can be achieved through OpenBTS' open authentication method.

Open authentication does not require prerequisite knowledge of Ki at the GSM tower level. OpenBTS open authentication automatically accepts the the signed response from the mobile equipment. This allows any device to successfully authenticate and then associate with the base station.

During normal authentication when the mobile device returns the signed response a tower would compare the signed response to the output of the tower's Ki and the original RAND challenge for validity. Open authentication accepts any response from the mobile device. This feature allows an attacker to circumvent any prior knowledge of Ki.

Ki is a shared private key stored on all SIMs, as well as stored in the home location registry portion of a GSM base station. This key is used for initial authentication between mobile equipment and tower. Ki never leaves the phone and is only used in authentication for signing the initial random data sent to the mobile device from the tower. Additional encryption measures on a SIM protect the Ki from being accessed. However, manufacturers that implement weak encryption are vulnerable to having the SIM card compromised.

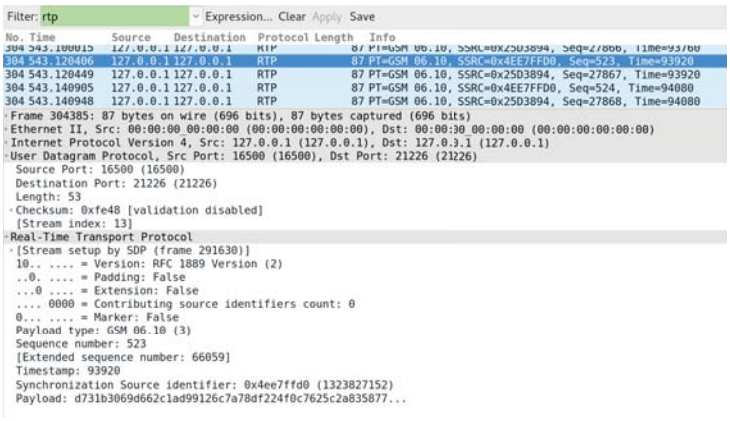


Figure 2: Wireshark RTP traffic.

It is important to note that according to *Getting Started with OpenBTS*, "OpenBTS has an alternative authentication method for this situation known as 'Cache Based Auth.' It performs an initial authentication exchange with the handset and records the results. It uses this same request and expects the same answer in the future. The method is not as secure as unique exchanges for each request but is still better than completely disabling authentication."[10].

6 Proof of Concept

The OpenBTS implementation, allows for packets to be captured on the interface connected to the software defined radio. By utilizing Wireshark, an open source packet capturing application, phone calls between mobile devices were captured. Wireshark provides the ability to examine the RTP streams containing GSM payloads. The figure below shows a screenshot of a packet capture using Wireshark [26].

The figure below displays output from a log file with decoded text messages containing "ICMP_REQUEST" and "ICMP_REPLY."

The figure below is also a screenshot on one of the devices that sent a text that was logged in the figure above.

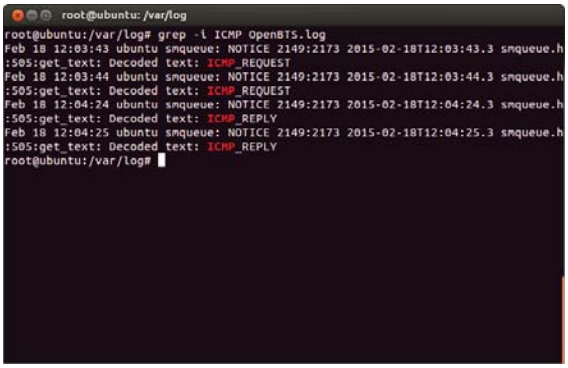


Figure 3: OpenBTS log of text messages sent.

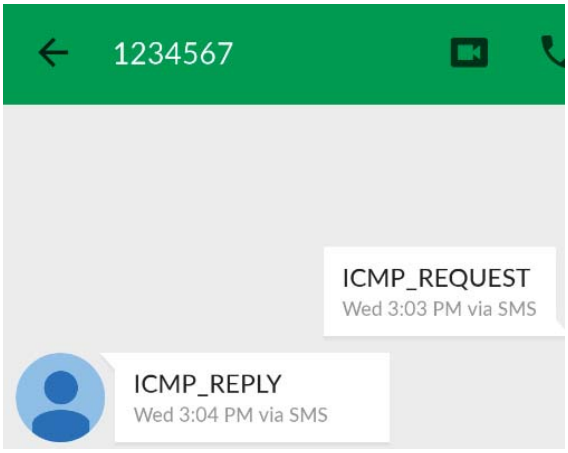


Figure 4: Text message application displaying a conversation.

7 Results

The experiment with open authentication on the OpenBTS tower resulted in several phones associating and authentication without any indication of foul play. Due to the GSM protocol allowing the tower to define which authentication and encryption methods to implement, it is trivial to configure a tower to use encryption methods with known attacks. However, utilizing A5/0 encryption and open authentication allows for plain text communication to occur with data in transit and the mobile device to associate without knowing the private key, Ki.

Additionally, modifying the mobile network code through the OpenBTS configuration command line interface can force the tower to appear as a major mobile carrier to all mobile devices. This code is represented as a two to three integer value accompanied with a mobile country code, also a two to three integer value. The mobile country code represents a geographic location numerically. By modifying these values, the tower can appear as any major network carrier from any geographic location.

8 Conclusion

Several methods have been proposed that attempt to change how mobile devices authenticate with networks. These changes have been proposed in order to prevent man in the middle attacks such as the IMSI Catcher method described in this paper. For example, Lee et. al propose that the HLR can give a VLR a temporary key that it may use to authenticate the mobile station without knowing its Ki[12]. The paper also discusses how this method can be used for "bilateral authentication" because the mobile station will be able to authenticate that the VLR is an imposter since it will have to get a secret from the mobile station's HLR. Chang et. al also propose a method to share a secret key between the mobile station and the visitor location registry in order for the device to be authenticated with the home location registry [4]. If implemented, these methods would be able to ensure that simple man in the middle attacks such as the IMSI Catcher detailed in this paper would not be possible to implement.

9 Future Work

Public awareness of GSM interception attacks is increasing, along with applications that aim to detect IMS catchers and rogue towers. Analysis of Android application IMSI detection techniques will be investigated for effectiveness and possible mitigations.

Future work will focus on deploying a multi-tower network to analyze network traffic overhead, and inter tower communication and how to intercept the handoff.

10 Acknowledgements

This research would not be possible without a grant from the Office of the Provost and the Rochester Institute of Technology [15]. This implementation supports the Rochester Institute of Technologys mobile security and forensic curriculum [5].

References

[1] ASTERISK.ORG. Asterisk, 2015.

[2] BETTSTETTER, C., VOGEL, H.-J., AND EBERSPACHER, J. Gsm phase 2+ general packet radio service gprs: Architecture, protocols, and air interface. *Communications Surveys & Tutorials, IEEE 2*, 3 (1999), 2–14.

[3] BIRYUKOV, A., SHAMIR, A., AND WAGNER, D. Real time cryptanalysis of a5/1 on a pc. In *Fast Software Encryption* (2001), Springer, pp. 1–18.

[4] CHANG, C.-C., LEE, J.-S., AND CHANG, Y.-F. Efficient authentication protocols of gsm. *Computer Communications 28*, 8 (2005), 921–928.

[5] EMBLING, E., GILBERT, S., AND MISLAN, R. Designing and implementing a wireless carrier topology in a lab environment. In *Global Wireless Summit* (2013).

[6] FCC.GOV. Protect your smart device, 2015.

[7] FEDERRATH, H. Protection in mobile communications.

[8] GOLDE, N., REDON, K., AND BORGAONKAR, R. Weaponizing femtocells: The effect of rogue devices on mobile telecommunications. In *NDSS* (2012).

[9] HERMANSSON, J., MANSSON, C., JACOBSSON, A., NYSTROM, Z., KARLSSON, B., PALMGREN, C., LEUHUSAN, G., AND ORNEHOLM, F. Digital mobile telephone system in which each subscriber is assigned a telephone number and several subscriber identity module (sim) cards, Aug. 12 1997. US Patent 5,657,373.

[10] IEDEMA, M. *Getting Started with OpenBTS*. O'Reilly Media, 2014.

[11] INTELLIGENCE, G. Definitive data and analysis for the mobile industry, 2015.

[12] LEE, C.-C., HWANG, M.-S., AND YANG, W.-P. Extension of authentication protocol for gsm. *IEE Proceedings-Communications 150*, 2 (2003), 91–95.

[13] LO, C.-C., AND CHEN, Y.-J. Secure communication mechanisms for gsm networks. *Consumer Electronics, IEEE Transactions on 45*, 4 (1999), 1074–1080.

[14] MEYER, U., AND WETZEL, S. On the impact of gsm encryption and man-in-the-middle attacks on the security of interoperating gsm/umts networks. In *Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004. 15th IEEE International Symposium on* (2004), vol. 4, IEEE, pp. 2876–2883.

[15] MISLAN, R. Mobisploit provost implementation grant, 2014.

[16] MURTO, J. Subscriber authentication in a mobile communications system, Nov. 23 1999. US Patent 5,991,407.

[17] OPENBTS.ORG. A platform for innovation. Online.

[18] RADIO-ELECTRONICS.COM. Gsm handover or handoff, 2015.

[19] RESEARCH, E. Usrp b210. Online.

[20] RESEARCH, E. Usrp n210. Online.

[21] RESEARCH, E. Vert900 antenna. Online.

[22] SIMMONS, G. J. Authentication theory/coding theory. In *Advances in Cryptology* (1985), Springer, pp. 411–431.

[23] SPECIFICATION, G. 11.10. *ETSI TC-SMG: Digital cellular telecommunications system (Phase 2+)* (2000).

[24] WELTE, H. Anatomy of contemporary gsm cellphone hardware. [Online. Accessed 15-February-2015].

[25] WILLASSEN, S. Forensics and the gsm mobile telephone system. *International Journal of Digital Evidence 2*, 1 (2003).

[26] WIRESHARK.ORG. Wireshark, 2015.

Covert Channel Using ICMPv6 and IPv6 Addressing

Geoffrey Ackerman
Department of Computing Security
Rochester Institute of Technology
gma8175@rit.edu

Daryl Johnson
Department of Computing Security
Rochester Institute of Technology
daryl.johnson@rit.edu

Bill Stackpole
Department of Computing Security
Rochester Institute of Technology
Bill.Stackpole@rit.edu

Abstract— *Internet Protocol version 6, the latest revision of the Internet Protocol (IP), is rising in popularity. Along with it has come ample opportunity for the discovery and utilization of fresh, new covert channels. This paper proposes a covert channel using this "IP Next Generation Protocol", widely referred to as IPv6, as well as its associated protocol ICMPv6. As a proof-of-concept, two hosts running respective sender and receiver python scripts will take advantage of ICMPv6 Echo messages and the IPv6 addressing scheme to send and receive data unbeknownst to any host, person, or other entity that may be monitoring or watching over the network.*

Keywords: IP, IPv6, ICMPv6, Covert Channel, Sniffing, Spoofing

1. Introduction

A covert channel is a communication channel that transfers information in ways prohibited by computer security policy and unspecified by the respective protocol. This can be accomplished through the use of the covert channel's structure to transfer small amounts of data at a time. While encryption denies a traffic observer knowledge of the contents of a conversation, the goal of covert channels is to deny knowledge of the conversation itself. Without knowledge of the operation of a covert channel, observation of the traffic would not reveal the channels existence or contents. Even if the message is seen, there is no way of knowing that the datagram is something out of the ordinary. It is an example of security through obscurity. This paper will discuss some IP-based covert channels that have been discovered and will propose, demonstrate, and evaluate a new channel using the IPv6 and ICMPv6 protocols.

1.1 Covert Channel Types

Covert channels are generally categorized as storage or timing channels. A storage channel "involves the direct or indirect writing of a storage location by one process and the direct or indirect reading of the storage location by another process" [1]. A timing channel involves transfers of information based on a chosen timing interval whose messages must be synchronized with a similar clock on each end. Additionally, some advocate for a third category called behavioral channels [2]. These channels use an alteration of internal states or behavior of an application to leak information.

1.2 Related Covert Channels

Joe Klein, a network expert with the North American IPv6 Task Force, said,

> "We are expecting a lot here to be discovered and disclosed. But just like the early implementation of any technology, we expect to find defects and covert channels." [3]

Many IPv6-based and IPv4-based covert channels have already been found. One such example of an IP-based covert channel is known as the Loki Project [4]. This channel uses the ICMPv4 protocol. Covert data is sent and received through Echo-Request and Echo-Reply packets. More specifically, the tool uses the data field of these echo messages, a field of both arbitrary length and content normally used for timing information, to hide the data. Since this field is both optional and not useful to most devices, it is not normally checked.

Another example is a tool called VoodooNet, or v00d00n3t [5], which is one of the most commonly referenced covert channels using IPv6. Created by R.P. Murphy and presented at Defcon 14 in 2006, this tool uses a technique known as 6to4 tunnelling by encapsulating IPv6 network traffic within today's standard, IPv4. Through optional extension headers in IPv6 packets, messages can be transferred from one host to another. This methodology of 6to4 tunneling was not revolutionary, as it had been demonstrated and used for years, but the use of it as a way to covertly send information was.

IPv6 implements "extension headers", which are used to carry optional Internet Layer information. No intermediary nodes read these headers as they are only meant for the destination node. These headers can be abused to create covert channels, as described in [6]. One example uses unusual options in the extension header so that when the destination node reads this option, it will skip the extension header and move on to the next one, thus allowing covert data to be stored in the data section of the skipped header. Another example uses the PadN option, which is used to align packet boundaries by inserting two or more octets of padding into a header's Options. Per the IPv6 RFC, this padding should consist of 0's. However, certain operating systems will accept non-zero padding, thereby allowing

arbitrary covert data.

An extensive study of IPv6-based covert channels can be found in the dissertation "Network-Aware Active Wardens in IPv6" by Grzegorz Lewandowski of Syracuse University [7]. In this paper, Lewandowski theorizes many fields in many different protocols that use IPv6 that may be used as a covert channel. He suggests the idea of setting a false source IPv6 address multiple times but does not explore the idea any further. To understand how this can be used as a covert channel, one must have an understanding of the protocols.

2. Protocol Overviews

2.1 History of IPv6

Due to the unprecedented expansion of Internet usage and the ever-increasing number of new devices being connected to the Internet, the impending shortage of IPv4 address space available for use was recognized. In response, the Internet Engineering Task Force (IETF) initiated the design and development of new protocols and standards to eventually supplant Internet Protocol version 4. The result, the "IP Next Generation Protocol", was developed with larger 128-bit addresses compared to the 32-bit addresses used by IPv4. This allows for an astoundingly larger address space, approximately $3.4x10^{38}$ addresses, as compared to the approximately $4.3x10^{9}$ addresses available in IPv4. The basic protocol was published in 1998 [8] and as of September 2013 the percentage of users using IPv6 reaching Google services surpassed two percent [9]. In 2006 the associated Internet Control Message Protocol (ICMPv6) specification [10] was published to serve as a critical component of IPv6.

2.2 ICMPv6

ICMPv6 uses Echo-Request and Echo-Reply messages in the same way as ICMPv4. Within these message packets there are six distinctive fields (*Table 1*). This covert channel will use the Identifier field, which is 16-bits in length, and the optional Data field of arbitrary length. The Identifier's purpose is to match corresponding Echo-Request and Echo-Reply messages with each other. This gives two hosts exchanging echo messages confirmation that they are talking with the intended target. The actual number used does not have an effect on the outcome of the communication. The data field consists of zero or more octets of arbitrary ASCII data, generally used for timing information (i.e. computing round trip time).

Table 1: ICMPv6 Packet

Type	Code = 0	Checksum
Identifier	Sequence Number	Optional Data

2.2.1 Neighbor Discovery Protocol

In version 6 of IP and ICMP, ARP no longer exists. As a replacement, there is the Neighbor Discovery Protocol [11] operating in the Link Layer of the Internet model [12]. Hosts and routers use this new protocol for address autoconfiguration, determining the link-layer addresses of neighboring hosts, to keep track of which neighbors can be reached, and to find routers that are available to forward their packets. The protocol uses Router and Neighbor Solicitations and Advertisements. Any host can issue a Router Solicitation to find any routers attached on a link. These packets are sent out periodically. A router can also advertise its presence on a link using Router Advertisements. Routers will advertise themselves periodically or in response to a specific solicitation. Similarly, any node can determine link-layer addresses of any neighbors using a Neighbor Solicitation. These are also used to determine if a known host is still reachable using a cached link-layer address. Hosts use Neighbor Advertisements to respond to Neighbor Solicitations. All of these packets play an integral role in the functioning of the ICMPv6 protocol, and therefore the IPv6 protocol.

2.3 IPv6

The IPv6 addressing scheme (*Table 2*) commonly consists of a 48-bit ISP-assigned "Site Prefix" field, a 16-bit "Subnet" field, and a 64-bit "Interface Identifier". An example address is: 21DA:D3:1:2F3B:2AA:FF:FE28:9C5A/64 (leading zeroes and groups of one or more consecutive zeroes can be omitted):

Each address consists of eight groups of four hexadecimal numbers. Each of these groups is 16 bits, or two octets. An address can be assigned through various means, including from the network interface's MAC address, from a DHCPv6 server, or through manual configuration. Every IPv6-enabled interface must have a link-local address, which has the prefix fe80::/64. The site prefix can be either link-local, unique local (equivalent to IPv4 private addresses), or global (equivalent to IPv4 Internet addresses). Global IPv6 addresses are globally routable and can be used to connect to addresses with a global scope anywhere, or addresses with link-local scope on the directly attached network. Global Unicast addresses have an IPv6 prefix of 2000::/3 [13].

Table 2: IPv6 Address

Site Prefix	Subnet	Interface ID
21DA:00D3:0001:	2F3B:	02AA:00FF:FE28:9C5A

3. New Covert Channel Process

The proposed covert channel is demonstrated as a proof-of-concept two-sided python script, which utilizes Scapy [14]. Scapy is a packet manipulation tool that can create and send custom packets containing arbitrary data. The python

script consists of a sender and a receiver on two different hosts, both run with root privileges.

The Identifier field of an ICMPv6 Echo-Request packet, which can be manually configured by the sender, and the Data field, which consists of optional arbitrary data of arbitrary length, will be modified to initiate a connection and to establish a dynamic alphabet by which messages can be encoded and decoded. The use of a dynamic alphabet is purely for the sake of complexity. If this channel were to be discovered, the randomization and changing of alphabets would help to mitigate any frequency analysis done on the source IPv6 addresses. Once the encoding step is complete, the sender will use a statically configured IPv6 source address to represent, transmit, and deliver a desired message via raw UDP packets. A few assumptions must be made for the purpose of this proof-of-concept: 1) both the sender and receiver have access to root privileges, 2) IPv6 is enabled on all devices that the covert channel traverses, 3) sender and receiver real IPv6 addresses must be a shared secret of both parties, and 4) UDP is not blocked.

The proposed covert channel uses the ICMPv6 protocol much in the same way as the Loki Project. However, it is unique in that the Echo-Request messages are solely used to establish the alphabet(s) that will be used to encode and decode the subsequent message. The message is created and transferred when this process is taken one step further using an alternative method. UDP datagrams with spoofed, unique source IPv6 addresses are used to transfer the message with the source addresses representing the actual message fragments. Every two octets of the interface identifier represents one character of a message that has been created using the alphabet defined in each Echo-Request packet. An example using three unique source IPv6 addresses sending "Hello World" is illustrated in *Fig. 1*.

IPv6 Address: 2000::4ca2:8df7:1589:1589

Translation:

4ca2	8df7	1589	1589
H	e	l	l

IPv6 Address: 2000::1617:9742:1617:a49e

Translation:

1617	9742	1617	a49e
o	W	o	r

IPv6 Address: 2000::7950:b701:0000:0000

Translation:

7950	b701	0000	0000
l	d		

Fig. 1: Hello World Example

Note: the use of simple hex-based encoding/decoding could potentially be replaced with a more complex encryption mechanism. The use of encryption with this channel would help to normalize character frequencies and eliminate the effectiveness of frequency analyses. That being said, this paper's focus is on making use of the spoofed IPv6 source addresses of UDP packets, not the variety of techniques that could be utilized to establish a way to obscure or obfuscate the intended message.

3.1 The Process

3.1.1 Initial Alphabet Definition

Since the encoded message will be an IPv6 address, each plaintext alphanumeric character must have an associated string of four hexadecimal values (the alphabet), as illustrated in *Fig. 2*.

A	5120	G	907d	M	640f	S	5d74	Y	9129
a	32bf	g	7e30	m	62da	s	4b6c	y	860e
B	bf05	H	4ca2	N	3f45	T	f43e	Z	1a89
b	36b7	h	ae28	n	e4a3	t	3f52	z	c0df
C	5c3e	I	1589	O	dd17	U	67dc	1	e285
c	51cf	i	3afd	o	1617	u	dd29	2	625d
D	429d	J	5653	P	bfec	V	ef2a	3	5519
d	b701	j	ffc9	p	6a70	v	88b0	4	abc5
E	e2f7	K	3096	Q	4b07	W	9742	5	5a67
e	8df7	k	56f9	q	9df4	w	d27a	6	1ed4
F	f68a	L	612b	R	c56c	X	7378	7	a841
f	ea53	l	7950	r	a49e	x	482d	8	9edb
								9	a075

Fig. 2: Example Alphabet

The receiving host, known as Bob, starts his receiving script, which begins by passively sniffing the network for ICMPv6 messages with his IPv6 address as the destination address. The sending host, known as Alice, reads in the message to send and chooses the number of UDP packets to send per alphabet. Alice then calculates the number of unique alphabets that are needed and creates them using a random 4-digit hex-string generator.

In *Fig. 3* Alice is seen inputting her message and deciding on the number of packets to encode and send per each unique alphabet. Then three variables, two of which are sent with the initial alphabet, are displayed. Once the appropriate number of alphabets are created and stored, the message needs to be encoded. The encoding occurs before anything is sent to Bob. Based on the chosen number of packets per alphabet, each alphabet is used to encode a certain number of characters in the original message. The encoding is accomplished character by character - each plaintext character is compared with each letter in the alphabet array and if they match, the associated hex string is used as that letter's encoding and as one octet of the IPv6 address to be spoofed.

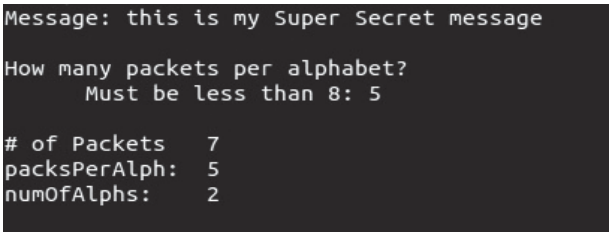


Fig. 3: Alphabet Creation

Now Alice is ready to initiate the message sending process. To start, an ICMPv6 Echo-Request is built with an arbitrary ID value (e.g. 0x62, or 98 in decimal) and the first alphabet is inserted into the Data field starting at the 98th position (Note: any follow-up alphabet Echo-Requests will have an ID value of 0x63. These values are arbitrary but they must be an agreed upon, non-random value). All data in positions 0-97 are randomly generated values. The Echo-Request message is then sent to the recipient's IPv6 address.

When Bob reads through all of the packets that have been captured by the filter he set, he looks for an Echo-Request destined for his address, with an ID Value of 0x62, and data in an alphabet format at position 98. The following is the alphabet as it is stored in the Data field of the Echo-Request, shown in *Fig. 4*.

A2e8c**a**927f**B**7c6d**b**6874**C**db94**c**8a2b**D**dfcd**d**d0b6**E**c42c
e9029**F**29a**ff**6d2**G**1958**g**a7d8**H**fbf9**h**e10a**I**c418**i**3890
J93ef**j**f048**K**598d**k**3095**L**af13**l**2ba9**M**cb13**m**d50c**N**9d03
nf0be**O**995c**o**1eb1**P**ff74**p**205e**Q**a9e6**q**1782**R**6b73**r**f4c2
Sc9d4**s**9cd4**T**b50b**t**24d8**U**7bc9**u**3a54**V**421c**v**fdfb**W**72c9
w10a1**X**e2eb**x**59d0**Y**f236**y**4ba0**Z**5206**z**67921fda424d10
31bd8**4**395a**5**9ac0**6**4f207f1938**8**41799f842**1715**

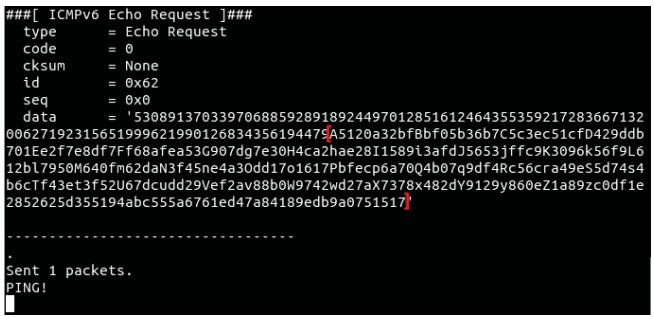


Fig. 4: Alphabet w/in ICMPv6 Echo-Request (Alice)

When this is found, the alphabet is stored and an Echo-Reply is crafted as an ACK to be sent back to Alice. Immediately after, Bob begins sniffing for UDP packets destined for him *(Fig. 5)*.

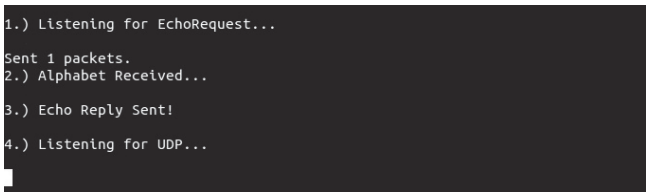


Fig. 5: Listening for UDP Message Fragments (Bob)

3.1.2 Message Sending

After sending the initial alphabet Echo-Request, Alice begins sniffing the network, awaiting the Echo-Reply acknowledging retrieval of the alphabet. Once the ACK is received, five UDP packets with Global prefixes are created from the first alphabet and sent to Bob. Four characters from the message are used per packet (i.e. the four groups of two octets used as the interface identifier in the IPv6 source address). Each address is created from the original message, spoofed as the source address, and sent to Bob via UDP *(Fig. 6)*.

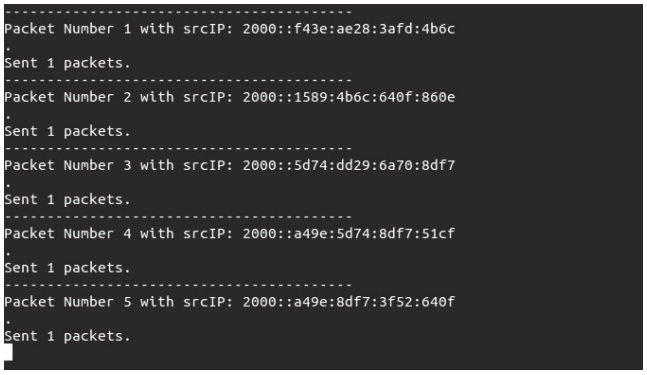


Fig. 6: Message Fragments Sent! (Alice)

3.1.3 Message Retrieval

Bob reads each source IPv6 address in each UDP packet addressed to his IPv6 address and parses out the interface identifier. These encoded message fragments are stored until the entire message is received. After all UDP packets have been inspected, Bob sends an Echo-Request with a message in the Data field signaling that the message fragments have been received and he is ready for the next alphabet to be defined *(Fig. 7)*.

```
1.) Listening for EchoRequest...

Sent 1 packets.
2.) Alphabet Received...

3.) Echo Reply Sent!

4.) Listening for UDP...


Sent 1 packets.

5.) Echo Request Sent!

    - Ready for next Alphabet
```

Fig. 7: Messages Received - Ready for next Alphabet (Bob)

3.1.4 Repeat Until Complete

From this point onward, assuming there is more message data to send, the cycle of (Alice) defining an alphabet, (Bob) ACK, (Alice) sending message fragments, (Bob) reading and storing IPv6 interface identifiers, and (Bob) requesting a new alphabet is repeated. Once the end of the message is sent and Bob requests a new alphabet, Alice sends an Echo-Request with an ID Value of 0x64, signaling the end of the message. Once the end of the message is signalled, Bob can decode the message using each alphabet he has stored for the chosen number of message fragments, as specified in the initial alphabet definition *(Fig. 8)*.

```
************************************************
********************DECODING********************
************************************************

Encoded Message:
f43eae283afd4b6c15894b6c640f860e5d74dd296a708df7a49e5d748df751cfa49e8df
73f52640f8df74b6c4b6c32bf7e308df7

Decoded Message:
This Is My Super Secret Message

************************************************
************************************************
************************************************
```

Fig. 8: Received Encoded and Decoded Messages (Bob)

4. Covert Channel Characteristics

To evaluate this covert channel, Eric Brown's criterion as described in [15] will be used. Using the following characteristics and metrics this channel's implementation is evaluated.

4.1 Type

This covert channel is considered a storage channel. While the described proof-of-concept's implementation depends on the sender and receiver agreeing on a set period of time to sniff the network while waiting for the necessary packets to come through, the actual data being sent is stored within a field of an IPv6-based UDP packet (i.e. the Source Address field). The timing itself does not contribute to the message being sent.

4.2 Throughput

The throughput of this covert channel was tested using a 2,000 character message, which required 500 UDP packets. The average time it took for these 500 packets to be sent using variable amounts of virtual RAM is shown in *Fig. 9*. The results show that the amount of vRAM allocated per VM is generally proportional to the channel's throughput. The tests conducted were performed on two Linux virtual machines running Ubuntu 14.04 in VMware Workstation. The host machine was a Lenovo U430p laptop with the following specifications:

Windows 8.1 (64-bit)
Intel Core i3-4010u @ 1.70 GHz
4.00 GB RAM w/ 99.73 MHz bus speed

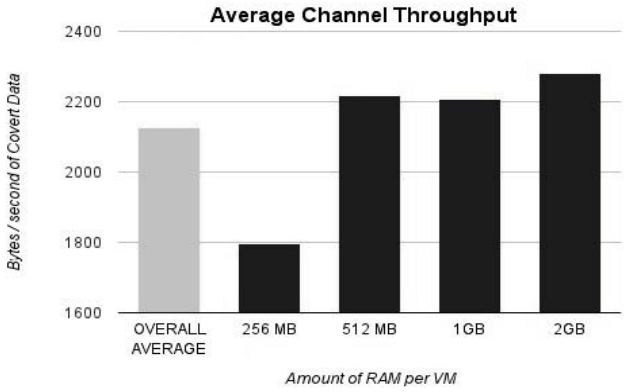


Fig. 9: Channel Throughput

The overall average time it took for each full message transmission was 3.799 seconds. Dividing the total number of characters in the message by the time it took to send equates to 531.2 characters per second. Since one character is represented by four bytes of data, the overall average bandwidth of this covert channel is 2124.938 bytes/second.

The calculated throughput of this channel may vary by the chosen time interval used by each party and the number of alphabets used. There is an inverse relationship between this time interval and the overall bandwidth of the channel. Additionally, an attacker would likely limit the successive transmission of packets in order to avoid detection due to floods of packets. Spreading out the transmissions over longer intervals of time and limiting the amount of UDP messages sent at once reduces the throughput while decreasing probability of detection. Finally, the overall bandwidth of the channel will be limited by 1) The third party application (Scapy) that is used to send and receive the packets as it will have a limit to how quickly it can do so, and 2) The natural latency caused by the infrastructure that the packets must traverse to and from the source and destination.

4.3 Robustness

The authors consider the survivability of this channel to be high. An extremely important aspect of this covert channel is its routability. Because the IPv6 address varies only in the interface identifier, the prefix can be link-local, unique local, or global, whichever the user decides would work best in the given situation. Encountering firewalls or proxy servers may cause problems, but due to the need for both IPv6 and ICMPv6 protocols in modern networks, these perimeter devices will likely allow all of this traffic in and out. With the increased number of Internet connected devices using IPv6, network managers will not want to implement a block of this protocol for fear of restricting their clients and other IPv6 users who are legitimately on the network. Additionally, if the user is aware of the unique local prefix of the site he/she is a part of, they would be able to circumvent a firewall restricting non-site specific addresses simply by using that unique local prefix instead of a global prefix for each spoofed source address. Moreover, sending the data over standard ports, such as port 80, 443, or 8080, would only decrease the likelihood of the data being blocked.

4.4 Detection

Little traffic generated by this channel looks out of place or malicious. The one packet-field that appears different from normal is the data field of the Echo-Requests that are sent to and from each host. However, this field is rarely inspected by any devices since its intended purpose is timing information. Using the standard Ping networking utility, this data field is padded with arbitrary data (e.g. on Windows, in both ICMPv4 and ICMPv6, the alphabet is used). However, there is no standard for this payload data and it can be 0 or more bytes in length. It would be possible for an intrusion detection system, such as Snort, an open source network IDS/IPS, to use manual rules that will check for anomalous payloads in particular packets. If the payload of the Echo-Request packets were to be checked for abnormal length or content, an alarm may be triggered. The default payloads for common operating systems could additionally be learned and used to detect any abnormal ICMPv6 packets. That being said, using standard default rule sets, Snort failed to detect anomalous payloads in [16] in multiple tests. On top of that, there is nothing abnormal about the IP addresses that are spoofed.

If this covert channel were implemented using a site-specific prefix, a site with a large amount of IPv6 traffic would need some sort of constantly-updating white list to keep track of every legitimate IPv6 address on the network. If properly implemented, anti-spoofing and anti-alien firewall rules could be used to detect site-specific addresses coming in from the Internet. If implemented using the global prefix, as demonstrated in the proof-of-concept, it may be possible for network border watchers to recognize non-local source addresses. However, being the equivalent of public IPv4 addresses, these global addresses are not likely to be seen as abnormal. Furthermore, the likelihood of IPv6 source addresses to be analyzed statistically and for abnormalities is low. Overall, the authors consider the probability of detection to be low-to-medium.

4.5 Prevention

One quick and easy way to prevent this channel's implementation would be disabling ICMPv6, eliminating the receiver's ability to decode the message that he receives. However, due to the importance and vitality of the ICMPv6 protocol, it cannot be blocked. If IPv6 is enabled, ICMPv6 must also be enabled (and vice-versa). Therefore, a user can enable IPv6 on his or her host machine (assuming it is not already enabled by default), establishing an open channel to anyone on the outside. Another way to prevent this attack is to totally block all IPv6 traffic from coming in or going out. While this may have been an applicable solution five years ago, prior to the depletion of IPv4 address space, it is no longer viable. Businesses must be aware of and ready and able to manage the use of IPv6 on their network. They must have the protocol enabled on all devices and network managers must monitor, analyze, and inspect this traffic thoroughly. It is common best practice to block ICMP that is initiated from an external host. This defense could prevent a dynamic alphabet from being possible. Completely disabling UDP would prevent the main covert channel from existing entirely.

Intrusion detection/prevention systems pose a threat to this covert channel. Snort can trigger alerts based on signature, protocol, and anomaly-based inspection. ICMPv6 payloads can be inspected for length and if they are longer than a specified value, an alert is triggered. With the alphabet in the Echo-Request payloads, this type of rule could possibly detect this aspect of the covert channel. However, the aforementioned tests show that Snort IDS fails to detect abnormal ICMP traffic, such as large packet sizes, using the standard rule sets that come with the software. So unless specific rules are applied to detect proper versus improper length of a payload or variations in how certain operating systems build these packets, even the most widely deployed IDS/IPS solution worldwide will not detect or prevent the dynamic alphabet aspect of this channel.

Due to the covert channel's mechanism, little will trigger an alarm at the network level. Therefore, on Brown's rating scale of hard, moderate, and easy this channel is considered hard to prevent.

5. Future Work

Further investigation into the IPv6 protocol should be performed to find additional storage fields that are variable and not normally considered worthy of analysis. These fields may become the basis for even more IPv6 covert channels. Automated timing of the channel has not been tested with

success; the users of the proof-of-concept must interact with the scripts by issuing a KeyboardInterrupt when needed (i.e. at each timing interval's end). Testing of Snort rules, other IDS/IPS rules, and firewall rules against the channel's mechanism need to be performed as well for more precise data on robustness, detectability, and preventability.

6. Conclusion

IPv6-based attacks are by no means uncommon and they will only become more and more prevalent as time goes on and IPv6 becomes more widely accepted and implemented. If network managers do not take IPv6 into account when they are building and monitoring their network infrastructure, they are likely to be a target of an IPv6-based attack. Whether a corporation sees IPv6 as a business driver or not, the protocol is probably running on their network. A network with IPv6-enabled routers, firewalls, and IDS/IPS can have rogue IPv6 traffic coming in and out of the network without being scrutinized. In this case, the new covert channel has free reign to run between any number of devices on the network. As demonstrated by this covert channel proof-of-concept, an open IPv6 channel presents a serious vulnerability to a company's network and can result in a dangerous exploitation. The covertness of this channel is such that with an IPv6 implementation, or lack thereof, there is a very serious risk of successful attack using these standard and essential "Next Generation" protocols.

References

[1] "Covert Storage Channel." ATIS Telecom Glossary. N.p., n.d. Web. 1 Dec. 2013.

[2] Johnson, Daryl, Peter Lutz, and Bo Yuan. "Behavior-Based Covert Channel in Cyberspace." Proc. of The 4th International Conference on Intelligent Systems & Knowledge Engineering, Belgium, Hasselt. N.p.: n.p., 2009. Print.

[3] Lemos, Robert. "Covert Channel Tool Hides Data in IPv6." www.securityfocus.com. N.p., 11 Aug. 2006. Web. 09 Oct. 2013.

[4] Loki Project Daemon9 AKA Route. "Project Loki." www.phrack.com. Phrack Magazine, Aug. 1996. Web. 12 Nov. 2013.

[5] Murphy, R. P. "IPv6 / ICMPv6 Covert Channels." Las Vegas: Defcon, Aug. 2006. PDF.

[6] Mavani, Monali, and Leena Ragha. "Covert Channel in IPv6 Destination Option Extension Header." Circuits, Systems, Communication and Information Technology Applications. International Conference. 2014. (CSCITA 2014). Proc. of 2014 International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA), India, Mumbai. Vol. 1. N.p.: Institute of Electrical and Electronics Engineers (IEEE), 2014. 219-24. Print.

[7] Lewandowski, Grzegorz. "Network-Aware Active Wardens in IPv6." Diss. Syracuse U, 2011. Print.

[8] Deering, S., and R. Hinden. "Internet Protocol, Version 6 (IPv6) Specification."Http://www.ietf.org/. N.p., Mar. 2006. Web. 30 Sept. 2013.

[9] Roberts, Phil. "Internet Society." http://www.internetsociety.org/. N.p., 24 Sept. 2013. Web. 02 Nov. 2013.

[10] Conta, Et Al. "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification." sww.ietf.org. N.p., Mar. 2006. Web. 30 Sept. 2013.

[11] Narten, T., E. Nordmark, W. Simpson, and H. Soliman. "Neighbor Discovery for IP Version 6 (IPv6)." IETF Tools. Internet Engineering Task Force, Sept. 2007. Web. 12 Nov. 2013.

[12] Braden, R., Ed., "Requirements for Internet Hosts - Communication Layers", STD 3, RFC 1122, October 1989.

[13] Hinden, R., OâĂŹDell, M., and S. Deering, "An IPv6 Aggregatable Global Unicast Address Format", RFC 2374, July 1998.

[14] "Scapy." Secdev.org, n.d. Web. 30 Sept. 2013. <http://www.secdev.org/projects/scapy/>.

[15] Brown, Eric, Bo Yuan, Daryl Johnson, and Peter Lutz. "Covert Channels in the HTTP Network Protocol: Channel Characterization and Detecting Man-in-the-Middle Attacks." N.p., n.d. Web. 22 Sept. 2013.

[16] Stokes, Kristian, Bo Yuan, Daryl Johnson, and Peter Lutz. "ICMP Covert Channel Resiliency." N.p.: n.p., 2009. PDF.

Covert Channels in SSL Session Negotiation Headers

Justin Merrill
Rochester Institute of Technology
justinmerrill978@gmail.com
Daryl Johnson
Rochester Institute of Technology
daryl.johnson@rit.edu

***Abstract*—The Handshake headers of the SSL/TLS protocol contain several multi-byte random data fields used in the generation of the encryption keys used during the session. This random data can be replaced with covert messages that can be intercepted on the wire using packet capture techniques. By encoding data into these fields, a modified SSL client can send messages to a legitimate destination, with legitimate application payload data and still leak covert messages to a receiver listening on the wire.**

Keywords: SSL, TLS, Random Field, Covert Channel

I. INTRODUCTION

The Secure Socket Layer (SSL), and its successor protocol known as Transport Layer Security (TLS), are an important and ubiquitous part of the current Internet landscape. It provides the underlying security that makes services such as ecommerce and secure online correspondence possible. The TLS protocol does this by acting as an intermediate layer between the transport protocol (TCP) and whatever application protocol is being used [1]. When the client connects to the server, a secure session is negotiated between them. After this takes place all application traffic is encrypted before transmission, preventing eavesdropping.

The TLS protocol performs two primary functions [1]. The first is ensuring the application data is protected from eavesdropping and manipulation while in transit. This is done using a combination of a cipher algorithm and hashing algorithm negotiated by the client and server during the initial session setup. The second primary function is allowing the client to authenticate the server's identity before connecting with it. This is done via X.509 certificates associated with each unique server [6]. Since these certificates must be signed and checked against a limited number of well known and auditied Root Certificate Authorities who verify the request is legitimate before issuing a certificate. This makes it much harder to maliciously impersonate a well known secure service.

While TLS vastly improves security, it can also make protecting a network more difficult. Since TLS sessions encrypt all of the application data traveling over them, they make it difficult to inspect packets for hidden or malicious payloads. Wrapping covert or malicious traffic will prevent the transmitted data itself from being monitored but still leaves a few key vulnerabilities open to detection. The destination IP address and TCP port are still visible. Such attempts can be monitored and thwarted by targeting suspicious destinations and ports. So, while encrypting covert traffic can help hide it from detection, it is itself not a covert channel. A covert channel is, as Butler Lampson states in one of the earliest papers on the subject, "not intended for information transfer at all" [3].

TLS has several potential covert channel opportunities during the initial connection process. During this time, a subset of the TLS protocol known as the TLS Handshake Protocol is used to negotiate the parameters for the session [1]. This paper will specifically focus on several fields of fixed length random bytes used in the process of generating the encryption keys used during the session.

II. AVAILABLE CHANNELS IN HANDSHAKE

When an SSL client first connects to a server, it goes through a process of negotiating the encryption suite to be used and verification of certificates. The process goes through the following steps [5]:

ClientHello
: Client opens a connection and informs the server of TLS functionality and cipher suites supported. Included is a random field that will be used to formulate the final key and prevent replaying of data.

ServerHello
: Server chooses the cipher it will use for the session, passes along its own info to the client. Server includes its own random value.

ServerCertificate
: Server passes along its public certificate so that the client may verify it.

ServerHelloDone
: Server announces to the client that it is done sending initial negotiation messages.

ClientKeyExchange
: Client exchanges pre-master secret, a value both the client and the server will use to generate the final symetric encryption keys.

ClientCipherSpec
: Client switches into secure communications mode.

Finished
: Client announces end of session negotiation.

ChangeCipherSpec
: Server switches into secure mode.

Finished
: Server also announces end of session negotiations. The pair is now ready to exchange application data.

Within this negotiation process there are three fields of random bytes that are used. Random fields make an excelent candidate for a covert channel. Since the data contained in the field is by definition supposed to be random, any value can be inserted into the field without interfering with the normal operation of the protocol. If steps are taken to ensure the covert messages are encoded or encrypted before being sent it will be very difficult to differentiate covert messages from random data.

During the ClientHello and ServerHello phases of the session setup, both sides generate a 28 byte random value and send it to the opposing side [1]. This value is combined with a 32-bit datetime stamp and sent over the wire in unencrypted plaintext. This technique has already been partially explored as a means to leak and intercept the encryption keys and perform an attack on the encrypted session [2]. Its usefulness is not limited to leaking keys, as a full covert message payload can be transmitted using the same channel.

One of the things that makes this channel particularly useful is that the data is sent in plaintext between the two endpoints. This means the party wishing to receive the covert message does not need to operate or hijack the sever endpoint in order to receive the message. As long as they are able to capture the TLS negotiation packets somewhere along the path between client and server, the message can be recovered. Another benefit is that, if both the client and server endpoints are compromised by the covert message sending party, a bi-directional data stream can be used as both client and server hello messages contain random fields.

Use of this covert channel presents a slightly more difficult vector to the receiver. The client encrypts the premaster secret before putting it on the wire. This requires the receiver to either control the sever being used as an endpoint or have copied the servers private key needed to decrypt the message. However, the encrypted premaster secret can also work in the covert message sender's favor, as it makes it very difficult to examine the contents and discover the covert message.

III. PROCEDURE

A. Design

After deciding on the random fields as potential covert channels, several obstacles needed to be overcome to make the channels usable. The first was to find a way of generating SSL packets with their ClientHello random value field filled with our desired covert message. This task is more difficult than it would first seem. The packet can not be modified outside of the original sending applications SSL library. If one was to modify the random value in transit, the client and server would generate different master keys and the session setup would fail. Therefore it was necessary to modify the SSL library itself to inject the data into the ClientHello message.

Modifying the SSL code is complicated by the fact that almost all applications using SSL do not implement the protocol themselves but rely on a shared library. The most common of these are OpenSSL with open source and Unix software and Schannel in the Microsoft DotNet framework. These were quickly ruled out for the first round of testing as they are shared across the system and would be difficult to modify. Microsofts Schannel is not possible to use as it is closed source and can not be modified.

What was needed was a self-contained SSL implementation with source code available that could be easily implemented. Enter the Legion of the Bouncy Castle. Bouncy Castle is an implementation of several cryptographic functions, including SSL/TLS support, in operating system independent libraries. It was created to produce crypto libraries, in C and Java, that would be free of backdoors and government meddling [4]. The rest of the projects code was planned to be written in C (because of its ease of use and rich framework) and the Bouncy Castle API was well documented so it was chosen as the SSL library for the tests.

B. Testing

The first test was to see if both random fields could be exploited to send data without affecting the ability to create a valid SSL session. After reading over the code, the spots where the random fields were formulated were located. They were each one simple line of code which invoked a random byte generating function to get the required number of bytes and insert them into the packet structure. A static variable containing a preformatted message was added to each class and the random generator functions were replaced with a simple array copy that inserted the pre-created message into the field. Some additional code was written to open up an SSL session to a webserver, running on the local LAN, on HTTPS port 443.

With the test code written, the Wireshark packet capture tool was started and a display filter was added to show only SSL traffic. When the test code was executed, it successfully opened a connection to the server, negotiated a complete SSL session, than disconnected. When the packet capture was examined, both fields had successfully carried their messages without disrupting the proper operation of the protocol.

For the second test, the process was automated more on the sending side to allow a larger user defined message to be sent. The sending program was modified to take a target webserver hostname on the command line and ask the user to enter a message. Due to the way the Bouncy Castle library was implemented and the impossibility of getting the private key for a major website, the second test focused only on the ClientHello random field channel. To ensure proper message transmission and make the channel more useful, the message data is encoded into a simple packet format before being inserted into the header field. A library was written to do the process of encoding the data and returning an array of packets. The program then quickly opens and tares down sessions until all packets have been sent. The packets were retrieved the same way, using Wireshark packet captures. This test was also

successful and the message was decoded properly by hand on the receiving end.

C. Implementation

Although it needed to be simple, a few fields were needed to comprehensively create the covert message transmission format. The specific byte layout is shown below in Figure 1. The first field is the message identifier field, to allow this to be identified as a covert packet. The value chosen for the second test was cc. That is followed by a message ID number, to help the receiver distinguish between multiple messages from the same client. Since this test code only sends one message at once, it is always set to zero. Next is the packet type field. It is either set to one, for a normal data packet, or two for the final packet carrying an MD5 hash of the complete message. This is followed by a sequence number, incremented once for each packet in the message and a length field for how many of the remaining bytes are data and how many are zero padding.

Field	Length
CC Message Identifier	2 bytes
Message ID Number	1 byte
Message Type	1 byte
Sequence Number	2 bytes
Data Length	2 bytes
Data [padded with zeros]	variable

Fig. 1: Covert packet format

IV. BANDWIDTH

The throughput of this covert channel is highly variable. After leaving space for the overhead bytes, the Hello message channel can carry 20 bytes a session and the premaster secret can carry 38 bytes a session. The number of sessions that can be opened is limited more by the client then any physical limitation of the system. It would not be unusual to have many sessions opening and closing at once as modern websites often load a lot of dynamic content and update items on the page automatically every few seconds. With that in mind, opening and closing connections as fast as possible for extended periods would look highly suspicious. It is a balance between thoughput and stealth.

V. FUTURE WORK

The ground work that has been laid in this paper proves these two channels are usable as a method of covert communications. There is still a lot of work that would need to be done to turn the concept into a usable implementation however. The biggest weakness currently is the necessity of using Wireshark and manual packet reassembly to extract the message at the other end. A program would need to be written to log all SSL Handshake sessions which are visible to the packet sniffer then examine them for covert messages. Once the messages are extracted, it is a simple process to reassemble and they can be verified as complete, due to the included MD5 hash.

Another problem is creating a stealthy transmission client. The current one is simple because it was built for experimental use. If this channel was actually to be used in the field, the covert channel should be worked into a system library like OpenSSL or a legitimate application like Firefox. If the pre-master secret channel is to be used, a decoy server the message sending party controls needs to be set up or the private keys of a legitimate site would need to be exfiltrated. Without that private key, the channel is not visible to the observing packet capture program.

VI. CONCLUSION

TLS, the current successor protocol to SSL, is a ubiquitous part of the modern Internet landscape. While using SSL to hide covert messages in the Application data is fairly obvious, there are a few novel ways of sending covert messages using the pre-session negotiation handshake packets. Specifically, three random data fields that can be used to carry covert messages. Using a specially written program, otherwise normal looking TLS packets can be copied and the hidden messages decoded. This could present a useful vector for exfiltrating data from a network without arousing suspicion.

REFERENCES

[1] T. Dierks and E. Rescorla. The Transport Layer Security (TLS) Protocol Version 1.2. RFC 5246 (Proposed Standard), August 2008. Updated by RFCs 5746, 5878, 6176.

[2] Eu-Jin Goh, Dan Boneh, Benny Pinkas, and Philippe Golle. The design and implementation of protocol-based hidden key recovery. In Colin Boyd and Wenbo Mao, editors, *Information Security*, volume 2851 of *Lecture Notes in Computer Science*, pages 165–179. Springer Berlin Heidelberg, 2003.

[3] Butler W. Lampson. A note on the confinement problem. *Commun. ACM*, 16(10):613–615, October 1973.

[4] Legion of the Bouncy Castle Inc. Bouncy castle. https://www.bouncycastle.org/", 2013.

[5] Robertckl. "how does ssl work? what is an ssl handshake?". http://www.symantec.com/connect/blogs/how-does-ssl-work-what-ssl-handshake, 2014.

[6] Carlos Scott. Network covert channels: Review of current state and analysis of viability of the use of x.509 certficiates for covert communications. Technical Report 1, University of London, Egham, Surrey TW20 0EX, England, 1 2008.

SESSION

COMPUTER SECURITY + SECURITY APPLICATIONS

Chair(s)

Dr. Jing-Chiou Liou
Dr. Giovanni Luca Masala

Intrusion Detection in the Cloud Environment Using Multi-Level Fuzzy Neural Networks

H. Akramifard[1], **L. Mohammad Khanli** [1], **M.A Balafar**[1], **R. Davtalab**[1]
[1] Faculty of Electrical and Computer Engineering, Tabriz University, Tabriz, East Azerbayejan, Iran

Abstract - *Today virtualization is one of last innovations in computer's world. Enterprises are attempting to reduce their computing cost using virtualization. Cloud computing is ultimate response to this request of the market. Growth in the number of companies, who want to employ cloud resources, turns the user's data protection into a significant issue. Concentration of this paper is on the security of enterprise's data by intrusion detection while employing cloud computing. The goal of this research is recognizing the security threats and introducing a security method to mitigate them in the cloud computing environment. The intrusion detection will be responsible for anomaly detection on the generated data from the collected transactions through the cloud. The captured data will be classified using Multi-Level Fuzzy Neural Networks to detect the appearance of intruders on the cloud computing network. This approach will consider different attributes of the data to investigate the user's behavior. The evaluations show Multi-Level Fuzzy Neural Networks have more efficiency and better accuracy in intrusion detection.*

Keywords: Cloud computing, security, classification, intrusion detection, anomaly-detection, Multi-level Fuzzy Neural Networks.

1 Introduction

Nowadays cloud computing [1] provides computing and data storage services through the Internet. The cloud computing has scalability, elasticity and speed, etc. The internet started to offer meaningful bandwidth in the nineties. Cloud computing is a general term for anything that involves delivering hosted services over the Internet and managed by the cloud service provider. The Cloud services allow businesses and people to use software and hardware infrastructure that are managed by third parties at remote locations. The cloud services include online massive computing, file storage, social networking sites, webmail, and online business applications. The cloud computing provides remote access to information and computer resources from anywhere that an internet connection is available. Figure 1 shows architecture and layers of a cloud computing environment.

Incr Increasing in amount of cloud users, raises the privacy and Security concerns. Data protection became the major issue as the user's data managed by a third party [2]. We have to design and implement Intrusion Detection Systems (IDS) to detect the malicious activity on a cloud environment that could detect intruders and generate the alarms at the occurrence of any illegitimate activity. The intrusion detection systems train with the both normal and malicious data.

Garcia-Teodoro et al. [3] divides anomaly detection techniques as below:

1. Statistical based
    a) Univariate
    b) Multivariate
    c) Time series model
2. Knowledge based
    a) Finite State Machines
    b) Description languages
    c) Expert systems
3. Machine learning based
    a) Bayesian networks
    b) Markov models
    c) Neural networks
    d) Fuzzy logic
    e) Genetic algorithms
    f) Clustering & outlier detection

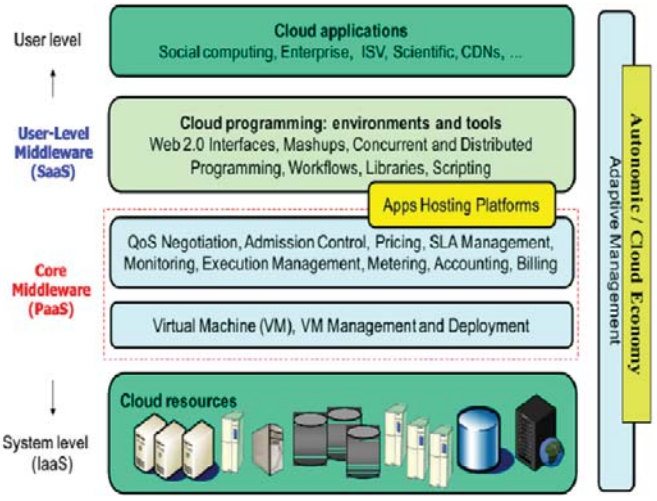


Fig. 1. Cloud Computing Architecture and its layers [4].

Section 2 represents the intrusion detection and related works in the cloud. In section 3, focus is on Multi-Level Fuzzy Neural Networks. An overview of the concepts of intrusion detection in cloud using MLF-NN is mentioned in section 4. In section 5 the proposed method and the evaluation are

explained. Finally the conclusion has been presented in section 6.

2 Intrusion Detection In Cloud

Recognizing malicious activities against the networking resources is known as intrusion detection. The recognition of any suspicious activity on the devices or networks is raised by an alert [5]. An Intrusion Detection System (IDS) in a cloud computing environment is for protecting each VM against the threat of malicious accesses. An Intrusion Detection System is a program that monitors the events at a machine or at a network automatically. It monitors the traffic at each machine, also monitors the network and makes records, to provide security to all the devices in the network [5], [6]. Environment of an IDS one of following groups [7]:

1) Host-based Intrusion Detection System:
   It monitors a specific host to detect if any program accesses some resources, it acts like a firewall.
2) Network-based Intrusion Detection System:

It monitors the network packets for specific network segments or points to recognize any suspicious action.

2.1 Categories of IDS

Intruder identification is one of the basic IDS operations [8]. The two main identified methods of IDS [5], [6] are as bellow:

1)Misuse-based Detection:

A misuse-based Intrusion Detection System stores signatures depicting attacks into a database. Signature of such attacks widely used systems where security threats are common. The pattern (signature) based IDS performs a depth inspection of the packets, for any spiteful patterns in the load or header.

2)Anomaly-based Detection:

An anomaly-based Intrusion Detection System protects a statistical model of custom patterns, patterns that describe the normal behavior of monitored users [5]. At the first training stage of this Intrusion Detection System, a similarity metric is used to compare an input with the normal model, then generates alerts for large deviation values.

Misuse-based IDS look like efficient and effective, but it shows two main problematic conditions, one, mistakes in detection of unknown attacks [9], [10], and second, pattern analysis defect. The first is due to the fact that misuse-based IDS relies on string comparison of previous attack patterns [11], thus the unknown attacks can show deviation from the comparison string to already known attacks, and thus are ignored from being detected, that is false negatives [12], and second, misuse-based IDSs have weakness in pattern analysis, and in rule writing methods as to capture all the defenselessness of attacks it mainly relies on the human ability.

We distinguish misuse and anomaly based principles mainly in the way of modeling of their behavior and way of defining of their normalcy. Those two method specify how is further processing of observed data too [13], [14].

Here is a short history of some related work about intrusion detection in literature:

Massimo Meneganti et al. used fuzzy logic for classification and detection of anomalies firstly in 1998 [15]. They utilized fuzzy neural networks to find anomalies in the cooling system of a blast furnace.

Pei-Te Chen et al. proposed the concept of security auditors, to discover the system weaknesses and modify the tested packets using fingerprints that can be detected and recognized by IDS in 2007, [16].

Jun-Ho Lee et al. proposed a multi-level method for IDS [17] in cloud computing system in February 2011. In their method all the users were bound to a security system on the basis of anomaly status. The system decides about anomalies on user's IP coverage, amount of ID/password failures, vulnerable ports, and etc. In June 2011, an intrusion detection model baseed on anomaly in an environment of SaaS application was presented by Gustavo Nascimento et al. [5].

Chirag N. Modi et al. integrated a Network based IDS system in Cloud that offered IaaS to detect network attacks in July 2012 [18]. In this system, they used Bayesian classification method with Snort. This module guarantees low false positives and negatives with acceptable cost. Ajeet Kumar Gautam et al. proposed a hybrid intrusion detection system in cloud computing, they used KFSensor and anomaly based IDS via FlowMatrix with honeypot technology, in 2012 [6]. They designed an architecture by providing and detecting various attacks. In September 2012, Amirreza et al. introduced a Cloud Intrusion Detection System Service (CIDSS) [19] to overcome the crucial challenge of securing the client from cyber-attacks. Three primary components of CIDSS:

1) A Service Agent for Intrusion Detection.
2) A Service Component (CCSC).
3) An Intrusion Detection Service Component (IDSC) that were used to incorporate information and after that test them.

In 2013, Ahmed Patel et al. proposed a model of Intrusion Detection and Prevention system (IDPS) in cloud computing [7]. The concepts of fuzzy theory, autonomic computing, risk management, and ontology were grabbed and combined acknowledge the requirements of an IDS. In that year P. Gupta et al. proposed behavior based IDS [20], the implementation was instructed in a real cloud IaaS environment. The framework was tested with NIDS to detect network based attacks.

In 2014, Harshit Saxena et al. proposed an intrusion detection system using K-means, PSO with SVM classifier [21] to detect various attacks at network. They has tried to design an IDS that is trained on the basis of Particle Swarm Optimization, executed on the KDD data.

3 Multi-Level Fuzzy Neural Networks

Here we will introduce two efficient types of fuzzy neural network, Fuzzy min-max neural network (FMM) and Multi-Level Fuzzy Min-Max Neural Network (MLF).

3.1 Fuzzy min-max neural network

Fuzzy min-max neural network (FMM) is a machine learning method that has been proposed by Simpson in 1992 [22]. It can be used for data classification. The learning phase includes only one pass over the learning data. In this method we use convex hyperboxes in the pattern space. Each hyperbox is determined by MN and MX points which, respectively, mention the min and max points of the hyperbox. A three-dimensional hyperbox has been shown in figure 1.

Each hyperbox covers a part of pattern space, and belongs to only one of the classes but can include more than one sample of that class, defined as (1) [23].

$$B_i = \{X, MN_j, MX_j, f(X, MN_j, MX_j)\} \qquad \forall X \in I^n \quad (1)$$

Where MN_j and MX_j are min and max corners of the hyperbox. X is an input vector, and n mention number of dimensions. Each class may have one hyperbox or more. Hyperboxes of the same class could overlap each other, but hyperboxes from different classes couldn't. Final hyperboxes of an example of FMM network in a 2-D binary classification have been shown in figure 2.

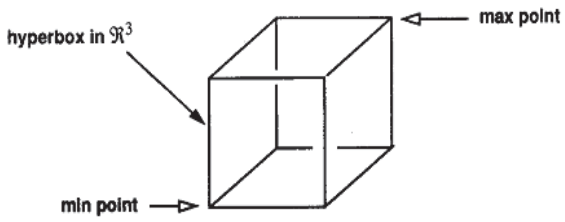


Figure 2. 3-D hyperbox and its min and max points [23].

Fuzzy set, that inputs classes belong to, includes union of hyperboxes of those classes. In the test stage, these hyperboxes and their membership function are used to detemine the classes. In this method, the size of hyperboxes is in range of [0, 1]. One of the possible membership functions is Simpson function is shown at (2):

$$b_j(X_h) = \frac{1}{2n}\sum_{i=1}^{n}\left[max\left(0, 1 - max\left(0, \gamma\, min\left(1, a_{hi} - mx_{j\,i}\right)\right)\right) + max\left(0, 1 - max\left(0, \gamma\, min\left(1, mn_{ji} - a_{hi}\right)\right)\right)\right] \quad (2)$$

Where $\boldsymbol{X_h} = (x_{h1}, x_{h2}, \ldots, x_{hn}) \in I^n$ is the *h*th sample and γ is in range of [0, 1] that determine how fast the membership values decrease as the distance between X_h and B_j increases. Figure 3 illustrate an example of two-classes fuzzy min-max hyperboxes, without overlapping between the classes.

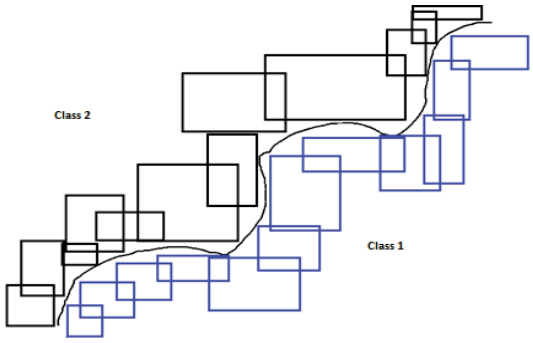


Figure 3. Final hyperboxes [23].

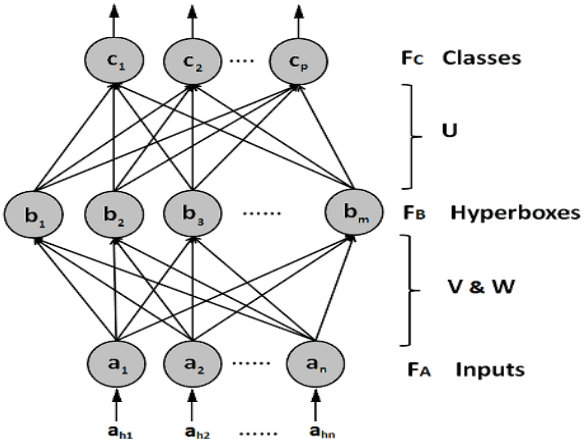


Figure 4. Structure of the classic FMM [23].

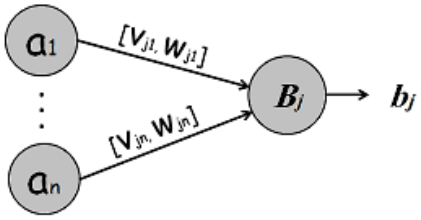


Figure 5. Details of a hyperbox [24].

FMM neural networks have three layers as shown in Figure 4, the first, input layer (FA), the second layer represents hyperboxes (FB), and third layer represents classes of each node (FC). Also each hyperbox locate in middle layer (FB), and the membership function of this hyperbox is the transition function of the Correspond node. Figure 5 demonstrate a hyperbox in details. Each node of input layer is connected to all nodes in the middle layer and each of these links has two weights (*Vji* and *Wji*), which are, respectively, the min and max points of the B_j hyperbox, and *i* is the index of the nodes in the first layer. Each node of the middle layer is also connected to all nodes in the output layer. Weights of those links are obtained from (3), and FC nodes outputs are provided by (2):

$$u_{ij} = \begin{cases} 1, & if\ b_j \in C_i \\ 0, & if\ b_j \notin C_i \end{cases} \quad (3)$$

All hyperboxes are created and adjusted in the learning step. The learning phase has three parts. Existance of a box that belongs to the same class and simultaneously the sample is in box area, will be checked for per sample (A_h). If a box is found, then no further processing is required and training goes on with the next sample. If there is no such hyperbox, following three steps are executed [24].

1. Expansion: In this stage, a hyperbox must be found to display the related class and also be capable of expansion to cover the input sample, the hyperbox size is limited to the θ parameter. If no such hyperbox is found, a new hyperbox is created with min and max points, relevant to this sample.
2. Overlap Test: In this step, the overlapping area of the extended hyperbox will check for all hyperboxes that belong to the other classes. In one case of (4), we can find overlap of two hyperboxes, after recognizing each dimension. To eliminate this overlap, the dimension Δ that has the least overlap will be selected for contraction.

Case 1: $v_{ji} < v_{ki} < w_{ji} < w_{ki}$

Case 2: $v_{ki} < v_{ji} < w_{ki} < w_{ji}$

Case 3: $v_{ji} < v_{ki} < w_{ki} < w_{ji}$

Case 4: $v_{ki} < v_{ji} < w_{ji} < w_{ki}$ (4)

For example of case 1: Min of 1st box (v_{ji}) less than min of 2nd box (v_{ki}), min of 2nd box (v_{ki}) less than max of 1st box (w_{ji}), max of 1st box (w_{ji}) less than max of 2nd box (w_{ki}), figure 6 shows the visualization of this case.

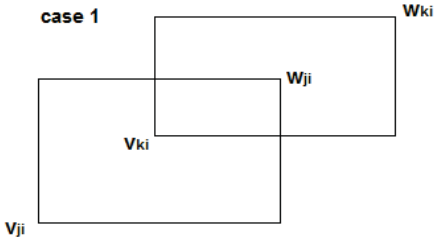


Figure 6. Illustration of contraction for case 1.

3. Contraction: If there is no overlap, this step is not necessary; else, considering the type of the overlap according to (4), one case of (5) will be executed.

Case 1: $v_{ji} < v_{ki} < w_{ji} < w_{ki}$

$v_{k\Delta}^{new} = w_{k\Delta}^{new} = \frac{v_{k\Delta}^{old} + v_{j\Delta}^{old}}{2}$ OR

$w_{j\Delta}^{new} = v_{k\Delta}^{old}$

Case 2: $v_{ki} < v_{ji} < w_{ki} < w_{ji}$

$v_{j\Delta}^{new} = w_{k\Delta}^{new} = \frac{v_{j\Delta}^{old} + v_{k\Delta}^{old}}{2}$ OR

$v_{j\Delta}^{new} = w_{k\Delta}^{old}$

Case 3: $v_{ji} < v_{ki} < w_{ki} < w_{ji}$

If $w_{k\Delta} - v_{j\Delta} < w_{k\Delta} - v_{j\Delta}$ Then

$v_{j\Delta}^{new} = w_{k\Delta}^{old}$

Else

$w_{j\Delta}^{new} = v_{k\Delta}^{old}$

Case 4: $v_{ki} < v_{ji} < w_{ji} < w_{ki}$

If $w_{k\Delta} - v_{j\Delta} < w_{j\Delta} - v_{k\Delta}$ Then

$w_{k\Delta}^{new} = v_{j\Delta}^{old}$

Else

$v_{k\Delta}^{new} = w_{j\Delta}^{old}$ (5)

Here, Δ denotes the selected dimension. These three steps are executed on every learning sample to obtain the required hyperboxes.

3.2 Multi-Level Fuzzy Min-Max Neural Network

In this article we will use multi-level fuzzy min-max neural network for IDS classification. This type of neural networks tries to better cover area of classes using more precise and smaller hyperboxes. Despite of classic FMM method, the contraction step do not handle the overlaps. The manner of MLF method is creation of hyperboxes in the first and the second levels, and the classification task are illustrated in figure 7.

Each node in the network of the MLF method is known as a subnet and is an independent classifier that classifies samples that belong to the defined region of pattern space. The first level classifier classify most of the region of pattern space, and the second level nodes take care of the remaining regions that are the same overlapped region of root subnet, as well each node in the ith level of the network classifies patterns of overlapped region in i−1th level of the network. Finally, the node that has the best output will select as the network's output.

In MLF, like in other FMM methods, all hyperboxes are created and adjusted during training phase and are used in test phase. FMM method handle overlap problem step by step just when an overlap is created; but in MLF overlap handling is done after creation and adjustment of all hyperboxes. This can reduce space and time complexity.

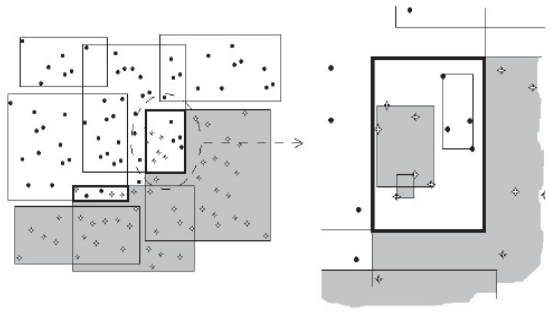


Figure 7. Different levels of classification in MLF [24].

4 Intrusion Detection In Cloud Using MLF-NN

For intrusion detection in this paper we proposed the concept of Multi-Level Fuzzy Min-Max Neural Network algorithm. Using MLF-NN we will classify criminal activities like unauthorized access and change in behavior of the user. This is accomplished by using data in the database. The algorithm design will be as follows:

START

1) Obtaining topology of devices on cloud environment.
2) Obtaining data set of the transaction through the running virtual machines.
3) Select the data to be investigated and normalize it on the basis of various attributes.
4) Training the MLF-NN using obtained dataset.
5) If classifier recognizes the activity as an attack, generate an alarm, else the user is genuine and no intrusion is detected.

END

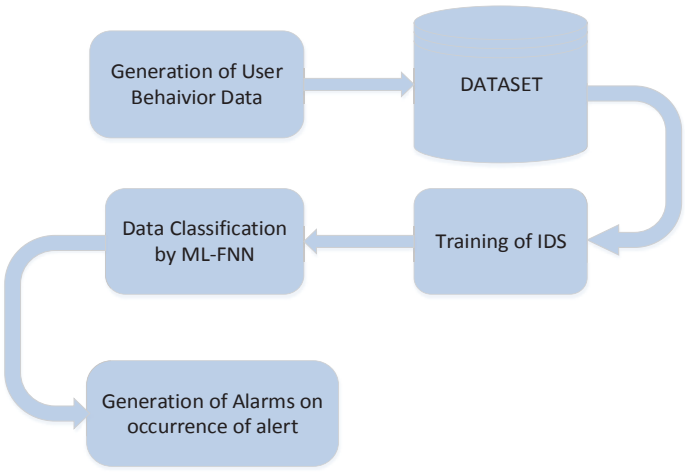


Figure 8. Architecture of intrusion detection using ML-FNN.

The proposed method comprises of three major steps of intrusion detection:

Step One: Data Generating

In this step a topology will be configured on an emulator. It lets the network to act on a virtual machine with the cloud environment. The operations will be logged for further observations.

Step Two: Dataset Making

In the second step generated data will be gathered. In this phase we will train the proposed IDS with the user behavior. The attributes types of the user's behavior in the cloud network are:

1. Basic features of individual TCP connections.
   a) Duration
   b) Protocol type
   c) Same host or not
   d) Number of data bytes from destination to source
   e) Number of data bytes from source to destination
2. Content features within a connection suggested by domain knowledge.
   a) Number of failed login attempts
   b) Success of login
   c) Number of "compromised" conditions
   d) Number of file creation operations
   e) Number of shell prompts
3. Traffic features computed using a two-second time window.
   a) Number of connections to the same host as the current connection in the past two seconds
   b) "SYN" errors
   c) "REJ" errors
   d) Number of connections to the same service as the current connection in the past two seconds
   e) Connections to different hosts
   f) Percent of connections to the current host having the same src port
   g) Percent of connections to the same service coming from different hosts
   h) Percent of connections to the current host that have an S0 error
   i) Percent of connections to the current host and specified service that have an S0 error
   j) Percent of connections to the current host that have an RST error
   k) Percent of connections to the current host and specified service that have an RST error

Step Three: Detecting

The final step will be the data analysis step where the data will be classified. Classifying is done with MLF-NN over a dataset. The result includes 21 attributes and two classes.

5 Evaluation Of Intrusion Detection Using MLF-NN

In this article the experimental data are from KDD dataset. We randomly select two groups without overlap from the data set; respectively denote them as INP and OUTP. INP uses to training and OUTP uses to test the model. There are intruders and normal users in the data set, we simulate the behavior of these two types of users for validating the ability of the model to identify the two types of users. Risk users' behaviors are normal in most cases, but they may be abnormal in some moments. The experiment simulate the behaviors by sending a large number of HTTP requests at a time, their behavior is similar to malicious users, but their attack time length is short.

We analyze the users' behaviors based on the data of INP obtain the evidences of users' behaviors. Behavior evidence including:

- ✓ Environmental attributes, such as network throughput, transmission delay, and IP loss ratio.
- ✓ Operational attributes, such as number of hits, pages accessed, important pages accessed, and time on page.

The environmental attributes principally used to determine the safety of the user's network environment. But operational attributes are mainly used to conclude consistency of user's behavior with his habits. About 95% of the users' behaviors are concentrated in the stable range [25]. User behavior hierarchical structure is shown in figure 9.

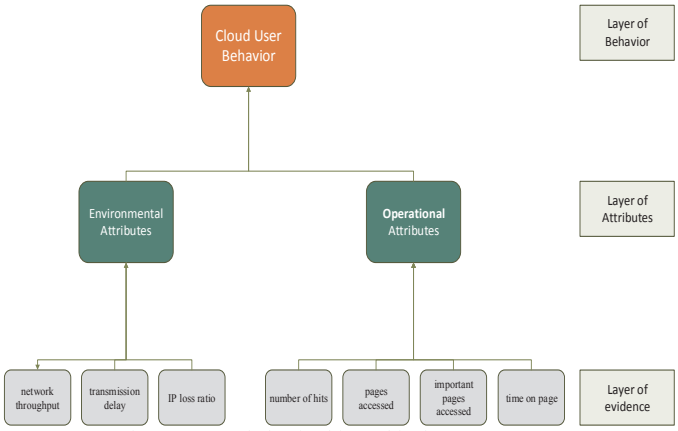


Figure 9. User behavior hierarchical structure [25].

The model have been tested by data set OUTP using MLF-NN and compare the model with the other classification methods. There are 25973 samples in OUTP, and the number of sample of attacks are 12075 and normal behaviors are 13898.

In anomaly detection, True Positive or Detection Ratio (DR) and False Positive Ratio (FPR) are two essential metrics. Here, the DR mainly mentions the amount of detected intrusions, and FPR mentions the false positives of recognized users as intruder. The DR and the FPR of the model using this model and comparing it with other classification methods, have been shown in figure 10 and figure 11.

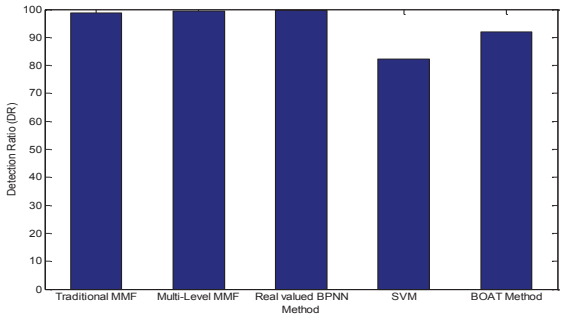


Figure 10. The DR of the four method.

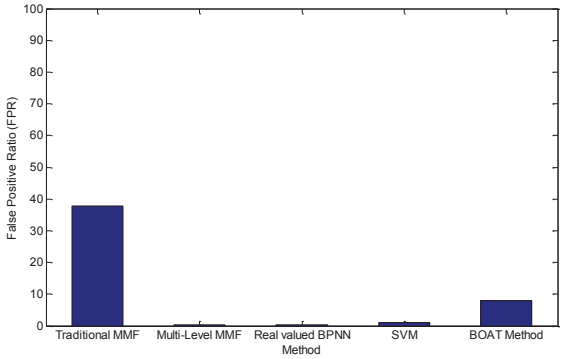


Figure 11. The FPR of the five methods.

As has been shown in figure 10 the Real valued BP-NN and MLF-NN respectively have 99.64% and 99.60% accuracy in DR and as has been figure 11 two above classification methods have 0.4% and 0.38% FPR, it shows the accuracy of the proposed model is better than other models like Bootstrapped Optimistic Algorithm for Tree Construction (BOAT) method came out to be 92.02% DR and 7.98% FPR.

Despite of near optimal ratio in detection and false positive, we chose MLF-NN over Real valued BP-NN because, because if there is some new data and if we want to train the network in Real valued BP-NN we must train all network, but using MLF-NN we can learn only new data to the network without changing all previous network trained data. Figure 12 has been represented comparation the Accuracy, Precision, Recal, and F-Score between four classification methods on proposed model. As we can see MLF-NN has the best result in comparision with other methods.

6 Conclusion

In this paper, we have presented a solution that detects malicious activities that masquerade in the system with the aim of violating the information. This solution uses MLF-NN to learn the behavior pattern of the user to detect malicious user in the system. The proposed solution proved to be effective in terms of reducing false positives rate and false negatives rate. The reduction of false positive and false negative rate indicates that, there is increasing in detection rate of intrusions. The results show that malicious users can be detected based on their behavior patterns.

7 References

[1] Anthony T. Velte, Toby J. Velte, Robert Elsenpeter, "Cloud Computing – A Practical Approach", Tata McGrawHill Edition, ISBN: 978-0-07-162695-8.

[2] Mell, Peter, and Tim Grance. "Effectively and securely using the cloud computing paradigm." NIST, Information Technology Lab 2009.

[3] P Garcia-Teodoro, J Diaz-Verdejo, G Macia-Fernandez, and E Vazquez. Anomaly-based network intrusion detection: Techniques, systems and challenges. Computers & Security, 28 (1 -2):18-28, 2009. doi: 10.1016/j.cose.2008.08.003.

[4] Rodrigo N. Calheiros, Rajiv Ranjan, Anton Beloglazov, César A. F. De Rose, Rajkumar Buyya, "CloudSim: A Toolkit for Modeling and Simulation of Cloud Computing Environments and Evaluation of Resource Provisioning Algorithms", Software: Practice and Experience, Volume 41, Issue 1, pp. 23–50, January 2011.

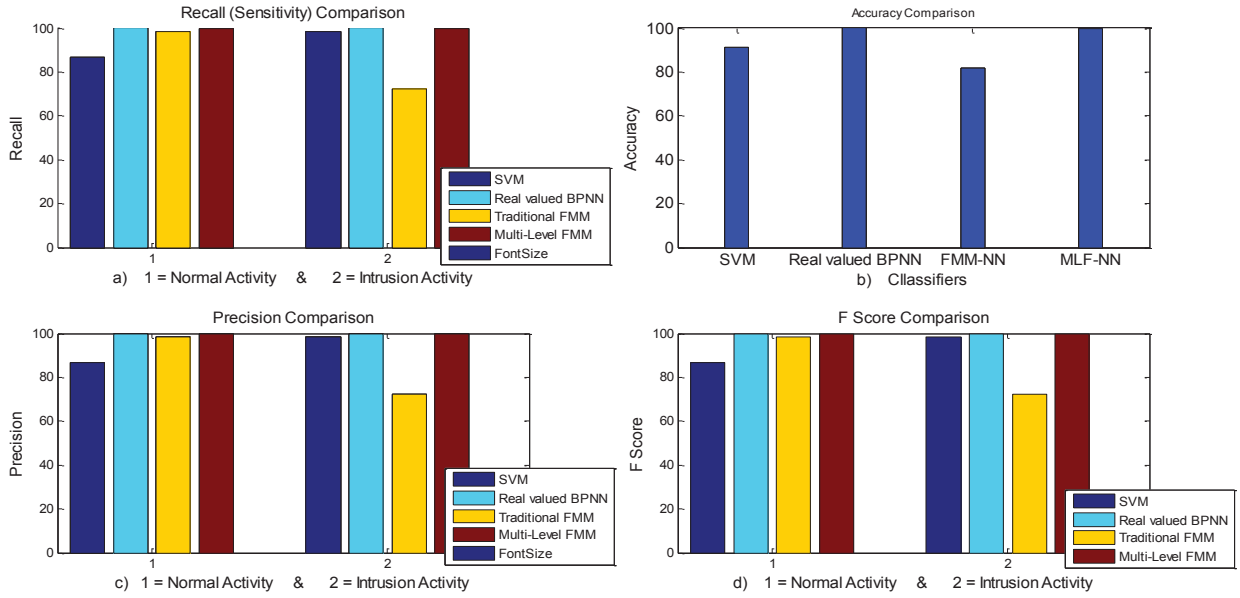


Figure 12. comparision between methods: a)Recal, b)Accuracy c)Precision d)F-Score.

[5] Nascimento, G., Correia, M., "Anomaly-based intrusion detection in software as a service", Dependable Systems and Networks Workshops (DSN-W), IEEE/IFIP 41st International Conference on, pp.19-24, June 2011.

[6] Ajeet Kumar Gautam, Vidushi Sharma, Shiva Prakash, "An Improved Hybrid Intrusion Detection System in Cloud Computing", International Journal of Computer Applications, Volume 53– No.6, pp. 1-13, September 2012.

[7] Ahmed Patel, Mona Taghavi, Kaveh Bakhtiyari and Joaquim Celestino Júnior, "An Intrusion Detection And Prevention System In Cloud Computing: A Systematic Review", Journal of Network and Computer Applications. Volume 36, Issue 1, pp. 25 -41, January 2013.

[8] Ms Deepavali P Patil, Prof.Archana C.Lomte, "Implementation of Intrusion Detection System for Cloud Computing", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 11, November 2013.

[9] David J. Day, Denys A. Flores, Harjinder Singh Lallie, "CONDOR: A Hybrid IDS to Offer Improved Intrusion Detection", IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications, pp. 931-936, 2012.

[10] Choudhury, A.J.; Kumar, P.; Sain, M.; Hyotaek Lim; Hoon Jae-Lee, "A Strong User Authentication Framework for Cloud Computing", Services Computing Conference (APSCC), IEEE Asia-Pacific, pp.110-115, December 2011.

[11] Tupakula, U, Varadharajan, V., Akku, N., "Intrusion Detection Techniques for Infrastructure as a Service Cloud", Dependable, Autonomic and Secure Computing (DASC), IEEE Ninth International Conference on , pp.744-751, December 2011.

[12] Hari Om, Aritra Kundu, "A Hybrid System for Reducing the False Alarm Rate of Anomaly Intrusion Detection System", In Proceedings of 1st Int'l Conf. on Recent Advances in Information Technology (RAIT-2012),IEEE, pp. 131-136, 2012.

[13] Herve Debar, Marc Dacier, and Andreas Wespi. Towards a taxonomy of intrusion-detection systems. Computer Networks, (October 1998), 1999. Anita K. Jones and Rrobert S. Sielken. Computer system intrusion detection: A survey. Computer Science Technical Report, pages 1- 25, 2000.

[14] Aleksandar Lazarevic, Levent Ertoz, Vipin Kumar, Aysel Ozgur, and Jaideep Srivastava. A comparative study of anomaly detection schemes in network intrusion detection. Proceedings of the . . . , pages 25-36, 2003.

[15] Massimo Meneganti, Francesco S. Saviello, and Roberto Tagliaferri, "Fuzzy Neural Networks for Classification and Detection of Anomalies", IEEE transactions on neural networks, vol. 9, no. 5, pp. 848-861 september 1998.

[16] Pei-Te Chen, Chi-Sung Laih, "IDSIC: an intrusion detection system with identi fication capability", Springer-Verlag, pp.185-197, June 2007.

[17] Jun-Ho Lee; Min-Woo Park; Jung-Ho Eom; Tai-Myoung Chung, "Multi-level Intrusion Detection System and log management in Cloud Computing", Advanced Communication Technology (ICACT), 13th International Conference on , pp.552-555, February 2011.

[18] Chirag N. Modil, Dhiren R. Patell, Avi Patel, Rajarajan Muttukrishnan, "Bayesian Classifier and Snort based Network Intrusion Detection System in Cloud Computing", Computing Communication & Networking Technologies (ICCCNT), 2012 Third International Conference on, pp. 1-7, July 2012.

[19] Amirreza Zarrabi and Alireza Zarrabi, "Internet Intrusion Detection System Service in Cloud", International Journal of Computer Science Issues, Vol. 9, Issue 5, No. 2, pp. 308-315, September 2012.

[20] Punit Gupta, Deepika Agrawal, "Behavior Based IDS for Cloud IaaS", International Journal of Software and Web Sciences (IJSWS), pp. 31-36, June-August 2013.

[21] Harshit Saxena, Dr. Vineet Richariya, "Intrusion Detection System using K- means, PSO with SVM Classifier: A Survey", International Journal of Emerging Technology and Advanced Engineering, Volume 4, Issue 2, pp. 653-657, February 2014.

[22] A. Joshi, N. Ramakrishman, E. N. Houstis, and J. R. Rice, "On neurobiological, neuro-fuzzy, machine learning, and statistical pattern recognition techniques," IEEE Trans. Neural Netw., vol. 8, no. 1, pp. 18–31, Jan. 1997.

[23] P. K. Simpson, "Fuzzy min-max neural networks. I. Classification," IEEE Trans. Neural Netw., vol. 3, no. 5, pp. 776–786, Sep. 1992.

[24] Reza Davtalab, Mir Hossein Dezfoulian, and Muharram Mansoorizadeh, "Multi-Level Fuzzy Min-Max Neural Network Classifier", IEEE transactions on neural networks and learning systems, vol. 25, no. 3, march 2014.

[25] Tian Junfeng, Cao Xun "A Cloud User Behavior Authentication Model Based On Multi-partite Graphs", IEEE, Innovative Computing Technology (INTECH), 2013 Third International Conference on , pp. 106-112, 2013.

An Overview of the Current Classification Techniques in Intrusion Detection

Buthina Al-Dhafian, Iftikhar Ahmad, Abdullah Al-Ghamid
Software Engineering Department, King Saud University
Riyadh, Saudi Arabia
b.r.al.dhafian@gmail.com, wattoohu@gmail.com, ghamdi@ksu.edu.sa,

***Abstract*—During the last decade, a lot of attention has been given to intrusion detection systems (IDSs) as another security tools used to detect attacks and make working in computer systems and network more efficient and stable. However, the current challenge in these systems is consider into which is an optimal classification technique that must be used to detect intrusion in high level of accuracy. Many classification techniques have been designed in IDSs to detect attacks, where the accuracy of IDSs depends mainly on them. Numbers of studies have been proposed to enhance the performance of IDSs by increased the detection rates (DR) and decreased the false alarms rats (FAR). In this paper, we present a review of the current classification techniques that are used during designing IDSs. We also provide a review of the current dataset that are used to train and test selected classifier. The main goal of our research is to provide a review of the current classification techniques in intrusion detection in order to enhance the performance of classifier by highlighting different issues, which need to be solved. This paper seeks to help the researchers to develop an optimal classification technique by eliminating the issues that reduces the accuracy of IDSs.**

Keywords—Intrusion detection; Intrusion detection systems; Classification techniques; Datasets

I. INTRODUCTION

Nowadays, the basic design of security in computer systems and network has been changed due to the huge number of attacks, which appeared because the increased in the numbers of internet users. Despite the existing tools of security systems, which protect these attacks such as firewalls, antivirus, and data encryption, it still hard to ensur that computer systems and network will be free of security flaws. IDSs are emerged as another technique which have increased the tools of security systems to monitor, identify, and detect intrusions in high level of accurecy [1].

The machanism of IDSs depends on the observation to classfiy data into normal or abnoraml behavoir. When an intrusion or suspect pattern is observed, an alarm is activated in order to take measures to maintain the integrity of the system [2]. Many IDSs have been designed to detect intrusions, although, maintaining the accuracy considered the main issue, which is mainly depends on optimal classifier selection [3]. Find the optimal classification techniques in IDSs considered as a critical issue as well, such that each classifier required to be trained with sample data in order to recognize the patterns and then it is tested with other samples in order to be efficient enough to perform well. Moreover, select the appropriate datasets, which are used in the testing and the training process considered as a dilemma [4], which is used to evaluate IDSs. Therefore, a number of studies have been proposed using multiple of classification techniques in order to design IDSs with high level of accuracy. The main goal of our research is to present a review of the current studies that aim to improve classification in IDSs by increasing the DR and decreasing the FA using optimal classifier technique. Moreover, to highlight different issues that need to be solved during develop an optimal classification technique in IDSs.

Besides this introductory section, the remaining of this paper is organized as follow. A background study in IDSs is given in section II. A review of current classification techniques in IDSs is presented in section III. An overview of the most popular standard datasets used in IDSs are explore in section IV. A comparative analysis is discussed in section V. Finally, conclusion is drown in section VI.

II. BACKGROUND

An intrusion can be described as any event that violates systems security, occurs by an intruder. An intrusion detection (ID) considered as one of the security systems' tools, which is used to detect intrusion in computer systems and network based on the hypothesis that the behavior of an intruder and a legitimate user vary from each other. IDSs have appeared to deal with vulnerabilities of systems security, where they are designed as complementary rather than alternative tools to these systems. The first concept of ID appeared early 1980 [5], which emphasized on a single computer system followed by the actual work done by Denning in 1987 [6] at SRI International, where intrusion detection is extended to address multiple computers in a distributed system. Few years later, many IDSs ware proposed to serve on both sides of researches and commercial world. However, they have operated based on general architectural framework [1], [2]. The main component of this architecture is a detector (also known as analysis engine), which is responsible for classifying data to normal or abnormal behavior, and considered the basis to determine the accuracy of IDSs.

IDSs work to identify the suspect behavior based on three types of detection methodologies [2], [7]: (i) Misuse-based detection (MD), which is used mainly to detect intrusions according to the predefined pattern of known attacks, the accuracy of this type considered good and theoretically, it has a very low FAR, however it cannot detect new attacks. (ii) Anomaly-based detection (AD), where detect intrusions based on a reference model of the normal behavior of the monitored

system, it has the capability to detect unknown attacks, but the FAR it very high. (iii) Stateful protocol analysis (SPA), which detects intrusion based on predetermined profiles of accepted definitions of normal protocol activity for each protocol state, it differs from AD as it depends on vendor-developed universal profiles that determine how particular protocols work. These types of detection methodologies can perform separately or integrated into one system, known as hybrid IDS [8], which constructed to avail from multiple approaches and overcame many of the issues by producing a much stronger IDS.

On the other hand, IDSs have many types of technologies, which can be categorized based on the scope of detection. These categories are [7], [9], [10]: (i) Host-based IDSs which analyze the activities that flow into the host to identify attacks, (ii) Network-based IDSs which analyze network packets that come from the outside to detect attacks, (iii) Application-based IDSs a partial set of HIDS, responsible for monitoring and analyzing the activities that have took place inside a software application. (iv) Wireless IDSs [11] which monitor and analyze the protocols of wireless networks to identify shady activities, and (v) Network Behavior Analysis (NBA) IDSs which inspect network traffic or statistics on network traffic to identify suspicious behavire. Some of these technologies can be adopted with each other to be known as Mixed IDS (MIDS) with a view to improve DR and make systems as much as possible free from attacks.

III. CURRENT CLASSIFICATION TECHNIQUES IN INTRUSION DETECTION

In literature, numerous of studies have applied different classification techniques to design IDSs; some studies have deigned IDSs by using single techniques (such as neural network, fuzzy techniques, support vector machines, etc...), and the other hand, some studies have deigned IDSs based on combining different techniques (such as hybrid or ensemble techniques). A brief overview of the current classification techniques in IDSs is listed below:

A. Approach-1

Tong et al. [8] have been proposed a hybrid RBF/Elman neural network model to be used for both AD and MD, which can efficiently detect temporally dispersed and collaborative attacks. They used a radial basis function (RBF) network as a real-time pattern classification and they applied the Elman network to restore the memory of past events. Their model takes an output of RBF as input of Elman network, while an Elman network restore each output of RBF network by keep memory of past misuse events. For their experiments, they have used DARBA dataset. The results showed that, their model can detect intrusions with higher RD and lower FPR compared with other IDSs that used neural network techniques. Additionally, the ability to determined DOS and probing attacks in IDSs is enhanced.

B. Approach-2

A new approach, which called FC-ANN have been proposed by Wang et al. [3] with the aim to enhance detecting precision for low-frequent attacks, detecting stability as well as achieving higher DR and lower FPR. Feed-Forward neural network (FFNN) and Fuzzy c-means clustering are used for designing their approach, which designed based on the following three phases. In the first phase, a fuzzy clustering technique is used to generate different training subsets to reduce the size and complexity, while in the second phase different ANNs are trained based on different training sets. Finally, in the last phase, a meta-learner, fuzzy aggregation module, is introduced to learn and combine the different ANN's results in order to eliminate the errors of different ANNs. For their experiment, KDD CUP 1999 dataset were used for the evaluation purpose. The results have demonstrated that, their approach has the effectiveness especially for low-frequent attacks, i.e., R2L and U2R attacks in terms of detection precision and detection stability.

C. Approach-3

A SVM-based intrusion detection system has been proposed by Horng et al. [12] with the aim of shorten the training time as well as to improve the performance of SVM classifier. Their approach combines three methods, which are a hierarchical clustering algorithm, a simple feature selection procedure, and the SVM technique. KDD Cup 1999 dataset is used for their experiments for evaluation purpose. First, they used BIRCH clustering algorithm to transform the KDD Cup 1999 dataset to a smaller sized dataset. Then, they trained SVM classifiers based on the reduced dataset with abstract data points. Finally, they used "leave-one-out" procedure in order to remove unimportant features from training set. The results have showed that the proposed system had the best performance to detect intrusion, practically; it had superior performance in the detection DoS and Probe attacks.

Approach-4

Ahmad et al. [13] have been proposed an intrusion detection system in order to overcome performance issues by using feature subset selection based on multilayer Perceptron (MLP). In their approach, Principle Components Analysis (PCA) are used for features transformation, while Genetic Algorithms (GA) are applied for search the principal feature space for a subset of features. For classification purpose, they used MLP. KDD cup 1999 was used for their experiments. Their approach was an initial effort for features subset selection in intrusion detection, which aims to find a subset of principle components by using GA to search the PCA space. The results showed that, the proposed approach have improved accuracy, simplified the architecture of intrusion detection, as well as decreased training time and computational overheads.

D. Approach-5

Chitrakar and Chuanhe [14] have proposed a new hybrid approach, which aims to make the classification operate in anomaly-based IDSs more accurate and efficient. They used K-Medoids Clustering approach to gathering similar data instances based on their behavior and applied SVM to classify data to normal or abnormal behavior. Their experiments have been evaluated based on Kyoto2006+ datasets.

The main procedure in their approach is converted the attribute of datasets into suitable types, and then normalized

them based on the need of the SVM kernel. The entire selected data are classified into [-1, 1], where -1 represents each sample with both known and unknown attack and 1 represents normal class. The experimental result referred to that, their approach has increased DR as well as decreased FPR in in superior level compared to the other hybrid approach.

E. Approach-6

A hybrid intrusion detection system for MD and AD have been proposed by Om and Kundu [15], which combines K-Means and two classifiers K-Nearest Neighbor (K-NN) and Naïve Bayes. KDD-Cup 1999 dataset are used for evaluation purpose. First, the entropy based feature selection algorithm is used to select the appropriate features. Then, k-means clustering algorithm is applied on the selected features to split the data records into normal and abnormal clusters. After that, the obtained data are classified into normal or abnormal clusters by using the hybrid classifier. The main goals in their approach were to reduce the FAR, detect the intrusions, and further classify them into four categories: DoS, U2R, R2L, and probe. As a result, they have found that the proposed approach is better than the other conventional approaches such as kMeans, kNN, and Naïve Bayes in terms of accuracy, DR, and FAR.

F. Approach-7

An optimized intrusion detection using soft computing techniques has been proposed by Ahmad et al. [16], with the aim to provide an optimal intrusion detection system that has ability to minimize amount of features and maximize DR. In their approach, KDD Cup 1999 dataset is used for evaluation purpose and PCA is applied to convert the input samples into a new feature space. Moreover, GA is used to find a suitable number of principal components, and for classification purpose, they have applied SVM. They have focused on comparing SVM performance on feature sets. First, they obtained 12 features from PCA and GA and classified them with SVM. Second, they collected 22 features directly from PCA output using the traditional method and classified them with SVM. The experimental results referred to that, the proposed method has provided an optimal intrusion detection, which is able to minimize amount of features and maximize the DR.

G. Approach-8

Kim et al. [17] have proposed a new hybrid intrusion detection which integrating hierarchically MD model and AD model to overcome performance issue. In their proposed approach, the MD has used the information of known attack to build a classifier while the AD has used information of normal traffic to build a classifier. First, the MD model is decomposed normal training data into disjoint subsets. Then MD model is applied for each separate subset of normal training data. The techniques that used for each model were C4.5 decision tree (DT) for MD model, and 1-class SVM to construct multiple AD models. For their experiments, they have used NSL-KDD data set. It has been found that the result of the proposed method is superior to the conventional methods with respect to performance for detecting unknown attacks, training and testing time.

H. Approach-9

Ahmad et al. [18] have proposed a novel method in intrusion detection to enhance the performance of the classifier, where PCA is applied for feature transformation. Moreover, GA is used to find the genetic principle components, which offer a subset of features with optimal sensitivity and the highest discriminatory power. For classification purpose, they have applied SVM, where KDD-Cup dataset is used for evaluation purpose. Their work has extended the previous work [16], and the results have showed that, the proposed method has enhanced the performance of classifier in intrusion detection by minimizing the number of features (up to 10) and maximizing the DR (up to 99.96 %).

I. Approach-10

Chaunhan et al. [19] have presented a comparison between different classification techniques, which are worked to detect intrusions and classify them into normal and abnormal behaviors. The algorithms that have been selected are J48, Naive Bayes, RIPPER (JRip), and One Rule (OneR). Their experiments were performed by using NSL-KDD dataset. WEKA platform was selected for the implementation of the selected algorithms. The results have showed that the best algorithm for classification purpose is OneR classifier, where it required the shortest time, which is around 0.45 s with 10-fold cross-validation, and 0.32 s with supplied test set compared with others classifiers.

IV. STANDARD DATASETS

One of the most important parameters which can affect the capability of the intrusion detection mechanism is dataset, where the performance of IDSs depends on its accuracy and vice versa. When the training dataset is optimally accurate with a rich content then, the efficiency of the trained system is improved. Thus, the collection of the data in order to train and test the different classfication techniques is a critical dilemma. There are three different methods for collecting data to be used for experiments in the IDSs, which are [4]: (i) real traffic, (ii) sanitized traffic, and (iii) simulated traffic. However, these methods still inefficient for training and testing classification techniques, where using the real traffic to collecting data can be very costly, sanitized traffic is more risky, while generated simulation traffic required a hard work that can make the standard constructed simulated datasets popular for evaluating IDSs. There is a number of standard datasets, which can be classified based on the network traffic such as DARBA [20], KDD-Cup [21], NSL-Cup [22], CAIDA [23], and Kyoto2006+ [24]. Accordingly, these data can be used for experiments in order to evaluate classification techniques in the field of IDSs.

A. DARBA Dataset

DARPA dataset [20], [25] is the first standard corpora for evaluating computer network IDSs, which has been collected and distributed by MIT Lincoln Laboratory. Each evaluation effort built to measure the possibility of detectiton and FA for each system under test using many types of attacks. DARPA dataset was collected by set up a test bed that simulated the operation of a typical US Air Force LAN for over two months

to structure audit data to be used for evaluating algorithms in IDSs.

B. KDDCUP1999 Dataset

KDDCUP1999 dataset [21] is a connection of data transfer collected from a virtual environment to be used for the Competition of the Third International Knowledge Discovery and Data Mining Tools. This standard dataset is gathered by Stolfo et al. [26], based on the pre-processing version of data built in DARPA 1998 [20]. Each connection record is about 100 bytes, consists of 41 features, and labeled as normal or as an attack. KDDCUP1999 dataset separated into two sets, which are training set and testing set.

C. NSL-KDD Dataset

NSL-KDD dataset appeared to fix the issues occurred in KDDCUP1999 dataset, which has highly affected the performance of the evaluated systems. It proposed by Tavallaee et al. [27] as a new revised version of KDDCUP1999 dataset, and it publicly available online on [22]. The main contribution of NSL-KDD dataset, it does not include redundant records whether in train or test sets.

D. CAIDA Datasets

CAIDA datasets [23] are a collection of several different types of data, resulting from both active and passive measurement of the internet. These datasets are available to the research community with retention the privacy of individuals and organizations who donate data or network access. Established in 1997 by Dr. Kc Claffy and Tracie Monk, and located in San Diego Supercomputing Center (SDSC). The data collection for each dataset is still active and has continuing, regularly scheduled groups, or terminated and will not be resumed.

E. Kyoto 2006+ Dataset

Kyoto 2006+ dataset [24], [28] is a connection of data transfer collected from honeypots and darknets data published by Kyoto University. It appeared to fill the gap in the existing evaluation datasets, such as KDDCUP1999 dataset. Kyoto2006+ dataset built from diverse types of honeypots over three years of real traffic data from 2006 until 2009. It consists of 14 statistical features, which are derived from KDDCUP1999 dataset ignoring other features that contain redundant. As well, it includes additional 10 features for more analysis and evaluation of NIDSs.

V. COMPARATIVE ANALYSIS

A. Datasets Comparison

Table I presents a comparison between five types of standard datasets, which are mentioned in section V. DARPA dataset considered as a popular dataset used in IDSs to measure DR and FA for any network traffic, which consists of four types of attacks (DoS, R2L, U2R, and Probing attacks). However, it faced a set of critiques [29], where it appeared early in 1998 and 1999, using very simple models to create background traffic, and the synthesized data it does not look like to be similar background traffic in real networks. Moreover, traffic collectors that used to collect data from network traffic (i.e. TCPdump) are extremely probable ignore packet during intensive traffic load [27]. KDDCUP1999 dataset also appeared early in 1999 as a preprocessing version for data in DARPA 1998 dataset, which classified records into 41 features that are not related to any critiques to DARBA dataset. In spite of KDDCUP1999 dataset including a huge number of attacks where the attack types in training set are not the same in testing set, it includes redundant and duplicate record, which cause overhead during the evaluation process [27]. NSL-KDD dataset also appeared as a new version from KDD Cup dataset, which has removed redundant or duplicate records in KDD-Cup dataset and represent the records in way that is more reasonable. However, it does not considered as the ideal way for representing the existing real networks [28]. The issue is not just exclusive on NSL-KDD dataset but also includes old version from DARBA 1998 and KDD Cup 1999 datasets. CAIDA datasets also appeared for evaluating IDSs, which consists of different types of datasets that are collected from the internet, which considered as a perfect resource for representing the real existing networks. Kyoto 2006+ dataset also appeared for evaluating NIDSs, and it is built by through ignoring features that contain redundant, as the previous mentioned datasets, this dataset is also the comprehensive representation for the real current networks. Although it is recently emerged, it does not mention information on particular attack types.

Standards datasets are not limited to these five types of evaluation. VELOS dataset [30] also appeared to evaluate the performance of IDSs, which includes approximately 10 gigabytes of normal and malicious traffic with nearly different kinds of the potential attacks, mainly web access attacks. This traffic is primarily in several pcap files and tcpdump text files.

B. Approaches Comparsion

Table II presents a comparison for the approaches that were proposed and applied for intrusion detection, which are mentioned in section IV. It provides details regarding the techniques that are combined with a view to strengthen the performance of the IDSs. Moreover, it highlights different issues that need to be solved. In the hybrid approach for Tong et al. [8], despite the ability of using SVM, MPM, soft-computing, and other pattern classification technology as a pattern classification module instead of RBF neural network, it considered not suitable for classification. Results have showed its weakness compared with the other hybrid approaches. This can be due to used full features, which caused redundant and introducing overheads. Moreover, the raw feature set can confounded the classifier due to the redundancy and results FA. Further, it increases training and testing overheads, reduces accurate DR, consumes more memory and computational resources, increases architectural complexity and malfunctioning the system. The same issue appeared in [3] where they used full features, which caused overhead , and found that redundancy have reduced accuracy of DR, as well as it considered a time consuming for the training process. Moreover, they have found that the used classifier it often converges to the local minima, which considered unsettled, it also considered unsuitable for nonlinearly separable problem, and slow for overtraining. Despite using clustering approach for

TABLE I: STANDARDS's DATASETS COMPARISON IN IDS

Dataset	Release Date	Prepared Data	Data Source	Features	Availability	Pros	Cons
DARPA	1998 1999	MIT Lincoln Laboratory	Simulated the operation of a typical US Air Force LAN	-	Public	• First standard corpora for evaluating IDSs • Consists of a broad range of attacks (DoS, Probing, R2L, U2R) • Measure DR and FA for any network traffic	• Models that used to generate background traffic were too simple [29] • Workload of the synthesized data does not simulate the background traffic in real networks [29] • TCPdump can cause overloaded and drop packets [27]
KDDCUP	1999	Stolfo et al.	Preprocessing version for data in DARPA 1998 dataset	41 features (basic features, content features, and traffic features): 32 numeric features, 9 categorical features	Public	• Used for evaluating AD systems • Attacks types in training set are distinctive from the testing set.	• Includes redundant and ~~or~~ duplicate records [27] • Does not reflect the modern environment [28]
NSL-CUP	2009	Mahbod et al.	An improved version of KDDCUP 1999 dataset	41 features (basic features, content features, and traffic features): 32 numeric features, 9 categorical features	Public	• Dose not includes redundant or duplicate records • The selected records of NSL-KDD dataset are inversely proportional to the percentage of records in the KDDCUP dataset • The number of the records is reasonable in the training set and testing set.	• Not perfect for representing the existing real networks [28]
CAIDA	1998	Dr. Kc Claffy and Tracie Monk	Depends on the dataset that is captured	Depends on the dataset that is captured	Public	• Collect different types of data from the online available sources • Retention the privacy of individuals who donate the data • Perfect in representing the existing real networks	• Some datasets are restricted by permission
Kyoto 2006+	2009	Kyoto University	Real traffic data in honeypot and darknets	24 features: 14 statistical features derived from KDDCUP1999, 10 additional features	Public	• Used for evaluating NIDSs • Ignored features that contain redundant • Perfect in representing the existing real networks	• Does not mention information on particular attack types

performance, still determine the suitable number of clustering remains an open problem. Although Horng et al. [12] have used "leave-one-out" procedure to ignore irrelevant features from dataset, it is much complicated and overheads. Despite the better results of K-NN and Naïve Bayes compared to the conventional as kMeans, kNN, and Naïve Bayes in terms of accuracy, DR, and FAR., K-NN and Naïve Bayes can be further explored with other feature selection techniques, where the algorithm that is used is characterized by simplicity for dealing with redundant and irrelevant records in KDD-Cup 1999 dataset. The hybrid approach that was proposed by Chitrakar and Chuanhe [14] also showed a better accurate result for DR and FPR. It used Kyoto 2006+ dataset for evaluation purpose, where it ignored features that contain redundant. However, SVM classifier that was used in their approach can be more efficient and stable when applying multiple kernel based SVM classification schemes. Additionally, the time complexity of k-Medoids clustering still needs to be decreased. On the other side, the approaches that presented by Ahmad et al. [13], [16], [18], which sought to improve the performance issues in IDSs, provided the better results for DR and FA. In [13] , their proposed approach was an initial effort for features subset selection, which presented in order to override feature selection issues. Their experiments showed that, the proposed approach have improved accuracy, simplified the architecture of intrusion detection, as well as decreased training time and computational overheads. However, MLP classifier not adequately explained with more experiments, such as examining the ability of the classifier to execute well on the original dataset, or execute well on transformed dataset; this can be due to the problem for local minimal and required overtraining. Therefore, their method needs more experimentation to verify it. In [16], Ahmad et al. have used SVM classifier for their method with the aim to provide an optimal intrusion detection system that has ability to minimize amount of features and maximize DR. However, their approach required more experimentation, which can verify it. Although Ahmed et al. [18] have provided a new approach to enhance the performance of SVM classifier, and it results have showed that the proposed approach enhanced the performance of SVM classifier in intrusion detection in terms of minimizing the number of features and maximizing the DR, SVM classifier is not suitable for multiclass. Moreover, their work also needs more experimentation to verify it. Furthermore, despite the better results that have enhanced the detection accuracy which were resulted from NSL-KDD dataset, in [17], decompose the normal data using C4.5 DT degrades the misuse detection performance, and unequal allocation of data instances hinders the reduction of the training and testing time. Moreover, in [19], the results of accuracy, and time complexity, can be enhanced

TABLE II: APPROACHES COMPARISON IN IDS

Authors	Year	Detection Patterns	Preprocessing Techniques	Classification Techniques	Data source	Accuracy %	DR %	FP %	Issues/Problems
Tong et al. [7]	2009	Hybrid (AD + MD)	-	RBF NN	DARBA 1999		95.3	1.4	• Used full features that leading to redundancy and results FA, Increases training and testing overheads, Reduces accurate DR, consumes more memory and computational resources, Increases architectural complexity and malfunction of the system
Wang et al. [3]	2010	AD	-	Feed-Forward NN	KDD Cup 1999	96.71	-	-	• Used full features, which caused overhead and redundant, and required more time for training • Identifying the suitable number of clustering still an open issue • The used classifier it often converges to the local minima, which considered unsettled, unsuitable for nonlinearly separable problem, and slow for overtraining
Horng et al. [10]	2011	Hybrid (AD + MD)	"leave-one-out" procedure	SVM	KDD Cup 1999	95.70	-	0.70	• More complicated and had overheads on massive dataset due to the methods that are used to determine the significant feature. • SVM classifier can be explored and compared with other feature selection techniques
Ahmad et al. [11]	2011	Hybrid (AD + MD)	PCA + GA	MLP	KDD Cup 1999	99.00	-	0.30	• MLP classifier not adequately explained with more experiments such as, its ability of executing it on original dataset, or its ability on executing along with a transforming dataset, due to the problem for minimal and overtraining. • The approach needs more experimentation to verify it
Chitrakar and Chuanhe [12]	2012	AD	Sampling and filtering	SVM	Kyoto 2006+	99.21	99.30	<1.00	• SVM classifier can be more efficient and stable when applying multiple kernel based SVM classification schemes • Increased time complexity through applying every data samples, one by one, during the implemented process
Om and Kundu [13]	2012	AD	Entropy based feature selection algorithm	K-NN + Naïve Bayes	KDD Cup 1999	99.00	98.18	0.83	• Hybrid Classifiers that used can be further explored with other feature selection techniques, where the algorithm that was used is very simple to deal with redundant and irrelevant records in KDD-Cup 1999 dataset.
Ahmad et al. [14]	2013	Hybrid (AD + MD)	PCA + GA	SVM	KDD Cup 1999	-	99.60	0.400	• The approach required more experimentation to verify it • SVM classifier can be explored and compared with other feature selection techniques
Kim et al. [15]	2014	Hybrid (AD + MD)	-	C4.5 DT + SVM	NSL-KDD	-	>99.0	<0.50	• Decompose the normal data using C4.5 DT degrades the misuse detection performance. • Unequal allocation of data instances hinders the reduction of the training and testing time • SVM classifier can be explored and compared with other feature selection techniques
Ahmad et al. [16]	2014	Hybrid (AD + MD)	PCA + GA	SVM	KDD Cup 1999	-	99.96	0.70	• The approach needs more experimentation to verify it • SVM classifier can be explored and compared with other feature selection techniques, • SVM classifier being inappropriate for multiclass
Chaunhan et al. [17]	2014	AD	-	J48	NSL-KDD	99.55	-	0.004	• The results can be enhanced and the timing for the training and the testing can be reduced by ignoring redundant and irrelevant features through using optimal technique for feature selections
				Naïve Bayes		89.59	-	0.106	
				JRip		99.58	-	0.004	
				OneR		96.10	-	0.036	

by ignoring redundant and irrelevant features using optimal technique for feature selections.

Therefore, the accuracy of IDSs depends on an optimum classification technique, which is mainly depends on optimal dataset selection. Many techniques for feature selection were applied on dataset with a view to enhance the accuracy of the classifier. A numbers of studies have been presented in order to enhance the performance of IDSs by increasing DR and decreasing FPR. Thus, finding the optimal classification technique to avoid current issues in the recent techniques, choosing suitable dataset, which includes rich types of recent attacks, and selecting a suitable features are the current important issues in the field of IDSs.

VI. CONCLUSION

Find the optimal classification technique to enhance the performance of IDSs by increasing the DR and decreasing the FA is still an ongoing area. In this paper, a review of the current classification techniques in IDSs is introduced. Moreover, a review of the most popular datasets used for train and test

selected classifier is also introduced. Each classification technique has its superiority and limitations during classify data into normal or abnormal, so that it is important to select an optimal one during intrusion detection process. Additionally, we have also discussed these reviews to conclude that, a set of issues must be take into consideration during development of classification techniques in IDSs, such as which is an optimal dataset that includes a rich types of recent attacks, and which features that must be selected without confused, overhead, and time-consuming selected classifier.

ACKNOWLEDGMENT

This research work is supported by Department of Software Engineering, CCIS, King Saud University, Riyadh, Saudi Arabia.

REFERENCES

[1] T. Verwoerd and R. Hunt, "Intrusion Detection Techniques and Approaches," *Computer Communications,* vol. 25, no. 15, pp. 1356-1365, September 2002.

[2] A. Lazarevic, V. Kumar and J. Srivastava, "Intrusion Detection: A Survey," in *Managing Cyber Threats*, vol. 5, Springer US, 2005, pp. 19-78.

[3] G. Wang, J. Hao, J. Ma and L. Huang, "A new approach to intrusion detection using Artificial Neural Networks and fuzzy clustering," *Expert Systems with Applications,* vol. 37, no. 9, pp. 6225-6232, September 2010.

[4] I. Ahmad, A. Abdullah and A. Alghamdi, "Artificial neural network approaches to intrusion detection: a review," in *Proceedings of the 8th International Conference on the World Scientific and Engineering Academy and Society*, Istanbul, Turkey, 2009.

[5] J. P. Anderson, "Computer security threat monitoring and surveillance," Fort Washington, Pennsylvania, April, 1980.

[6] D. E. Dorothy , "An Intrusion-Detection Model," *Software Engineering, IEEE Transactions on,* vol. 13, no. 2, pp. 222-232, February 1987.

[7] H.-J. Liao, C.-H. R. Lin, Y.-C. Lin and K.-Y. Tung, "Intrusion Detection System: A Comprehensive Review," *Journal of Network and Computer Applications,* vol. 36, no. 1, pp. 16-24, 1 January 2013.

[8] X. Tong, Z. Wang and H. Yu, "A research using hybrid RBF/Elman neural networks for intrusion detection system secure model," *Computer Physics Communications,* vol. 180, no. 10, pp. 1795-1801, Octobar 2009.

[9] Y. Bai and K. Hidetsune, "Intrusion Detection Systems: technology and development," in *Proceedings of the 17th International Conference on Advanced Information Networking and Applications*, March, 2003.

[10] K. Scarfone and P. Mell, "Guide to Intrusion Detection," National Institute of Standards and Technology, February, 2007.

[11] R. Mirchell and I.-R. Chen, "A Survey of Intrusion Detection in Wireless Network Application," *Computer Communications,* vol. 42, pp. 1-23, 1 February 2014.

[12] S.-J. Horng, M.-Y. Su, Y.-H. Chen, T.-W. Kao, R.-J. Chen, . J.-L. Lai and C. D. Perkasa, "A novel intrusion detection system based on hierarchical clustering and support vector machines," *Expert Systems with Applications,* vol. 38, no. 1, pp. 306-313, January 2011.

[13] I. Ahmad, A. Abdullah, A. Algamdi, K. Alnfajan and M. Hussain, "Intrusion detection using feature subset selection based on MLP," *Scientific Research and Essays,* vol. 6, no. 34, pp. 6804-6810, December 2011.

[14] C. Huang and R. Chitrakar, "Anomaly detection using Support Vector Machine classification with k-Medoids clustering," in *Proceedings of the 3rd International Conference on Asian Himalayas*, Kathmandu. Nepal, November, 2012.

[15] H. Om and A. Kundu, "A hybrid system for reducing the false alarm rate of anomaly intrusion detection system," in *Proceedings of the 1st International Conference on Recent Advances in Information Technology (RAIT)*, March, 2012.

[16] I. Ahmad, A. Abdullah and A. Alghamdi, "Optimized intrusion detection mechanism using soft computing techniques," *Telecommunication Systems,* vol. 52, no. 4, pp. 2187-2195, April 2013.

[17] G. Kim, S. Lee and S. Kim, "A novel hybrid intrusion detection method integrating anomaly detection with misuse detection," *Expert Systems with Applications,* vol. 41, no. 4, pp. 1690-1700, March 2014.

[18] I. Ahmad, M. Hussain , A. Alghamd and A. Alelaiwi, "Enhancing SVM performance in intrusion detection using optimal feature subset selection based on genetic principal components," *Neural Computing and Applications,* vol. 24, no. 7-8, pp. 1671-1682, June 2014.

[19] H. Chauhan, V. Kuma, S. Pundir and E. S. Pilli, "Comparative Analysis and Research Issues in Classification Techniques for Intrusion Detection," in *Proceedings of the International Conference on Advanced Computing, Networking, and Informatics,*, India, 2014.

[20] "DARPA Intrusion Detection Evaluation Program," MIT Lincoln Labs, 1998. [Online]. Available: http://www.ll.mit.edu/mission/communications/cyber/CSTcorpora/ideval/index.html. [Accessed February 2015].

[21] "KDD Cup 1990 Data," 1999. [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. [Accessed February 2015].

[22] "NSL-KDD Dataset," 2009. [Online]. Available: http://nsl.cs.unb.ca/NSL-KDD/ . [Accessed February 2015].

[23] "CAIDA Dataset," CAIDA, [Online]. Available: http://www.caida.org/data/overview/ . [Accessed February 2015].

[24] "Kyoto2006+ Dataset," [Online]. Available: http://www.takakura.com/Kyoto_data/ . [Accessed February 2015].

[25] R. Lippmann, D. Fried, I. Graf, J. Haines, K. Kend, D. McClung, D. Weber , S. Webster, D. Wyschogrod, R. Cunningham and M. Zissman, "Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation," in *Proceedings of the International Conference on DARPA Information Survivability Conference and Exposition*, 2000.

[26] S. J. Stolfo, W. Fan, W. Lee, A. Prodromidis and P. K. Chan, "Cost-based modeling for fraud and intrusion detection: results from the JAM project," in *Proceedings of the International Conference on DARPA Information Survivability Conference and Exposition*, 2000.

[27] M. Tavallaee, E. Bagheri, W. Lu and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proceedings of the Second IEEE international conference on Computational intelligence for security and defense applications*, Ottawa, ON Canada, July, 2009.

[28] H. Takakura, Y. Okabe, M. Eto, D. Inoue and K. Nakao, "Statistical analysis of honeypot data and building of Kyoto 2006+ dataset for NIDS evaluation," in *Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security*, 2011.

[29] J. McHugh, "Testing Intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory," *ACM Transactions on Information and System Security,* vol. 3, no. 4, pp. 262-294, November 2000.

[30] " VELOS Dataset," [Online]. Available: http://velos-dataset.appspot.com. [Accessed February 2015].

Evaluation on Malware Classification by Combining Traffic Analysis and Fuzzy Hashing of Malware Binary

S. Hiruta[1], Y. Yamaguchi[2], H. Shimada[2] and H. Takakura[2]
[1]Graduate School of Infomation Science, Nagoya University, Nagoya, Aichi, Japan
[2]Information Technology Center, Nagoya University, Nagoya, Aichi, Japan

Abstract— *Recent cyber attacks frequently use variants of malware programs which update existing functions drastically and implement new functions. Not only in functional viewpoint, recent malware programs improves their secrecy in variants, such as obfuscation, encryption, and changing thair behavior by inspecting their execution environment. But the number of skilled malware analysts is limited. So, a method to reduce expensive cost of manual analysis is widely explored in order to fight against huge amounts of malware programs. In this paper, we propose an integrated approach of dynamic traffic analysis and static program analysis. Similar to other conventional methods, the former part performs feature extraction, clustering, and labeling to summerize traffic data into sequence of characters. The latter part applies Fuzzy Hashing to malware programs which can effectively represent identical partial part in malware programs. We evaluated three integration patterns such as prioritize dynamic analysis result, prioritize static analysis result, and utilize mean of two analysis result. From the experimental results by using 340 malware samples and their traffic data, our method can correctly identify 61.1% of malware.*

Keywords: Malware Classification, Dynamic Analysis, Static Analysis, Similarity Measure

1. Introduction

Recent cyber attacks often use variants which modified existing malware. The infection technique has also been sophisticated by using targeted email attacks or watering holing, and efficient countermeasures are required on the assumption that malware has already intruded[1][2].

For the malware analysis among these countermeasures, it is important to estimate the behavior of malware, the purpose of attacks and the damage caused by malware activity. However, the obfuscation and encryption is commonly applied to malware programs and current malware changes its activities infected by PC(Personal Computer) environment. Therefore, it is difficult to determine the similarities and differences by comparing the malware program with existing ones. The type of malware is increasing, and the effort of individual analysis else continues to increase. On the other hand, malware analyst is insufficient overwhelmingly, and it is not able to deal with the current situation in the analysis by hand. To solve this problem, we have to prepare a system to classify a large number of malware fast and accurate. The analyst asks the system to examine malware carefully and report the malware which is not classify as a known family.

Malware used in targeted attacks, such as RAT(Remote Administration Tool), always perform some communication in order to scout and penetrate to target network, steal confidential information, destroy the system by communicating with C&C(Command and Control) servers. In many cases, malware uses communication protocols which are commonly used such as 80/tcp(http), 443/tcp(https), 443/udp(SPDY/QUIC). Such communications have various communication patterns, and large amount, and are encrypted if necessary. Furthermore, examples of superimposing the stolen information in the SYN packets is also observed. Also, it is not effective to analyze malware communications by IP(Internet Protocol) address or domain name, URL(Uniform Resource Locator), etc. because C&C servers are myriad prepared.

In this paper, we propose a malware classification method combining communication behavior analysis and binary code analysis. First, we perform malware classification based on similarity to known malware by using malware traffic data(pcap format). Second, we do clustering for feature vector generated on a per-packet from traffic data. Third, convert traffic data to cluster sequence based on a clustering result. Finally, calculate similarity of cluster sequence by using n-gram algorithm. If traffic data is similar to multiple malware families, we apply the Fuzzy Hash for malware binary and calculate similarity based on hash value as a second stage.

The rest of paper is organized as follow. In Sec. 2, we present related works. In Sec. 3, we introduce our proposal in detail. The effectiveness of our proposal is proofed with evaluation results in Sec. 4. Finally, in Sec. 5, we conclude this paper and present future works.

2. Related Works

Malware classification method is divided into a dynamic analysis and a static analysis. Dynamic analysis is a method

that classifies malware by features obtained by actually malware execution such as process generations, resource modifications, and observation of system call. Static analysis is a method that classifies malware using features from structure of program code.

2.1 Dynamic Analysis

Dynamic analysis is divided into system behavior based one and network behavior based one. As examples of former one, Bayeret et al. proposed profile based method which records files and processes modification footprint under malware execution[3]. Fujino et al. proposed that records API(Application Programming Interface) calls under malware execution[4].

Recently, malware classification method focuses on the network behavior based one because malwares get accustomed to hide its activity in the system. Nari et al. proposed a method based on graph of utilizing[5]. Lim et al. proposed a method which utilize session sequence data that characterize transition of malware traffic[6].

However, since a number of malware has been created for a same purpose, many malware families show similarity both system and network behavior if we apply simple analyze.

2.2 Static Analysis

Zhong et al. proposed a method which calculates the similarity between the opcode of known malwares and unknown malwares[7]. Iwamoto et al. classified malware by finding specific API calls from the program code[8].

In recent years, malware classification method by using Firstly, Fuzzy Hash applies hash to the program code so that it contribute to the analysis speed. Secondly, Fuzzy Hash is expected to contribute to the accuracy of malware classifications because it has a feature that hash values from partially identical data gives similar ones. As an example, Kamii et al. determined similarity of malware structures by applying the Fuzzy Hash to the program code [9] . On the other hand, [10] calculated similarity by applying hash values to byte stream corresponding to the individual functions of malware instead of applying to the entire malware

However, it is difficult to classify only static analysis because recent malware is subjected encryption or obfuscation. Also some recent malwares include the ability to self-modifying program according to analysis environments.

3. Proposed Method

3.1 Overview

In this method, we propose three classification methods based on dynamic analysis and static analysis.

1) Dynamic First Static Later(DFSL).
2) Static First Dynamic Later(SFDL).
3) Integration of Static and Dynamic(ISD).

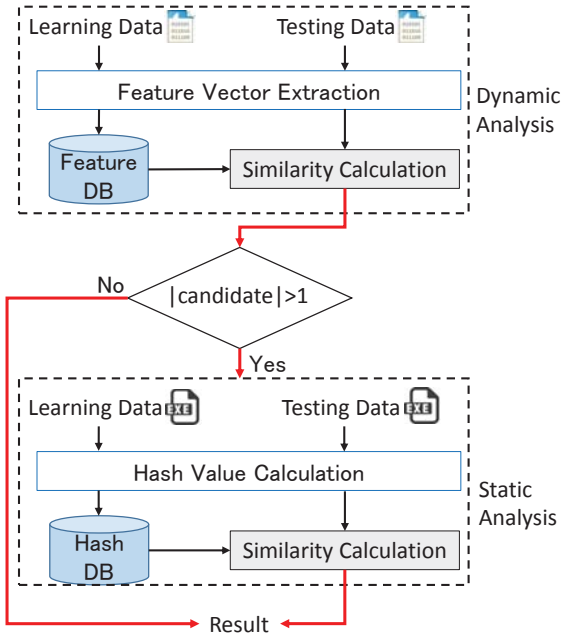


Fig. 1: Overview of Dynamic First Static Later

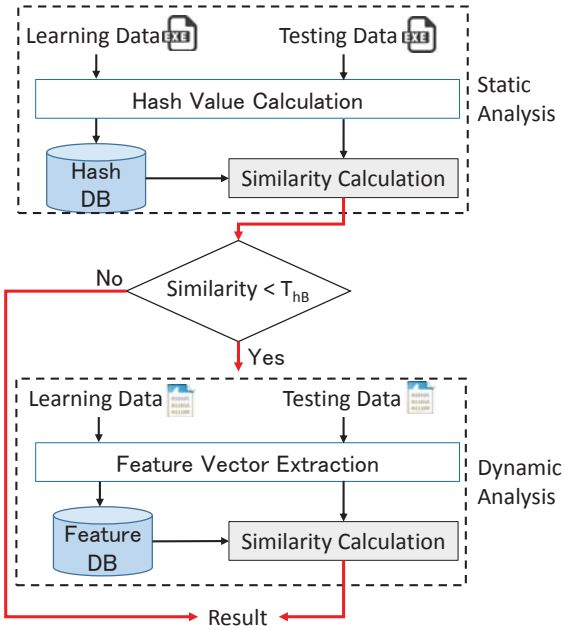


Fig. 2: Overview of Static First Dynamic Later

Fig. 1 shows a flow of the DFSL and Fig. 2 shows a flow of the SFDL. Each procedure in the figures are as follows.

- Feature Vector Extraction: Extract the per-packet feature vector on from traffic data of malware.
- Clustering and Labeling: Assign the feature vector to a nearest cluster and apply labeling. As a result, we can represent the malware traffics with cluster sequences.
- Traffic Similarity Calculation: Classify malware based

Table 1: Generated Cluster Sequences

Malware Name	Cluster Sequence
Adware.Win32.Fiseria 1	BBCBBICHCBGBBGBBHCBCGBAAAAAAAAABBBBAAAAA...BEEEEDEEDDDDDDDDEEEE
Adware.Win32.Fiseria 2	BBCBBICHCBGBGBBHCBCGAAAAAAAAABBBBAAAAAAA...EEDDDDDDDDEEEEEEEEEE
Trojan-Spy.Win32.SpyEyes 1	BBBBEEEEEEBCEEEHCBCHGGGCBGBBGCCHCBHGGGC...EEEBEEEEDEEDDDDDDDDE
Trojan-Spy.Win32.SpyEyes 2	BBBBCBBBBBEEEEEEBCEEEHCBCHGGGCBGBBGCCHC...EDEEDDDDDDDDEEEEEEEE

Table 2: Fash Values of Malware Using Ssdeep

Malware Name	Hash Value
Trojan.Win32.Agent 1	T9AhE7rLFl4AAQML253uqkMbLFw5t7LD3KE2YFKIWXFeLL0CIo
Trojan.Win32.Agent 2	/9AhE7rLFl4AAQML253uqkMbLFw5t7LD3KE2YFKIWXFeLL0CIq
HERU:Downloader.Win32.LMN 1	sqGTrxWcwEUfMxJRCo6tn2nXngXGMX6Al3goCJw6JpqH8i8UgwyWO6i
HERU:Downloader.Win32.LMN 2	VqGTrxWcwEUfMxJRCo6tn2nXngXGMX6Al3goCJw6JpqH8i8U4wyWO6o

on similarity of cluster sequences.

- Hash Value Calculation: Apply Fuzzy Hashing to malware binaries.
- Binary Similarity Calculation: Classify malware based on similarity of malware binaries.

3.2 Feature Vector Extraction

We expected that features of network traffic contains beneficial information for malware classification. We estimated that similar malwares generate partially similar traffics because malware with same purpose are usually generated by same tool. Therefore, we generated feature vectors containing the following features:

- Sevice Port Number
- Packet Length
- Communication Protocol

In addition to ICMP, ARP, TCP, and UDP, we used DNS and HTTP as independent features because they are communication protocols which are largely related to malware activities.

3.3 Clustering and Labeling

Feature vectors ontained by feature extraction are classified by k-means++[11]. k-means++ is a non-hierarchical clustering algorithm and classifies given data into k clusters. After clusters generation, we assign labels to them and convert traffic data to cluster sequences based on labeling. Table 1 shows cluster sequences of 4 types of malware traffic data. In this example, we assigned 9 to k and labeled 'A' to 'I' for each cluster. From this example, it can be seen that malware can be classified based on them.

3.4 Traffic Similarity Calculation

Calculate similarity base on cluster sequences obtained in Sec. 3.3. Similarity R_P of cluster sequences s_1 and s_2 are as follows,

$$R_P = \frac{c}{(t_1 + t_2) - c} \quad (1)$$

$$t_i = |s_i| + (n-1) \quad (2)$$

Where t_i is the number of tokens given by n-gram of cluster sequence s_i, and c is the number of common tokens between s_1 and s_2.

3.5 Hash Value Calculation

Then, we apply Fuzzy Hashing to malware binaries and obtain hash values of malware. The obtained hash values are stored into Hash DB. We utilize `Ssdeep`[13] as an algorithm of Fuzzy Hashing that calculates CTPH(Context Triggered Piecewise Hashes)[12]. Hash values are constructed from block size, hash1 and hash2. Upper limit of hash1 length is 64 words, hash2 length is 32 words. In this method, we use hash1 as hash values. `Ssdeep` separates input data into several blocks. The concatenation of the strings becomes calculated hash value. Therefore, similar hash values are obtained from input data if input data contain much identical partial data. `Ssdeep` outputs both maximum 64 word hash value and maximum 32 word hash value. In this method we only use maximum 64 word hash value.

Table 2 shows hash values of 2 types malware as examples. From this table, we estimated that we can be correctly classify malware using hash values.

3.6 Binary Similarity Calculation

We utilize Levenshtein Distance(LD)[14] for similarity measurement of hash values and defined similarity of binary as follows.

$$R_B = 1 - \frac{LD}{\max\{|s_1|, |s_2|\}} \quad (3)$$

LD represents the amount of the different symbols between strings s_1 and s_2. LD also represents the minimum number of symbols to be modified which is required to convert s_1 to s_2. Fig. 3 shows an example of LD counting.

s_1:	a	b	c	d	e	f
	=	=	=	≠	≠	≠
s_2:	a	b	c	g	h	
LD count	0	0	0	1	2	3

Fig. 3: Example of LD counting

3.7 Integration of Dynamic and Static Analysis

As mentioned in Sec. 3.1, we prepared three combination or integration classification methods.

In 1st method which is named as Dynamic First Static Later (DFSL) firstly classifies malwares based on R_P. If there are many malwares those R_P values show close to maximum R_P values, we utilize R_B value as a second stage (Fig. 1). In the second stage, we calculate R_B between malware binary and whole learning data. The testing malware is classified into the family which shows highest R_B value. The detailed algorithm is shown as follows.

$$\mathbf{R}_{Pi} = \{R_{Pi-1}, ..., R_{Pi-J}\} \quad (4)$$

$$R_{Pi-max} = \max\{R_{Pi-1}, ..., R_{Pi-J}\} \quad (5)$$

$$|R_{Pi-max} - T_{hP}| < R_{Pi-j} \quad (6)$$

First, we calculate similarities by comparing testing data m_i and learning data $l_j (1 \leq j \leq J)$ by dynamic analysis. In similarity results $\mathbf{R}_{Pi}$(4), we identify the maximum similarity R_{Pi-max}(5). Finally, if the number of R_{Pi-j} that satisfies (6) is more than 1, we consider that the candidates are multiple and calculate similarities by comparing the m_i and the l_j by static analysis. T_{hP} is threshold that we determine.

In 2nd method which is named as Static First Dynamic Later(SFDL) firstly classifies malware based on R_B. R_B is calculated between testing malware binary and whole learning data binaries. If whole R_B value shows less than threshold T_{hB}, we classify malware base on R_P In the second stage, we calculate R_P between hash values of testing data and whole learning data. The testing malware is classified into the family which shows highest R_P value. The detailed algorithm is shown as follows.

$$\mathbf{R}_{Bi} = \{R_{Bi-1}, ..., R_{Bi-J}\} \quad (7)$$

$$R_{Bi-max} = max\{R_{Bi-1}, ..., R_{Bi-J}\} \quad (8)$$

$$P_{Bi-max} < T_{hB} \quad (9)$$

First, we calculate similarities by comparing testing data m_i and learning data $l_j (1 \leq j \leq J)$ by static analysis. In similarity results $\mathbf{R}_{Bi}$(7), we identify the maximum similarity R_{Bi-max}(8). Finally, if R_{Bi-max} satisfies (9), we calculate similarities by comparing the m_i and the l_j by dynamic analysis.

In 3rd method which is named as Integrated Static and Dynamic(ISD) which utilize average value of R_P and R_B. Both R_P and R_B are calculated between testing malware and whole learning data. Then, we obtain R_A which is an average value of R_P and R_B and classified into the family which shows highest R_A value.

4. Evaluation

In order to evaluate the classification performance of the proposed methods, we experimented using malware samples and traffic date of them.

4.1 Experimental Data

We used 340 malware samples as experiment data which are collected from April 2014 to October 2014. The traffic data is obtained by executing the malware samples with the sandbox[15]. The 340 malware samples were classified into 48 families by `Kaspersky`.

We divided 340 malware samples into 227 learning data and 113 testing data. Also, we generated 9 clusters by k-means++ from the malware traffic data from $801,790$ packets including traffic of both learning data and testing data. Also, we calculated the Fuzzy Hashing values of malware. After prior pre-processing, we performed classification based on similarity determination between learning data and testing data by the proposed three methods. If our classification results matched `Kaspersky` classification results, they were correct answers.

4.2 Results

4.2.1 Dynamic First Static Later(DFSL)

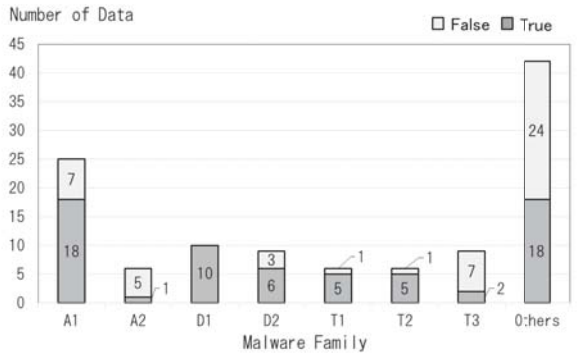
As mentioned in Sec. 3.7, DFSL firstly applies dynamic analysis and then applies static analysis if there are many candidates in dynamic analysis. We used 0.05 for T_{hP} which is defined by preliminary evaluation and applied static analysis to 92 malware samples.

Table 3 shows the number of correctly classified testing data (True) and incorrectly classified testing data (False) which is classified with first stage only ($|Candidate| = 1$) and with both stages ($|Candidate| > 1$). As shown from Table 3, 21 testing data finishes only first stage and 15 testing data are correctly classified. The classification accuracy becomes 71.4%. 50 testing data were correctly classified. The classification accuracy becomes 54.3%. The overall accuracy becomes 57.5%.

Fig. 4 shows the result of detailed True / False numbers in malware family level. The horizontal axis shows malware families and the vertical axis shows the number of testing data belongs to that family. The detailed discussion will be done in Sec. 4.3 with comparing to other methods (SFDL and ISD).

Table 3: Detailed Classification Results by Dynamic First Static Later

	$\|Candidate\| = 1$	$\|Candidate\| > 1$	Total
True	15	50	65
False	6	42	48
Total	21	92	113

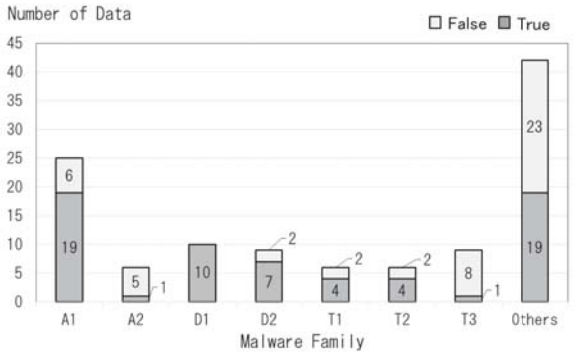


A: Adware, D: Downloader, T: Trojan

Fig. 4: Classification Results by Dynamic First Static Later

Table 4: Detailed Classification Results by Static First Dynamic Later

	$R_B >= 0.90$	$R_B < 0.90$	Total
True	44	21	65
False	9	39	48
Total	53	60	113



A: Adware, D: Downloader, T: Trojan

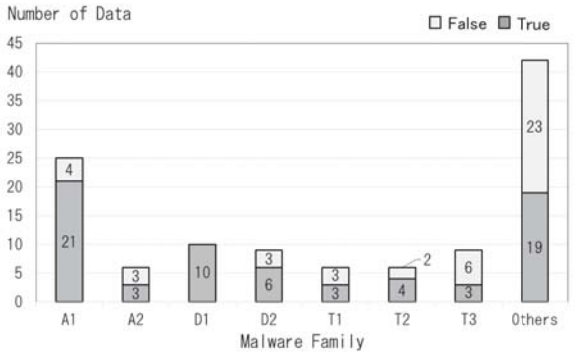
Fig. 5: Classification Results by Static First Dynamic Later

4.2.2 Static First Dynamic Later(SFDL)

As mentions in Sec 3.7, SFDL firstly applies static analysis and then applies dynamic analysis. We used 0.90 for T_{hB} which is defined by preliminary evaluation and applied dynamic analysis to 60 samples.

Table 4 shows the results with first stage only ($R_B \geq 0.90$) and with both stages ($R_B < 0.90$). As shown from Table 4, 53 testing data finishes only first stage and 44 testing data are correctly classified. The classification accuracy becomes 83.0%. Left 60 testing data are sent to second stage and 21 testing data are correctly classified. The overall accuracy becomes 57.5% which is identical to DFSL.

Fig. 5 shows the result of detailed True / False numbers in malware family level which organization is identical to Fig. 4. The detailed discussion will be done in Sec. 4.3 similar to Fig. 4.



A: Adware, D: Downloader, T: Trojan

Fig. 6: Classification Results by Integration of Static and Dynamic

4.2.3 Integration of Static and Dynamic(ISD)

In this method, we used the similarity R_A which is an average of R_P and R_B.

$$R_A = \frac{R_P + R_B}{2} \tag{10}$$

In ISD, 69 testing data are correctly classified and 44 are missed. So, the classification accuracy becomes 61.1%. But 55 testing data gives $R_B = 0$ which represents no similarity in malware binary. So, these testing data almost classified with only traffic data so that the improvement of ISD becomes comparatively small.

Fig. 6 shows the result of detailed True / False numbers in malware family level which organization is identical to Fig. 4.

4.3 Consideration

Fig. 4, Fig. 5 and Fig. 6 the classification results of DFSL, SFDL, and ISD. Classification results of malware that many one belong to the malware family are shown individually in Fig. 4, Fig. 5 and Fig. 6. From Sec. IV-B, classification accuracies of Fig. 4 and Fig. 5, there are little differences between correctly classified malware by DFSL and SFDL, such as A1, D1, T1 and T2. We can be considered that these differences were caused as follows.

Table 5: Classify Adware.Win32.Agent by Dynamic Analysis

Malware Family	R_P
Adware.Win32.Fiseria	0.986
Downloader.Win32.Morstar	0.971
Adware.NSIS.Agent	0.954

Table 6: Classify Adware.Win32.Agent by Static Analysis

Malware Family	R_B
Adware.Win32.Fiseria	0.380
Downloader.Win32.Morstar	0.750
Adware.NSIS.Agent	0

Table 7: Classify Trojan.Win32.Inject by Static Analysis

Malware Family	R_B
Trojan.Win32.Inject	0.490
Trojan-Spy.Win32.Zbot	0.430

Table 8: Classify Trojan.Win32.Inject by Dynamic Analysis

Malware Family	R_P
Trojan-PSW.Win32.Tepfer	0.891
Trojan.Win32.Inject	0.016
Trojan-Spy.Win32.Zbot	0.694

In DFSL, there are some examples that are correctly classified by dynamic analysis, but are misclassified by static analysis. we show DFSL result of Adware.Win32.Feseria in Table 5 and Table 6. As shown in the Table 5, the similarity between the Adware.Win32.Feseria was 0.986 which is the highest similarity. However, as shown in the Table 6, the similarity between the Downloader.Win32.Morstar was the highest similarity. In consequence, Adware.Win32.Feseria was misclassified to Downloader.Win32.Morstar.

Similarly, in SFDL, there are some examples that are correctly classified by static analysis, but are misclassified by dynamic analysis. As a sample of this pattern, we show SFDL result of Trojan.Win32.Inject in Table 7 and Table 8. As shown in the Table 7, the Trojan.Win32.Inject was 0.490 which is the highest similarity. However, as shown in the Table 8, the similarity between the Trojan-PSW.Win32.Tepfer was the highest similarity. In consequence, Trojan.Win32.Ingect was also misclassified to Trojan-PSW.Win32.Tepfer.

Therefore, even if the final accuracy is identical ones, the little differences between DFSL and SEDL are occurred.

In contrast, ISD takes both the similarity R_P and R_B into consideration so that it can deal with these malwares. Therefore, this method shows the most effective results. However, there are still malwares that could not be classified so that another classification algorithm against to those malwares are required.

5. Conclusion and Future Works

In this paper, we proposed malware classification methods based on dynamic analysis and static analysis.

In dynamic analysis, first, we extract the per-packet feature vector from traffic data of malware. Second, we assign the feature vector to a nearest cluster and apply labeling. Third, we represent the malware traffics with cluster sequences. Finally, we classify malware based on similarity of cluster sequences using n-gram.

In static analysis, first, we apply Fuzzy Hashing to malware binaries. Then, we classify malware based on similarity of malware binaries using Levenshtein Distance(LD).

We evaluated our methods with 340 malware samples and traffic data of them. The accuracy rate of the classification method DFSL was 57.5%. The accuracy rate of the classification method SFDL was also 57.5%. The accuracy rate of the classification method ISD was 61.1%. From these results, the third method was the most effective for classifying malware.

As our future works, first, we will apply the our proposed methods to much more malware and evaluate the performances. Second, we will improve cluster sequences,L representation, for example, compressing the same traffics in them. Finally, we will refine the extracting feature vector and the similarity calculate algorithms.

Acknowledgment

This work is supported by R&D of detective and analytical technology against advanced cyber-attacks, administered by the Ministry of Internal Affairs and Communications.

Also, we thank NTT Secure Platform Laboratories for our collaborative research.

References

[1] Information-technology Promotion Agency, "Design and op-erational guide to protect against advanced persistent threats 2nd edition," 2011. Available: https://www.ipa.go.jp/files/000017299.pdf

[2] P. Cichonski, T. Millar, T. Grance, and K. Scarfone, "Computer Security Incident Handling Guide," Technical report, SP 800-61 Rev. 2, Gaithersburg, MD, United States, 2012.

[3] U. Bayeret, P. M. Comparetti, C. Hlauschek, C. Kruegel, and E. Kirda. "Scalable, Behavior-Based Malware Clustering," Network and Distributed System Security Symposium, vol. 9, 2009.

[4] A. Fujino and T. Mori, "Analysis of massive amount of API call logs collected from automated dynamic malware analysis systems (In Japanese)," Available: http://www.iwsec.org/mws/2013/manuscript/3A1-4.pdf

[5] S. Nari and A. Ghorbani, "Automated malware classification based on network behavior," Computing, Networking and Communications 2013, pp. 642-647, 2013.

[6] H. Lim, Y. Ymaguchi, H. Shimada, and H. Takakura, "Malware Classification Method Based on Sequence of Traffic Flow", Proceedings of 1st International Conference on Information Systems Security and Privacy, pp. 230-237, 2015.

[7] Y. Zhong, H. Yamaki, Y. Yamaguchi, and H. Takakura, "ARIGUMA Code Analyzer: Efficient Variant Detection by Identifying Common Instruction Sequences in Malware Families", 2013 IEEE 37th Annual Computer Software and Applications Conference, pp. 11-20, 2013.

[8] K. Iwamoto and K. Wasaki, "Malware classification based on extracted api sequences using static analysis," Proceeding of the Asian Internet Engineering Conference, pp. 31-38, 2012.
[9] K. Kamii, M. Terada and, J. Chao "A Proposal of Malware Information Support System using Similarity of File Structure(In Japanese)", CSEC:Computer Security Group 2011, vol. 52, pp. 1-6, 2011.
[10] D. Raygoza, "Automated Malware Similarity Analysis," DEFCON17, 2009. Available: http://www.blackhat.com/presentations/bh-usa-09/RAYGOZA/BHUSA09-Raygoza-MalwareSimAnalysis-PAPER.pdf
[11] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding,"SODA '07 Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, pp. 1027-1035, 2007.
[12] J. Kornblum, "Identifying almost identical files using context triggered piecewise hashing," in Digital Investigation, vol. 3S, pp. 91-97, 2006.
[13] J. Kornblum, "Fuzzy Hashing and ssdeep," Sourceforge, 2012. Available: http://ssdeep.sourceforge.net/
[14] V. I. Levenshtein, "Binary Codes Capable of Correcting Deletions, Insertions, and Reversals," Cybernetics and Control Theory, pp. 707-710, 1966.
[15] K. Aoki, T. Yagi, M. Iwamura and M. Itoh, "Controlling Malware HTTP Communications in Dynamic Analysis System using Search Engine", The 3rd International Workshop on Cyberspace Safety and Security, 2011.

A Polyscale Autonomous Sliding Window for Cognitive Machine Classification of Malicious Internet Traffic

Muhammad Salman Khan, Ken Ferens, and Witold Kinsner
Dept. of Electrical and Computer Engineering, University of Manitoba, Winnipeg, MB, Canada
muhammadsalman.khan@umanitoba.ca, ken.ferens@umanitoba.ca, witold.kinsner@umanitoba.ca

Abstract—*Features of an Internet traffic time series can be estimated using dynamical systems. Dynamical systems may exhibit chaos and strange attractors [1] [2]. Since Internet traffic shows non stationarity and long term dependence among data samples, a cognitive polyscale approach should be taken to analyze the hidden features in a nonlinear data time series. It is necessary to estimate a reasonable window of time series so that the polyscale analysis can be performed without violating the statistical bounds of the analysis. In this work, a feature extraction algorithm is developed using variance fractal dimension trajectory and the statistical parameters of the calculation are validated using an autonomous varying window of data samples. Our analysis shows promising results since the algorithm is able to capture the presence of DNS denial of service attack and has extracted the bursts of data sample accurately.*

Keywords— Cognitive machine learning, Fractal, Polyscale, DNS DDoS amplification attacks, Anomaly detection, Cyber threats, Variance fractal dimension, Non stationary trend analysis.

I. INTRODUCTION

Analysis of internet traffic requires converting the traffic parameters into a time series. With the careful implementation of sampling intervals, the time series of the traffic depicts the estimated behavior of the internet traffic [3]. If the data series is composed of *N* independent features, then we can represent the time series in *N* dimensional space. Analysis of time series [4] can be categorized as (i) time analysis, (ii) frequency analysis and, (iii) multiscale time and frequency analysis (wavelet analysis). There is a fourth category introduced by Witold Kinsner in [5] and called as polyscale analysis. Wavelets and Multifractal analysis may be used to illustrate the scale invariance and long-term memory properties of time series and objects. Both wavelets and multifractal analysis characterize the traffic using multiscale analysis; However, multifractal analysis considers the information at multiple scales simultaneously, while wavelets considers information at different scales independently [6] [5]. Therefore, multifractal analysis is also called polyscale analysis. This term was first coined and then conceptualized by Witold Kinsner in [5] [7] to signify the difference from wavelet multiscale analysis. In addition, internet traffic streams are well characterized by self-similarity and long-term memory properties [8] [9] [10]. Polyscale analysis uses various measures of complexity to extract features of the time series. Polyscale analysis is different from traditional mono-scale analysis, such as statistical methods and Fourier analysis, in that polyscale methods not only calculate statistical information at multiple scales, but they also measure the connecting factor, the fractal dimension, which is typically obtained through a log-log plot across these scales simultaneously. [11] [12].

Polyscale analysis of a time series provides significant tools to analyze the non-linearity of the series using non integer fractal dimensions and multifractal analysis. Multifractal analysis is used to detect complexity using the self-similarity or self-affinity features of time series at different scales [13]. Mathematically, we can calculate fractal dimensions by finding the exponent of the power law relationship of the multiscale coordinates over a log-log plot [14].

Contemporary statistical analytical models estimate the statistical characteristics of a time series using probability distributions, hypothesis testing and/or various probabilistic learning techniques. Alternatively, a time series may be modeled and analyzed using dynamical systems where fractal analysis plays a significant role [4] [15]. As multifractal/polyscale

analysis considers non-integer dimensions embedded within topological or integer dimensions, the hidden complexities or features can be extracted by estimating the exponent of log-log relationship.

The distributed denial of service (DDoS) DNS amplification attack exploits the DNS protocol to amplify the payload of DNS packets. These packets do not contain any useful information and thus reduce the available and useful bandwidth of the network. In this type of attack, the attacker broadcasts a control message (a DNS request) to authentic computing nodes over the network in the disguise of originating from an authentic DNS server. It manipulates the source and destination IP address such that the victim node does not send the request towards the DNS server and the attacker. Rather, the DNS server traffic is directed towards the victim's computer [16] [17]. Now there is a one way route from a group of authentic nodes towards the victim node in such a way that the DNS recursive server sends responses towards the victim's node. Since these requests come from many authentic nodes continuously, the response of the DNS server directed towards the victim node is overwhelming and the victim node faces reduction in availability of the bandwidth and ultimately faces denial of service state and becomes unable to communicate to any network request.

As the attacker cannot be traced because the attack is launched using authentic nodes and the attacker remains anonymous, it is important to analyze the traffic continuously and extract the features based on varying characteristics of the traffic. There are various methods to detect DNS DDoS amplification attacks. The authors in [17] describe a method of mapping and monitoring the DNS mechanism of requests and responses to detect anomaly in the packet flows. This method shows better results in detection, but is limited due to scaling issues in a large network. Moreover, it is useful for local DNS servers only. In [18], the authors utilized hardware based Bloom filters to analyze DNS packets to detect DNS amplification attacks. Also, as mentioned in [16] [19], there are location based and time based methods to detect DNS DDoS amplification attacks. There are various methods to detect the attacks and include packet based signature analysis and node based collaborative techniques.

In this work, we applied variance based polyscale feature extraction mechanism to detect DNS attacks in a nonlinear internet time series with one selected and observed feature i.e. samples of DNS packet count. This technique is effective as it is scalable and works accurately for long duration attacks. Also it provides a unique measure of complexity introduced by the attack i.e. variance fractal dimension. Moreover, this work provides an important application of polyscale analysis in detecting malicious anomalies (attacks) in a nonlinear time series. Also, this work validates that a non-stationary data time series can be analyzed by autonomously estimating a set of subsample window with weak sense of stationarity.

II. VARIANCE FRACTAL DIMENSION AND TRAJECTORY

Variance fractal dimension analysis is a class of information based fractal analysis where second order statistics of the data samples at multiple scales are used simultaneously to estimate the power law relationship among the scales. For a single parameter/attribute/feature data time series, the variance fractal dimension is embedded within the topological dimension of 1 (a line) and 2 (an area) [12] [14]. Variance fractal dimension is calculated by estimating the Hurst exponent which is characterized by the fractal Brownian motion process [20]. Hurst exponent of 0.5 represents standard Brownian motion process. It is mandatory to ensure stationarity of time series before applying variance fractal dimension analysis [21].

Let $x(t)$ represents a periodically sampled data time series. It is important to note that the sampling frequency should be considered such that it follows the Nyquist sampling criterion [22] [23]. Moreover, since internet data time series consists of digital information packets therefore, we should consider sampling the data time series such that the original information should be preserved i.e. Nyquist sampling criterion. Moreover, as internet data time series already contains analog information i.e. speech signal represented by VoIP packets, we should consider the characteristics of digital process in such a way that the packet flow information should be preserved inside a digital sample [24]. In the current data set, we have considered DNS based packets which require a maximum round trip time (RTT) of 100ms. Although, we can configure higher round trip time since DNS packets are UDP packets, but our analysis of the data set provided us the maximum figure of 100ms for the DNS packets in the data set.

For the process $x(t)$, the variance of the data samples is:

$$var[f(x(t))] = E[(f(x(t)) - \overline{f(x(t))})^2] \quad (1)$$

where E[.] is the statistical expectation operator and

$$\overline{f(x)} = E[f(x(t))] \quad (2)$$

The $f(x(t))$ represents any function to calculate magnitude of samples in the multiscale calculations. For example, [12] considered $f(x(t)) = x(t_1) - x(t_2)$, where $x(t_1)$ is the first sample of the sub-window and $x(t_2)$ is the last sample of the sub-window. For our work,

$$f(x(t)) = range(x(t)) \quad (3)$$

$$range(x(t)) = \max(sample\ sub - window) - \min(ample\ sub - window) \quad (4)$$

According to power law [25],

$$var(f(x(t))) \sim |t_2 - t_1|^{2H} \quad (5)$$

where H is the Hurst parameter and is bounded between 0 and 1. If H=0, the process exhibits long range negative autocorrelation i.e. if current sample is low valued then future sample will have high value with high probability. If H=1, the process shows long range dependence and exhibits persistence of the trend i.e. if current sample is high valued then the future sample will have high value with high probability. If H=0.5, then the process exhibits no autocorrelation and samples.

Now using log,

$$\log[var(f(x(t_1) - x(t_2)))] \sim 2Hlog[\Delta t] \quad (6)$$

Therefore,

$$H = \frac{1}{2} \lim_{\Delta t \to 0} \frac{\log[var(f(\Delta x_{\Delta t}))]}{\log[\Delta t]} \quad (7)$$

The variance fractal dimension is related to H as follows [12] [25]:

$$D_v = E + 1 - H \quad (8)$$

where D_v is the variance fractal dimension embedded between integer dimension 1 and $E - 1$. For a single feature time series (i.e. DNS packet count time series), $E = 1$. Therefore,

$$D_v = 2 - H \quad (9)$$

Variance Fractal Dimension Trajectory

In order to calculate variance fractal dimension of a non-stationary time series in a continuous fashion, it is required to divide the time series in windows of data samples where each window should be chosen such that stationarity of the chosen samples is preserved. Then, VFD is calculated over each window continuously. The plot of VFD is termed as Variance Fractal Dimension Trajectory (VFDT).

Following are the important considerations for the correct calculation of variance fractal dimension in a given internet data time series:

1) Data series over which variance fractal dimension calculation is considered must show stationarity in the weak sense of second order statistics.
2) Sampling interval must be equal and should contain information of the samples reasonably i.e. round trip time (RTT) of the DNS packets. In general, the necessary condition for sampling is to know that Nyquist criterion is fulfilled. Moreover, the sufficient condition is to ensure that details of protocols and applications are analyzed properly and embedded in the data sampling interval.
3) If variance fractal dimension calculation violates the bounds of embedding dimensions i.e. calculations show negative dimension or values greater than the upper bound, then it is a sign of non stationarity and the window of samples should be varied accordingly.
4) Log-Log plot should not have saturation points in the calculations since saturation points introduces bias in Hurst value and do not contribute any polyscale information. If there are such points, consider removing them first.
5) Outliers in the calculations of variance fractal dimension should be considered as noise and the window size should be varied to remove those outliers. As the window progresses, these outliers will eventually be included in the overall calculations of VFD.
6) Only steady state samples should be considered in the data time series. Initial transient samples should be removed before applying VFDT.
7) The number of samples should be sufficiently large.

III. DATA SET

The data used in this work was the PREDICT ID USC-Lander/ DoS_DNS_amplification-20130617 (2013-06-17) to (2013-06-17) [26]. There are 19 ERF packet capture files with anonymized IPs. There are total 59,928,920 (~ 60 million) packet counts out of which there was a total of 358019 DNS packets. Out of 358019 DNS packets, 340865 packets were DNS attack packets. The total capture file size was 5.3 GB. The first packet in the file started at June 17, 2013, 21:52:45.395326000 and the last packet ended at June 17, 2013, 22:25:32.859674000. The first DNS attack packet arrived at 22:00:12 and the last DNS attack packet arrived at

22:15:34. According to the USC-Lander, this data set was composed of one DNS Denial of Service Amplification attack staged between USC/ISI, Marina del Rey, California to CSU, Fort Collins, and Colorado. The attack was performed on a single destination IP. The attacker IP was not present in the data set which used 6 DNS servers to generate a botnet network.

IV. ALGORITHMS

Data Parsing and Generating Time Series

1) Collect the PCAP or ERF capture file using Wireshark or Tshark.

2) Break the file into small chunks of 100,000 packets per chunk.

3) Read Timestamp, Source IP, Destination IP, Packet size, packet info fields into Matlab data structures.

4) Create a DNS time series of DNS packet flow.

Adaptive Window Algorithm

1) Set the following parameters:
   a. Data pointer: d_p
   b. Window size: lag
   c. Window= d_p+ lag

2) Initialize $\boldsymbol{d_p}$ at first sample of the data series.

3) Run a loop till the end of data series.

4) Pass this window through **VFDT()** function and get estimated variance.

5) Check if estimated **variance** falls within embedding integer dimensions of 1 and 2 (check for stationarity):

   a. If **variance** > 2, increase **lag** by 64 and recalculate variance. Do not increment $\boldsymbol{d_p}$. Do this till valid variance is returned from **VFDT()**.

   b. If **variance** < 0, decrease **lag** by 64 and recalculate variance. Do not increment $\boldsymbol{d_p}$. Do this till valid variance is returned from **VFDT()**.

*Variance Fractal Dimension Trajectory – **VFDT()***

1) Let N samples are considered in a given window of data. Let's call it *main window*.

2) Select the largest size of cover (samples per cover) that corresponds to the largest scale. Let's number it scale 1. It should be chosen such that the main window of samples should provide at least 30 covers in the first scale i.e. scale 1.

$$K_H \geq \left\lceil \frac{log30}{logb} \right\rceil$$

Therefore, we chose the following relation to calculate K_H as the total number of scaling levels:

$$K_H = \left\lceil \frac{logN}{logb} \right\rceil - \left\lceil \frac{log30}{logb} \right\rceil$$

3) Select the lowest size of cover such that at least 2 samples per cover are available.

$$K_L \geq 1$$

Therefore, we chose the following relation as the lower bound of scaling level:

$$K_L = 1$$

4) Run main loop for K from K_H till K_L.

5) Now set the following parameters:

   a. Total number of samples per cover at level K

$$n_K = b^K$$

   where b is a generic number base. We use b=2.

   b. Total number of cover at K- level are:

$$N_K = \left\lfloor \frac{N}{n_K} \right\rfloor$$

6) In each iteration of the main loop, run second loop from 1 till N_{K_H}. This loop calculates the difference in maximum variation in each cover as follows:

$$\Delta y_{K_{H_i}} = \max(sample\ sub - window) - \min(ample\ sub - window)$$

   where $\boldsymbol{i}$ will run from 1 till n_{K_H}.

7) When the second loop is completed, calculate the variance at level K_H as follows:

$$var(\Delta y_{K_H}) = \frac{1}{N_k - 1}\left[\sum_{j=1}^{N_k} (\Delta y_{K_H})^2 - \frac{1}{N_k}\left(\sum_{j=1}^{N_k} \Delta y_{K_H}\right)^2\right]$$

8) Calculate the coordinates on log-log plot as follows:
$x_k = \log(n_{K_H})$
$y_k = \log(var(\Delta y_{K_H}))$

9) Reduce the value of K by 1 till it reaches K_L. Return to step 5.

10) From the set of coordinates, calculate slope of the log-log plot and Hurst parameter of the *main window*.

V. EXPERIMENTAL RESULTS

In this work, an autonomous sliding window algorithm is developed to characterize a non-stationary data time series using a range based variance fractal dimension trajectory. The PREDICT data set [26] was used to perform the analysis. This dataset contains 19 ERF capture files with anonymized IPs. Moreover, the attack was recorded for 10 minutes and the packets were captured for 32 minutes and 47 seconds. Each file has more than 3.5 million packets. One target IP and six DNS server IPs are known a-priori. The files contain both attack and legitimate packets, including DNS packets. The algorithm was implemented using Matlab, and the data parsing was done by breaking a file into multiple parts, where each part contained 100,000 packets. From this dataset, we generated a time series plot of the DNS traffic.

The experiment investigated the effect of applying different window sizes and lag values over the data set. As shown in Fig. 1, the time series of DNS packet counts contains both normal and attack packets. The attack start time and end time are shown by a blue arrow. Moreover, we observe a large spike at the start of the time series (sample number 75); this represents the start of the system where nodes broadcast to DNS server for their query resolution. It is a normal process and as expected; this spike is resolved within 100ms of the time sample and the series settles down with small values. Fig. 2 shows an edited version of original time series in Fig. 1, where the early spike is removed. We can observe very high varying data series having multiple bursts of DNS packets. The attack started at sample number 729 and ended at sample number 9068. The series climbed up in the vicinity of 729 and settled down again in the neighborhood of 9068. As shown in Fig. 3, the variance fractal dimension trajectory (VFDT) of this data series (without removing the early spikes) is generated when the window size of 256 samples are chosen with a single sample sliding window.

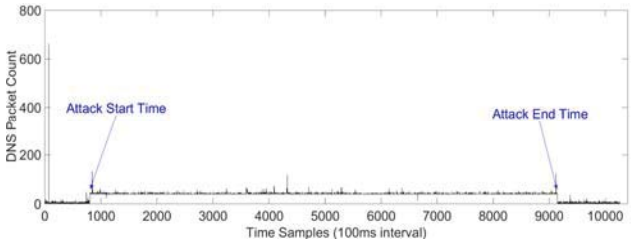


Fig. 1 DNS packet count time series.

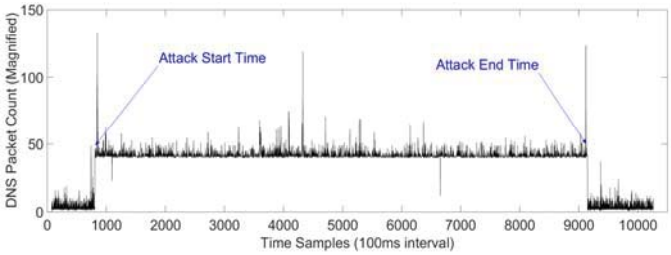


Fig. 2 DNS packet count time series (w/o first spike).

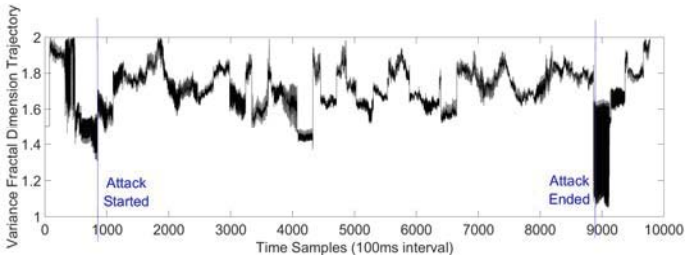


Fig. 3 VFDT, Window size=256, Lag=64.

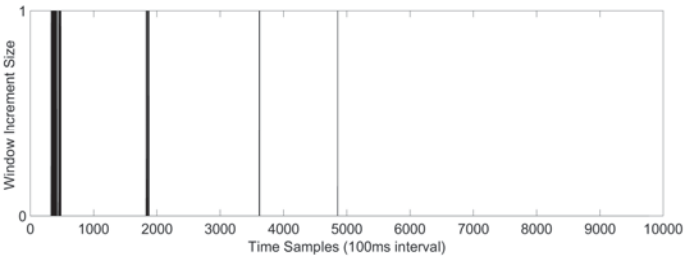


Fig. 4 Window size=256, Window breathing pattern.

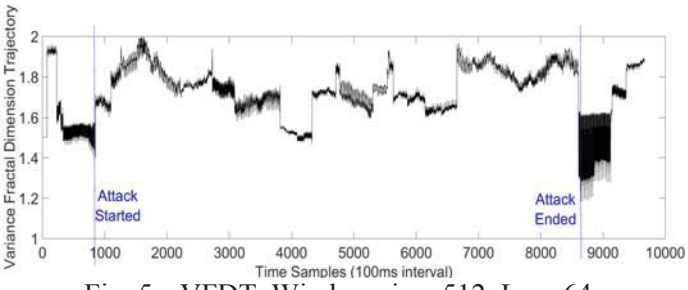


Fig. 5 VFDT, Window size=512, Lag=64.

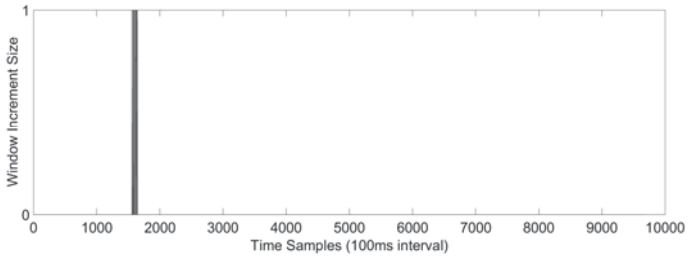


Fig. 6 Window size=512, Window breathing pattern.

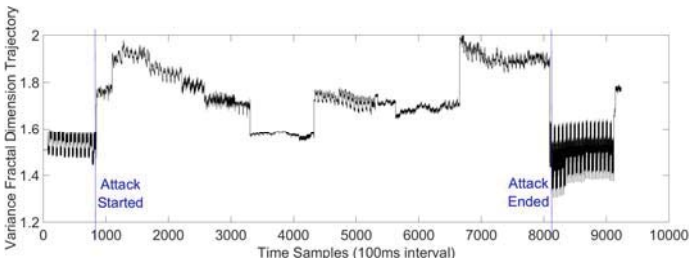


Fig. 7 VFDT, Window size=1024, Lag=64.

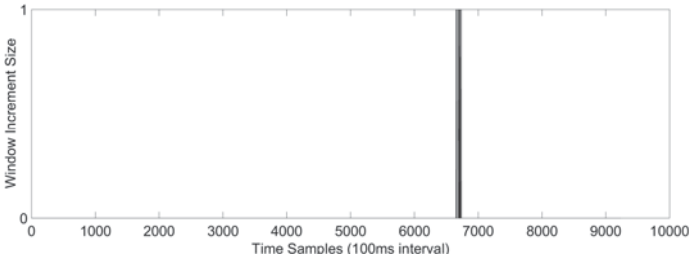


Fig. 8 Window Size=1024, Window breathing pattern.

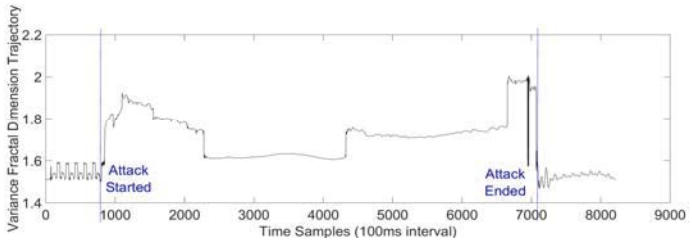


Fig. 9 VFDT, Window size=2048, Lag=64.

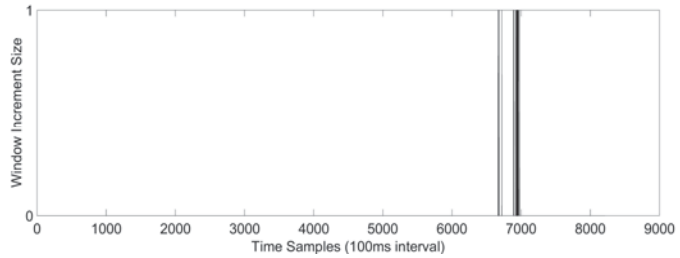


Fig. 10 Window size=2048, Window breathing pattern.

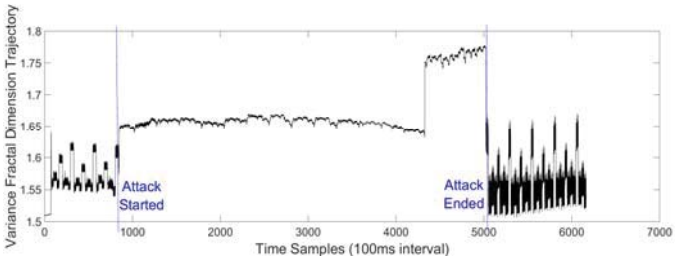


Fig. 11 VFDT, Window size=4096, Lag=64.

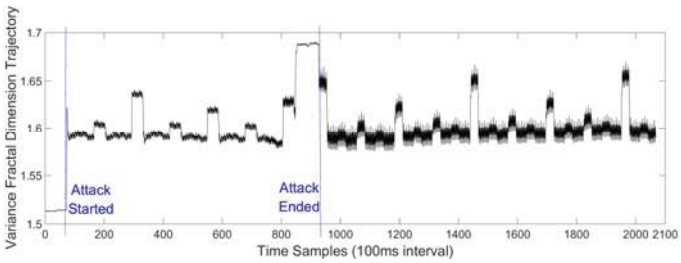


Fig. 12 VFDT, Window size=8192, Lag=64.

The VFDT remains within valid topological dimension of 1 and 2. As shown in Fig. 4, the autonomous increment of 64 samples in window size is performed in order to ensure stationarity in the weak sense of second order statistics. Moreover, it is observed that there are lot of variations in the VFDT that show the independence of calculation of variance among adjacent window sizes. With this trajectory of 256 samples per window and sliding window of single sample, the features of the data series are not only extracted but amplified. If we visually compare this trajectory with the original data series in Fig. 1, the time interval before the start of attack shows very highly varying samples with one outlier peak. Range calculations of variance based multifractal with small window size of 256 samples takes into account the presence of this outlier peak and as a result we see that VFDT stays close to the topological dimension of 2. However, after the start of the attack, the range of peaks remain comparatively low varying and therefore, we see that the trajectory falls below the fractal dimension of 1.4. Further, we see that the data series variance becomes very low varying and then starts having relatively highly varying data series. Therefore, within the time samples of 1000 and 2000, we see an increase in the VFDT trend so that it tends to be close to the topological dimension of 2. Same argument can be extended to the rest of the time series. Moreover, Fig. 4 shows the band of window size increments by 64 samples when the variance fractal dimension calculation shows invalid value i.e. greater than topological dimension of 2. We see that this autonomous breathing of window size is prominent when the data series shows sudden increase in variance that in

turn increases the trajectory close to the topological dimension of 2.

As indicated in Fig. 3, the start of attack is obvious when there is an increase in the variance fractal dimension trajectory greater than 1.4. Moreover, it indicates the end of attack when the variance fractal dimension trajectory gets below 1.2. The high change in variance occurs since the window increments are done using a single sample increment (fractal amplification). When the window pointer reaches to the normal sample, it again starts increasing since the variations in the normal data series are quite high.

As shown in figures, 0, Fig. 7, Fig. 9, Fig. 11 and Fig. 12, increasing the window size (512, 1024, 2048, 4096 and 8192 samples respectively) significantly change the envelope of variance fractal dimension trajectory. As observed, the wiggling of VFDT decreases with increasing window size and then increases in Fig. 11 and Fig. 12. Moreover, the start of attack time i.e. change of variance fractal dimension trajectory from approximate 1.5 to 2 shows dependence on the size of window and the range of variability in each window size. Also, as indicated, Fig. 12 shows start of attack at very early stage of the data series which is due to the large window size of 8192 samples. Likewise, the end time is indicated earlier following the same argument. In addition, it is also observed that increasing window size also flattens the VFDT to the extent that it no more reaches the topological dimension of 2 in the vicinity of start time of attack. This happens due to the averaging of the variance calculations at multiple scales and the presence of multiple coordinates on the log-log linear plot. Also, since we are using linear regression to estimate the slope of the log-log plot, therefore, at large window sizes, there are various points on the log-log plot and the estimation of slope is biased towards higher values. This in turn, reduces the fractal dimension calculations away from topological dimension of 2. Carrying on the same argument, window size of 4096 and 8192 samples shows no autonomous variation of window size that is another validation of our argument.

Therefore, it can be deduced that for this data set, a window size of 2048 is sufficient to show variance fractal dimension trajectory while it is able to indicate the presence of attack accurately i.e. dimension increases towards topological dimension of 2. However, the end of attack is indicated quite earlier which is also attributed to the termination of data time series (Fig. 1) that flattens the variance calculations. Moreover, during the duration of attack, the variance dimension trajectory shows lower dimensions but still remains above the topological dimension of normal data series.

VI. Conclusions

This paper has described a new variance fractal dimension trajectory calculation for internet data time series that shows non stationarity and contains malicious attacks. A new range based variance fractal dimension calculation method is presented to generate variance fractal dimension trajectory. If the variance fractal dimension trajectory reaches to the topological dimension of 2 and/or shows greater dimension than the dimension of normal traffic, we can predict the presence of an attack quite accurately. Also, we have described the method of varying data window size autonomously based on testing stationarity using weak-sense second order statistics. Our algorithm is capable of capturing highly varying data samples and is prone to correlation effects of data samples in previous windows.

VII. Acknowledgement

This work is supported in part through a research fellowship from Mitacs-Accelerate Canada. Authors are also thankful to PREDICT USA for providing state-of-the-art data sets.

References

[1] Muhammad Salman Khan, Ken Ferens and Witold Kinsner, "A chaotic complexity measure for cognitive machine classification of cyber-attacks on computer networks," *International Journal of Cognitive Informatics and Natural Intelligence (in print),* vol. 9, 2015.

[2] Muhammad Salman Khan, Ken Ferens and Witold Kinsner, "A chaotic measure for cognitive machine classification of distributed denial of service attacks," in *Proc. 13th IEEE Intern. Conf. Cognitive Informatics and Cognitive Computing, ICCI*CC 2014*, London, UK, 2014.

[3] Thanasis Vafeiadis, Alexandros Papanikolaou, Christos Ilioudis and Stefanos Charchalakis, "Real time network data analysis using time series models," *Simulation Modelling Practice and Theory,* vol. 29, pp. 173-180, 2012.

[4] Holger Kantz and Thomas Schreiber, Non Linear Time Series Analysis, Cambridge University Press, UK, 2004, pp. 87-100.

[5] Witold Kinsner, "It's time for polyscale analysis and synthesis in cognitive systems," in *IEEE 10th Intl. Conf. Cognitive Informatics & Cognitive Computing (ICCI*CC11)*, Banff, AB, 2011.

[6] Witold Kinsner, "It's time for multiscale analysis and synthesis in cognitive systems," in *IEEE 10th Intl. Conf. Cognitive Informatics & Cognitive Computing (ICCI*CC11)*, Banff, AB, 2011.

[7] Witold Kinsner, "Polyscale analysis and fractional operators for cognitive systems," in *IEEE 13th Intl. Conf. Cognitive Informatics & Cognitive Computing (ICCI*CC14)*, London, UK, 2014.

[8] Will E. Leland, Murad S. Taqqu, Walter Willinger and Daniel V. Wilson, "On the self-similar nature of Ethernet traffic (extended version)," *IEEE/ACM Transactions on Networking (TON),* vol. 2, no. 1, February 1994.

[9] Mark E. Crovella and Azer Bestavros, "Self-similarity in World Wide Web traffic: Evidence and possible causes," *IEEE/ACM Transactions on Networking,* vol. 5, no. 6, pp. 835-846, December 1997.

[10] Patrice Abry, Richard Baraniuk, Patrick Flandrin , Rudolf Riedi and Darryl Veitch, "Multiscale nature of network traffic," in *IEEE Signal Proc. Mag.*, 2002.

[11] Changzheng Chen, Zhong Wang, Yi Gou, Xinguang Zha and Hailing Miao, "Wavelet based multifractal analysis to periodic time series," *Journal of Computational and Nonlinear Dynamics,* vol. 10, no. 1, September 2014.

[12] Witold Kinsner and Warren Grieder, "Amplification of signal features using variance Ffactal dimension trajectory," *International Journal of Cognitive Informatics and Natural Intelligence,* vol. 4, no. 4, pp. 1-17, Oct. 2010.

[13] Yingxu Wang, Jean-Claude Latombe, Du Zhang and Witold Kinsner, "Advances in Cognitive Informatics and Cognitive Computing," *International Journal of Cognitive Informatics and Natural Intelligence,* vol. 3, no. 4, pp. 91-95, 2009.

[14] Witold Kinsner, "A unified approach to fractal dimensions," *Int'l Journal of Cognitive Informatics and Natural Intelligence,* vol. 1, no. 4, pp. 26-46, 2007.

[15] Robert L. V. Taylor, "Attractors: Nonstrange to Chaotic," *Society for Industrial and Applied Mathematics, Undergraduate Research Online,* pp. 72-80, 21 6 2011.

[16] Saman Taghavi Zargar, James Joshi and David Tipper,, "A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks," in *IEEE Communications Surveys & Tutorials*, 2013.

[17] Georgios Kambourakis, Tassos Moschos, Dimitris Geneiatakis and Stefanos Gritzalis, "Detecting DNS amplification attacks," *Lecture Notes in Computer Science,* vol. 5141, pp. 185-196, 2008.

[18] Changhua Sun, Bin Liu and Lei Shi, "Efficient and low-cost hardware defense against DNS amplification attacks," in *IEEE GLOBECOM*, 2008.

[19] Tao Peng, Christopher Leckie and Kotagiri Ramamohanarao, "Survey of network-based defense mechanisms countering the DoS and DDoS problems," *Journal ACM Computing Surveys (CSUR),* vol. 39, no. 1, 2007.

[20] Masahiro Nakagawa, "A critical exponent method to evaluate fractal dimensions of self-affine data," *Journal of Physic Society Japan,* vol. 62, 1993.

[21] Angkoon Phinyomark, Pornchai Phukpattaranont and Chusak Limsakul, "Fractals, Applications of Variance Fractal Dimension: A Survey," *Complex Geometry, Patterns, and Scaling in Nature and Society,* vol. 22, no. 1, 2014.

[22] Athanasios Papoulis and S. Unnikrishna Pillai, Probability, Random Variables and Stochastic Processes, 4 ed., McGraw-Hill, 2002.

[23] Alan V. Oppenheim, Ronald W. Schafer and John R. Buck, Discrete-Time Signal Processing, 2 ed., Upper Saddle River, NJ: Prentice-Hall, Inc., 1999.

[24] Peng CK, Havlin S, Stanley HE and Goldberger AL., "Quantification of scaling exponents and crossover phenomenon in non stationary heart beat times eries," *Chaos,* vol. 5, no. 1, 1995.

[25] Brandon J. Whitcher, Simon D. Byers, Peter Guttorp and Donald B. Percival, "Testing for homogeneity of variance in time series: Long memory, wavelets, and the Nile river," *Water Resources Research,* vol. 38, no. 5, 2002.

[26] PREDICT-USC-Lander-DoS_DNS_amplification, "Scrambled Internet Measurement, PREDICT ID USC-Lander/ DoS_DNS_amplification-20130617 (2013-06-17) to (2013-06-17) provided by the USC/Lander Project.," 2013.

Deception in Dynamic Web Application Honeypots: Case of Glastopf

B. Mphago, O. Bagwasi, B. Phofuetsile, and H. Hlomani
College of Information Communication and Technology
Botswana International University of Science and Technology
Palapye, Botswana

Abstract— *Websites contain critical information to both the organization and the customers. With the cyber security threats currently on the rise, websites are getting easily compromised which prompts administrators to find ways to secure them from the black hat community. The use of honey pots as an alternative to other methods of securing web sites had brought with it advantages as well as disadvantages. Glastopf as one of the web application honeypots brings with it features that emulates a real server. It replies to the attack using the response that the attacker is expecting from his attempt to exploit the real server. However, Glastopf has its disadvantages too. Once deployed, Glastopf can be easily identified by the attackers due to the simplicity and static nature of its web-page templates. This paper seeks to improve the camouflage nature of the honeypot by proposing a new frame work which will produce dynamic web pages that will completely disguise the fake pages from the users.*

Keywords: honeypot security, deception, dynamic web pages, cyber attack

1. Introduction

Honeypots have emerged as great tools in tracking hackers and learning their attack methods. Traditional security tools such as firewalls, intrusion detection systems, and proxy servers, and as such, a tool that learns hacking mechanisms becomes evidently important to the security community. Many honeypots designs have been proposed and some of their configurations have been a challenge in many aspects [1]. The main intended purpose of a honeypot is to deceive an attacker by re-directing him/her to spoof hosts lines that will provide phoney information that appears to be informative or important after an analytic process was done on the attack signature. This suggest that the honeypot needs to be as dynamically deceptive as possible in all aspects in-order to achieve its goal of making attackers believe they have managed to gain access to real system, and most of honeypots are unable to achieve dynamicity to conceal the fact that they are honeypots, Glastopf being an example hence failing its primary task as a honeypot.

2. Background

As part of changing technologies, a honeypot can be involved in different aspects of security such as prevention, detection, and information gathering [2]. Lance Spitzner, defined a honeypot as a security resource whose value lies in being probed, attacked, or compromised [3], and for the purpose of this paper, we will adopt this definition mainly because it covers all the aspects that we believe a honeypot should be. A web application honeypot (WAH) in particular, is a basic web server with an attack surface [4]. This attack surface is the public HTML content which is indexed by search engines, and it contains links to files with known vulnerabilities.

Honeypots can be classified according to their purpose (research and production honeypots) and the level of interaction (low, medium, and high). A research honeypot is designed to gain information about black-hat community and does not add any direct value to an organization [3]. They are used to gather intelligence on the general threats organizations may face, allowing the organization to be better protected against those threats. A production honeypot is one used within an organizational environment to help protect the organization and mitigate the risk. Honeypots deployed in a production environment serve to alert administrators to the potential attacks in real time [5]. Low interaction honeypots are primarily production honeypots that are used to help protect a specific organization [3]. They only simulate services that cannot be exploited to gain total access to the honeypot. Medium interaction honeypots are slightly more sophisticated than low interaction honeypots but less sophisticated than high interaction honeypots [2], and like low interaction honeypots, they do not have an operating system installed, but simulated services are more complicated technically. High Interaction honeypots are actual systems with full-blown operating systems and applications. They are the extreme of honeypot technologies.

Currently, there are five major web application honeypots that are published and made available to the security community: HIHAT [6], DShield Web Honeypot Project [7], Google Hack Honeypot [8], PHPHoP [9], and Galstopf, being the most recent and the most sophisticated ever produced by The Honeynet Project. The first four honeypots above have one

thing in common: all of them use modified templates from real web applications to pretend that they are vulnerable and attractive to attackers. Thus, all of these honeypots are static, meaning you have to write new templates to support new vulnerabilities, and this can be time consuming and is a reactive process. However, the advantage to the template is that the honeypot looks very similar to a real victim and eventually will entice more manual and more complex attacks. The static limitation of this honeypots led to the development of another web application honeypot (Glastopf), which is a dynamic low-interaction web application honeypot capable of adapting to new and changing environments, thus making it a more reliable web application honeypot. The second reason that led to the development of Glastopf was the limited ability of the previously mentioned honeypots to deal with multistage attacks [9]. However, despite all the goods and praises about this honeypot, Glastopf still have some limitations in some quarters, the most notable one being deception by its webpage templates.

3. Motivation

In this paper, we investigate the Glastopf's deceptive qualities when queried through specially crafted requests. Our main focus is on the webpage templates it supplies when queried. The avenue for this is to theorize on how Glastopf web application honeypot can better deploy and avail its webpage templates and still stay deceptive enough from the black-hat community. We discovered that as of current, the webpages supplied by Glastopf are too basic for experienced hackers to see that they are not coming from a real system. This may be due to, as stated by Cohen [10], that the creators of Glastopf (The Honeynet Project) were not directed so much at deception to defeat the attacker in the tactical sense as at intelligence gathering for strategic advantage, but rather, unlike most historic honeypots, the creators are dedicated more at learning about the tool, the tactics, and motives of the black-hat community and sharing lessons learnt. This is despite the fact that a honeypot that can not hide itself entirely loose its value once detected. Its analogous to a trap without camouflage, attackers will simply avoid it, and as such it will not serve its purpose. In order to perform its function, a honeypot has to avoid being discovered. Responses given by a honeypot need to ensure they mimic that of the original system well so as not to raise suspicion, and this is normally the most difficult part when dealing with an attacker who has intimate, expert knowledge of a service or particular protocol[11].

4. Conceptual Framework

This section discusses the theoretical framework advanced by this paper. This is discussed by first exposing the limitations in the current set of Glastopf and further detailing a framework to address the identified weaknesses.

4.1 The Current Status

Glastopf is a low-interaction, dynamic web application honeypot capable of emulating thousands of vulnerabilities to gather data from attacks that target web applications [6]. It accomplishes its goal by deceiving the attackers that it is a hosting application with a list of vulnerable paths/ scripts, often referred to as dorks by its developers. These dorks are published in search engines and crawlers, which are then indexed and included in search results that attackers collect when they search for vulnerable paths in the web. So, when an attacker finds these published paths in search engines, he will, most probably, attempt to perform an attack on them, among other vulnerable paths that were not listed in the search engine. When these new vulnerable paths are detected, they will also be advertised and indexed in the search engines, thereby attracting new attackers looking for those paths.

Now, in the current situation, the webpage templates supplied by Glastopf when queried are static and too basic for experienced hackers to see that they are not legitimate web pages from a legitimate web-server. What happens now is when you install Glastopf you find templates in the data file within the honeypot, that can be customized to suit your needs. This means once customized, the template becomes static and does not necessarily provide everything the attackers wants to see.

4.2 The Proposed Framework

In this paper we therefore propose a framework which will provide a more secure way to camouflage webpage templates from the black hat community by introducing dynamic web pages as a replacement to the current static web pages. We employ a Content Management System, Wordpress to be precise, in designing these templates, which will give them a more realistic look that they are real web pages. Once the honeypot determines what the hacker wants, it emulates a response, either being the webpage itself or vulnerable scripts within the page, and send it back to the attacker. In our proposed solution, we build another module that tells the honeypot to auto-populate the templates with the information the hacker wants to see. Fig. 1 depicts the proposed model overview of how the dynamic web page can be integrated to the honeypot. This typically starts with an attacker sending a query to the honeypot (1). Whereupon the honeypot determines what kind of an attack this is and sends a request for a web page template to the template distribution module (2). The template distribution module then communicates with the template population module, giving it information on how to populate the required webpage template (3). The population module then interfaces with the inference engine and the knowledge base for the creation of the template (4). The inference engine makes use of the dummy database , UI Elements and Vulnerable code to create the desired template

(5). The reverse of these steps end up with a template that was requested by the honeypot being sent to the attacker.

An attacker normally sends a request to the honeypot with the belief that they are attacking a real system. The honeypot with its intelligent ability to process requests sends a request to the template population module. This is an expansion to the already existing mechanism of processing requests and returning the dummy webpages to the attacker. At the moment, the web pages generated by this honeypot are static, this is the main loophole that makes the honeypot less deceptive, and in the proposed frame work we seek to address this issue by building the template with a CMS and then build another module that directs the honeypot to populate the template with what the attacker is expecting to see.

The principle behind Glastopf is that a reply is sent to the attacker using the response the attacker is expecting to see from his attempt to exploit the web application. Our query processor which is made of two parts, the inference engine and the knowledge base lies as a middle framework between the template population module and the web page content. Once an attacker makes a request to the honeypot, the honeypot already knows the kind of response to give back to the attacker. Before giving the response to the attacker, the Query processor is responsible for auto populating the web template created using a CMS like word press. The inference engine acts as an expert system which contains rules responsible for mapping the page design commands with retrieval of page elements from the UI Elements store. The page elements which are prone to exploits are mixed with the dummy data and other elements are combined using rules defined in the inference engine. The knowledge base provides the facts that are known to the world and these facts are dynamically added to the page template as defined by the rules in the inference engine. After the Query processor successfully creates a dynamic web page, it is sent back to the honeypot containing elements which have been requested by the attacker.This process is depicted in Fig. 2.

5. Discussion

In this section we discuss and compare Glastopf with one of the closely related adaptive web application honeypots concepts, Heat Seeking Honeypots, and highlight on the limitations depicted by both of the honeypots. The actual reason for the comparison of the two honeypots is based on the fact that they are the two widely known adaptive and intelligent web application honeypots as of current.

Heat-seeking honeypots is not a honeypot per-se, but rather a framework in adaptive web application honeypots. These types of honeypots actively adapt to emulate the most popular exploits and construct pages that are similar to the ones targeted by attackers [9]. Heat-seeking honeypots have four components: first, they have a module to identify web pages that are currently targeted by attackers. Second, web pages are generated based on the queries the honeypot gets without manually setting up the actual software that is targeted. Third, these links are advertised through search engines and all the received attack traffic is logged. And finally, the honeypot logs are analyzed to identify attack patterns.

Fig. 3 summarizes the overall functionality of heat-seeking honeypots. First, attacker queries from the feed are searched using search engines, and the pages from the search results are collected. Then the pages are encapsulated and put on the heat-seeking honeypots, along with real software installed on virtual machines. The next stage would then be to advertise the pages in search engines and crawlers. In this case, when attackers issue similar queries to search engines, the honeypot pages are returned in the search results, and then the interactions with the attackers are logged for further analysis.

Just like heat-seeking honeypots, Glastopf is also an adaptive and intelligent web application honeypot; Both Glastopf and Heat-seeking honeypots have vulnerable scripts in their webpages which are advertised in the search engines for the attackers to attack. Unlike heat-seeking honeypots where vulnerable pages similar to the ones attackers are looking for being fetched from the web, in Glastopf, the webpage templates are not fetched from the web, but rather, they are pre-installed in the honeypot itself as static web page templates. These templates can be customized by Glastopf user according to their own specifications during installation, and be fed to the attackers when queried.

The authors believe that the addition of these dynamic auto populated web templates will fully give the honeypot what it is lacking at the moment, complete deception.

6. Conclusion

Honeypots provide a very good starting point of protecting web applications against malicious attackers. They accomplish this by hiding themselves from the attackers and respond as if they are a compromised system, and by so doing , they learn how an attacker is exploiting it and reports back to the security administrators who can make use of this information to secure their web applications. Glastopf is the only adaptive and intelligent web application as of current, which is believed to be the future of web application honeypots. However, Glastopf also lacks in the aspect of deception as it provides basic and static web pages which experienced hackers can easily identify. The improvement of this feature would be the production of dynamic web pages which will completely emulate real web pages from a real system, and as such the attacker has less chance noticing that they are interacting with a honeypot rather than a real system. The success of the implementation of the proposed framework will not only be a breakthrough to combat cyber security but will also serve as a platform for greater innovations in the security industry as a whole.

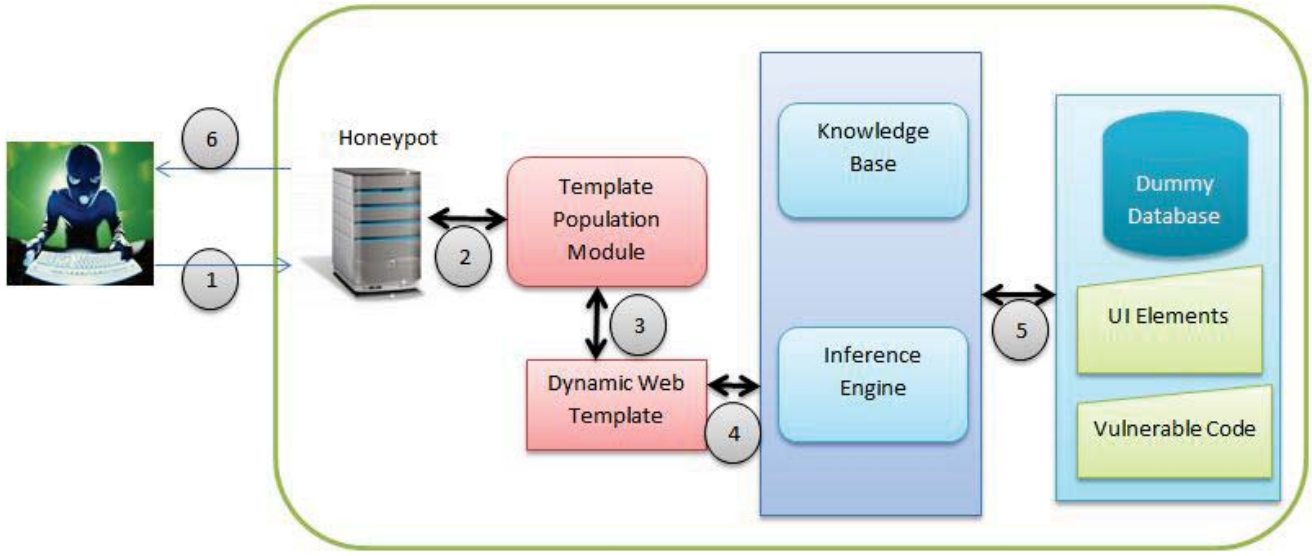


Fig. 1: Generic architecture for dynamic web page generation

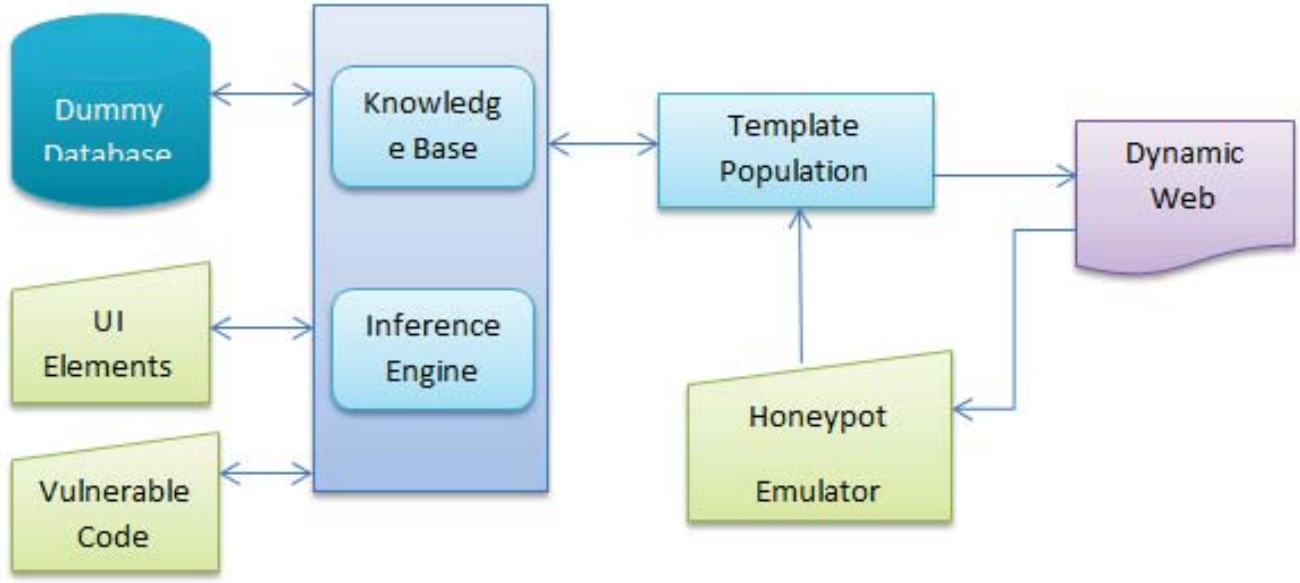


Fig. 2: An overview of the proposed model for dynamic web page integration into the honeypot

7. Future Works

In this paper, we discussed theoretical ideas on how Glastopf webpage templates can be improved and made dynamic in order for them to stay deceptive. We introduced Word-press to build these webpage templates and then discuss auto template population module for populating our templates. In our next paper, we will extend these ideas by employing empirical analysis and design of the proposed ideas. The idea is to create a python script that will direct the honeypot on what to populate the webpage template based on what the attacker expects to see. Then we would measure and analyze the response patterns of our content population module, hence compare it with that of a normal web server. This would basically give us an understanding that our proposed ideas have been a success.

8. Acknowledgment

Our research activities are sponsored by the Botswana International University of Science and Technology (BIUST) and hence we would like to extend our gratitude to the university for affording us the opportunity to contribute to the body of knowledge web and cyber security.

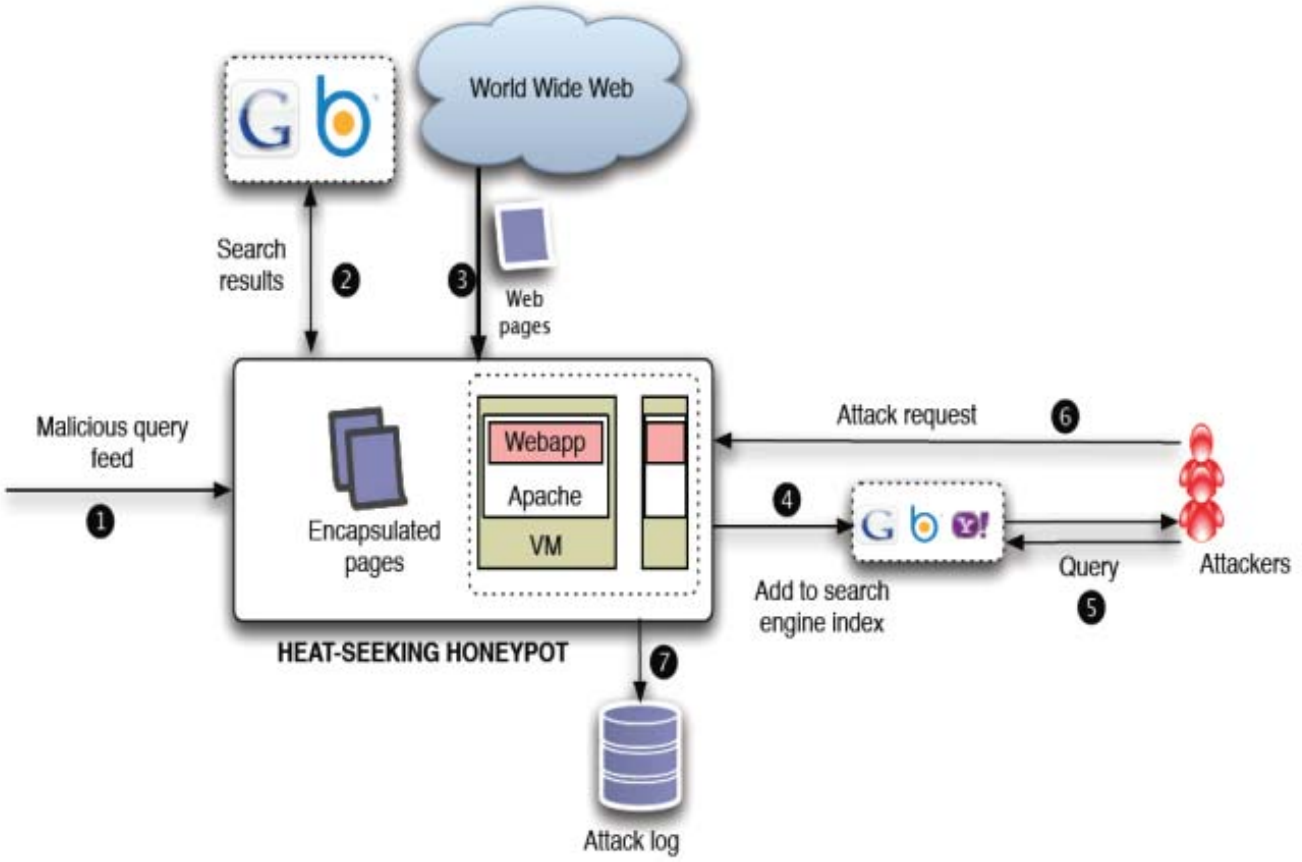


Fig. 3: Heat Seeking Honeypots Functionality

References

[1] G. Wagener, S. Radu, E. Thomas, and A. Dulaunoy, "Adaptive and self-configurable honeypots," in *12th IFIP/IEEE International Symposium on Integrated Network Management*, Luxembourg, 211, pp. 345–352.

[2] I. Mokube and M. Adams, "Honeypots: Concepts, approaches, and challenges," in *Proceedings of the 45th Annual Southeast Regional Conference*, ser. ACM-SE 45. New York, NY, USA: ACM, 2007, pp. 321–326.

[3] L. Spitzner, *Honeypots:Tracking Hackers*, 1st ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2002.

[4] Honeypot Project. (2012) Cyber fast track: Web application honeypot. [Online]. Available: https://honeynet.org/files/CFT-WAH-FinalReport.pdf

[5] M. Gibbens and R. Vardhan. (2012) Honeypots. [Online]. Available: http://www.cs.arizona.edu/ collberg/Teaching/466-566/2013/Resources/presentations/2012/topic12-final/report.pdf

[6] HIHAT. (2007) High-interaction honeypot analysis tool. [Online]. Available: http://hihat.sourceforge.net/index.html

[7] DShield Web Honeypot Project. DShield Web Honeypot Project. [Online]. Available: https://sites.google.com/site/webhoneypotsite/

[8] R. McGeehan, G. Smith, B. Engert, K. Reedy, and K. Benes. GHH, The Google Hack Honeypot. [Online]. Available: http://ghh.sourceforge.net

[9] L. Rist, S. Vetsch, M. Koğin, and M. Mauer. (2010) Know your tools: A dynamic, low-interaction web application honeypot. [Online]. Available: https://www.honeynet.org/sites/default/files/files/KYT-Glastopf-Final_v1.pdf

[10] F. Cohen, "The use of deception techniques: Honeypots and decoys," *Handbook of Information Security*, vol. 3, pp. 646–655, 2006.

[11] S. Innes and C. Valli, "Honeypots: How do you know when you are inside one?" in *Australian Digital Forensics Conference*, 2006, p. 28.

SESSION

SECURITY APPLICATIONS I

Chair(s)

Dr. Levent Ertaul

Cloud Storage Client Application Evidence Analysis on UNIX/Linux

R. Malik[1], N. Shashidhar[1], and L. Chen[2]
[1]Department of Computer Science, Sam Houston State University, Huntsville, Texas, USA
[2]Department of Information Technology, Georgia Southern University, Statesboro, Georgia, USA

Abstract - *The research proposed in this paper focuses on gathering evidence from devices with UNIX/Linux systems (in particular on Ubuntu 14.04 and Android OS) in order to find artifacts left by cloud storage applications that suggests their use even after the deletion of the applications. The work performed aims to expand upon the prior work done by other researches in the field of cloud forensics and to show an example of analysis. We show where and what type of data remnants can be found using our analysis and that can be used as evidence in a digital forensic investigations.*

Keywords: cloud storage forensics, cloud application artifacts, data remnants, data carving, UNIX/Linux forensics, digital forensic investigations.

1 Introduction

The evolution of cloud computing [1] has certainly increased the importance of cloud forensics in the past few years. In particular, many businesses are now shifting their services from standalone computer devices to the cloud environment [2]. Cloud computing has changed the way by which digital data is stored, processed, and transmitted. The increased use of the cloud environment has brought forth many issues and challenges for digital forensics experts [3]. In fact, the National Institute of Standards and Technology (NIST) has identified 65 of these challenges that needs to be addressed [4]. Nowadays, most people download on their computer or other devices, such as smartphones or tablets, one or more of the popular cloud storage services applications, such as Google Drive, Dropbox, and Microsoft OneDrive. However, there are alternatives to these cloud services that are gaining greater attention by the public. When performing a digital investigation on these user devices, it is possible to find artifacts that suggest the use of any of the above mentioned applications. Previous research has confirmed that these applications do leave a trace even after they are removed on the client devices. These traces, which contain account information, settings, and user activities can be helpful when conducting a digital forensic investigation. However, prior work in this area has only been done on Windows systems, in particular on a Windows 7 operating system. No work so far has been done on UNIX/Linux, even though UNIX/Linux-based operating systems are commonly used as well, especially when providing applications on a server. The research proposed in this paper expands upon the work done by prior research. The chosen operating system is one of the most known Debian-based Linux distributions, Ubuntu. It is very likely that in the most of the cases, a user downloads the client application on multiple devices. For example, the client application can be downloaded on both a PC and a smartphone. Therefore, a second objective of this research was to perform the gathering of evidence on a smartphone, in particular on Android OS. There are multiple cloud storage services options on UNIX/Linux, however, not very surprisingly, Microsoft OneDrive does not provide a client, and very surprisingly enough, at the time of this writing, an official client of Google Drive has not been released for UNIX/Linux yet. Therefore, in order to perform this research, the best alternative to the mentioned client applications were used, such as Copy and OwnCloud. In addition, Dropbox was also analyzed since it represents the most used cloud storage service [5].

2 Prior Work

The following literature review, explores the procedures and approaches used by other researchers in this particular field. Three main prior and related researches were analyzed to discover the approach taken in order to collect artifacts. Artifacts collected can be files either accessed or modified by the cloud storage applications on the client devices, or artifacts related to web-based cloud storage services (which are accessed through a web browser). Two main approaches were identified. The first approach represents a presumption of where artifacts should be located on a device, and then perform a search in those specifics locati ons, based on the examiner's knowledge. Meanwhile, the second approach is based on the use of programs and tools, such as Process Monitor from the Sysinternals Suite [6] to determine the location, in a dynamic manner, of the artifacts and data remnants. All the prior work done in this field was performed on a Windows 7 system using virtual machines. The following is a brief discussion of the prior work.

The paper by H. Chung, J. Park, S. Lee and C. Kang [7] provides a procedure to investigate devices such as PCs and smartphones. According to this procedure, the investigator collects and analyzes data from all devices that a user has used to access a cloud storage service. Based on the type of the device that is being analyzed, the procedure can take a different approach. Simply put, if the device is a PC then it is very important to collect volatile data from physical memory (if live forensic analysis is possible) and nonvolatile data such as files, directories, internet history, and log files. Physical memory contains useful information about users and their activities. For example, physical memory can contain login attempts and login credentials used to access cloud storage accounts through a web

browser. If the device is instead a smartphone, and if the system is running Android OS, after rooting, it is possible to collect data from the main system folders. In the case of an iPhone, after connecting the device to a PC, important data of user activity related to the cloud can be found in backup files or in data synchronized with iTunes. Once all data is collected an analysis in order to find useful artifacts is performed. According to the paper, cloud-storage services can be web-based services accessed through a web browser or client applications installed on the device. In the first case, it is essential for an expert to analyze data such as web browser log and database files (cache, history, cookies, and downloaded files) that are stored in the user profile directory on a Windows system. Cache files include downloaded image files, text files, icons, HTML files, XML files, download times, and data sizes. History files contain visited URLs, web pages titles, visit times, and the number of visits. Cookie files store information about hosts, paths, cookies modification times, cookies expiration times, names, and values. Download lists include local paths of downloaded files, URLs, file size, download times, and whether downloaded files were successful. Through such web browser files an expert can identify a user's activity, including access or logins to a cloud storage service. However, when a client application is installed on a Windows system, traces of it are left in the registry, log files and database files. Mac systems have similar traces except registry files. These files are essential during a digital forensic analysis, since they provide proof of the use of a cloud storage service. These log files contain information such as logins attempts, if and when services were used, and times of synchronized files. Database files contain information about synchronized folders and files (creation times, last modified times, and whether files were deleted) on a PC. All this information can be used to create a timeline of the user activity. In a smartphone device traces are left in database files, XML files, and plist files (which contain information about a user account). Finally, the rest of the paper provides examples of forensic analysis and shows where data is found on a PC or a smartphone. The cloud services that were used in this work, are Amazon S3, Dropbox, Google Docs, and Evernote. In the research work done by M. Katz and R. Montelbano [8], to obtain the locations of artifacts, Process Monitor is used. The result is filtered to show the file system activity, and changes to the registry and files. The cloud storage applications used in this research are SkyDrive (now OneDrive), Dropbox, and Google Drive. When SkyDrive was installed 4959 artifacts were either created or modified. Presence of the files modified using the client, was found in unallocated space, $Recycle.Bin CSV files, pagefile.sys, and inside the AppData folder. In the case of Dropbox installation, 4163 artifacts were either created or modified. Evidence of deleted files was found in unallocated space and in pagefile.sys. During Google Drive installation, 9438 artifacts where either created or modified. Evidence of files modified or deleted was found in unallocated space, $Recycle.Bin CSV files, pagefile.sys, and configuration files. The result of this research proved that a large number of files are affected during the installation of the application, and a large number of files are left behind, once the uninstallation process is completed. Evidence of file manipulation was mainly found in the unallocated space, $Recycle.Bin CSV files, and pagefile.sys. The type and number of artifacts varied depending on the application, but evidence of the use of the cloud application was still present after uninstallation of the client in all the cases.

The following research, performed by D. Quick, B. Martini, and R. Choo [9], provides a well formatted methodology and a very exhaustive analysis of data remnants left by cloud storage applications. This research is performed on a Windows 7 machine and the cloud storage services analyzed are Microsoft SkyDrive (again, now OneDrive), Dropbox, and Google Drive. The objective of the research was to solve questions, such as, which data remains on the hard disk after a user used the client software? Which data is left once the user has had access to the cloud storage through a web browser? What is the location of the data remnants on the operating system and in the memory? Other questions that were attempted to answer relate to network traffic data and smartphones. Based on the work of this research, artifacts of files either access or modified, and data remnants left behind by the applications are found inside prefetch files (which are used to analyze the software activity, such as the number of times the software has run or the associated files used by the application), registry files (they can contain references, activities, settings or other information), link files (files' shortcuts), thumbnails pictures within the thumbcache, event logs (which contain information relating to system, software, and other events recorded by the operating system), and finally, directory lists file ($MFT files). Forensic analysis has been also performed on the memory, $Recycle.Bin (in order to find deleted files), client applications (analysis of installation path, sample files, synchronized files and folders), on the account accessed through a web browser (can contain information about the number and the type of devices used to access the storage space), and finally, on the files related to the browser. Network traffic was also captured and analyzed to find activity related to login sessions. To conclude, data carving was performed through allocated and unallocated data. Thumbnails icons and large size pictures were recovered. This research, used a dynamic approach: tools to dynamically find evidence that were used were Process Monitor, Wireshark, among others. Another research was performed by M. Epifani et al [10], on Microsoft SkyDrive (OneDrive), Google Drive, Dropbox, and iCloud. Again in this case, the collection of artifacts left behind by the applications was performed on a Windows 7 system. To track the disk usage DiskPulse was used (to determine information related to created, modified and deleted files), Regshot, and RegFromApp were used to track registry changes. By monitoring the registry changes, researchers were able to obtain installation locations and installed client applications versions. Other useful data was collected from configuration files present in the installation folder (inside the user profile), from online accounts (information about deleted files, devices connected to the account, version history for every file, and last browser sessions), from the memory (it can contain user email, display name, filecache.dbx path, server time, file list, deleted files, username and passwords in the case of a web-based storage access), from Hiberfil.sys and pagefile.sys, link files, browser history and cache, registry point, and volume shadow copies. As

we can see, this research collected evidence from the same locations as the previous researches.
To conclude this literature review, we can assert that the procedures and approaches taken in these prior research works are in concordance with each other. Even if the approaches were of two different types, the locations analyzed and the data remnants found were similar.

3 Methodology

To perform our research, different programs and tools were used. *VMWare Workstation 10* was used to create a virtual machine of *Ubuntu 14.04 ("Trusty Tahr")* 64-bit. Once the installation was complete, a snapshot was taken in order to revert to a clean state once the analysis of the client application was concluded. After the client application was installed, a keyword search on the system was used to find the locations of the files and other artifacts. Specifically, the command "*find / print | grep –i 'keyword'*" was executed on the terminal, and the output was filtered to gather paths, files, and other artifacts. Unfortunately, there is not a version of *Process Monitor* (from the *Sysinternals Suite*) for Linux. Nevertheless, the command *lsof* (list open file) can be executed to list files opened by the client process, by using its PID. The tool used to acquire memory is *LiME* [11]. In addition, a keyword search (using for example login credentials) on the memory was performed in order to find artifacts.
Ubuntu 14.04 64-bit uses *Ext4* as a default file system and to find where the locations of files on the file system, *TheSleuth Kit* [12] was used. In particular, the *istat* tool is useful to find *inode* information, status of directories, and status about files inside the main directory that are created by the cloud storage application in order to synchronize files with web-based the account. The *istat* tool was also used to determine the group to which the *inode* belongs, and to find the block address where the content of the data is stored. Using the *dd* command it is possible to create an image of the partition or, as explained later, to image ranges of blocks and groups. The images can be viewed with a *hex editor*. *Foremost* [13] is an example of a data carving tool that makes it possible to recover deleted files. Finally, *SQLite Browser* [14] was used to open database files created by a web browser, such as *Mozilla Firefox,* or by cloud storage applications. The *Android OS* analysis was performed on an *Android OS x86 RC1* virtual machine.

4 Main Research

4.1 Research

As pointed out in the previous work section, there are two means to connect to the cloud storage account. The first is through a web browser, while the second is through a client application. It is important, therefore, to gather information in both cases. This section contains many different subsections that explain where evidence can be found on both Mozilla Firefox (which is installed by default on Ubuntu operating systems), and in the main directories of the client applications. As previously seen, another source of useful information is found on a running system inside the physical memory. Therefore, a subsection is dedicated to the volatile information found in the physical memory. One subsection will also briefly describe where other artifacts can be found on the file system for all the applications and, finally, the last two subsections are dedicated to the possibility of recovering deleted files, and the gathering of data on Android OS.

4.2 Web Browser Analysis

All the web-based accounts for all the applications were accessed through Mozilla Firefox and, after operations such as uploading, deleting, and modifying files, the memory was captured and analyzed. Inside the memory it is possible to find in plain text users' credentials such as email addresses, usernames and passwords in plain text, user IDs, first names, last names, logins attempts, timestamps of logins, paths on the server, server names or addresses, references to files accessed, timestamps of creation times, data of last synchronization, files uploaded, files sizes, files modification times, files deletion times, messages and actions taken by the server. Accessing the hidden folder inside the user's home directory, by default named *.mozilla*, it is possible to find database files that store useful information. The database file *contentprefs.sqlite* contains preferred websites, such as the cloud storage websites accessed. The file *cookies.sqlite* stores cookies that are created when the cloud storage website was accessed. The *formhistory.sqlite* database file contains the history and tracks the websites visited by the user, the number of visits, along with the last visited time. The database file *places.sqlite* contains multiple tables where evidence of the use of the cloud storage services websites can be found. For example, the table *moz_places* stores the name of the server, paths, and files accessed on the server, while the table *moz_hosts* and *moz_favicon* store hosts names and icons paths. Finally, the file *login.json* stores logins attempts, hostnames, usernames, URLs of the forms in which credentials were inserted, encrypted passwords, and MAC times. Inside the user's home directory, the hidden directory *.cache* contains thumbnails that show pictures of websites and accounts accessed.

4.3 Client Application Analysis: Copy

When the Copy client application is executed it launches a process is called *CopyAgent*, which functions as a background process and enables the application to properly run and execute its operations. Once the location of the data remnants was found, by sorting through the artifacts, it is possible to determine that the main evidence resides in the Copy main folder and the hidden folder *.copy*, both present in the user's home directory. The Copy main folder contains the files that are synchronized with the account, a hidden folder called *.copycache*, which contains temporary files (references to these files were found during the memory analysis), and a hidden text file called *.user_info*, which does not contain any useful information. The hidden *.copy* folder contains two directories: *cache* and *resources*, however they are both empty. The most interesting information is contained in the following files: *config.ini* contains the host *UUID*, which is an alphanumeric string of 32

characters; the file *config.db* contains settings, such as the root cache path, the database version, the cloud server address, the root folder path, the authentication token, the client ID, the user ID, the user first name, last name, email address, the push token and push URL, among other settings. The file "*copy <user_email>.db*" (in this research the email address used is *cloudstorage.test.mail@gmail.com.db*) contains a list of files that are uploaded on the cloud storage space or the files that are synchronized with the local folder. The *file* table in this database, contains the files metadata such as the file path, name, parentID, volumeID, inode address, attributes, mtime, rstate, ctime, size, and child count. Other tables in this database contain more useful information such as the owner ID and the file fingerprint. The *synclog.txt* file, logs the application operations such as logins attempts, uploaded files, deleted files, and modified files, along with other timestamps. Finally, the file *trace.txt* stores information regarding the user, the process (*CopyAgent*), the operating system, and the server. Another Copy configuration file is contained inside */home/rakesh/.config/Barracuda Networks, Inc\Copy.conf*. Once the client application is removed, the *find* command revealed folders and files left on system. Since *CopyAgent* was never installed but simply run from the Copy client downloaded folder, once it is removed, it will not delete the Copy main folder; however, a user can simply delete the Copy main folder found in user's home profile. If the user is not aware of the hidden *.copy* folder, this will remain on the system, along with all the information contained, which represents a great deal of useful information during a forensic analysis. Even if the Copy main folder is deleted, both the inode entries and the data content are still present on the hard disk, and unless it is wiped or the data is overwritten, recovery of deleted files is possible (see *Recovering Deleted Files subsection*).

4.4 Client Application Analysis: ownCloud

Once the client application of ownCloud is installed, the main folder is by default present in the user's home directory. Unlike Copy, there is no hidden folder, but hidden files are present in the client main folder inside the user's home directory. This folder contains the files that are synchronized with the account, along with the hidden database file *.csync_journal.db*. The table *metadata* of the database file, stores valuable information regarding files, such as paths, inode addresses, UIDs, GIDs, MAC times, md5 hashes, among other information,. The table *version* contains the version of the client application. The hidden file *.owncloudsync.log* is a log file that stores timestamps, operations executed by the application (instructions), the length of the operations, the names of the files involved, MAC times, sizes, the file IDs, among other useful information. The configuration file found in the following path and named */home/<user_name>/.local/share/data/ownCloud/owncloud.cfg* contains the URL of the ownCloud server, the username, authentication type (http), and other settings. After removing the application, the files on the local system will still be present. Therefore a user has to manually delete the ownCloud main directory. Unfortunately, from a forensics point of view, the database containing the metadata and the log files, even if are hidden, are removed with the directory, that includes also the stored files. However, inode entries and data content are still present on the hard disk, so as for Copy, unless it is wiped or overwritten, recovery is still possible.

4.5 Client Application Analysis: Dropbox

Dropbox, unlike Copy and ownCloud, encrypts the database files and log files for security purposes. The encryption keys are not released even to the user. However, a tool exists for Windows versions that decrypts these files. Unfortunately, this tool does not have a counterpart for UNIX/Linux, nevertheless, according to the Magnet Forensic team, a version of the decryptor should be available for UNIX/Linux in the future [15]. Once the application is installed and the account is set, the Dropbox main folder is created in the same location as Copy and ownCloud, which is the user's home directory. The main Dropbox folder maintains a hidden cache directory that stores the files synchronized with the account. In addition to the main folder, there is a hidden folder named *.dropbox* inside user's home directory, which contains multiple interesting files and subdirectories: a file named *info.json* stores the path of the dropbox root directory and the file *dropbox_pid* that contains the Process ID, the sub-directory *instance1* stores all the encrypted files so it is not possible to collect evidence unless the files are first decrypted. One of these file, is *aggregation.dbx,* contains timestamp values, server paths, and a blocklist value. The encrypted files are *config.dbx* (contains configuration settings, user, host machine and server information), *deleted.dbx*, *filecache.dbx*, (contains files metadata), *notifications.dbx*, *sigstore.dbx*, *PENDING_cHDqrT*, *TO_HASH_3WP99a*, and *UPDATE_XyrFxy*. The folder *instance_db* stores another encrypted file named *instance.dbx*. When uninstalled, Dropbox does not delete the main Dropbox directory in the /home/rakesh/ folder. Even if the folder is deleted, a user might be unaware of the presence of the hidden folder *.dropbox*, and this will be helpful if the database files and logs can be decrypted. Like Copy and ownCloud deleted file are possible to recover.

4.6 Physical Memory Analysis

As mentioned in the Methodology section, physical memory was acquired using LiME and it was analyzed with keyword searches. Volatile evidence found includes names of files present in the application's main folder, recently accessed folders, name and client application main process information, process instructions (for example, in the case of Copy, *copy-sync, copy-update, copy-paused, or owncloud_init, owncloud_commit, owncloud_opendir* in the case of owncloud, etc.) icons paths, hosts names, loaded libraries and modules used by the process, libraries imported through the server, temporary files created by the client application main process in the respective application cache folders, database files accessed, log files accessed, log files entries, logins attempts and credentials (for example, during the memory analysis after the use of Copy, the string *–Loginsuccess U:*

'cloudstorage.test.mail@gmail.com' was found.

4.7 File System Analysis

Several artifacts are found in various locations of the file system. However, most of these files are not useful during a digital forensic investigation, but to only demonstrate the installation and use of the client applications by the user. The directories */usr/share/man* and */usr/share/doc* on Ubuntu contain the man pages and documentation of the applications, while the directories */bin/* and */usr/bin* store executables or daemons files used to launch the applications. A great number of icons related to the cloud storage applications are found inside */home/<user_name>/.icons*. Information can also be found inside systems logs: for example, the log auth.log inside */var/log* stores command issued on the terminal, along with the working directory and it can be used to analyze the user recent activity. The log file */home/<user_name>/.bash_history* contains a list of the recently commands issued by the user. The *dpkg.log* contains references to packages installed on the system, and it can reveal if any cloud storage client application was installed from the command line. Other possible locations that may contain useful evidence are the folders */home/<user_name>/.local/share/Trash,* which stores deleted files, and */home/rakesh/.cache/thumbnails*, which contains thumbnails associated with the client applications. If an application during its execution should crash, the *syslog* and *kern.log* files found inside */var/log* store references of the crash. The log file *history.log* contains the recently installed packages and the command issued to install them, while *term.log* contains libraries used to install the applications.

4.8 Deleted Files Analysis

We clearly established in the Prior Work section that it is possible to find artifacts of cloud applications inside the unallocated space. Therefore, it is possible to recover the data stored inside this space. This subsection will be dedicated to the recovery of deleted files, which is a process known as data carving. Recovering deleted files can be done using two different methods. The first is to recover the deleted files from the server, but it requires the application to be logged in, or the account to be accessed through a web browser. Copy and ownCloud applications both allow the user to recover deleted files with a feature known as '*undelete*'. Dropbox does not have a similar feature, but it is possible to recover deleted files or previous version of files on an account accessed through a web browser. However, unlike Copy and ownCloud, Dropbox allows a permanent deletion option, that will permanently delete a file from an account. The second way to recover deleted files is through the use of a data carving tool on the local device.

All the client applications analyzed in this research, when they first synchronized with the server, downloaded a copy of each file and stored them locally. As mentioned earlier, even if the files are deleted, the *Ext4* file systems does not delete the *inode entry*, so both the *inode entries* and the files' *data content* are still present on the hard drive. To see how the recovery is done, first we have to analyze how the file are allocated. Using the *istat* tool from the *TheSleuth Kit* an examiner is able to find the *inode addresses* of the files, the *group* to which they belong, and the *block address* of where the *data content* of the files is stored. The files are usually (but not necessarily) stored in contiguous *inode addresses*. Using the *blkstat* tool on the *block addresses* that contain the files' *data content*, the output shows that the *blocks* are in fact allocated. By executing the *fsstat* tool it is possible to find the *inode range address* of the *group* and the range of the *blocks* where the *data content* is present (usually stored in a different group). Using *dd* on these ranges it will allow a forensic expert to create a small image that can be further analyzed with a hex editor. Now, we can examine if the files are still present once they are deleted. After deleting the files, *blkstat* will return that the *blocks* that store the *data content* are not allocated anymore. However, by opening the images with a hex editor, it is possible to determine that the data is still present inside the *block*. In addition, the inode entries are still present (a character has been added before the file name to indicate that the files pointed by the inode have been deleted). At this point, using a simple data carving tool such as *Foremost* or *icat* from the *TheSleuth Kit*, recovering the deleted files is fairly easy.

4.9 Android OS Analysis

A common featre of cloud storage services is that they allow multiple devices to be synchronized. Therefore, it is very likely that one user will synchronize the same cloud storage account on multiple devices. Thus, in order to gather as much data as possible, it is important to analyze not only workstations, desktops, and laptops but also devices such as smartphones and tablets. An operating systems that comes as default on many smartphone is Android OS. Client applications were both available for Copy and Dropbox, however, an official client at the time of the writing of this paper is not available for ownCloud. Nevertheless, an unofficial but very efficient client application has been published by the BezKloboukuNos team [16]. In the case of Copy the main files are found inside the folder */data/media/0/Android/data/com.copy/files*. Among these files, the *configuration.ini* file contains different configuration and account settings, while the file *copy.db* stores names of the synchronized files along with the metadata of the files. The log file *trace.log* stores information regarding logins, synchronized files, timestamps, account information, and the client application's process operations such as uploading, deleting and/or modifying files. The *download* directory inside the *temp* subdirectory contains the files that were locally saved, while the directory *thumbnails-cache* contains thumbnails. The file *com.copy_preferences.xml* in the directory */data/data/com.copy/shared_prefs* stores other useful data such as the last opened file along with its timestamp. When the Copy application is deleted from Android OS, all the files are also deleted. However, information about the use of the application can be found in logs, such as *logcat*, which contains the system recent activity. Operations and actions executed by the Copy application, files and directory accessed were found inside *logcat*. ownCloud does not have an official client, but as

mentioned above, an unofficial client is available. Once the application is installed, the directory */data/media/0/owncloudApp* stores the subdirectory *owncloud_user@server_address*, which stores files that were downloaded locally on the device. The database file named *filelist* in the folder */data/data/com.owncloud.androidApp/databases/* stores the list of the files synchronized along with the type of the file, the username, server name and address. The file *com.owncloud.android/App_preferences.*xml stores as well, the username and server address. When the application is removed, the directory */data/media/0/owncloud* still contains the files that were locally stored, along with other temporary files. The rest of the files created by the application are deleted. Also in this case, recent activity can be found in logs such as *logcat.* The directory */data/data/com.dropbox.android/* stores local files, and the subdirectory *databases* contains database files. Unfortunately, also in the case of Android OS most of these files are encrypted. A file named *361753330-db.db* inside the directory, stores the names of the synchronized files, and the file *prefs-shared.db* stores account preferences. Finally, recent activity of the user, including the use of Dropbox, can be found using *logcat*. A lack of analysis tools for Android OS, did not let us perform a dynamic search to gather the useful data remnants. In fact, the files above mentioned and listed were found through a manual search.

5 Conclusions and Future Work

The research proposed in this paper highlighted the main locations and files on a client device with an Ubuntu 14.04, and more generally, a UNIX/Linux operating system. These files can be used as evidence related to the use of cloud storage service client applications. Although the operating system is different, as well as the file system, the results obtained by previous researches, and the results of this research are in accordance and similarities arise. Differences arise due to the different feature of the Windows 7 operating system and the UNIX/Linux OS, therefore locations and file types may vary. In fact, the registry, which is a classic feature of Windows is not present on UNIX/Linux systems so there will be no evidence related to it. Unfortunately, the amount of tools used to analyze dynamically a process, such as Process Monitor does not have a valid counterpart for UNIX/Linux, therefore a dynamical analysis to gather evidence is a bit harder. Nevertheless, it was still possible to find evidence in the hidden directories or hidden files created by the application, as well as in database or log files, inside web browser files, in the memory, and in both allocated and unallocated space. Many other locations and files on the file system, such as the cache, system logs, and configuration files stored useful data. Files stored on Android OS are similar to the files stored on laptops or desktops. However, one big difference is that, in the case of Ubuntu, the cloud application downloads and stores a copy of each file locally, while in the case of Android OS, a copy of a file is stored locally only if the file is accessed by the user. The fact that the files are stored locally, is really helpful during digital investigations since deleted files and their metadata are still present on the hard disk, unless the unallocated space is wiped or overwritten. Recently, most of the main cloud storage services, such as iCloud or Dropbox, started to encrypt their files, which makes it harder for digital experts to gather useful evidence. With a proper search warrant, the cloud service provider may be able to decrypt the files and allow forensics experts to gather evidence. Another key factor that can be seen in this analysis, is that even if the applications are different, the configuration files, database files, and log files (and use of hidden files and directories, cache directories and temporary files, database and log files) are similar. Finally, this research highlighted that a great deal of evidence can still be found on the Ubuntu system once the application is uninstalled or simply removed.

6 References

[1] Columbus, L. "Predicting Enterprise Cloud Computing Growth". Sepember 4, 2013. Available: http://www.forbes.com/sites/louiscolumbus/2013/09/04/predicting-enterprise-cloud-computing-growth/

[2] Cohen, R. "The Cloud Hits the Mainstream: More than Half of U.S. Businesses Now Use Cloud Computing". April 6, 2013. Avaialable: http://www.forbes.com/sites/reuvencohen/2013/04/16/thecloud-hits-the-mainstream-more-than-half-of-u-s-businessesnowuse-cloud-computing/

[3] NIST. "Drafts Cloud Forensics Standard". Information Management Journal, September, 2014.

[4] NIST. "Cloud Computing Forensic Science Challenges". Draft NISTIR 8006, NIST Cloud Computing Forensic Science Working Group Information Technology Laboratory, NIST, June, 2014. Available: http://csrc.nist.gov/publications/drafts/nistir8006/draft_nistir_8006.pdf

[5] Henry A. "Most Popular Cloud Storage Provider: Dropbox". Lifehacker, August 2, 2013. Available: http://lifehacker.com/most-popular-cloud-storageproviderdropbox-644624306

[6] Process Monitor v3.1, TechNet, Available: http://technet.microsoft.com/en-us/sysinternals/bb896645

[7] Chung, H., Park, J., Lee, S., & Kang, C. "Digital forensic investigation of cloud storage services". *Elsevier*, May 4, 2012. [8] Katz M., Montelbano R. "Cloud Forensics", The Senator Patrick Leahy Center for Digital Investigation, Champlain College, 4 November 2013.

[9] Quick, D., Martini, B., & Choo, R. "Cloud Storage Forensics", 1st ed., p. 208, Syngress, 2013.

[10] Epifani, M. "Cloud Storage Forensics", SANS European Digital Forensics Summit, Prague, 2013, Available: https://digital-forensics.sans.org/summitarchives/Prague_Summit/Cloud_Storage_Forensics_Mattia_Eppifani.pdf

[11] LiME, Linux Memory Extractor, Available: https://github.com/504ensicsLabs/LiME

[12] Carrier, B. The Sleuth Kit: Download. Available: http://www.sleuthkit.org/sleuthkit/download.php

[13] Foremost, SourceForge.net, Available: http://foremost.sourceforge.net

[14] DB Browser for SQLite. Available: http://sqlitebrowser.org/

[15] Dropbox Decryptor: A Free Digital Forensics Tool. Available: http://www.magnetforensics.com/dropbox-decryptor-a-free-digital-forensics-tool/

[16] Client for ownCloud, BezKloboukuNos, 2014, June 29 Available: https://play.google.com/store/apps/details?id=com.owncloud.androidApp&hl=en

Storing Credit Card Information Securely using Shamir Secret Sharing in a Multi-Provider Cloud Architecture

Levent Ertaul, William Marques Baptista, Rishi Maram
CSU East Bay, Hayward, CA, USA.
levent.ertaul@csueastbay.edu, william.marquesbaptista@gmail.com,
rmaram2@horizon.csueastbay.edu

***Abstract*— Cloud storage and Online database services allow information to be accessed from virtually any location around the world with Internet access. While this makes information readily accessible and shareable, it also exposes data to the hostile environment that is the Internet, rendering it vulnerable to malicious attackers. Credit card information is one type of data that can be targeted by malicious attackers. This paper proposes an implementation that could store credit card information securely on a multi-provider cloud architecture by using Shamir secret sharing as the foundation and alternative to traditional encryption. The primary objective of this work is to implement the Shamir secret sharing algorithm in multi-provider cloud architecture. The Java programming language is used to implement a proof-of-concept application that connects to Amazon RDS and Google SQL databases. Performance analysis of the implementation is discussed, demonstrating its efficiency and revealing where bottlenecks may be encountered when processing credit card numbers. Finally, some remarks are made regarding the evolution of the implementation throughout the undertaking of the project, and additional improvements to the mechanisms are proposed.**

I. Introduction

Cloud storage brings a competitive advantage to smaller businesses that are constrained by a limited budget and employees as a cost-efficient means of competing [1]. Advances in cloud storage services such as Amazon RDS [2] or Google Cloud SQL [3] among others, along with the ever-growing market of cloud-storage solutions [4], gives smaller businesses the flexibility to choose between different service providers.

Customer credit card information is an example of sensitive data which is often transmitted over the Internet, whether while shopping Online, transferring funds to and from virtual wallets [5], or paying with a smart phone at a checkout counter [6]. Not only are there third parties which provide businesses with a service to handle credit card transactions [7], a security standards council [8] also exists that develops various standards related to data security and payment applications. However, there have been doubts as to the effectiveness of this council itself [9], which would have a significant impact on not only how users shop, but also how businesses and banks handle monetary transactions.

Instead of trusting a single entity with credit card information, said information could be split into multiple pieces and stored across multiple database service providers. In this scenario, small businesses not only use cloud services as a cost-effective competitive advantage [1], but also decide who to trust with storing their customer's credit card information.

Traditional encryption is computationally expensive [10], so to counter that cost this implementation suggests an alternative solution. It is both based on work in [10][11], which serve as a foundation to store credit card information securely by means of the Shamir secret sharing scheme [11] in a multi-provider cloud architecture. This implementation takes a credit card number as input, generates multiple pieces called shares, and stores those shares across multiple databases from multiple cloud providers. In doing so, it is capable of eliminating the informational value of any given share should there be a security breach at any of the database service providers' location.

Section II of this paper addresses obstacles that may be encountered regarding the storage of sensitive data. Section III will discuss the method of creating shares as a form of encryption, presenting the general idea behind secret sharing. Section IV will highlight the details of how the methods were implemented, and provide the reasoning behind the decisions. Section V will present performance analysis comparing different numbers of shares. Section VI highlights changes made to the implementation during the undertaking of the project, and includes remarks as to what aspects of the implementation may be improved. Section VII concludes with final remarks on the project.

II. Traditional Encryption vs. Secret Sharing

Businesses may sometimes opt for some form of traditional encryption when storing data on local databases, especially those containing customer information that includes credit card numbers. However, doing so may sometimes require special hardware such as Hardware Security Modules, and traditional methods of encryption can be computationally expensive [10].

Attackers often need to breach just a single entity to obtain customer data, but any one of these breaches can prove to be disastrous: Heartland Payment Systems-134 million cards exposed[9][12]; TJX Companies Inc.-94 million card numbers exposed[13]; Card Systems Solutions-40 million card accounts exposed[12]. While these statistics only mention large entities, smaller entities are just as vulnerable to attacks, especially if all the required information is stored in a single provider location, let alone locally in a single database.

An alternative solution would be to store data remotely across multiple providers vs. locally. This would increase the number of targets an attacker would have to breach to obtain the desired data. A secret sharing scheme [11] could be used to break credit card information into multiple pieces and store those pieces at different locations. Ideally, different providers would be used to further reinforce the concept of mutually suspicious entities with conflicting interests [11].

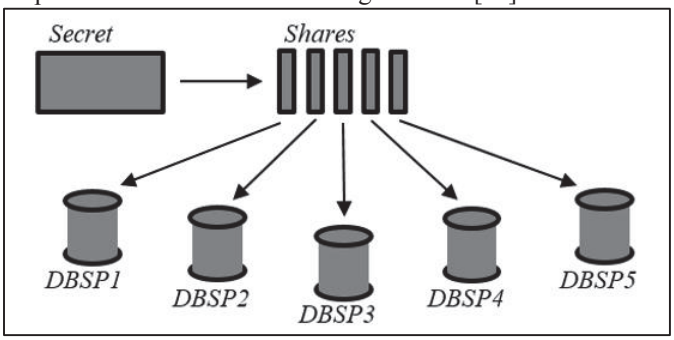


Fig. 1. A secret is split into shares, and distributed to different database service providers (DBSP).

Figure 1 depicts an example of what our implementation accomplishes. This implementation takes a credit card number and splits it into multiple shares by means of a secret sharing scheme [11]. The shares are then stored across different cloud databases obtained from different service providers. The greatest strength of our implementation is that an attacker must know a) how many shares a particular secret has, b) where these shares have been stored and have access to them, c) the corresponding pairs of each share and how they are calculated, d) the secret prime used during the initial computation, and e) parameters specific to the polynomial used to produce the shares. It becomes clear when presented with these obstacles that a secret sharing scheme can be an effective solution in securing credit card information.

III. Shamir Secret Sharing

In [11] Shamir proposes a mechanism for robust key management schemes, where a numerical value is split into pieces, and can be easily reconstructed with a subset of those pieces. This is known as a threshold scheme [11] where a subset *k* of *n* total pieces is required to reconstruct the original secret. The implementation in this paper requires that all *n* pieces be available to reconstruct the original secret, in other words, as opposed to a *(k, n)* threshold scheme, this implementation is a *(n, n)* scheme.

A. Creating the Shares

To create the shares, let *k* represent the number of pieces, or shares, to be produced where *k=n total pieces*. Then, a prime *P* must be chosen where *P>M* and where *M* is the numerical value or secret being split into shares. A polynomial of *k-1* degree is then constructed with *k-1* random coefficients *C*, where *P>C>0*.

$$f(x) = M + C_{(1)}x^{(1)} + C_{(2)}x^{(2)} + \ldots + C_{(k-1)}x^{(k-1)} \quad (1)$$

Polynomial (1) is the result of this construction, where the secret *M* is the term C_0x^0, each *C* is a randomly generated coefficient, and substituting *0* for *x* in the polynomial returns this secret value *M*. Pairs are created as an *x* input into (1) resulting in the corresponding output share *f(x)*, thus forming the pair *(x, f(x))*. Any number of pairs can be produced or even replace existing pairs. Once the shares have been created, we no longer have use for the random coefficients as they are randomly chosen for every secret. The resulting pairs and the prime used to select random coefficients are the only information needed to be recorded at this time.

B. Reconstructing the Secret

All *k* pairs *(x, f(x))* are required to reconstruct the secret using Lagrange polynomial interpolation [11][14].

$$M = \sum_{k=1}^{n} y_{(k)} \prod_{i=1, i\neq k}^{n} \frac{-x_{(i)}}{x_{(k)} - x_{(i)}} modP \quad (2)$$

We then expand (2) using the pairs that were originally created to recover the secret.

C. Unrecoverable Secret:

We should note that, as mentioned in the beginning of this section, we need all pieces to reconstruct the secret. If, for instance, we were missing one or more pairs, our result would be quite different. Some alternatives to this limitation are suggested in section VII.

IV. Implementation Methods and Reasoning

This project was implemented on a notebook computer with a 1.6GHz quad core processor, 8 GB of RAM, and running a GNU/Linux OS [15] with kernel version 3.13.0-46-generic. NetBeans 8.0.2 [16] integrated development environment (IDE) was used to implement the project using the Java [17] programming language. The IDE was used to create a proof-of-concept Swing GUI application, taking advantage of an integrated profiler for performance analysis. The profiler allows us to monitor the performance characteristics of different methods in the implementation without making changes to the original source code.

Eleven databases were created, 5 on Amazon RDS [2] using MySQL 5.6.22 [18], and six on Google Cloud SQL [3] using MySQL 5.5 [18]. A MySQL JDBC Driver library [19] was used to interface between the proof-of-concept application and the remote cloud databases. Each database contains one table, and a user was specifically created to access these databases with ***SELECT***, ***INSERT*** and ***DELETE*** privileges.

As opposed to [10], there is no need to evaluate max/min values, or perform calculations on ranges of values stored in the databases. Apart from the GUI-related code, there are four prominent code portions categorized as 1) global variables and structures, 2) generators and equations, 3) database operations, and 4) additional methods.

A. Global Variables and Structures

The global variables in the implementation are the username and password information to access the different databases, a prime *P*, the number of shares *K*, and the database *IP* addresses. There are also three array list structures: a

database address array list, a coefficient array list, and a share array list.

The database username and password information was hard-coded to facilitate the interaction between the proof-of-concept application and the databases. As a security precaution, these credentials should not be stored within a production version of this proof of concept, but instead an alternative method of authenticating a user or application with the different databases should be implemented. That, however, is outside the scope of this project.

The prime number was implemented as a *BigInteger*, which allows the use of integers larger than the 64-bit limitations of the *long* data type [20]. It was hard-coded as a 2048-bit value for the purpose of this work, but should be kept secret at all times. This large prime ensures that the coefficients generated are also very large, making it difficult for an attacker to recover the secret due to the discrete logarithm problem [21]. How the prime number should be securely stored, encrypted, or recovered in a production implementation is outside the scope of this paper, as that may vary from one application to another.

The number of shares *K* denotes how many shares a secret will be divided into. This variable is also used throughout the application as an iterative loop parameter for database connections and generation of the various lists.

The coefficient array list stores randomly generated numbers and is accessed during share generation. After generating all shares, this list is no longer needed, so the contents can be overwritten if another secret is being processed.

The share array list is populated during the generation of shares and used to commit records on the different databases. It is cleared and re-populated during the recovery process of a secret as shares are collected from the different databases.

The database address list contains addresses to the different databases, and they are also used as one of the parameters to create record name fields for each database's respective share.

B. Generators and Equations

The three generators in this implementation are the database address generator, the secure random coefficient generator [22], and the share generator. The database address generator runs only once, whereas the remaining two run once for every secret that is processed.

1) Database Address Generator

```
private static void enumDBList() {
        int i, n;
        if (K % 2 == 0) {
            n = K/2;
        } else {
            n = K/2+1;
        }
        for (i = 0; i < K; i++) {
            if (i<n) {
                dbList.add(dbGoogle + (i+1));
            } else {
                dbList.add(dbAmazon + (i+1-n));
            }
        }
    }
```

The database address generator is a simple algorithm that enumerates the database address list, and is directly dependent on the number of shares. The database names used at both service provider locations are named from secsharedb1 to secsharedb5 for Amazon RDS [2], and secsharedb1 to secsharedb6 for Google SQL [3], for a total of eleven databases. The IP addresses are hardcoded global variables with the first portion of the database name. They are then referenced using the variables *dbGoogle* and *dbAmazon* for short while the database numbers are concatenated at the end of the string within the algorithm's iterative loop.

We wanted to make sure that both providers were always used, so this algorithm distributes the number of databases based on the value of shares *K*: for an even number of shares, the same number of databases is used at each provider location; for an odd number of shares, there is one more database used at Google's location. There is no particular reason for choosing a Google database over an Amazon database other than to fill the gap. Although a seemingly simple and innocuous algorithm, it is safe to say that this is what controls which databases are being accessed in the different provider locations. More on this algorithm is discussed in Section VI.

2) Random Coefficient Generator

```
private void genCoefs(int shares) {
        BigInteger r;
        coefs.clear();
        int i;
        for (i = 0; i < shares - 1; i++) {
            r = new BigInteger(2047, new SecureRandom());
            if (r == BigInteger.ZERO) {
                r = r.add(BigInteger.ONE);
            }
            coefs.add(r);
        }
    }
```

The coefficient generator above is used to generate *K-1* 2047-bit cryptographic-strength pseudo random numbers [22], and this process is repeated for every secret being split into shares. More on the decision for its size is discussed in section VII. During the test, smaller coefficients were used, and we noticed that there were times where a zero was returned as a secure random number. This would have changed an entire term to zero during multiplication, so a check was added to add the value one to the secure random number when this was the case. The coefficients are stored in the coefficient array list and only accessed when generating shares, after which they are no longer useful since a different set of coefficients will be generated when processing another secret.

3) Secret Share Generator

```
private void genShares(int shares) {
        long secret = Long.parseLong(jTextField2.getText());
        BigInteger coef, term, result;
        secShares.clear();
        int i, j;
        for (i = 0; i < shares; i++) {
            result = BigInteger.ZERO;
            for (j = 0; j < shares-1; j++) {
                coef = (BigInteger) coefs.get(j);
                term = (BigInteger.valueOf(i+1)).pow(j+1);
                result = result.add(coef.multiply(term));
            }
```

```
            result = result.add(BigInteger.valueOf(secret));
            secShares.add(result);
        }
    }
```

The share generator above produces shares corresponding to each *x* input. It uses a polynomial such as (1) using the generated coefficients mentioned earlier, and produces as many shares as indicated by the global variable *k*. In this implementation, the *x* values are generated incrementally with a loop, and more remarks regarding this process will be mentioned in section VI. The resulting shares vary in size, depending on the number of shares being produced, but their sizes increase as the number of shares increase. The shares were also implemented as a *BigInteger*, again due to the 64-bit limitations of the *long* data type [20].

4) Secret Recovery Algorithm – summation portion

```
private void secRecover() {
        BigInteger pair, term;
        lagSum = BigInteger.ZERO;
        int i;
        for (i = 0; i<K; i++) {
            pair = (BigInteger) secShares.get(i);
            term = pair.multiply(lagrange(i+1));
            lagSum = lagSum.add(term.mod(P));
        }
        if ((K % 2) == 0) {
            jTextArea1.append("M: " +
P.subtract(lagSum.mod(P)) + "\n");
        } else {
            jTextArea1.append("M: " + lagSum.mod(P )+ "\n");
        }
    }
```

The secret recovery algorithm above was implemented iteratively as it was the simplest implementation method at the time. A recursive method was not tested. This first algorithm implements the summation portion of the terms in (2). During the test trials, we noticed that for an even number of shares, the result resembled the prime rather than the secret. After closer inspection, when using an even number of shares the summation result has to be subtracted from the prime to obtain the secret back. For this reason a test is performed where if *k mod 2* is *0*, then the summation result is subtracted from the prime to obtain the secret.

5) Secret Recovery Algorithm – cross product portion

```
private BigInteger lagrange(int curShare) {
 BigInteger numer = BigInteger.ONE;
 BigInteger denom = BigInteger.ONE;
 BigInteger result;
 int i;
 for (i = 1; i < K+1; i++) {
  if (i != curShare) {
   numer = numer.multiply(BigInteger.valueOf(i));
   denom = denom.multiply(BigInteger.valueOf(curShare-i));
  }
 }
 result = numer.multiply(denom.modInverse(P));
 return result;
}
```

An important component of the secret recovery algorithm is the cross product portion of (2) which is implemented as a sub-method and called within an iterative loop. This decision made it easier to read the code in terms of the summation and cross product portions of (2). Both the summation and cross product portions take advantage of the iterative loop used to generate the corresponding *f(x)* share for a given *x* value. This simplifies the code implementation. However, doing so raises issues which will be covered in section VI.

C. Database Operations

This portion of the code consists of 5 methods, of which 4 are basic database operations that could be implemented in a production environment. The operations are responsible for committing a record, fetching a record, deleting a record, and deleting all records.

Figure 2 below depicts the "add" button, which ensures that neither the name field nor the credit card number field is empty before proceeding. It then triggers the random coefficient generator, the share generator, and finally calls the record commitment method which is one of the 4 database operations. As mentioned previously in this section, *x* values for (1) is generated incrementally within a loop. For this reason, the databases are accessed in the same order every time. More remarks on this mechanism are found in section VI.

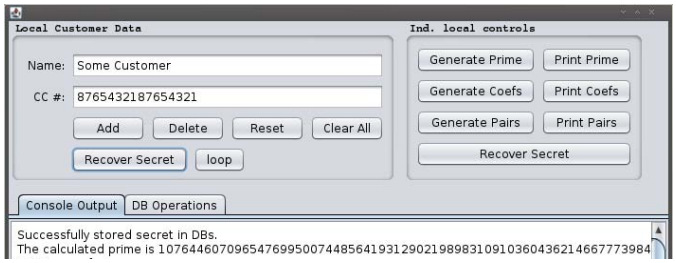


Fig. 2. Proof of concept Application depicting some of the control buttons. Add, Delete, Recover Secret, and Loop all operate on the databases, while the independent local controls do not interact with any database, and are used for debugging.

The record commitment method stores the various shares in the different databases. First, an SQL command is built, and a hash along with the corresponding share are concatenated to the SQL command. This command is then executed within a loop, where a different database is accessed, on each iteration. If a duplicate name field is found in the database, then the error is caught and signaled as an output in the DB Operations output area whose tab is shown in figure 2. Additional remarks on how *x* value generation may be improved are discussed in section VI.

Fetching a record works similarly as the record commitment function, where each database is accessed incrementally by means of a loop. Because every table only has 2 fields, name and share, the name field is used to search a corresponding share. For this work, the name field is built as a hash of the customer name times the hash of a password times the hash of the database where the corresponding share is to be fetched. More remarks on how the record name field is created are made in section VI. As the shares are fetched, they are placed in the share list to be used during the secret recovery step. If a record is not found, the error is caught and fetching the record fails.

Deleting a record works similarly as the record commitment method in that each database is accessed incrementally within a loop. The record name field is searched by calculating the

hash of the customer name times the hash of a password times the hash of the database name where the corresponding share is being searched, and once found, the record entry is deleted.

Deleting all records in all databases is trivial, where every database is accessed and data in their respective tables is dropped. While this wouldn't seem to be a necessary or routine part of the implementation, this procedure may be used as part of a kill-switch in case of emergencies.

D. Additional Methods

The remaining methods in the code consist of the detection of a duplicate entry when committing a record, a function that checks for valid input, a function that creates the name fields for each share using hashes of different data, and a test method that was used during the performance analysis. Details on the performance of the implementation will be reviewed in section V.

V. Performance Analysis and Results

The code was implemented in Java [17] using NetBeans 8.0.2 [16] as the integrated development environment. The reasoning behind this decision was that the profiler available in NetBeans was the least intrusive means to measure the performance of the individual methods, capable of collecting CPU timing information at a fine granularity with no changes to the original code. The average CPU times in establishing connections are times to connect to the set of Amazon RDS [2] and Google SQL [3] databases as dictated by the number of shares *k*.

A test button was implemented which loops through 3 principle methods: ***addRecord()***; ***getRecord()***; ***secRecover()***;. Of these three, we will focus on two, namely ***addRecord()*** and ***secRecover()***, since these include the sub-methods and results we are more interested in.

As mentioned earlier in section IV, the ***addRecord()*** method is responsible for generating secure random coefficients[10], generating shares using a corresponding *x* value for each database, and committing the changes to the databases. There were 8 runs of the test loop to gather data on the generation of coefficients and pairs, each run executing the three methods stated earlier once. The times were then recorded and averaged on a spreadsheet.

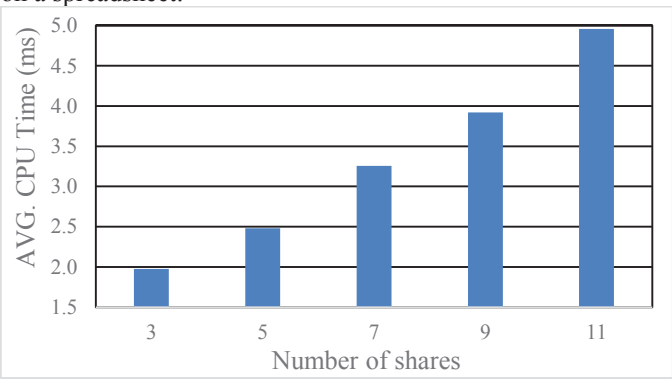


Fig. 3 Average CPU time to generate secure random coefficients. The number of coefficients generated per group of shares is (number of shares-1).

In figure 3, we can observe the performance results for the generation of secure random coefficients. On average, it took a mere 4.96ms to generate eleven 2047-bit secure random numbers. The fastest time was recorded at 3.93ms, whereas the slowest was 5.91ms. The results for any given number of shares will tend to vary slightly depending on the availability of resources on the test notebook computer; however, we can observe a clear trend that as the number of shares increase, the time it takes to generate secure random coefficients increases almost linearly.

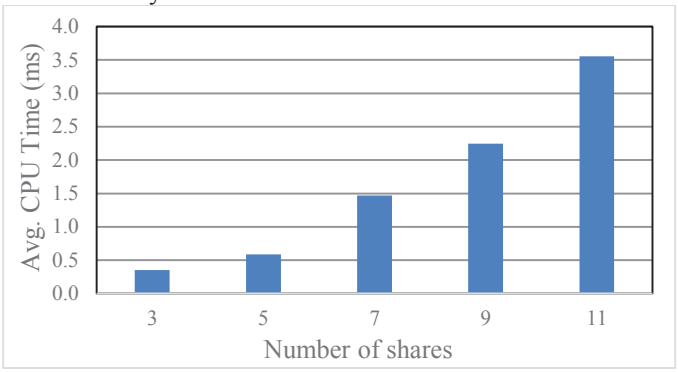


Fig. 4 Average CPU time to generate shares.

In figure 4 above, we can clearly observe that as the number of shares generated increase, the time to generate said shares increases exponentially. This is due to the fact that, because we are implementing [10][11] as a (*n, n)* scheme where *k=n*, the polynomial used to generate a corresponding share will increase in degree for every additional share *k*. No tests were performed with methods using a fixed-sized polynomial, as that would counter our implementation of a (*n, n)* scheme. During the tests, eleven shares were produced at a mere 2.11ms at its fastest, compared to 6.10ms at its slowest which is more than double the fastest time. These results also vary due to the availability of resources on the notebook computer.

During these tests, the results that stood out the most were the database connection times. The ***addRecord()*** method contains three sub-methods: ***genCoefs()*** which generates the secure random coefficients; ***genPairs()*** which generates the corresponding shares to an *x* input; and ***commitRecord()*** to establish the database connections and store the information.

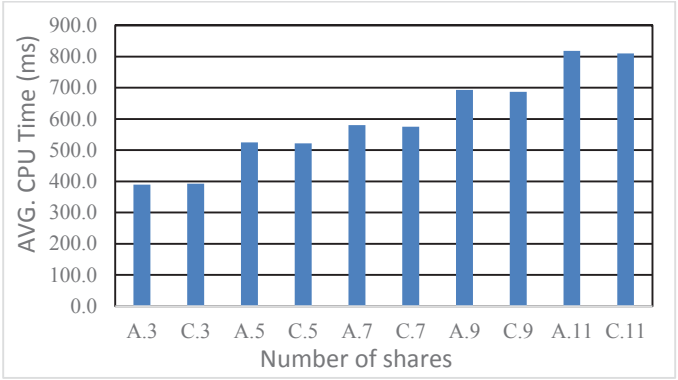


Fig. 5 Average CPU time to add a record 'A.' vs. portion of time spent establishing database connections 'C.' per number of shares.

In figure 5, we can observe how much time the ***commitRecord()*** method takes in the process of adding records to the databases. For each set of shares, the left column denoted by 'A.*x*' shows the average CPU time spent executing the three sub-methods, while the right column 'C.*x*' shows what portion of that time is spent establishing the

connection and sending the SQL ***INSERT*** command. From figure 5 we can conclude that the numbers of shares, and consequently the number of database connections being established, tend to have a more significant impact with respect to time than the methods involved in generating the secure random coefficients, and generating shares.

The objective in the following test was to analyze the average CPU time it took to recover a secret based on the number of shares used. The secret recovery method produces the summation of the *f(x)* terms multiplied by the cross product performed in the sub-method. This was done by means of an iterative loop, which called the sub-method at each iteration.

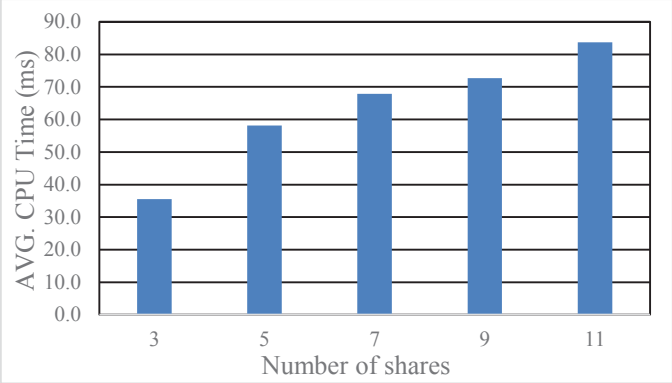


Fig. 6 Average CPU time to recover a secret per number of shares.

As we can observe in figure 6 above, the average CPU times to compute the secret increases as the number of shares increase. During the tests, it took 78.50ms at best to recover the secret for eleven shares, compared 90.60ms at its slowest. While these results are for almost four times the number of shares as the test for three shares, we can also observe that there is not a significant impact on the average CPU time to recover a secret based on the number of shares. For three shares, it took an average of 35.48ms to recover the secret, whereas for eleven shares it took an average CPU time of 83.68ms, which is a ratio of ~1:2.3.

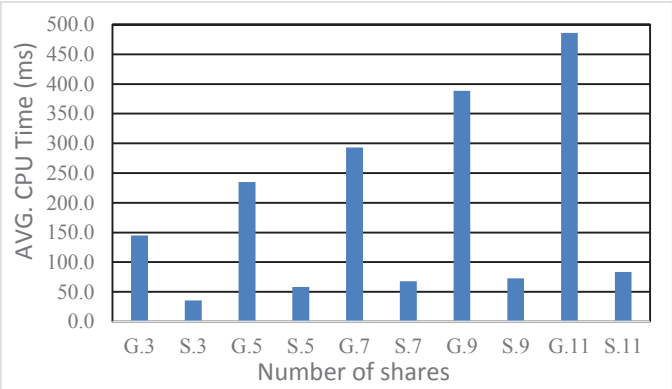


Fig. 7 Average CPU time to fetch shares denoted by 'G.' vs. average CPU time to recover secret denoted by 'S.'.

In figure 7 above, we can observe that the average CPU time to recover a secret denoted by 'G.*x*' does not produce as much of an impact as does the average CPU time spent establishing a connection denoted 'S.*x*' to recover the shares from the different databases. For example, using three shares as the data with the smallest gap, it took an average CPU time of 144.72ms to fetch all shares from the different databases, compared to an average CPU time of 35.48ms to recover the secret. In other words, it took roughly 25% of the average CPU time to recover the secret than it did to retrieve all shares from the different databases.

While the average CPU times during these tests were calculated by making connections to eleven databases across two providers as stated in the implementation details in section IV, these times may be expected to be different if the connections were to be made to more databases and providers around the world. In that case, we would mostly be concerned with the database connection times to recover the different shares, whereas the process to create the different shares or to recover the original secret would remain relatively insignificant in comparison.

VI. Suggested Improvements

A. Generating x Inputs

Generating *x* values for (1) were fairly simple as they were generated by means of a loop, generating incremental values of *x* from 1 to *k*. A security improvement to this method would be to use larger values of *x* which are either database or service provider-specific. Using this method would not only allow databases to be accessed randomly, it would also reduce the correlation between different databases in terms of the order in which they are accessed.

B. Generating Secure Random Coefficients

Initially, 2048-bit coefficients were being generated. This quickly proved to be problematic since there may be times at which the coefficients were larger than the prime being used, which would violate the rule in [11] of $P>C>0$. A check was then implemented to compare the size of the prime and the coefficient, where coefficients would be regenerated if they were larger than the prime. This additional step was deemed unnecessary, resulting in the decision of removing it altogether and simply generating 2047-bit coefficients. This large coefficient still enforces the discrete logarithm problem [21].

C. Generating Record Name Fields

For the purpose of this paper, the record name field is built as a hash of the customer name times a hash of the customer password times a hash of the database address where the particular share is being stored. This was done in an effort to suggest that the correlation between record name fields for a particular secret across all databases be reduced, or different for every database. However, this particular method is not sufficient since customers may have several credit cards whose shares may be stored in a particular database. A solution would be to also include a hash of some combination of numbers from a specific card. For the purpose of this paper it was enough to show that shares for a particular secret need not have the same identifying fields across all databases, however, a better mechanism should be used to enable a customer to have multiple credit card shares stored in the same database without record name field conflicts.

D. Shuffling Records

An additional security measure would be to shuffle the

records in any one of the databases to reduce the correlation among records across different databases. If records corresponding to a same secret are located in the same region or have the same order in any particular database, and the number of shares needed to reconstruct the secret is known, and the respective databases are known, then it would be easier for an attacker to assemble the required shares in attempt to reconstruct a secret. The task would still be very difficult, but this additional layer of security in just one of the databases would be enough to render the task even more difficult.

E. Diversity in Providers

This project advocates the use of multiple providers in effort to not only reduce the risks associated with a breach, but also to ensure availability. Although only two providers were used in this implementation, more could be used if desired, which would also afford new availability features. More on availability will be explained later in this section. Additionally, as suggested by [11], we would like to view providers as mutually suspicious with conflicting interests, which is why we do not want to store all shares for a particular secret within one same provider.

F. Planning For Redundancy

Although this implementation isn't a traditional *(k, n)* threshold scheme, that concept may be extended to the number of providers. For instance, for a secret that is split into fifteen shares and stored across three different databases or providers, it only takes one of those databases or providers to be unavailable for the secret to be unrecoverable. Instead, a threshold implementation would maybe create thirty shares to be spread over five providers for instance, and randomly accessing any combination of three providers would be enough to recover a secret. Together with the suggestion in 'F.', whenever the number of shares needed to reconstruct the secret is reached, all other connection attempts can be aborted. Additionally, having shares across different providers ensures that not all shares required to recover a secret are stored within one same provider.

VII. CONCLUSION

We have shown that implementing Shamir's secret sharing scheme to store credit card information on a multi-provider cloud architecture can be a viable solution. The performance tests show that the process of generating shares and recovering the secret is relatively fast and efficient when compared to the time spent establishing database connections, and this is valid for any application needing to connect to remote databases. The security feature of not being able to recover credit card information should any of the databases be breached brings an advantage that single-provider or singe local databases do not. This paper has shown that the Shamir secret sharing scheme is fast, reliable and secure, but most importantly that it is applicable. The suggested improvements in section VI along with additional contributions could produce a more secure and production-ready implementation for multiple environments.

VIII. REFERENCES

[1] K. Bessai, S. Yousef, A. Oulamara, C. Godart, S. Nurcan, "Scheduling Strategies for Business Process Applications in Cloud Environments," International Journal of Grid and High Performance Computing, Volume 5 Issue 4, pp. 65-78 October 2013.

[2] Amazon RDS: Relational Database Service. https://aws.amazon.com/rds/

[3] Google Cloud SQL. https://cloud.google.com/sql/docs

[4] Sage, Cloud Storage Market to Grow by 2019. http://na.sage.com/us/articles/technology/cloud-storage-market

[5] The PNC Financial Services Group, Inc. Virtual wallets. https://www.pnc.com/en/personal-banking/banking/checking/virtual-wallet.html

[6] Anonymous, "Young People in Particular are Annoyed by Queues at the Cash Register - Germans are Open to Paying by Smartphone," PR Newswire Association LLC, 01 Jul 2014, ProQuest Newsstand, 01 Jul 2014.

[7] Anonymous, "USA ePay Joins the Secure Vault Payments Network," Business Wire, 16 Nov 2010, ProQuest Newsstand, 16 Nov 2010.

[8] The PCI Security Standards Council, "Verify PCI Compliance, Download Data Security and Credit Card Security Standards," Accessed 15 Mar 2015, https://www.pcisecuritystandards.org/organization_info/index.php

[9] D. Wolfe, "Breach Revives Doubts About Card Industry Security Standard," American Banker, 03 Apr 2012, ProQuest Newsstand, 17 Apr 2012

[10] D. Agrawal, A. El Abbadi, F. Emekci, A. Metwally, "Database Management as a Service: Challenges and Opportunities," IEEE 25th Int'l Conf. Data Engineering (ICDE 09), IEEE CS Press, 2009, pp. 1709–1716.

[11] A. Shamir, "How to share a secret," Commun. ACM, vol. 22, no. 11, pp. 612–613, 1979.

[12] B. Krebs, "Payment Processor Breach May Be Largest Ever," The Washington Post, 20 Jan 2009, Accessed 15 Mar 2015. http://voices.washingtonpost.com/securityfix/2009/01/payment_processor_breach_may_b.html

[13] T. Armerding, "The 15 worst data security breaches of the 21st Century," CSOOnline, IDG Enterprise, 15 Feb 2012, Accessed 15 March 2015. http://www.csoonline.com/article/2130877/data-protection/the-15-worst-data-security-breaches-of-the-21st-century.html

[14] Lagrange polynomial interpolation. http://www2.lawrence.edu/fast/GREGGJ/Math420/Section_3_1.pdf

[15] Free Software Foundation. What's in a Name? https://www.gnu.org/gnu/why-gnu-linux.en.html

[16] Oracle Corporation, NetBeans. https://netbeans.org/about/index.html

[17] Oracle Corporation, Java. https://www.oracle.com/java/index.html

[18] Oracle Corporation, MySQL. http://www.mysql.com/

[19] Oracle Corporation, MySQL JDBC Driver. http://www.mysql.com/products/connector/

[20] D. Flanigan, Java in a nutshell: a desktop quick reference. BigInteger subclass, Sebastopol, CA: O'Reily Media, Inc. Mar 2005, pp. 546.

[21] R. Barbulescu et al., in A Heuristic Quasi-Polynomial Algorithm for Discrete Logarithm in Finite Fields of Small Characteristic, 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, 2014 © International Association for Cryptologic Research. Doi: 10.1007/978-3-642-55220-5_1

[22] D. Flanigan, Java in a nutshell: a desktop quick reference. SecureRandom subclass, Sebastopol, CA: O'Reily Media, Inc. Mar 2005, pp. 639

[23] B. M. Brosgol, "A Comparison of the Mutual Exclusion Features in Ada and the Real-Time Specification for Java," http://www.adacore.com/uploads/technical-papers/MutEx-Ada-RTSJ-paper.pdf

A Hybrid Approach to Improving Cloud Data Security

E. Nwafor[1], **L.Burge**[1]
[1]Department of Systems and Computer Science, Howard University, Washington, DC, USA
ebelechukwu.nwafor@bison.howard.edu, blegand@scs.howard.edu

Abstract - *Cloud computing has been widely accepted by consumers and business organizations. It can be seen used in day to day activities from online movie content delivery to banking transactions. It has revolutionized the way software is being created, managed and distributed. It offers convenience to most business organizations, alleviating the need of having to maintain physical servers. It is also economical since you only pay for what you use. For public cloud, data from cloud consumers are outsourced to cloud service provider(s) who are responsible for managing data as long as it remains in the cloud. As with any computer system, there are security issues that exist with cloud computing. This paper talks about the security issues related to cloud computing and also proposes a hybrid approach that ensures the confidentiality, and integrity of cloud data using moving target defense, message authentication code, and symmetric encryption.*

Keywords- Cloud data security, moving target defense, Message authentication code, Data Encryption

1. Introduction

In recent years, cloud computing has become the "next big thing". Companies such as Google, Amazon are actively advertising the use of these products. More consumers are beginning to become aware of the benefit of this technology and how this could be utilized to their advantage. Cloud computing offers ease of access since all of the resources can be readily available on request. Consumers can rapidly deploy their applications to the cloud and do not have to manage physical servers which are handled by the cloud service providers. It also ensures that consumer's only pay for the services utilized. A huge component of cloud computing is Virtualization. Virtualization simply can be defined as running a virtual instance of a resource (hardware, software) on a single device. Virtualization is beneficial to cloud computing because it offers more use of hardware resources since multiple virtual instances of computer resources are running on a single hardware. This also poses a potential security risk since most of the virtual instances could possibly be from different users and issues could arise with one virtual instance overlapping into the memory location of another virtual instance (buffer overflow issues).

As to every technological development, there exist security issues with cloud computing. Some of which are traditional issues that exists with client server technologies (man-in-the-middle attack, Denial of Service). Data Storage is a major concern. How do users trust the service providers with their sensitive data? Also, how do we ensure that cloud users have control of the privacy of data stored in the cloud at all times (either at rest or in transit). These are some of the major questions researchers have been trying to unravel since its invention.

In order to address the issue of data storage, we propose a hybrid approach which involves the use of message authentication codes, moving target defense and symmetric encryption. This ensures the confidentiality and integrity of cloud data while at rest or in transit and also gives the cloud users control over the privacy of their data. This paper is divided into three sections: Section 1 gives a brief overview on Cloud computing, Section 2 talks about the issues that exists with cloud computing focusing on cloud data storage, Section 3 talks about related research done on cloud security while the last section focuses on our proposed model.

2. Overview of Cloud Computing

Cloud computing [11] can be defined as a model for enabling ubiquitous network access to a shared pool of computer resources (e.g. network servers, storage applications and services) .It involves the rendering of services (i.e software, virtual severs) over the internet/intranet. Cloud computing is composed of service models and deployment models, and essential characteristics [11]. There are three types of cloud computing service models:

i. *Infrastructure as a Service (IaaS):* This service model involves the use of equipment provided by the cloud service provider for business operations. In this model, consumers can use cloud resources to run computations remotely. The consumer does not manage the cloud infrastructure but can control certain features on the cloud system such as network, operating system, applications deployed.

ii. *Software as a Service (SaaS):* This is the most popular form of cloud service model. It allows consumers to remotely use software located on a vendors' system. Users might be charged based on the amount of time spent using the service, an example of such service is Netflix, an application that allows movie streaming over the internet.

iii. *Platform as a Service (PaaS)* involves the rendering of software application development services (i.e. software application design, software application development) an example of this is Google App Engine. This model is mainly tailored to applications developers.

Cloud Deployment model are as follows:

Public Cloud: This cloud infrastructure is open to the general public. Anyone can use this cloud to store, process data.

Private Cloud: This cloud infrastructure is provisioned for use by a select organization. This cloud model might be restricted to a select network. It may be run by an organization, a third party , or a combination of both the organization and a third party [11].

Hybrid Cloud: This cloud infrastructure is a combination of both public and private cloud. Most organizations seem to employ this model because it allows a way of keeping non-sensitive portions of data on the public cloud and keeping sensitive data in the private cloud.

Community cloud: This is a cloud deployment model which is made for use by a specific community with similar interest. It may be owned or managed by the community, a third party, or one or more organizations contained in the community [11].

Some of the essential characteristics of cloud computing are described below:

i. *Pay as you go service:* Users only pay for the amount of services utilized.
ii. *Elasticity:* Cloud computing offers the ability to rapidly release resources at any time. This allows consumers to be able to request resources at any time and release back to the resource pool once finished.
iii. *On-demand self-service:* Resources such as storage space could be assigned automatically by a click of a button without having to involve human intervention.
iv. *Shared Resources* in which no resource is dedicated to the user. These resources could be used by multiple users concurrently thereby increasing the efficiency of the resource.

3. Security Issues with Cloud Computing

As to every technological innovation, there are many issues with Cloud computing. Some of these issues are traditional internet issues (e.g. issues arising from of passing data over a network, Malicious software attacks), While others are as a result of the implementation of the cloud computing model and issues arising with data storage. Some of the known issues are listed in the next section.

3.1 Issues with Client-Server Technology

Most of the issues that arise with cloud computing are typical client- server issues that have been in existence for a while. Some of these attacks are hard to detect which makes it difficult to mitigate. Some known issues with client-server technology are described below:

SQL injection attacks involve inserting malicious code into standard SQL query. This is with the aim of gaining unauthorized access to the database which might contain sensitive information. In some cases, an attacker can modify the contents of the database.

Man –in – the middle attack is a classic attack scheme in which an attacker eavesdrops on a communication between client-server and modifies contents of the communication. In most cases, both parties are unaware of the communication eavesdrop by the attacker.

Cross Site Scripting (XSS) attacks involves injecting malicious scripts into websites. This allows the attacker to inflict havoc in a client machine. There are two ways of injecting malicious code: Stored XSS and reflected XSS. Stored XSS involves permanently inserting the malicious code into the web content. In Reflected XSS, the malicious code is not permanently stored on the web content.

Denial of Service (DoS): DoS attack involves overloading a server with request messages. This reduces the efficiency of the server and makes it venerable to attacks from hackers. It also can lead to a system crash. This could be a financial burden because the more the information requested, the higher the fees charged by a cloud vendor.

3.2 Issues with Data Storage

This is one of the major issue facing the development of cloud computing. How data is stored is an essential part of cloud computing. Most business co-operations are reluctant to store their data in a public cloud since users have little to no determination on how and who has access to the data stored in the cloud, an issue arises with trust. Users have to trust the third party (Cloud providers) rendering these services hoping the data would be securely stored on their virtual servers. This might not be the case. For data at rest or in-transit to be secure it has to contain the three essential characteristics

Confidentiality: Data must be protected from unauthorized access. This can be achieved by using encryption techniques.

Integrity: The data sent must not be tampered with by anyone. This can be achieved by using digital signatures.

Availability: Data must be readily available at all times. To this end, we propose a hybrid approach which ensures the confidentiality and integrity of data in cloud systems while at rest or in transit.

4. Related Works

The authors of [9] propose a Self-Cleansing Intrusion tolerant system, a method to mitigate system vulnerabilities that exists in an application. According to the authors, not all intrusions can be detected and blocked in a timely manner. There should be an assumption that a compromise of the system exists regardless of if intrusion was detected or not hence a periodic performance of a self-cleansing operation which wipes information from the system and reinstalls based on a system checkpoint [9]. The device which contains sensitive information is mirrored to another trusted device in a secured

manner. Periodically, the system is automatically brought off-line for self-cleansing and integrity check [9] while the systems mirror which serves as the backup is rebooted. This is accompanied by a system recovery, a checkpoint, rollback, and data integrity check routines. This technique of intrusion tolerance can be applied to not just servers but routers and most computer systems. This model is particularly useful to computer system that contain static or semi-static data, computer systems that are state-less, and servers that handle short session (request-response task).

Kurra et al [7] proposes a resilient cloud storage architecture which ensures the confidentiality, integrity and availability of cloud data. This is achieved using moving target defense mechanisms and key hopping techniques. A client makes a request to access a cloud data, the information containing the client's role is check to see if it is authorized to access the information requested. The certificates containing information about the public key of the client is also verified. If the authentication is unsuccessful, the client system is added to a block list. The data is partitioned into an arbitrary amount of partitions [7] and encrypted using different symmetric keys. This key is periodically changed based on a specified time window. According to the authors, this technique improves the performance by 50% when using a key of length 512 bytes as compared to the traditional certificate techniques which uses a key of length 2048 bytes [7].

Haadi et al [8] looks at Moving target defense technique involving the random mutation of host IP address based on spatial randomness as well as time. This technique limits the timeframe an attacker has if it gains access to the network since the IP address of each host on the network is constantly changed. Each host is assigned with a set of IP addresses known as ephemeral IP address (eIP). The ephemeral IP address is mapped to the actual IP address. The binding of real IP address to eIP is achieved by randomly selecting an eIP from an unused address space based on uniform distribution [8]. Each host can only reach other host in a network during a specified time interval based on the eIP currently assigned to the system. This is referred to as spatial- host IP binding and is unique for each host contained in the network. The IP mapping from real IP address to ephemeral IP address contains a Time to live (TTL) component and this IP binding is frequently changed at an assigned time interval.

The authors of [10] look at a technique of evading port and network address attack using port and address hopping. The addresses and port numbers are mapped to a randomly generated address. This information is sent to the server through the Network Address Translator (NAT) contained in both the server and client portions of the network. NAT contains information of the real address of the server and maps the randomly generated port and IP to the real address. This random address and port number is often changed based on a specified time interval. An attacker is only able to see port and IP address which can only be used for a limited amount of time.

5. Discussion

Contrary to what most people believe, cloud computing is not a new technology. It stems from pre-existing technologies which spans various areas of computing with distributed systems being one of the major areas. Since data is mainly outsourced to a third party which are the cloud service providers, the need to properly secure consumer data is of utmost importance. We present a hybrid approach to cloud security by using techniques such as moving target defense, message authentication code, and symmetric encryption to ensure the confidentiality, and integrity of cloud data. Some of the key concept used is discussed below:

5.1 Moving Target Defense

Most systems are developed with static configurations. That is, they have the same configurations throughout their lifecycle [12]. This makes it easier for an attacker to study the vulnerabilities that might exist in these systems and exploit them. Moving target defense is a technique of making it difficult for an intruder to understand the vulnerabilities that might exist in a system by constantly changing the configuration of the system [10]. Our proposed model will address the network layer by constantly changing the configuration of the network layer using network address hopping techniques thereby increasing the complexity of an attacker gaining unauthorized access to a system.

5.1.1 Types of Moving Target Defense

Homogenous moving target defense: This is a network in which all nodes on the network employ the moving target defense scheme. Since all nodes contained in the network are constantly changing, this makes it difficult for an attacker to study the vulnerabilities that might exist in the system.

Heterogeneous moving target defense: This involves configurations in which a combination of static nodes and nodes with Moving target defense are deployed in the network. This form of network is not ideal because an attacker can study the network to determine what nodes have static configurations and try to exploit them.

5.2 Message Authentication Code (MAC)

MAC, also known as Keyed Hash function [13], is a method of ensuring cloud data integrity and authenticity. This is achieved with a combination of a key and a cryptographic hash. The algorithm accepts as input a symmetric key with the data to be protected and produces a message authentication code. The message to be sent is run through the MAC algorithm with a symmetric key which is agreed on by the sender and the receiver. The output generated is a Hash value known as a tag. This information together with the message is sent to the receiver. The receiver, on retrieval, verifies the message received using the symmetric key. The receiver generates a second MAC for the message received which is compared to the tag that was initially received from the sender. If the message has been intercepted or tampered, both

tags will differ. This ensures the integrity of the message received.

5.3 Symmetric encryption

Symmetric encryption involves the use of a single key in the decryption and encryption process. A secret key is generated which is used to protect information. This ensures the confidentiality of cloud data. The algorithm used is for our proposed model.

5.4 Asymmetric Encryption

This involves the use of separate keys (private and public keys) for encryption and decryption. The private key is used for encryption while the public key is used of decryption.

6. Our proposed model

Our proposed model uses techniques such as Message authentication code, symmetric encryption, Moving target defense to ensure the confidentiality, and integrity of cloud data. The IP address assigned is mapped to a randomly generated IP address. This ensures that the critical services such as DNS servers are not disconnected also, to reduce the complexity of dynamically updating and changing all addresses contained in the network. This IP mapping is undergone internally by a Translator service which contains information about various mapping from actual IP to generated IP address. This information is further protected using SHA-1 256 bit Digital signature and AES 256 bit symmetric encryption to ensure that it is not accessed by an unauthorized party. The random IP address is generated by the IP generator component. This is achieved by using a random number generator. If an attacker gains access to the network, the information contained is encrypted and signed also, the IP address is only valid for a limited period of time (e.g 3 mins). Due to the dynamic nature of the IP configuration, this makes network reconnaissance by an attacker difficult or near to impossible.

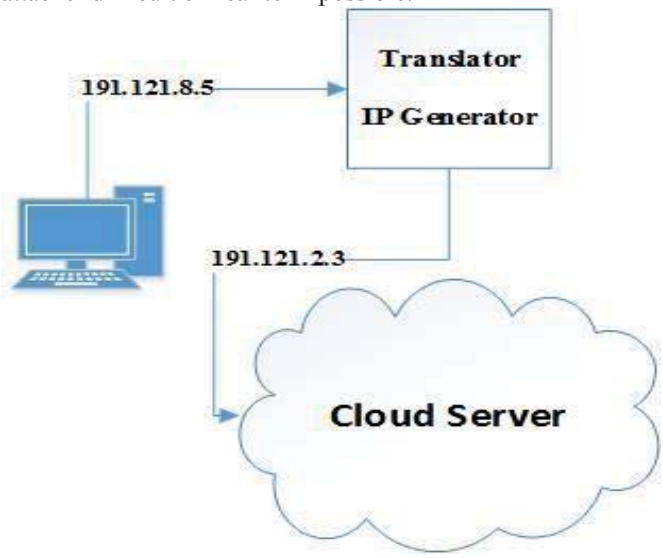


Figure II: Design Implementation for Moving Target Defense scheme

Data sent to the cloud server is partitioned into different data fragments ($B_1, \ldots, B_z$)
.Each data fragment is hashed using Message authentication code and encrypted using 256-bit AES symmetric encryption. $E(H(B_i))$. i represents an arbitrary index for each data fragment partition, E and H represents the encryption and Hash function of the partitioned data fragments respectively. A unique identifier, A_i, is generated for each data fragment and mapped to its respective data fragment. This allows for search of each data component based on its respective identifier. The identifier is encrypted using RSA asymmetric encryption, $E(A_i)$ and the public key of the server is shared with the client. Even if the data in the cloud is compromised from the cloud server in a data breach, an attacker can only obtain information about the identifier but not the encrypted partitioned data segment. The contents are encrypted with respective symmetric keys which are stored in the client system. Also as an extra security measure, the partitioned data fragments are uploaded to various cloud vendors. This increases the complexity of getting all of the data fragments at once in a case of a data breach. Decryption of data is done in the client system. This gives clients complete control of their data since encryption and decryption of data fragments are done on the client side with the ecnryption keys stored in a private cloud server on the client network.

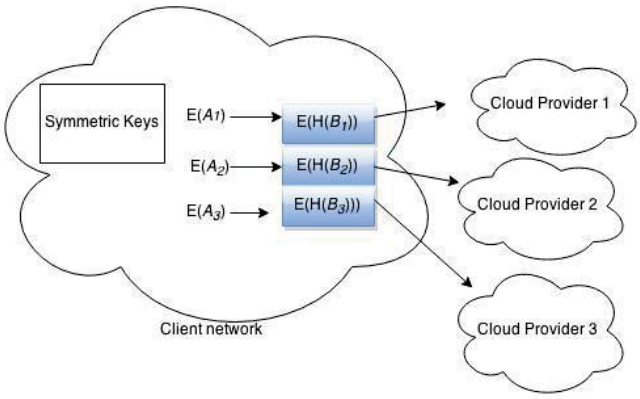


Figure III: Data partition security Architecture

The combination of these security schemes ensures the confidentiality and integrity of data stored in the cloud either in transit or at rest.

7. Future works

A working prototype for the proposed model is currently being developed to demonstrate the functionality of the proposed system. This model will be compared with similar security models to determine its effectiveness. Also, we will run test to see if there are any additional computational overhead incurred by using this approach.

8. Conclusion

This paper gives an overview of cloud computing and talks about some of the security issues associated with cloud computing. It also introduces a hybrid approach to cloud data security which utilizes some techniques such as moving target defense, message authentication code and symmetric encryption. This ensures the confidentiality and integrity of the data stored in the cloud.

9. References

[1]. Kamal Dahbur, Bassil Mohammad, Ahmad Bisher Tarakji, "A Survey of Risks, Threats and Vulnerabilities in Cloud Computing" Proceedings of the 2011 International Conference on Intelligent Semantic Web-Services and Applications, April 2011.1-6

[2]. Richard McDougall, Jennifer Anderson, "Virtualization Performance Perspectives and Challenges Ahead" SIGOPS Operating Systems Review, Dec 2010.

[3]. John C. Roberts, Wasim Al-Hamdani, "Who Can You Trust in the Cloud? A Review of Security Issues Within Cloud Computing" Proceedings of the 2011 Information Security Curriculum Development Conference. Sep, 2011.

[4]. Karissa Miller, Mahmoud Pegah "Virtualization, Virtually at the Desktop" Proceedings of the 35th annual ACM SIGUCCS fall conference.2007

[5]. Jens-Sönke Vöckler, Gideon Juve, Ewa Deelman, Mats Rynge, Bruce Berriman, "Experiences using cloud computing for a scientific workflow application" Proceedings of the 2nd international workshop on Scientific cloud computing. June 2011

[6]. Huijun Xiong, Xinwen Zhang, Danfeng Yao, Xiaoxin Wu, Yonggang Wen "Towards End-to-End Secure Content Storage and Delivery with Public Cloud" Proceedings of the second ACM conference on Data and Application Security and Privacy.2012: Pages 257-266

[7]. Hemayamini Kurra, Youssif Al-Nashif, and Salim Hariri. 2013. "Resilient cloud data storage services". In Proceedings of the 2013 ACM Cloud and Autonomic Computing Conference *(CAC '13).* ACM, New York, NY, USA.

[8]. Jafar Jafarian, Ehab Al-shaer. Qi Duan. "Spatio-temporal Address Mutation for ptotective cyber Agility against sophisticated Attackers" In Proceedings of the First ACM workshop on moving Target Defense. Pages 67-78

[9]. Yih Huang and Arun Sood, "Self-Cleansing Systems for Intrusion Containment", Proceedings of Workshop on Self-Healing, Adaptive, and Self-Managed Systems (SHAMAN), New York City, June 2002

[10]. M. Atighetchi, P. Pal, F. Webber, and C. Jones. Adaptive use of network-centric mechanisms in cyber-defense. In ISORC '03, page 183. IEEE Computer Society, 2003.

[11]. Peter Mell, Timothy Grance, "The NIST Definition of Cloud Computing" NIST Special Publication 800-145. September 2011.

[12]. David J. John, Robert W. Smith, William H. Turkett, Daniel A. Cañas, and Errin W. Fulp. 2014. Evolutionary based moving target cyber defense. In Proceedings of the 2014 conference companion on Genetic and evolutionary computation companion (GECCO Comp '14). ACM, New York, NY, USA,

[13]. The Keyed-Hash Message Authentication Code (HMAC). Federal Information Processing Standards Publication. Information Technology Laboratory, National Institute of Standards and Technology, July 2008.

Security Overlay for Distributed Encrypted Containers

Florian Patzer[1], Andreas Jakoby[2], Thomas Kresken[1], Wilmuth Müller[1]
[1]Fraunhofer Institute of Optronics, System Technologies and Image Exploitation – IOSB
Karlsruhe, Germany
[2]Bauhaus Universität, Weimar, Germany
florian.patzer / thomas.kresken / wilmuth.mueller@iosb.fraunhofer.de
andreas.jakoby@uni-weimar.de

Abstract: *Storage services enable a high potential for time and location independent access to information particularly combined with smart mobile devices. In combination with corporate and local storage, those services can be a powerful extension to available storage in enterprise or governmental environments. In contrast, common secure storage strategies like encrypted partitions or disks are static and remotely inaccessible, but are comfortable to use in a local scenario. However, storing sensitive data on public servers is not an option due to the possibility that an unauthorized third party can access it. Generally security policies like corporate compliance prohibit those services explicitly. Thus, sensitive data has to be encrypted to allow its storage on public servers.*

The paper at hand describes a security overlay using a trusted environment to build a distributed virtual encrypted container that supports OTFE (on-the-fly-encryption). For this purpose, an easily extendable security overlay is introduced where each file or data set is encrypted independently. The overlay provides a hierarchical key structure, which hierarchically controls access to uploaded data and maps the data structure at the same time. Additionally, the directory structures and the meta-data are protected against unauthorized access. Therefore, the presented concept enables the creation of a deniable distributed file system that can enable an implementation to make strong security promises.

The trusted environment can be provided by a device called CyphWay®, which has been developed at the Fraunhofer IOSB and presented at ICCWS 2014. The device guaranties that cryptographic keys are only available within a Hardware Security Module. Thus, the whole key structure and the keys themselves are protected even against the user devices, which is important regarding potentially insecure mobile platforms.

Unlike several encrypted container solutions the presented system allows to distribute encrypted data over a huge number of divergent publically available storage services, like cloud storages. In addition, it is possible to combine those storages with private or corporate storage.

Keywords: *Security Architecture, Secure Cloud Storage, Mobile Security, Information Security Management, Secure Distributed Storage*

1. Introduction

Within the last years, storage became available anywhere at any time. The booming Storage-as-a-Service (*STaaS*) market, advanced possibilities to access corporate storage remotely and mobile devices as smartphones, tablets or laptops make this flexibility possible. But with the increasing amount of mobility and storage diversity the types and numbers of possible attacks on the stored data and the corresponding keys increases also, even when modern cyber suits are applied. Classical strategies like encrypted containers, partitions and disks need to be mapped on mobile and remote usage. Distributed storage is attractive to private as well as professional users. STaaS or *cloud storage* providers offer clients to remotely access their storage, which is often enabled through applications for different platforms like Android, Windows, Ubuntu and (platform independent) web browsers. In addition, most of the cloud providers offer different amounts of space in their prizing model and some advertise small storages free of charge. This makes them interesting for all kinds of users. However, because of security policies and corporate compliance, the usage of such services is not common in corporate and governmental environments. This is mainly because sensitive data can be easily read by the storage providers. Many applications and techniques addressing this problem do not consider the storage provider as an attacker and are, for that reason, no solutions for professional environments. As an example, US cloud providers are forced to release even sensitive data of their customers to the US government due to the USA Patriot Act.

However, users who are allowed to work with cloud, local and corporate storage have to use several clients to manage their data. There are very few solutions that allow to combine those storages,

especially in mobile operating systems. Systems that provide this desirable feature, do not meet security requirements for sensitive data and are therefore often useless in professional environments.

Lately, new possibilities of assembling cloud storage on the client side have been created ([1], [7], [3]). Those clients are not satisfying in terms of the provided security. Additionally, none of the available products and concepts provides a satisfying directory structure that contains distributed elements and can be used as one single directory structure, like local encrypted containers. In advance, some use cases might require client software, where users do not have to care about the distribution of their data over different storage locations.

On the server side assembling storage that is situated on different locations is currently done by distributed file systems like Andrew File-System [5] or Google File System [4]. Those have been developed to fulfill the needs of data centers and are mostly limited to the file system used by the respective center. A user accessing those systems is bound to their technology.

There are techniques available that partially provide more flexibility. Distributed file systems such as Tahoe [10] and its advancements like [9] allow the combination of diversely located storages with different underlying file systems. However, as every location needs to run the same virtual file system or server software, those techniques do not address the desired usage of STaaS, which requires the support of diverse interfaces. That is why they are only used in environments where a standardized file system or server software is deployed. In the domain of the global web business, this alignment would be against the desires of one provider to dissociate oneself from the competition, keeping their customers dependent. Additionally, a migration could imply tremendous costs.

There is also work available that concentrates on the security of distributed file systems [8]. Those methods depend on a trustworthy environment that is assumed as given when deploying the client software. The work that is been made on the field of security of distributed file systems does not fulfill the need of an appropriate level of security when mobile devices are used as clients. That is because the trustworthy environment that is used to perform cryptographic operations in the mentioned concepts is the mobile device itself. The security of those devices is widely contentious. Especially law enforcement and military usage of storage services needs a higher level of protection in a mobile environment. This can be achieved by using an external hardware device like the CyphWay® that provides such a trusted environment and is explained later in this document.

The paper at hand provides a technique to map classical encrypted containers to modern storage strategies, like the storage of critical data on public services. The document presents a security overlay that can be applied to create a distributed virtual encrypted container to achieve a high degree of security and good user experience when combining several unprotected storage locations. A storage location is defined as unprotected, if an unauthorized individual can access a dataset that is stored on that location or available anywhere within the network. We suppose that any remote storage and any channel is insecure. Additionally, we admit that user devices can be compromised as mobile devices tent to have many security issues. Therefore, the data to be stored needs to be protected carefully by encryption and the keys, applied to those encryption instances, need to be protected even against the user devices. The overlay is supported by an external trusted hardware device (comparable to a hybrid of TPM[1] and HSM[2]) to perform cryptographic operations. This device is the only location where keys ever appear in plaintext. The resulting environment is supplemented by a specially designed key management system. The stored data and its meta-data are being encrypted and controlled by access rights. In addition, the overlay achieves the protection of the data structure by hiding it completely from unauthorized individuals.

2. Overlay Structure

At first, the overlay structure of our concept is presented. This is the key to allow the desired high level of security and uses the necessary level of abstraction to build a distributed container. Therefore, it is shown how the data and also the meta-data get protected and hidden by this technique. This will be achieved by separating the meta-data from the actual data.

Data Structure: Let $G = (V, \ E)$ be a directed graph, where the vertices denote directories or files. Edges represent associations in the following way:

> If $(u, \ v) \in E$ then u represents a directory which includes a sub-directory or a file represented by v. In other words, u is parent of v.

Note that files are always represented by sinks and that in this document file-level granularity is used, but in general different granularity levels are conceivable. This might even be a recommendable parameter for implementations. In addition, a bijective function

[1] Trusted Platform Module
[2] Hardware Security Module

$i: V \cup E \to N$ is defined that maps the graph elements to a set of indices, in order to identify them.

Now, a simple notation to navigate through G is provided. This will later be mapped on a key-based navigation. For $v \in V$ let $\Gamma^-(v) = \{w|(v,w) \in E, w \in V\}$ and $\Gamma^+(v) = \{w|(w,v) \in E, w \in V\}$ denote the outgoing end entering edges of v. For every vertex $v \in V$ it is assumed that there exists a given order within $\Gamma^-(v)$ and within in $\Gamma^+(v)$. Let $\text{in}(v)$ denote the first edge in $\Gamma^+(v)$ and let $\text{out}(v)$ denote the first edge in $\Gamma^-(v)$. For edges $e_1 = (u, \quad v) \in E$ let $\text{in}(e_1)$ denote the direct successor of e_1 within the edges in $\Gamma^+(v)$ and let $\text{out}(e_1)$ denote the direct successor of e_1 within the edges in $\Gamma^-(u)$.

Meta-data Structure: The meta-data, mentioned before, includes information like file and/or directory names, additional access restrictions, location of storage, etc. In the following, meta-data will be extracted from the respective file or directory in order to build a meta-graph.

The partial functions $\sigma: N \to M_V$ and $\rho: N \to M_E$ represent mappers that link the real data to the related meta-data, where N denotes a universe of indices, M_V denotes the universe of the meta-data extracted from the vertices of G and M_E denotes the universe of meta-data extracted from the edges of G. Consequently, the resulting meta-graph is defined as $G_M = (M_V, M_E)$. Furthermore, in favor of an intuitive understanding $M_v \in M_V, M_e \in M_E$ are written as synonyms for $\sigma(v) \in M_V, \rho(e) \in M_E$.

The location of a real file is part of its meta-data. Therefore, the function $\tau: N \to S$ is needed which maps an index $n \in N$ onto a storage location $s \in S$ where S denotes the set of all storage locations that shall be included in the desired storage distribution. To map the storage location $\tau(n)$ onto the graph element and its meta-data we define $\varphi: S \to (V \text{x} M_V) \cup (E \text{x} M_E)$. As a result, we can use $\tau(n)$ to access the storage location where the graph element $\varphi(\tau(n))$ is stored for any $n \in N$.

Design: Because every stored structure and information within this overlay will be protected by using encryption, the following abstract encryption scheme (Enc, Dec) is introduced. It is intentional that the overlay does not rely on a particular encryption scheme and by abstracting from such schemes, their exchangeability is guaranteed. Let Enc denote the encryption and let Dec denote the decryption function. The encryption of the data d is denoted by $Enc(k_{Enc}, \quad d) \to c$ and the decryption is denoted by $Dec(k_{Dec}, \quad c) \to d$ where k_{Enc} and k_{Dec} are the respective encryption and decryption keys. In favor of simplicity we assume $\Pr[Dec(k_{Dec}, \quad Enc(k_{Enc}, \quad d)) \to d] = 1$ for all possible data instances d. The encryption strategy works as follows:

- M_v is encrypted using an (approximately) unique key pair (k_{Enc}^v, k_{Dec}^v), for every vertex $v \in V$.
- M_e is encrypted using an (approximately) unique key pair (k_{Enc}^e, k_{Dec}^e), for every $e \in E$. This is a simplified exposition of the key structure. It will be extended within the next paragraphs.
- The meta-data set of each vertex includes keys that are needed to encrypt and decrypt the meta-data of the incident edges. For example, let $e = (v,w) \in E$ then M_V contains (k_{Enc}^e, k_{Dec}^e).
- The meta-data set of each edge includes keys that are needed to decrypt the meta-data of its connected vertices.
- In addition, the meta-data set of any edge is encrypted by an access-right key which guarantees that only users with access rights to the end-vertices can gain any information about the corresponding vertex.

Consequently, the minimal content of each $M_v \in M_V$ and each $M_e \in M_E$ is set as follows: The meta-data set M_v of a vertex v contains

- The name of the represented directory or file.
- The values $\text{in}(v)$, $i(\text{in}(v))$, $\tau\left(i(\text{in}(v))\right)$ as well as $\text{out}(v)$, $i(\text{out}(v))$, $\tau\left(i(\text{out}(v))\right)$.
- A set of key pairs K^{in} used to encrypt the meta-data of the edges in $\Gamma^+(v)$ and a set of key pairs K^{out} used to encrypt the meta-data of the edges in $\Gamma^-(v)$. These entries are of the form (k_{Enc}^e, k_{Dec}^e) for the adjacent edges.

The meta-data set M_e of an edge $e = (u, \quad v)$ contains:

- The name of the directory or file represented by u and v.
- The values $i(\text{u})$, $\tau(i(\text{u}))$ as well as $i(v)$, $\tau(i(v))$.
- The values $\text{in}(e)$, $i(\text{in}(e))$, $\tau\left(i(\text{in}(e))\right)$ as well as $\text{out}(e)$, $i(\text{out}(e))$, $\tau\left(i(\text{out}(e))\right)$.
- A key pair $(k_{Enc}^{\text{head}}, \quad k_{Dec}^{head})$ used to encrypt the meta-data of u and the key pair $(k_{Enc}^{\text{tail}}, \quad k_{Dec}^{\text{tail}})$ used to encrypt the meta-data of v.

As mentioned, there is more to the encryption of M_e. An additional encryption of the meta-data edges is needed to handle access rights (see Section 3). Thus, every $M_e \in M_E$ will be encrypted using an additional key pair $\left(k_{Enc}^{U(v)}, \; k_{Dec}^{U(v)}\right)$ where $e = (v,w) \in E, v,w \in V$, before encrypting it using (k_{Enc}^e, k_{Dec}^e). Therefore, the meta-data structure can be implemented as tuples as follows:

- $\langle i(\mathrm{v}), \; Enc(k_{Enc}^v, \; M_v)\rangle$ for every $v \in V$
- $\langle i(e), \; Enc(k_{Enc}^e, \; \mathrm{in}(e)), \; Enc(k_{Enc}^e, \; \mathrm{out}(e)), \left(k_{Enc}^{U(v')}, k_{Dec}^{U(v')}\right) E\left(k_{Enc}^{v'}, M_e\right)\rangle$ for every $e = (v,w) \in E, w \in V$

At this point, the original data structure is fully mapped by the meta-data structure. As a result, the original data can now be diversified on arbitrary storages. This data has to be encrypted. Therefore, one may add additional keys and encryption schemes, but it is suggested to use the key pair (k_{Enc}^v, k_{Dec}^v) for a file or dataset represented by a vertex $v \in V$.

The presented overlay allows the implementation of an encrypted container that can be used on several devices simultaneously and not only stores the data in a cloud but facilitates the user to include multiple corporate and external STaaS entities. Furthermore, the overlay is designed to create comfortable client software which provides a container that can be utilized like a local directory.

Algorithmic Examples: The following algorithms are minimalistic in favor of simplicity and focus on the overlay manipulation. Some details like extractions through decryption are avoided as they are implicitly clear. The content of a directory associated with a given vertex v is computed as follows:

determineDirectoryContent($i(v)$, (k_{Enc}^v, k_{Dec}^v))
begin
decrypt $Enc(k_{Enc}^v, \; M_v)$ using k_{Dec}^v
e = $(v,w) \leftarrow \mathrm{out}(v)$
determine $k_{Dec}^{\mathrm{e}} \leftarrow K^{out}$
loop until the successor edge e is not defined
determine $Enc(k_{Enc}^e, \; M_v)$ by decrypting $E\left(k_{Enc}^{\mathrm{e}}, \; E\left(k_{Enc}^{U(w)}, \; M_e\right)\right)$ from
$\langle i(e), \; Enc(k_{Enc}^e, \; \mathrm{in}(e)), \; Enc(k_{Enc}^e, \; \mathrm{out}(e)), \; Enc(k_{Enc}^e, \; E(k_{Enc}^{U(w)}, \; M_e))\rangle$
if the user has access to $k_{Dec}^{U(w)}$ then
decrypt $Enc\left(k_{Enc}^{U(w)}, \; M_e\right)$
add the necessary content information from M_e
to the content of v
end
$e = (v,w') \leftarrow \mathrm{out}(e)$ by decrypting $Enc(k_{Enc}^e, \; \mathrm{out}(e))$
end
end

A new vertex v' can be added to $v \in V$ as follows:

appendFileOrDirectory($i(v)$, $i(v')$, (k_{Enc}^v, k_{Dec}^v))
begin
determine M_v
create $M_{v'}$ M_e where $e = (v,v') \in E$
generate $\left(k_{Enc}^{U(v')}, k_{Dec}^{U(v')}\right)$ $(k_{Enc}^{v'}, k_{Dec}^{v'})$ (k_{Enc}^e, k_{Dec}^e)
add (k_{Enc}^v, k_{Dec}^v) as $(k_{Enc}^{head}, k_{Dec}^{head})$ and $\left(k_{Enc}^{v'}, k_{Dec}^{v'}\right)$ as $\left(k_{Enc}^{tail}, k_{Dec}^{tail}\right)$ to M_e
add (k_{Enc}^e, k_{Dec}^e) to K^{out} of M_v and to K^{in} of $M_{v'}$
determine in(e) and do $Enc(k_{Enc}^e, \; \mathrm{in}(e))$
determine out(e) and do $Enc(k_{Enc}^e, \; \mathrm{out}(e))$
do $Enc\left(k_{Enc}^e, \; E\left(k_{Enc}^{U(v')}, \; M_e\right)\right)$
end

Moving a vertex v' from parent v to parent w can be easily done by applying the following algorithm:

appendFileOrDirectory($i(v)$, $i(v')$, $i(w)$, (k_{Enc}^v, k_{Dec}^v), $\left(k_{Enc}^{U(v')}, k_{Dec}^{U(v')}\right)$, (k_{Enc}^w, k_{Dec}^w))
begin
determine M_v, M_e, M_w where $e = (v,v') \in E$
remove (k_{Enc}^v, k_{Dec}^v) (resp. $(k_{Enc}^{head}, k_{Dec}^{head})$) from M_e
remove e from $\Gamma^-(v)$
remove (k_{Enc}^e, k_{Dec}^e) from K^{out} of M_v
add e to $\Gamma^-(w)$
add (k_{Enc}^e, k_{Dec}^e) to K^{out} of M_w
add (k_{Enc}^w, k_{Dec}^w) as $(k_{Enc}^{head}, k_{Dec}^{head})$ to M_e
end

To speed up the access on objects that are usually used at the same time, like the edges which represent the content of a directory, we can group these objects and store them within a meta-object at the same location.

If a user navigates through the visible directories he gets some knowledge on the hidden vertices of the directory tree, like the number of entries or the number of predecessors. To hide this kind of information we can add some dummy entries to our system. A dummy entry consists of a random edge "cipher", which appears within the lists $\Gamma^-(v)$ and $\Gamma^+(v)$ of a vertex v, but has no meaningful decrypted meta-data, i.e. the ciphertext will be a random string.

3. Handling Access Rights

In Section 2, the basic overlay structure and its encryption was presented. However, there is still no access handling regarding different users. To control the access rights of an individual user it has to be distinguished between two scenarios:

1. The initial access to a directory or file within the container, i.e. the first access of the user to an element within the file system.
2. The user has currently access to a directory. Therefore, her or his access rights regarding the directory content and the parent directories has to be controlled.

The control of the initial access within the first scenario follows the method to control the access on data records in a cloud as presented in [6]. An example can be found within Figure 1

For each user u a user key pair $(k^u_{Enc}\ \ k^u_{Dec})$ is introduced. It is assumed that this key pair is stored in such a way that no unauthorized person has access to this pair (i.e. within the trusted environment that unlocks the key pair, if the user has authenticated himself via fingerprint). To manage the access rights it might be possible that some kind of a super-user within an organization must have access to these keys and is able to add new user keys (i.e. in the trusted environment).

If a user u has access to a vertex v within the file graph structure, the key pair $(k^v_{Enc},\ k^v_{Dec})$ and the corresponding storage location are stored as an entry point. This kind of information is called adaptor data A^u_v and is protected through encryption $Enc(k^u_{Enc},\ A^u_v)$. Thus, if a user u would like to access a directory or a file represented by a vertex v, he reads the corresponding encrypted adaptor data $Dec(k^u_{Dec},$ $Enc(k^u_{Enc},\ A^u_v)$. After extracting $(k^v_{Enc},\ k^v_{Dec})$ he or she can access the meta-data set M_v.

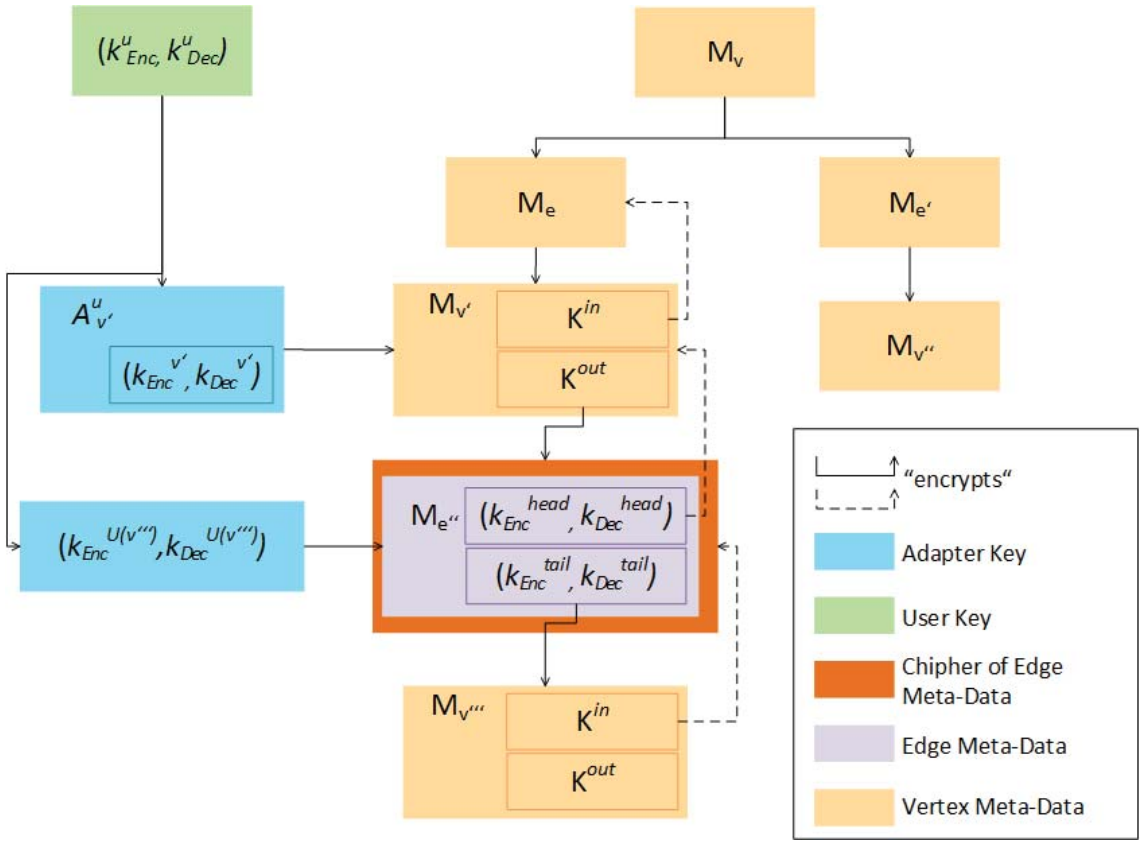


Figure 1: Illustration of an encrypted meta-data tree with an access example

To add another entry point for a user one can simply add the corresponding encrypted adaptor data for this user to the system. Analogously, if one would like to remove an entry point of a user one can remove the corresponding encrypted adaptor data from the system.

To control the access rights within the first scenario the key pairs $\left(k_E^{U(w)}, \quad k_D^{U(w)}\right)$ for $e = (v.w) \in E$ have been introduced.

It is possible to group the graph elements according to users who can access them. For any vertex v let $U(w)$ denote the set of users that have access to M_w. Then $\left(k_{Enc}^{U(w)}, \quad k_{Dec}^{U(w)}\right)$ represents the access key pair that is used to encrypt the meta-data of the edges with tail w. Analogue to the adaptor data these keys are stored encrypted by the user keys k_{Enc}^{u} for every user $u \in U(v)$.

4. Trusted Environment

As mentioned before, the trusted environment needed for cryptographic operations and access control within the presented overlay, can be provided by a hardware security device like the CyphWay®. The CyphWay® was developed at the Fraunhofer Institute of Optronics, System Technologies and Image Exploitation. The implemented demonstrator uses Android devices to visualize the data and is implemented on Raspberry Pi's using FPGAs for the necessary cryptographic operations and for the storage of administratively entered or cached keys. Overview of this demonstrator system is illustrated within Figure 2.

To access data the demonstrator tries to decrypt the corresponding encrypted meta-data. If the required key is not locally available for the CyphWay®, it tries to fetch the necessary key. Therefore it invokes a remote storage lookup to obtain a cipher of this key that can be decrypted by applying the current user key. If a user would like to access a directory where she or he is not authorized to access all subdirectories and included files, the CyphWay® filters the elements of the directory. It only forwards that information to the user which has been decrypted successfully. Consequently, elements the user is not authorized to see will be invisible for him or her.

The smartphone as well as any other final user device is only used to connect the CyphWay® to the storage and to manage the content of the directories and files. Plaintext keys are never available outside the the hardware security module of the CyphWay®. In addition, no cryptographic operations take place on the user device, which only signals encryption and decryption interests to the CyphWay®.

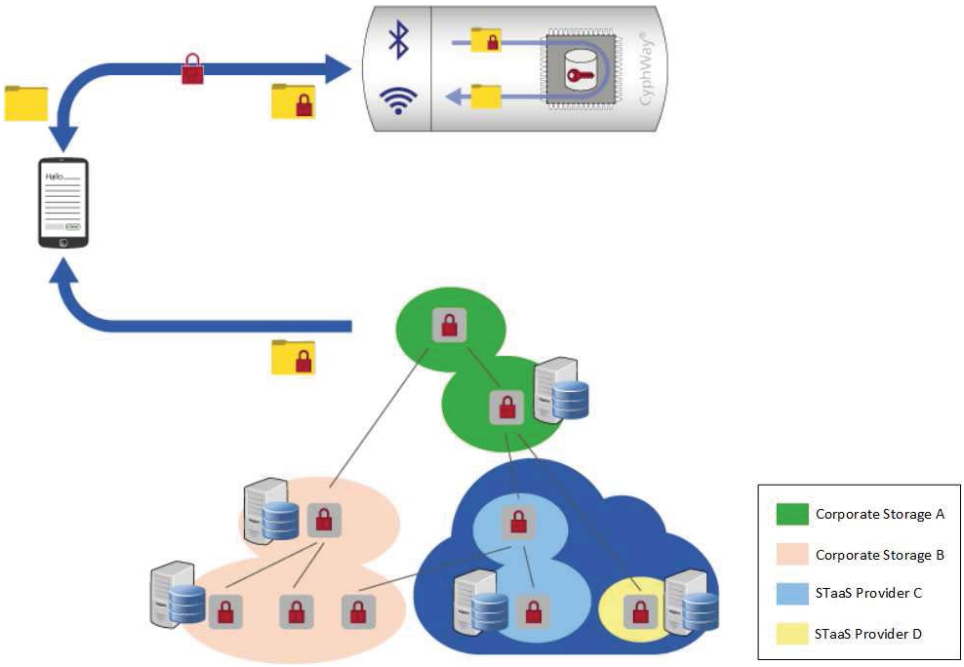


Figure 2: Illustration of the IOSB demonstrator and the distributed file system

Figure 2 illustrates the access to a specific file. The encrypted file is send from the storage to the smartphone and from the smartphone (e.g. via an encrypted Bluetooth channel) to the trusted hardware environment. There, the file is decrypted. In the next step the file is sent (e.g. via an encrypted Bluetooth channel) back to the smartphone, where it can be visualized and accessed.

Within the demonstrator the trusted environment is partitioned into a connector module and a core crypto module (the HSM). Thus, the used Bluetooth channel can be easily replaced against an arbitrary other secured communication channel, e.g. WLAN (which might lead to high energy consumption), or the crypto device can be connected directly to the smartphone or any other device via USB.

No keys will ever be available outside the trusted environment. As a result, even active attacks like Man-in-the-Middle attacks or information gathering Malware are not useful to extract any of the keys, if the underlying encryption scheme is sufficiently secure.

5. Conclusion

In this paper we presented a security overlay that allows the implementation of a distributed virtual encrypted container which supports OTFE. Because of the design of this concept, the overlay can be applied on a variety of underlying platforms, operating systems and file systems. The shown security overlay allows the combination of several storages, like STaaS, corporate datacenters and private clouds. Every data that gets stored within the virtual container gets encrypted, access controlled and is, therefore, protected from Dolev-Yao attackers [2] and even the storage owners. Additionally, the keys and meta-data are protected and access controlled. Utilizing the described demonstrator CyphWay® as a trusted environment makes it possible to protect the keys against every adversary, even against the owner's system. Therefore, we suggest the proposed security overlay for modern distributed storages that are accessed by mobile or other insecure clients. Since storing data on one encrypted partition is not a common use case anymore the presented technique can be used to meet the needs of modern storage strategies. In the future we will work on an implementation of a distributed virtual encrypted container using the overlay to demonstrate the potential of this concept.

References

[1] CloudFuze (2014) [online], https://www.cloudfuze.com/

[2] Dolev, D., Yao, Andrew C., "On the security of public key protocols", in: IEEE Transactions on Information Theory, Vol. 29, Issue 2, IEEE 1983, pp. 198-208

[3] Dongju, Y., Chuan, R. (2014) "VCSS: An Integration Framework for Open Cloud Storage Services", Proceeding of 2014 IEEE World Congress on Services (SERVICES), (pp. 155-160). Anchorage, AK.

[4] Ghemawat, S., Gobioff, H., Leung, S.-T. "The Google File System" Proceedings of the nineteenth ACM symposium on Operating systems principles SOSP '03 (29-43). New York, USA: ACM

[5] Howard, J. et al. (1988) "Scale and performance in a distributed file system". ACM Transactions on Computer Systems (TOCS), Volume 6 Issue 1, Feb. 1988, pp. 51-81.

[6] Jakoby, A., Müller, W., and Vagts, H. "Protecting Sensitive Law Enforcement Agencies Data - Data Security in the Cloud", Proc. of International Conf. on Cyber Warfare and Security (ICCWS 2014).

[7] Machado, G. S., Bocek, T., Ammann, M., Stiller, B. (2013) "A Cloud Storage Overlay to Aggregate Heterogeneous Cloud Services" Proceedings of the 38th Conference on Local Computer Networks (pp. 597 - 605). Sydney, NSW : IEEE.

[8] Pletka, R., Chachin, C. (2007) "Cryptographic Security for a High-Performance Distributed File System" Proceedings of 24th IEEE Conference on Mass Storage Systems and Technologies, 2007, MSST 2007. (S. 227 - 232). San Diego, CA : IEEE.

[9] Tseng, F.-H., Chen, C.-Y., Chou, L.-D., Chao, H.-C. (2012). "Implement A Reliable and Secure Cloud Distributed File System". Proceeding of the 2012 International Symposium on Intelligent Signal Processing and Communications Systems (ISPACS), (pp. 227 - 232). New Taipei : IEEE.

[10] Warner, B., Wilcox-O'Hearn, Z. (2008) "Tahoe – The Least-Authority Filesystem", Proc. of the 4th ACM international workshop on Storage security and survivability (pp. 21-26). Alexandria, VA, USA

A Study of Basic Architecture for Big-Data Security Analysis in SDN Environment

Seong-Ho Choi[1], Jun-Sub Kim[2], and Jin Kwak[3]
[1]ISAA Lab., Department of Information and Computer Engineering, Ajou University, Suwon, Korea
[2]IT Convergence Research Institute, Sungkyunkwan University, Suwon, Korea
[3]Department of Information and Computer Engineering, Ajou University, Suwon, Korea

Abstract - *The first purpose of our project is to find a way to reduce overhead for big-data security analysis in an SDN environment. The second purpose of our project is to support a multi-SDN environment. The SDN is a technology that enables users to control networks by software. Accordingly, the security services would be provided by security software of the SDN controller. However, the process of data analysis for a network security service can generate a large overhead. In addition, big-data security analysis requires more data. This problem such as large overhead and suspension of control systems can arise. Therefore, we need an architecture to reduce the overhead in the controller system. In this study, the architecture is based on a distributed system. It operates on the basis of a virtualization OS. As a result, the architecture uses a disjunct system that consists of control system and data analysis areas. This concept can be developed into a cloud environment, and multiple controllers can be installed and used. A test for the architecture is carried out by a simulation on the basis of the distribution system architecture. We built a distributed system based on KVM, and we was each configure for System of Security, Hadoop, control. As a result, we could reduce the overhead in the control system area. In addition, we could add a new SDN controller.*

Keywords: Distributed System, Virtualization, Cloud, Big-Data Security Analysis, Security Service, SDN

1. Introduction

The number of network devices is growing rapidly. Accordingly, SDN technology is drawing more attention than ever as a means of mitigating problems caused by variable traffic from by diverse types of packet and network environments. SDN is technology that controls the network by using software, it is a concept that separates the control plane from the transport plane. By using this information, software in the control area enables users to configure various networking movements, simplify complex network environment, and manage variable traffic and diverse types of packets efficiently [1-4].

Recently, there is a trend of more security threats as the number of network-connected devices increases. Security threats is evolve through diverse environments. It problem will be limiting the signature based security operation service.

Currently, security operation services detect security threats by filtering the signature. However, detecting the attack from a non-registered signature is difficult. This problem can be resolved with big-data security analysis, which provides functions to collect various service and security event logs generated from the network, and discover symptoms by a correlation analysis of collected data.

In an SDN environment, where users can control a network with software, the security service provided by an existing security device can be configured as software. This characteristic makes it possible to provide a security service such as IDS/IPS, implement a software firewall, form special systems, and execute big-data security analysis. However, overhead interruption can occur during the data analysis for the security service and the big-data security analysis. The overhead causes a system load for the network controller that results in system suspension. Therefore, we must be Study for decrease to overhead.

In this paper, we study a basic Architecture to distributed system for decreased overhead. This paper is organized as follows. In Section 2, we study for distributed SDN Controller, and Big-data Security Analysis. In Section 3, we studied the basic architecture for Big Data security analysis. Section 4 shows the results of the implementation and testing of the architecture. Our conclusions are presented in Section 5.

2. Background Study

2.1. Distributed Security in SDN Controller

An SDN environment is a next-generation network technology in which a network can be controlled by software. A security service provided by existing security devices can be configured as software by this technology and the security policy on a network device connected with a controller is applicable [5-8].

Controllers in an SDN environment can provide security service after collecting and analyzing the packets flowing through the network. However, a large overhead against the controller can occur during the process of data collection and analysis. The solution to this problem is to convert the controller system into a distribution system [9-12]. When the controller system is converted into a distribution system, users can protect the control function from the overhead by distinguishing the core area, which controls the

network directly, from the analysis area. Table I shows the result of the provision of the security service to the network through the controller after simple realization of IDS and the comparison between the share of CPU and that of memory.

The test is executed with the normal application model applied by a distribution environment, which is configured with a simple IDS module from the controller system [13-16]. Consequently, a high share has been recorded from the IDS, which operates in the single controller system and a significantly low share has been shown when operated as a system in a distributed environment. In conclusion, it is more efficient to provide service within the distributed environment to supply security service in the SDN environment.

TABLE I. RESULTS OF PERFORMANCE ANALYSIS

Attack name	Switch	HOST	Single Controller	Distributed Controller
			CPU Shared	CPU Shared
SYN Flooding	1	2	1.7%	1.0%
	5	2	3.0%	1.4%
	10	2	6.1%	2.3%

2.2. Big-Data Security Analysis

Big-data security analysis is a technology that can overcome the limits of existing network security analysis. Various solutions including SIEM are in use and are considered important in overcoming the flows of existing network security technology [17-20]. A summary of big-data security analyses is shown in Figure 1.

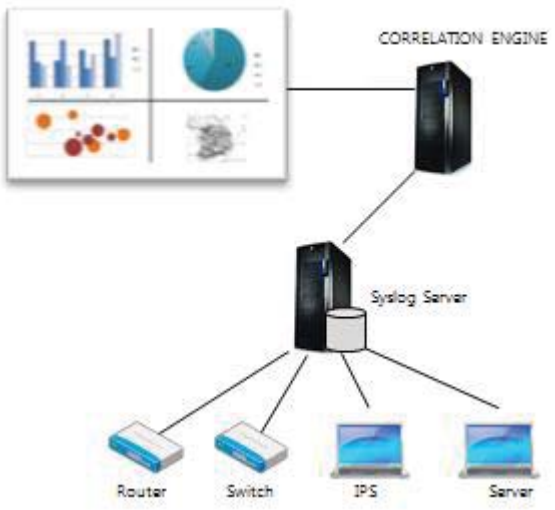


Fig. 1. Big-Data Security Analysis System

The big-data security analysis model detects symptoms through the collection of network control and security devices, log data such as the server, and then through a correlation analysis engine. This provides the functions to discover symptoms from an undetected cyber-attack in the security equipment. Therefore, the big-data security analysis platform will be an essential factor in future security monitoring and control systems.

3. Architecture of the Prototype

This paper discusses how to implement the big-data security analysis in an SDN environment. The architecture can execute big-data security analysis and the security service for the device connected with the network through the controller.

The controller is essentially an important system that controls the network. Therefore, the overhead in the controller should not increase during the security analysis process. The previous study reached a conclusion, after realizing the simple IDS in the controller, that the CPU and memory use increase rapidly when analyzing in the single controller. Problems such as system interruption can occur if an analysis system is applied that deals with a large amount of security data. Therefore, a distributed system is needed to minimize the negative effect on the controller system when analyzing data for security. This study hereafter focuses on the core controller system, which controls the network by configured controller system in a distributed environment, and on decreasing the overhead generated from it. The architecture of the prototype is as follows.

3.1. Architecture of Prototype

In this research, the controller system is configured with a distributed environment by a simple method derived from advanced research. The architecture is composed of a network control area, and the normal security service area, such as the IPS/IDS, firewall, and big-data security analysis area. See Fig. 2 for the architecture.

An explanation for each system area follows.

■Control area

This system is a core function of the controller and it provides network device management, host management, and various network services. It can be considered as a normal controller environment that controls connected network resources.

■Security service area

The security service area forms a system by using the RPC protocol to minimize the effect on the core controller. The security service, which is executed in the form of an application, records the security events in a log after analyzing packets for the network.

■Big-data security analysis area

The big-data security analysis area conducts correlation analysis with the collected data generated from the core controller system and the security event log information recorded in the distributed application system area. This area is managed separately from the controller because it is an environment that can be applied in the network environment, where single controllers and controllers connected with multiple domains control the network. The controllers connected with multiple domains may need to share the outcomes with the big-data security analysis system area. This architecture provide multi SDN environment. Just we will the connected other Controller with This architecture.

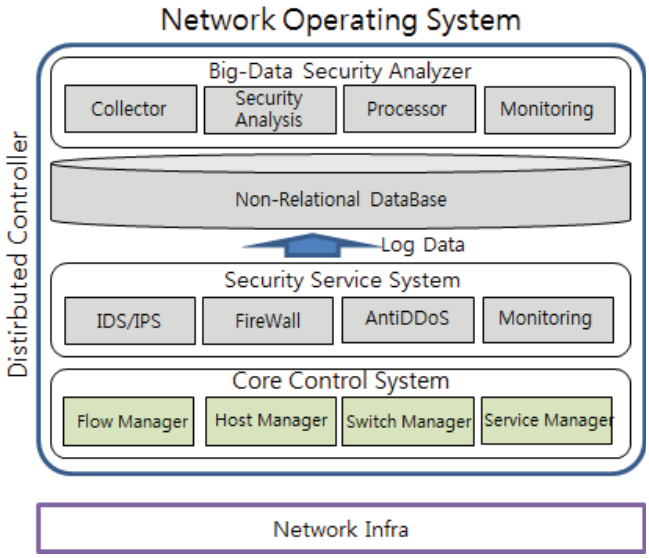


Fig. 2. Architecture of Prototype

3.2. Multi-SDN Environment

A multi-SDN can provide an environment with which it is possible to execute big-data security analysis by using the prototype architecture. It can also collect data needed for the big-data security analysis by gathering a log for each network by using the controllers in a multi-domain environment. It helps to increase the accuracy of the anticipation of possible attacks by collecting data to detect symptoms more efficiently. This means that users can transform the big-data security analysis system into a server and form it as a single data center in the prototype architecture. The controller can remotely connect from the OFswitch for network control. Another approach is that the controller system may be connected to a remote server with big-data security analysis. However, we need more research for effective measures.

See Fig. 3 for the implementation of big-data security in the environment composed of a multi-SDN controller.

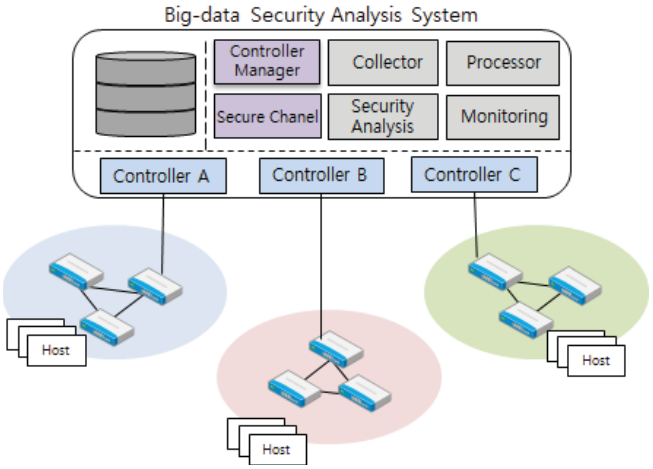


Fig. 3. Multi SDN Environment

4. Implementation

This section describes tests carried out on the basis of the architecture introduced in Section 3. It presumes that the security service operates well and that log data for various network services from the controller system can be collected; it also illustrates the process of detecting symptoms by transmitting data generated from the SDN controller to the big-data security analysis server. An overload test for the controller system and the analysis of CPU and memory share are included in the process.

An SDN environment forms the status in which an SDN controller-based network environment and interhost correspondence are available. It shows the procedure of a SYN flooding attack from host A to host B and collects event log data about the attack. It also generates massive data to show the process of producing log data in the controller and functions as a data transmitter to the security analysis server. The big-data security analysis server collects security event logs and massive files and is saved as a log file in the non-relational database. Big-data security analysis is a condition precedent.

4.1. TEST

The test realizes each system as a virtual environment. We realize each system in the architecture on virtualization. We used KVM technology for this environment [21-24]. The core control system and security analyzer system are configured in VM instances on the Hadoop file system for big-data analysis. In addition, we configure Mininet from the Hadoop system, and mininet is connecting the core control system of VM instance environment from Hadoop. We used network bridge technology in this architecture. Next, we obtained the result of the SYN flooding attack and log file for the network service from the Hadoop file system. In addition, we checked the CPU and memory share. The realized function is for checking the effect of the overhead from the controller system.

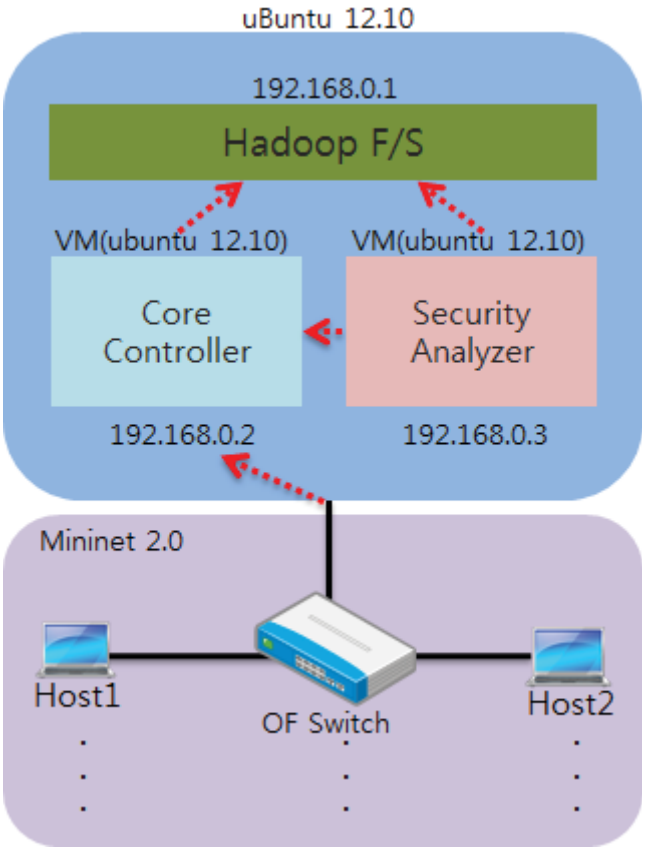


Fig. 4. Test Environment

See Fig. 4 for the network structure. This Result is CPU and memory usage from the core control system.

After setting up the test environment as seen in Fig 4, we generated a large amount of log data as more network devices were connected to the controller, and then proceeded to test the CPU and memory share while sending the log data to the big-data security analysis server. The comparison target is the result of CPU and memory, which provide a single system security service. See Table II for the test results. This result is CPU and memory usage from the core control system.

TABLE II. RESULT OF PERFORMANCE

Switch	Host	Single Controller System		Distributed Controller System	
		CPU	Mem	CPU	Mem
Switch-1	2	2.0%	8.0%	4.0%	9.2%
	4	2.6%	7.8%	4.5%	9.3%
	8	4.2%	7.7%	4.7%	9.3%
Switch-5	1	3.0%	9.9%	4.8%	10.1%
	4	5.3%	10.2%	5.0%	10.1%
	8	6.7%	10.3%	5.2%	10.5%
Switch-10	1	7.5%	10.8%	6.1%	10.7%
	4	8.0%	11.2%	6.3%	10.7%
	8	14.3%	12.3%	6.5%	10.8%

Figs. 5 and 6 graphically depict the change in usage.

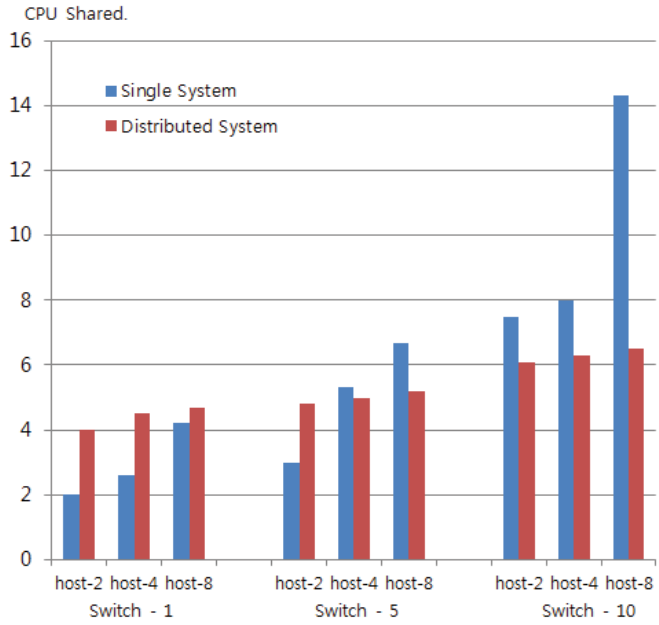


Fig. 5. Cpu Usage

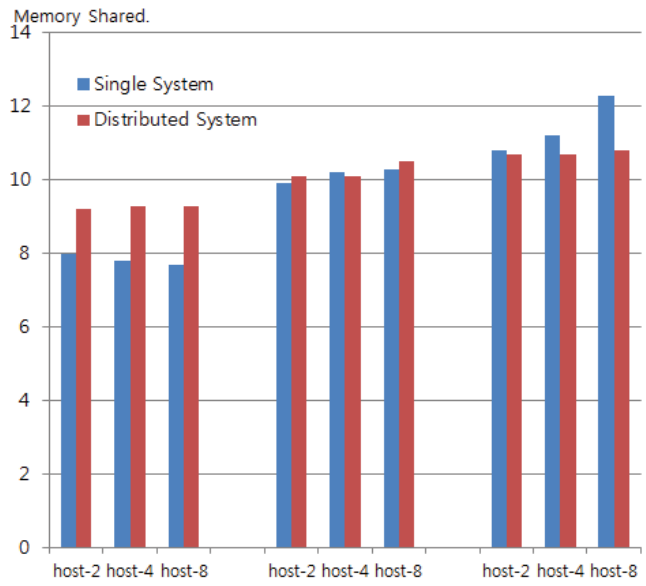


Fig. 6. Memory Usage

We were observe that when the Data analysis function is executed. Single System is CPU usage amount increase that if the number of switches and hosts increases, or if the

Submit a lot of data. But, Distributed System is CPU usage the little increases. We reached the conclusion that forming an individual service system with a distributed system and running it helped to reduce the overhead.

5. Conclusion

In this paper, we carried out a study of an architecture for implementing big-data security analysis in an SDN environment. We studied the architecture for the distributed system environment. In addition, we studied a multi-SDN environment for services provided by the big-data security analysis system. We formulated the control system as a distributed system to reduce the overhead generated from the big-data security analysis, and showed the possibility through simulation testing. Security service and big-data security analysis functions were not formulated in the simulation test. Nonetheless, we succeeded in reducing the overhead from the control system by handling the overhead with the distributed system. Even though the individual security service and big-data security analysis system were formed, possible problems were removed by minimizing the overhead on the control area system. However, the security service system and the big-data security analysis system should be formed with a high quality system that is dependent on the number of connected network devices.

This paper, in its initial procedure, would not provide significant help to the industry, but it suggests a method for reducing big-data analysis overhead. We hope that further study on this subject will enable a stable big-data security analysis through the expansion and verification of the architecture.

6. Acknowledgment

This work was supported by the ICT R&D program of MSIP/IITP, Republic of Korea. [13-912-06-003, Development of Mobile S/W Security Testing Tools for Detecting New Vulnerabilities of Android]

7. References

[1] Yu, J. H., Kim, W. S. and Yun C. H., "A Technical Trend and Prospect of Software Defined Network and OpenFlow," KNOM Review, 4.2014, 1-22.

[2] Lantz, Bob, Brandon Heller, and Nick McKeown, "A network in a laptop: rapid prototyping for software-defined networks," Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks. ACM, 2010.

[3] Handigol, Nikhil, et al., "Where is the debugger for my software-defined network?," Proceedings of the first workshop on Hot topics in software defined networks. ACM, 2012.

[4] ONF, "Software-Defined Networking: The New Norm for Networks," ONF White Paper, 4 (2012), 3-12.

[5] Kim, H. and Feamster, N., "Improving Network Management with Software Defined Networking," Communications Magazine, IEEE, 51 (2013), 114-119.

[6] Fernando, N. N. Farias, Joao J. Salvatti, Eduardo Cerqueira, and Antonio Jorge Gomes Abelem, "Management of the Existing Network Environment Using Openflow Control Plane," IEEE NOMS, 2012, 1143-1150.

[7] Oueslati, S. and Roberts, J., "A New Direction for Quality of Service:Flow-aware Networking," In Proc. NGI, 2014, 226-232.

[8] Hata, H., "A Study of Requirements for SDN Switch Platform," ISPACS 2013, 2013, 79-84.

[9] Dixit, Advait, et al., "Towards an elastic distributed SDN controller," ACM SIGCOMM Computer Communication Review. Vol. 43. No. 4. ACM, 2013.

[10] Schmid, Stefan, and Jukka Suomela, "Exploiting locality in distributed sdn control," Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking. ACM, 2013.

[11] Berde, Pankaj, et al., "ONOS: towards an open, distributed SDN OS." Proceedings of the third workshop on Hot topics in software defined networking. ACM, 2014

[12] Phemius, Kévin, Mathieu Bouet, and Jérémie Leguay. "Disco: Distributed multi-domain sdn controllers," Network Operations and Management Symposium (NOMS), 2014 IEEE. IEEE, 2014.

[13] Gupta, P., "SS-IDS: Statistical Signature Based IDS, " ICIW '09, 2009, 407-412.

[14] Ozgur Depren, Murat Topallar, Emin Anarim, M. Kemal Ciliz, "An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks," Expert Systems with Applications, November,2005, 713–722

[15] S. Staniford-Chen, S. Cheung, R. Crawford, M. Dilger, J. Frank, J. Hoagland, K. Levitt, C. Wee, R. Yip, D. Zerkle, Gr., "IDS-A GRAPH BASED INTRUSION DETECTION SYSTEM FOR LARGE NETWORKS," NISSC '96, 1996, 407-412.

[16] Weijian Huang, YanAn, Wei Du, "A Multi-Agent-Based Distributed Intrusion Detection System," ICACTE,2010 3rd International Conference on, 2010, 141-143.

[17] Wang, Guohui, T. S. Ng, and Anees Shaikh, "Programming your network at run-time for big data applications," Proceedings of the first workshop on Hot topics in software defined networks. ACM, 2012.

[18] Qin, Peng, et al., "Bandwidth-Aware Scheduling with SDN in Hadoop: A New Trend for Big Data," arXiv preprint arXiv: 1403.2800 (2014).

[19] Tankard, Colin. "Big data security." Network security 2012.7 (2012): 5-8.

[20] Bhatti, Rafae, et al., "Emerging trends around big data analytics and security: panel." Proceedings of the 17th ACM symposium on Access Control Models and Technologies. ACM, 2012.

[21] Jain, Raj, and Subharthi Paul., "Network virtualization and software defined networking for cloud computing: a survey," Communications Magazine, IEEE51.11 (2013): 24-31.

[22] Lin, Pingping, Jun Bi, and Hongyu Hu., "VCP: A virtualization cloud platform for SDN intra-domain production network," Network Protocols (ICNP), 2012 20th IEEE International Conference on. IEEE, 2012.

[23] Habib, Irfan. "Virtualization with kvm." Linux Journal 2008.166 (2008): 8.

[24] Kivity, Avi, et al., "kvm: the Linux virtual machine monitor," Proceedings of the Linux Symposium. Vol. 1. 2007.

SESSION

NETWORK SECURITY I

Chair(s)

Dr. Xinli Wang
Dr. Rob Byrd

Abnormal VoLTE Call Setup between UEs

Sekwon Kim, Bonmin Koo, and Hwankuk Kim
Mobile Security R&D Team, Korea Internet & Security Agency, Seoul, Korea

Abstract - *As the mobile environment has been rapidly changing recently due to advances in mobile communication technology, mobile traffic has been sharply increasing around the world. To respond to the increasing traffic, Korea's mobile carriers have been trying to build out their 4G networks early on rather than upgrading their existing 3G networks. However, due to the early build out of the LTE network and competition to improve it, network security was not sufficiently taken into consideration. Also, as the LTE network provides both data and VoLTE services on the All-IP-based network, it is exposed to the same types of security threats likely to occur on IP-based networks, such as forgery, alteration of information, and eavesdropping. This paper analyzes a particular security threat which is the vulnerability to hacking of call setup between terminals using VoLTE service in Korea, and proposes a counter technology.*

Keywords: LTE; VoLTE; Threat; Abnormal Call Setup.

1 Introduction

Recently the mobile environment has been changing rapidly due to advances in mobile communication technology. High-performance smartphones and personal tablets have become very popular, and as various mobile services have increased, anyone can now use high-speed mobile communication networks. Also, as an increasing number of customers, who used to be satisfied with downloadable-type contents only, are now using on-demand or streaming contents, mobile traffic is sharply increasing around the globe[1].

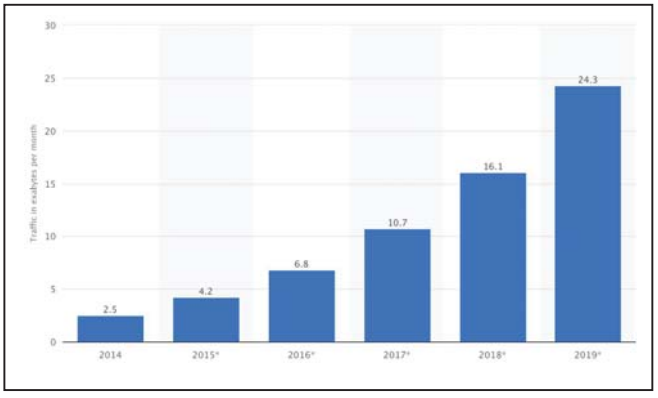


Fig. 1. Global mobile data traffic from 2014 to 2019 (in exabytes per month).

To respond to the increasing traffic, Korea's mobile carriers decided to install and build out 4G networks early instead of upgrading their existing 3G networks. As a result, LTE service began in Korea in 2011, and as of now Korea is the most advanced country in the world in terms of the LTE market and technology, i.e. Korea has become a global reference country for LTE.

However, in their scramble to publicize their technology and gain subscribers as quickly as possible they completed their LTE networks earlier than scheduled and launched services, and network security was not sufficiently taken into consideration due to the competition for network enhancement such as introduction of the LTE-A technology. Also, as the LTE network provides data and voice services on the All-IP-based network, it is exposed to security threats likely to occur on IP-based networks, such as forgery and alteration of information, and eavesdropping. In particular, if the SIP control messages for VoLTE service are forged or altered, then the result could be that voice call tolls could be used for crimes like voice phishing[2][3].

This paper will analyze the security threat to abnormal call setup between terminals using the VoLTE service, and propose a counter technology. This paper is organized as follows. Chapter 2 describes the LTE network, GTP protocol, IMS network and SIP Protocol. Chapter 3 analyzes the security threat to abnormal VoLTE call setup, and Chapter 4 proposes a counter technology. Lastly, Chapter 5 brings this paper to conclusion.

2 Background Information

2.1 LTE Network

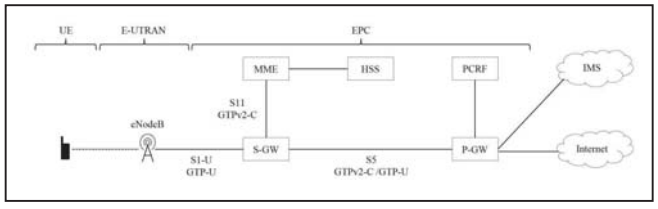


Fig. 2. LTE Network Structure.

LTE is a network infrastructure designed to provide all types of telecommunication services including voice calls, video calls, SMS and various mobile multimedia services, such as wireless Internet, to mobile terminals. As illustrated in Figure 2, it consists of an Access Network (E-UTRAN) that manages terminals and wireless resources, and a Core Network (EPC) that handles data transmission, authentication and billing.

The E-UTRAN, which provides the mobile communication environment, exists between the EPC and terminals. eNodeB allocates mobile resources to terminals, and manages them with certain coverages in each region.

LTE EPC consists of several key pieces of equipment. The MME, the S-GW and the P-GW each play important roles for providing data services, e.g. the mobile Internet. The MME authenticates the UE and manages the bearer. The S-GW is the terminal point of the E-UTRAN and the EPC. The P-GW is in charge of allocating terminal IP addresses and IP routing/ forwarding. In addition, there is the HSS that serves as the subscriber information DB, and the PCRF that determines the service quality policy for each subscriber[4].

2.2 GTP (GPRS Tunneling Protocol)

The GTP is the tunneling protocol for delivering the data sent by the UE on the LTE network. Equipment like the eNodeB, the MME, the S-GW and the P-GW use the GTP to create GTP tunnels for delivering data from equipment to equipment and communicate. The GTP is divided into the GTP-C for control (Create, Delete, Modify/Release) of GTP tunnels and the GTP-U for user IP packet transmission[5][6].

Figure 3 shows the GTPv2-C header used in the LTE network. Here, the Tunnel Endpoint Identifier (TEID) is a unique factor used to distinguish the GTP tunnels for individual UEs on the LTE network. For example, If 100 UEs are connected to the same S-GW and P-GW, one or more GTP tunnels will be created for each UE and more than 100 GTP tunnels will be created in total, with each GTP tunnel being identified with the TEID. And, the message type is a factor for distinguishing the GTP-C. Key message types are shown in Table 1[5].

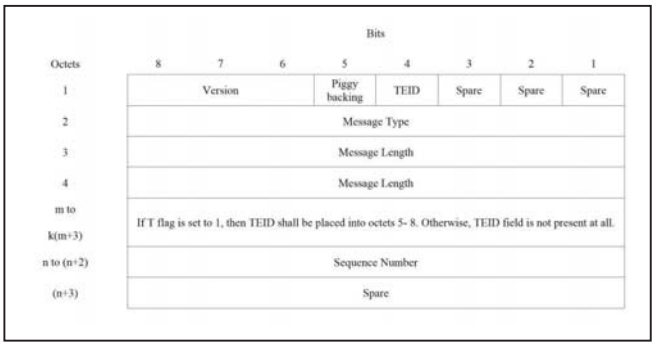


Fig. 3. General format of GTPv2 Header for Control Plane.

TABLE I. MESSAGE TYPES FOR GTPV2

Message Type	Message	Description
32	Create Session Request	Creates GTP tunnels
33	Create Session Response	
34	Modify Bearer Request	Modifies GTP tunnels
35	Modify Bearer Response	
36	Delete Session Request	Deletes GTP tunnels
37	Delete Session Response	

2.3 IMS (IP Multimedia Subsystem)

The LTE network is an All-IP-based network. Unlike the 3G network, it does not have a separate voice network, but rather LTE interworks with the IMS network to provide VoLTE, the voice service. As VoLTE supports the 50~7000Hz bandwidth, which is much wider than the 3G voice call bandwidth, clear high-quality voice calls are possible. Also, it is possible to switch to a video call in the middle of a voice call and easily share photographs, images and location information by interfacing with various data services. As VoLTE exchanges voices through the IP-based data network it is similar to VoIP technology, yet stable high-quality call service is possible thanks to separate quality management when data gets congested[7].

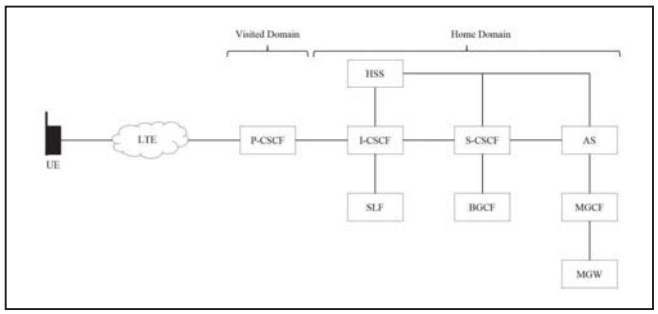


Fig. 4. IMS Network Structure.

Figure 4 illustrates the IMS network structure. VoLTE service is provided through the Call Session Control Function (CSCF) that handles the call and session control in the IMS network. The CSCF consists of equipment for processing the calls and sessions of IP-based multimedia services. It manages the registered information of VoLTE terminals, connects calls, and relays voice call origination and termination data. The CSCF can be divided into the P(Proxy)-CSCF, I(Inter-rogating)-CSCF, and the S(Serving)-CSCF depending on the function being referred to. The P-CSCF is the first point that the UE encounters when connecting to the IMS for the time. It serves as the proxy or user agent. The I-CSCF serves as the contact point for all incoming calls for connecting to subscribers in the network, queries the HSS to determine the S-CSCF, and allocates the S-CSCF to the UE in the registration process. Lastly, the S-CSCF performs key functions for call processing, and is responsible for all functions related to providing services like interfacing the service platform and providing service-related information. The AS is the service platform for providing service, the SLF provides HSS addresses to the CSCF. In addition, the BGCF, the MGCF and the MGW provide such functions as protocol and signaling conversion for interworking with other voice networks such as the PSTN[8].

2.4 SIP (Session Initiation Protocol)

VoLTE, the voice service through IMS network, uses SIP text-based signaling protocol the same as VoIP does to provide voice service over the Internet. The SIP is used to

control voice calls, e.g. voice call origination, termination and end-of-call, and is divided into the header and the body. The SIP header includes the Call-ID unique to each call and originating/ terminating MSISDN as well as the method field that defines the SIP message type. The body includes the media codec used for voice and video calls and information on IP and Port for sending and receiving the RTP Voice traffic. The key SIP method and its uses are shown in Table 2 below[9][10][11].

TABLE II. SIP METHOD AND USES

Method	Description
REGISTER	Registers the address listed in the To header field with a SIP server
INVITE	Indicates that a client is being invited to participate in a call session
SUBSCRIBE	Subscribes to event notification
NOTIFY	Notifies the subscriber of a new Event
REFER	Asks recipient to issue a SIP request (call transfer)

Figure 5 illustrates the procedures for VoLTE service. The UE registers itself with the CSCF, and uses the VoLTE service through the Call Setup process

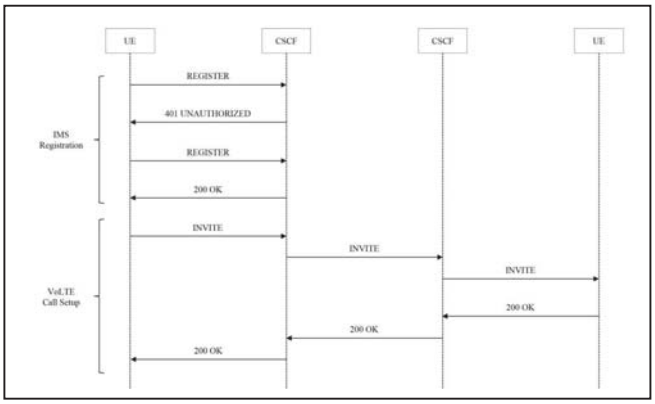


Fig. 5. IMS Procedures for VoLTE.

3 Abnormal VoLTE Call Setup

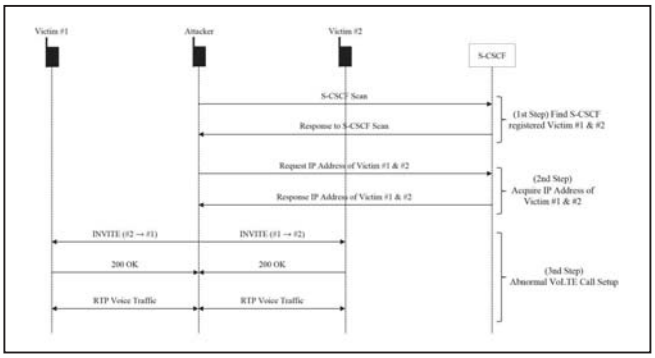


Fig. 6. Procedures for Abnormal VoLTE Call Setup.

The SIP is a text-based protocol that can be forged and altered easily. This chapter deals with the security threats that can connect phone calls abnormally between VoLTE users by altering the MSISDN in the SIP Header and the IP and port information for sending and receiving the RTP Voice traffic in the Body. Figure 6 shows the procedure.

For Abnormal VoLTE Call Setup, the attacker searches for the S-CSCF registered target UEs (Victim #1 & #2) by S-CSCF scanning. The attacker then obtains the IP addresses of victims from the S-CSCF. Finally, the attacker, disguised as the CSCF, sends each the altered SIP Invite to victims, and if victims terminate the Call, an abnormal call is set up between victims. As the RTP voice traffic between victims passes through the attacker at this time, eavesdropping is possible. The details are as follows:

3.1 Find S-CSCF Registered Victim

If the UE turns on the VoLTE, the IMS registration process will be carried out. At this time, the I-CSCF queries the HSS to determine the S-CSCF and then allocates the S-CSCF to the UE.

Fig. 7. Find IP Address of S-CSCF in SIP 200 OK packet.

The attacker can check the IP address of the S-CSCF allocated to his/her UE (x.x.227.129) in the SIP 200 OK packet during the VoLTE registration process as illustrated in Figure 7, and guess the IP address band of the S-CSCF on this.

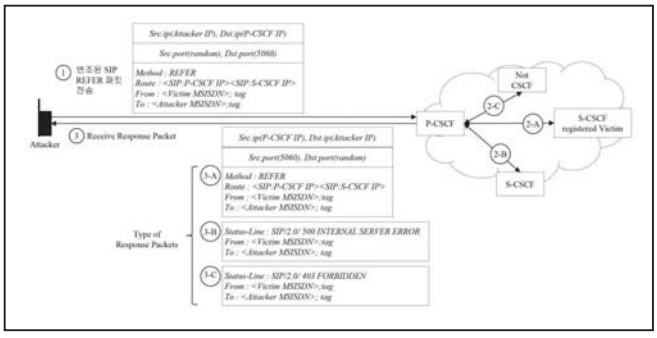


Fig. 8. Procedures for Finding S-CSCF Registered Victim by S-CSCF Scanning.

Figure 8 shows how the SIP REFER can be used to scan the IP address band of the S-CSCF and check the IP address of the S-CSCF which victim is registered by analyzing the response packets. The attacker sends SIP REFER packets as shown in Figure 9 in which the From field (Caller's MSISDN) and P-Preferred-Identity field were altered to the Victim's

MSISDN (800), the To field (Callee's MSISDN) and Refer-To field were altered to the attacker's MSISDN (203), and the Route field was altered to be the IP address of the S-CSCF (x.x.227.2~254) to the P-CSCF. The P-CSCF receives the packet sent by the attacker then forwards it to the IP address of the Route field (the IP address of the S-CSCF altered by the attacker).

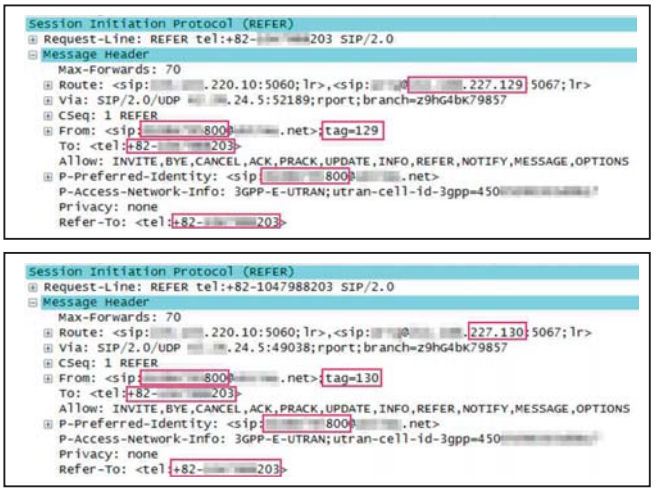


Fig. 9. Example of Altered SIP REFER Packets.

The attacker received three types of response packets as shown in Figure 10. "500 INTERNAL SERVER ERROR" means that the server that sent the response packet is not the CSCF, "403 FORBIDDEN" means that the CSCF has no registered victims, and "REFER" means that the S-CSCF has registered victims.

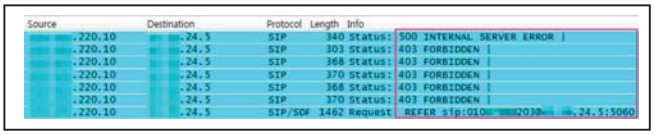


Fig. 10. Response Packets to S-CSCF Scanning.

Here, as the destination IP of the scanning traffic sent by the attacker is a P-CSCF IP, the Source IP of the response packet is also a P-CSCF IP. In other words, the attacker cannot use the Source IP of the response packet to check the S-CSCF IP with victims registered. The attacker can use the tag value of the From field in the received REFER packet to check the S-CSCF IP address with victims registered. Among the packets sent by the attacker, the S-CSCF IP address of the packet whose From field tag value matches the From field tag value (129) in the received REFER packet is the S-CSCF IP (x.x.227.129) with victims registered.

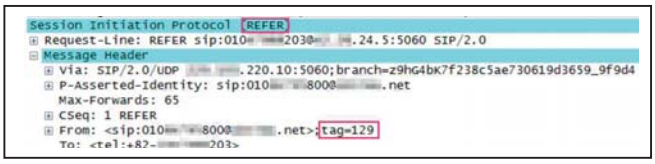


Fig. 11. Response Packet Details.

3.2 Acquiring IP Addresses of Victim

The SIP SUBSCRIBE message is used for requesting the CSCF for the status of VoLTE registered terminals. The CSCF sends the SIP NOTIFY message containing the registration status, including the IP address, in response to the SUBSCRIBE message. The attacker can obtain the IP address of the victim by transmitting the SUBSCRIBE message with an altered MSISDN to the S-CSCF with victims registered as illustrated in Figure 12.

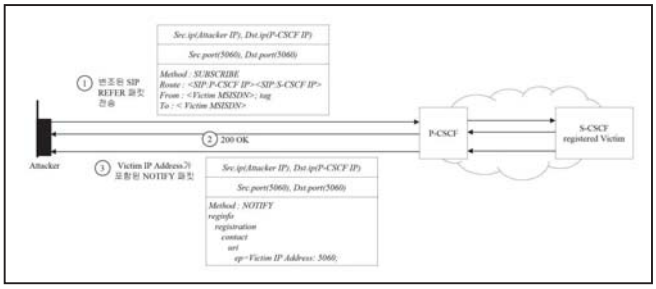


Fig. 12. Procedures for Acquiring IP Addresses of Victim.

The attacker sends the SIP SUBSCRIBE packet as shown in Figure 13, in which the Route field was altered to the IP address of the S-CSCF with victims registered (x.x.227.130), and the From field and To field were altered to the MSISDN of the victim (223).

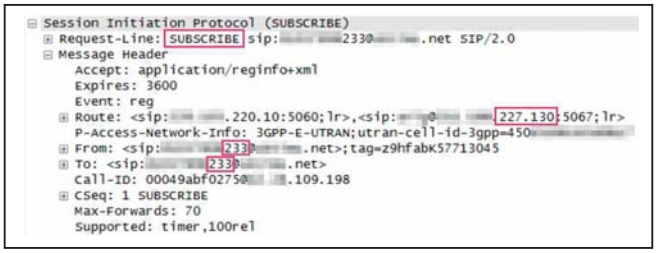


Fig. 13. Example of Altered SIP SUBSCRIBE Packet.

The P-CSCF receives the packet sent by the attacker and forwards it to the S-CSCF IP address in the Route field. The S-CSCF then transmits 200 OK and NOTIFY to the attacker in response.

Fig. 14. SIP NOTIFY Packet Included Victim's IP Address.

The attacker can obtain the victim's IP address (x.x.135.169), included in the SIP NOTIFY packet, from the response packet as shown in Figure 14. This IP address matches the IP address

for the IMS, which is verified through the Network Info app installed in the victim's UE as illustrated in Figure 15.

Fig. 15. Victim's IP Address.

3.3 Abnormal VoLTE Call Setup

UEs using the VoLTE service receive the S-CSCF in the registration process, and if they terminate any VoLTE Calls, the UEs will receive SIP INVITE packets from the allocated S-CSCF server. At this time, however, the UEs do not test the integrity of the S-CSCF. That is, they do not check whether the S-CSCF, which transmitted SIP INVITE to them, matches the S-CSCF allocated to them in the registration process, and simply receive SIP INVITE unconditionally and process it. The attacker abuses this, and as shown in Figure 16, the attacker can eavesdrop on the RTP voice traffic by setting up abnormal calls between the two victims.

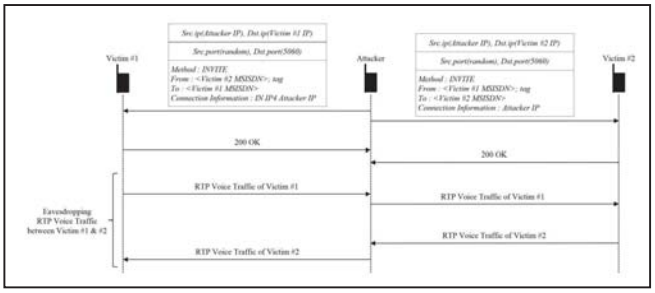


Fig. 16. Procedures for Abnormal VoLTE Call Setup between Victims.

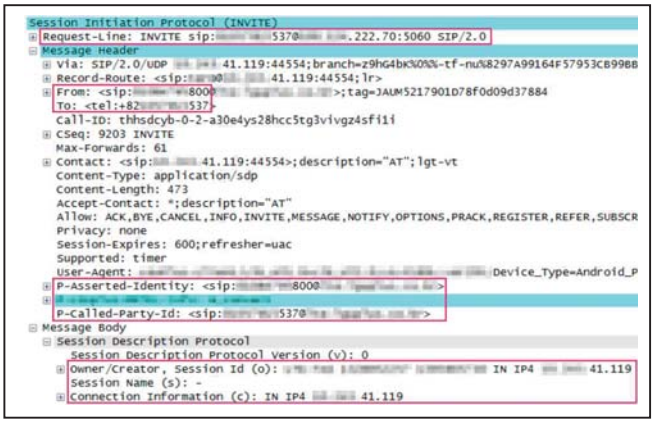


Fig. 17. Example of Altered SIP INVITE Packet.

The attacker sends the SIP INVITE packet as shown in Figure 17, in which Caller's Information (From field, P-Asserted-Identity field, etc.) was altered to Victim #2's MSISDN, Callee's Information (To field, P-Called-Party-ID field, etc.) was altered to Victim #1' MSISDN, and the IP address for receiving the RTP Voice Traffic was altered to the attacker's IP address (x.x.41.119), to Victim #1. The attacker sends the SIP INVITE packet, in which the Caller and Callee information is switched, to Victim #2.

The victim who received the altered SIP INVITE packet sent by the attacker will see a screen that says the VoLTE Call request was sent by the other victim, not the attacker, and if both victim #1 & #2 terminate the Call, the abnormal call setup will be completed. At this time, as the IP address for the RTP voice traffic in the SDP of the SIP INVITE packet is set up as the attacker's IP address, the RTP voice traffic between victims will pass through the attacker as illustrated in Figure 18. The attacker can demodulate it and eavesdrop on the call between victims.

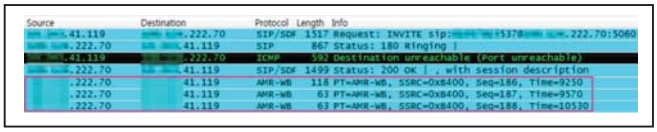


Fig. 18. RTP Voice Traffic between Victims Passed through The Attacker.

4 Counter Technology

The SIP is a text-based protocol that is easy to forge and alter. Chapter 3 described security threats whereby phone calls between VoLTE users can be hacked by altering the MSISDN in the SIP Header, the IP for sending and receiving the RTP voice traffic in the Body, and Port information.

The SIP standard recommends using TLS or IPSec for security and S/MIME for message integrity and confidentiality[12][13]. Actually, T-Mobile of the US uses TLS and Japan's NTT Docomo uses IPSec for communication between the UE and the CSCF to encrypt data in response to security threats. Also, the SIP-based VoIP system uses the SIP Digest Authentication function based on HTTP Authentication to authenticate all SIP Request messages[14]. As these encryption and authentication mechanisms slow down VoLTE service, however, they may cause some degree of dissatisfaction among LTE service subscribers who want and expect fast service.

It is possible to respond to security threats due to forged and altered SIP messages by adding security functions to the CSCF that controls calls and sessions in the IMS network. In other words, the IP addresses and MSISDN that the CSCF allocated to UEs will be managed separately, and thus make it possible to analyze whether the MSISDN is altered for all SIP Request messages and block fraudulent ones. However, as mobile communication networks provide "Always on" service, it is difficult to add functions without shutting down equipment, and service failures may result due to unexpected errors and equipment malfunction in the process of adding the functions. And functions added to equipment will inevitably increase the load on existing equipment. Increased load will eventually deteriorate availability and the introduction of additional CSCF may lead to increased costs.

This chapter proposes a technology for detecting SIP Request messages with forged and altered originator information by managing the UEs in the LTE EPC.

4.1 TEID-based UE Session Management

The S11 (MME↔S-GW) interface of the LTE EPC collects the GTP-C for creating, deleting and modifying GTP tunnels, analyzes it, and manages the session table of the TEID-based UEs. The management method consists of two stages: (1) pairing GTP-C Requests and Responses and (2) processing the GTP-C to manage the session table.

In the first stage, GTP-C Requests and Responses are paired through the buffer. This stage will check whether Requests and Responses are normally exchanged. Here, the buffer key is the combination of the MME IP and the Sequence Number included in the GTP-C. Figure 19 shows the procedure in detail.

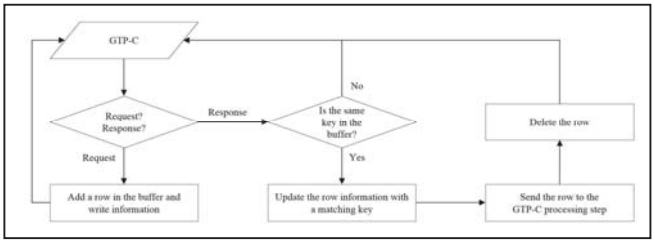


Fig. 19. Procedures for Pairing GTP-C Requests and Responses.

1) Receive the GTP-C Create Session, Modify Bearer and Delete Session.

2) If it is a Request, add a new row in the buffer, and write GTP-C information.

3) If it is a Response and the same key exists in the buffer, update the information in the row which matches the key, then transmit the information of the row to the second stage and delete the row.

4) If it is a Response and the same key does not exist in the buffer, receive the next GTP-C.

Figure 20 shows the changes of the buffer due to the creation, modification and deletion of GTP tunnel of a UE.

	(Key) MME IP + Sequence Number	Timestamp	Message Type	EBI	S11 S-GW GTP-C TEID	S1-U S-GW GTP-U TEID	MSISDN
(1) Create Session Request	112976012	1308010728018344	Create Session	5			
(2) Create Session Response	112976012	1308010728018597	Create Session	5	52297760	72406262	820000000203
(3) Modify Bearer Request	113185261	1308010728024719	Modify Bearer	5	52297760		
(4) Modify Bearer Response	113185261	1308010728024931	Modify Bearer	5	52297760	94628484	
(5) Delete Session Request	119158192	1308010730146153	Delete Session	5	52297760		
(6) Delete Session Response	119158192	1308010730146375	Delete Session	5	52297760		

Fig. 20. The Changes of The Buffer Due to The Creation, Modification and Deletion of GTP Tunnel.

The second stage processes the GTP-C when the GTP-C Request and Response were paired in the first stage, and manages the session table. The tables for managing sessions consist of the UC table for managing the control tunnels of the UE, and the UD table for managing data tunnels and detecting SIP packets with altered origination information. Here, the UC Table Key is a combination of the S11 SGW GTP-C TEID and EBI (EPS Bearer ID), and the UD Table Key is the S1-U SGW GTP-U TEID. Figure 21 shows the procedure in detail.

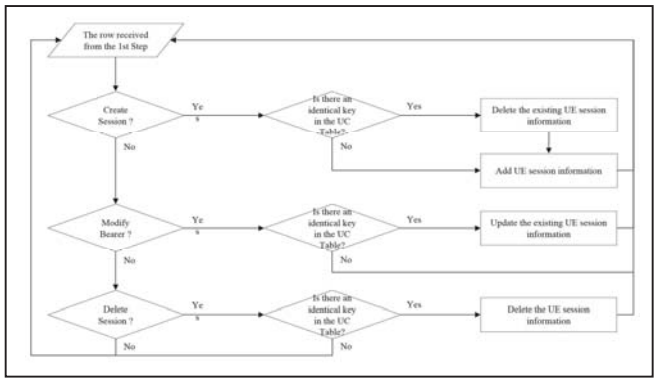


Fig. 21. Procedures for Processing The GTP-C to Manage The Session Table.

1) Receive the information on the GTP-C whose Requests and Responses were paired in the first stage.

2) If the Message Type is Create Session and the same key exists in the UC Table, delete the matching rows in the UC and UD Tables, and write the received Session information in the UC and UD Table.

3) If the Message Type is Create Session, and the same key does not exist in the UC Table, write the received GTP-C Create Session information in the UC and UD Tables.

4) If the Message Type is Modify Bearer, and the same key exists in the UC Table, update the received GTP-C Modify Bearer information in the UC and UD Tables with matching keys.

5) If the Message Type is Delete Session, and the same key exists in the UC Table, delete the rows in the UC and UD Tables with the matching keys.

6) After the above process is completed, receive the following GTP-C information from the first stage.

Figure 22 illustrates the changes in the UC and UD tables due to the creation, modification and deletion of the GTP tunnel of a UE.

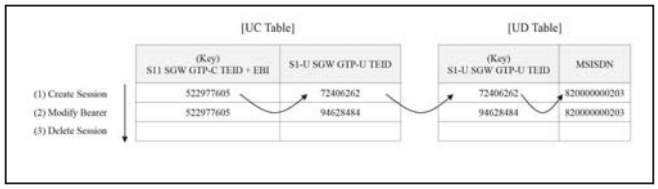


Fig. 22. The Changes in the UC and UD Tables due to The Creation, Modification and Deletion of The GTP Tunnel.

4.2 Detecting Abnormal SIP Packet

The GTP-U will be collected from the S1-U (eNodeB↔ S-GW) interface of the LTE EPC, and the MSISDN in the SIP will be compared with the value in the UD Table to detect

abnormal SIP packets with altered MSISDN. Figure 23 shows the procedure in detail.

1) Receive the GTP-U whose user packet payload is the SIP, and extract the TEID from the GTP Header and MSISDN information from the SIP Header.

2) Use the TEID to query the UD Table, and extract the value (MSISDN) from matching rows.

3) Compare the MSISDN extracted from the SIP Header with the value extracted from the UD Table.

4) If they match, and if they are judged to be normal but do not match, regard it as an abnormal SIP.

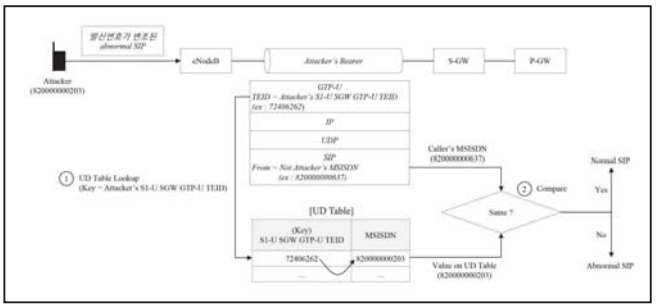


Fig. 23. Procedures for Detect Abnormal SIP Packets with Altered MSISDN.

5 Conclusion and Future Plan

The mobile environment has changed rapidly due to advances in mobile communication technology, and mobile traffic has been increasing sharply around the globe. Accordingly, mobile carriers are introducing LTE networks to secure network availability, and VoLTE, the voice service through the LTE network, has also become popularized. However, as the LTE network provides data and voice service on the All-IP-based network, it is exposed to security threats likely to occur on IP-based networks, such as forgery and alteration of information and eavesdropping. In particular, if the SIP control message for VoLTE service is forged or altered, it may open voice call tolls to crimes like voice phishing.

This paper analyzed the security threat that exists in VoLTE call setup due to the vulnerability of the procedure for checking the S-CSCF registered by the VoLTE UE and obtaining the IP address of the UE to being hacked and offers a countermeasure. The proposed technology can be easily implemented and used for an effective response to VoLTE security threats. Actually, the authors of this paper implemented the proposed technology, installed it on the LTE network of one of the mobile carriers in Korea, and are currently testing the performance. Also, as the proposed technology was implemented in the form of a module, it can be used to supplement the functions of the existing LTE network security equipment.

In the future, if the results of the trial test show a deterioration of performance, the authors are planning to enhance the proposed technology, and will, in any event, continue to conduct research on any security vulnerabilities of VoLTE.

ACKNOWLEDGMENT

This research was funded by the Ministry of Science, ICT & Future Planning, Republic of Korea, as part of its ICT R&D program for 2015.

6 References

[1] http://www.statista.com/statistics/271405/global-mobile-data-traffic-forecast/.

[2] Voice over LTE, Acme Packet, LTE World Summit 2014.

[3] Joo-Hyung Oh, Sekwon Kim, Myoungsun Noh, Chaetae Im, "Phone Number Spoofing Attack in VoLTE," 16th International Conference on Computer Networks and Security, vol. 08, pp. 1151–1153, December 2014.

[4] 3GPP TS 23.401: "General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access".

[5] 3GPP TS 29.274: "General Packet Radio System (GPRS) Tunneling Protocol for Control Plane (GTPv2-C)".

[6] 3GPP TS 29.281: "General Packet Radio System (GPRS) Tunneling Protocol User Plane (GTPv1-U)".

[7] Mike McKernan, "VoLTE vs. VoIP: What's the Difference?" SPIRENT 2012.

[8] 3GPP TS 23.228: "IP Multimedia Subsystem (IMS); Stage 2".

[9] Internet Engineering Task Force (IETF) RFC 3261: "SIP: Session Initiation Protocol".

[10] Internet Engineering Task Force (IETF) RFC 3265: "Session Initiation Protocol (SIP)-Specific Event Notification".

[11] Internet Engineering Task Force (IETF) RFC 3515: "The Session Initiation Protocol (SIP) Refer Method".

[12] Kent, S., and Atkinson, R. "Security Architecture for the Internet Protocol" (RFC 2401, November, 1998).

[13] Ramsdell, B., "S/MIME version 3 message specification", 1999

[14] Franks, J., Hallam-Baker, P., Hostetler, J., Lawrence, S., Leach, P., Luotonen, A., and Stewart, L., "HTTP authentication: Basic and digest access authentication" (RFC 2617, June, 1999).

An Improvement of Efficient Dynamic ID-based User Authentication Scheme using Smart Cards without Verifier Tables

Jongho Mun, Jiye Kim, Donghoon Lee, Jaewook Jung, Younsung Choi and Dongho Won*
College of Information and Communication Engineering
Sungkyunkwan University
2066 Seobu-ro, Jangan-gu, Suwon-si, Gyeonggi-do, 440-746, Korea
{jhmoon, jykim, dhlee, jwjung, yschoi, dhwon}@security.re.kr

***Abstract*—Remote user authentication scheme is one of the most convenient authentication schemes to deal with secret data over insecure channel. During the last couple of decades, many researchers have proposed a remote user authentication schemes which are ID-based, password-based, and smart card-based. Above all, smart card-based authentication schemes are becoming day by day more popular. One of the benefits of smart card-based authentication scheme is that a server does not have to keep a verifier table. In 2015, Lee improved Lee's scheme, and claimed that their scheme is more secure and practical remote user authentication scheme. However, we found that Lee's scheme is still insecure and has no wrong password detection mechanism. In this paper, we propose an improved remote user authentication scheme which is aimed to withstand possible attacks. Then, we show that the proposed scheme is more secure and supports security properties.**

***Keywords*—Remote User Authentication, Smart card, Network Security**

I. INTRODUCTION

Password-based authentication scheme has been widely used during the last couples of decades. Since Lamport [1] proposed the first password-based authentication scheme with insecure communication in 1981, password-based authentication schemes[2]-[10] have been extensively investigated. However, a problem of password-based authentication scheme is that a server must maintain a password table for verifying the legitimacy of a login user. Therefore, the server requires additional memory space for storing the password table. For this reason, many researchers has proposed a new remote user authentication scheme by using smart card. One of the benefits of the smart card-based authentication scheme is that a server does not have to keep a password table. This means that administrative overhead of server remarkably reduced. In the view of the fact that many remote user authentication schemes using smart card [11]-[16] have been proposed. In 2012, Lee[18] demonstrated that the authentication scheme of Das et al.[17] cannot resist password guessing attacks and impersonation attacks, and then proposed an improved scheme for security enhancement. The improved scheme also tried to use a dynamic ID and a nonce for each login in order to prevent adversary from traceability such that an adversary cannot trace the users. Lee[19] in 2015 showed that the authentication scheme of Lee[18] used a common secret key to encrypt each user's password such that any malicious legal user can employ the common secret key to perform off-line password guessing attacks, impersonation attacks and modification attacks, and to trace the other users and proposed authentication scheme based on quadratic residues and solves the security problems. However, the authentication scheme of Lee[19] is still insecure and has no wrong password detection mechanism. To overcome the drawback, we proposed a more secure remote user authentication scheme which is an improvement of Lee's scheme[19]. The remainder of the paper is organized as follows. We begin by reviewing Lee's remote user authentication scheme in Section 2. In Section 3, we describe security weaknesses of Lee's scheme. Our proposed scheme is presented in Section 4. Security analysis of our proposed scheme is given in Section 5. Finally, we conclude this paper in Section 6.

* Corresponding Author: Dongho Won

II. REVIEW IN LEE'S SCHEME

This section reviews the dynamic ID-based remote user authentication scheme proposed by Lee in 2015. As previous researches, Lee's scheme consists of three phases: registration, authentication and password update phases which as follows. The notations used in this paper are summarized as Table 1.

TABLE I. NOTATIONS USED IN THIS PAPER

Notations	Description
U	A qualified user
S	A authentication server
ID, PW	Identity and Password
x	S's master secret key, which is kept secret and only known by S. $\|x\|$ is a security parameter.
DID	Dynamic identity
$nonce$	A random number
T	A timestamp
$\oplus$	The bitwise XOR operation
$h(\cdot)$	A collision resistant one-way hash function
$A \rightarrow B : M$	A sends M to B through a common channel
$A \Rightarrow B : M$	A sends M to B through an authenticated and private channel

A. Quadratic Residue Assumption

Let $n = p \times q$, where p and q are two large primes. The symbol QR_n denotes the set of all quadratic residues in $[1, n-$

1]. If $y = x^2 \ mod \ n$ has a solution, i.e. $\exists$a square root for y, then y is a quadratic residue modulo n. Assume that $y \in QR_n$. It is computationally infeasible to find x satisfying $y = x^2 \ mod \ n$ without the knowledge of p and q since no polynomial algorithm has been found to solve the factoring problem [20]-[22].

B. Registration phase

The registration phase is operated when the user U_i initially registers to the server S and is described as follows.

1) $U_i \Rightarrow S : PW_i$
   User U_i chooses his/her password PW_i and sends PW_i to the server S over a secure communication channel.
2) $S \Rightarrow U_i$: smart card
   Upon receiving the registration message from U_i, S computes $M_i = h(ID_i \oplus x)$, $N_i = ID_i \oplus h(PW_i)$, installs $\{M_i, N_i, h(\cdot), n\}$ in the smart card, where p and q are two large primes and $n = p \times q$, and sends the smart card to U_i.

C. Authentication phase

The authentication phase also comprises login phase and verification phase, which describe as follows.

Login phase
U_i inserts his/her smart card into card-reader, inputs his/her password PW_i, and performs the following steps.

1) Compute $ID_i = N_i \oplus h(PW_i)$, $b = h(M_i \oplus T)$ and $DID_i = ID_i \oplus b$, where T is the current timestamp.
2) Compute $B_i = h(N_i \oplus h(N_i \oplus h(x)) \oplus R)$.
3) Compute $C_i = b^2 \ mod \ n$.
4) $U_i \rightarrow S : \{DID_i, C_i, T\}$

Verification phase
Upon receiving the login message from U_i, S performs the following steps.

1) Verify the validity of timestamp T.
2) Solve C_i by using the Chinese Remainder with p and q to obtain four (b_1, b_2, b_3, b_4).
3) Determine b by checking $h(h(b_i \oplus DID_i \oplus x) \oplus T) = ?b$. If successful, accept this login request; otherwise, reject this request.

D. Password change phase

Users are allowed to freely update their passwords by performing the following steps.

1) U_i inserts his/her smart card into the card-reader and inputs his/her password PW_i.
2) U_i chooses a new password $PW_{i(new)}$.
3) Next, the smart card computes $N_{i(new)} = N_i \oplus h(PW_i) \oplus h(PW_{i(new)})$.
4) Finally, the smart card updates N_i as $N_{i(new)}$. Then U_i can use the new password $PW_{i(new)}$ to login the authentication server S.

III. SECURITY ANALYSIS OF LEE'S SCHEME

In this section, we point out security weakness of Lee's scheme.

A. Denial-of-service via wrong password login

This is the type of attack when a legal user is denied access to services which are meant for him. Suppose U_i inserts wrong password PW_w in login phase. Smart card has no mechanism to detect it, then U_i sends wrong login request $\{DID_w, C_w, T_i\}$ to server S. On receiving $\{DID_w, C_w, T_i\}$ as login request, when S checks the equivalence $h(h(b_i \oplus DID_w \oplus x) \oplus T_i) = ?b$, clearly, it will not hold. As a result, S will terminate the session and U_i will face the DoS.

B. Not support mutual authentication

The mutual authentication called two-way authentication is a process in which both entities in a communication link authenticates each others. Thus, mutual authentication is one of the most important properties of a user authentication protocol. After user and server achieve the mutual authentication, then both them are sure that are the legitimate. However, Lee's scheme cannot achieve explicit mutual authentication between user U and server S.

IV. OUR PROPOSED SCHEME

In this section, we describe more secure remote user authentication scheme which improves Lee's scheme. Our improved scheme can provide mutual authentication. It consists of four phases. Our scheme works as follows.

A. Registration phase

The registration phase is operated when the user U_i initially registers to the server S and is described as follows.

1) $U_i \Rightarrow S : PW_i$
   User U_i chooses his/her password PW_i and sends PW_i to the server S over a secure communication channel.
2) $S \Rightarrow U_i$: smart card
   Upon receiving the registration message from U_i, S computes $M_i = h((ID_i||nonce) \oplus x)$, $N_i = ID_i \oplus h(PW_i)$, $R_i = h(h(ID_i) \oplus PW_i) \oplus M_i$, installs $\{h(\cdot), M_i, N_i, R_i, n\}$, in the smart card, where p and q are two large primes and $n = p \times q$, and sends the smart card to U_i.

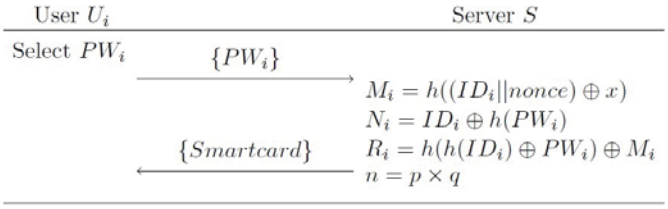


Fig. 1. Registration phase

B. Login phase

If U_i wishes to login S, U_i inserts his/her smart card into the card-reader and enters the password PW_i. Then, the smart card performs the following steps.

1) Smart card computes $ID_i = N_i \oplus h(PW_i)$ and $V_i = h(h(ID_i) \oplus PW_i) \oplus M_i$, then compares V_i with stored R_i. If it holds, smart card computes $b = h(M_i \oplus T_i)$, $DID_i = ID_i \oplus b$ and $C_i = b^2 \ mod \ n$, where T_i is the current timestamp. Otherwise, smart card rejects login request.
2) Then, smart card sends login request $\{DID_i, C_i, T_i\}$ to S through a common channel.

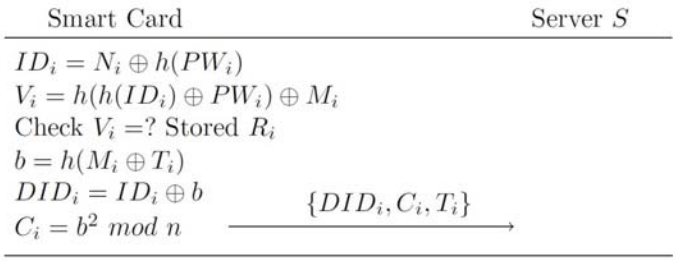


Fig. 2. Login phase

C. Authentication phase

Upon receiving the login request message from U_i, S performs the following steps.

1) S checks freshness of timestamp T_i.
2) After checking the freshness of timestamp, S solves C_i by using the Chinese Remainder with p and q to obtain four (b_1, b_2, b_3, b_4). Then, S determines b by checking $h(h(b_i \oplus DID_i \oplus x) \oplus T_i) =? b$. If successful, accept this login request. Otherwise, reject this login request.
3) S computes $ID_i = DID_i \oplus b$, $R_s = h(b \oplus h((ID_i || nonce) \oplus x) \oplus T_s)$, where T_s is the current timestamp and sends response message $\{R_s, T_s\}$ to smart card via a common channel.
4) When receiving the response message from S, smart card checks the freshness of T_s. Then smart card calculates $V_i^{'} = h(b \oplus M_i \oplus T_s)$. U_i confirms that S is valid if $V_i^{'}$ is equal to received R_s, otherwise U_i terminates this session.

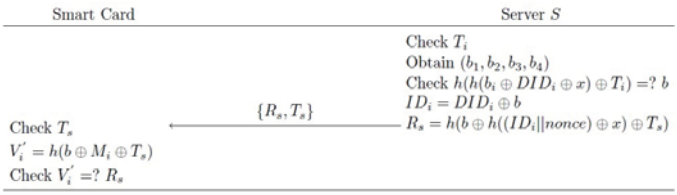


Fig. 3. Authentication phase

D. Password change phase

If U_i wants to change his/her password, he/she inserts his/her own smart card into a card reader, then enters password PW_i. After receiving password PW_i, smart card performs the following steps.

1) Smart card computes $ID_i = N_i \oplus h(PW_i)$ and $V_i = h(h(ID_i) \oplus PW_i) \oplus M_i$, then compares V_i with stored R_i. If it holds, smart card accepts U_i to enter a new password $PW_{i(new)}$. Otherwise, smart card rejects password changing request.
2) After receiving new password $PW_{i(new)}$, smart card computes $N_{i(new)} = N_i \oplus h(PW_i) \oplus h(PW_{i(new)})$, $R_{i(new)} = h(h(ID_i) \oplus PW_{i(new)}) \oplus M_i$ and updates N_i, R_i as $N_{i(new)}$, $R_{i(new)}$. Then U_i can use the new password $PW_{i(new)}$ to login the authentication server S.

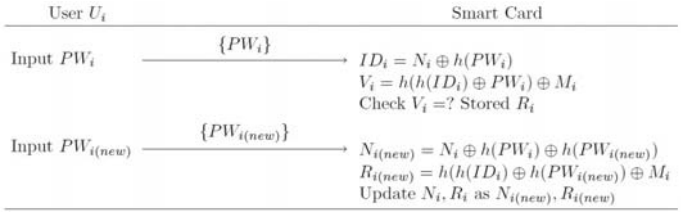


Fig. 4. Password change phase

V. SECURITY ANALYSIS OF OUR PROPOSED SCHEME

In this section, we demonstrate that our scheme can withstand several possible attacks. We also show that our scheme supports several security properties. In Lee's scheme, he demonstrated that their scheme can resist various attacks such as impersonation attacks, replay attacks, password guessing attacks, smart-card-theft attacks, modification attacks, and so on. Our proposed scheme keeps the merits of Lee's scheme. Now, we point out security analysis of our proposed scheme under following two assumptions.

1) An adversary can intercept all messages communicated among smart card and S.
2) An adversary can steal smart card of legitimate user U_i, and he/she can obtain the parameter of smart card.

A. Support Mutual Authentication

A mutual authentication refers to both the server and the user authenticating each other suitably and one of the most important properties of a user authentication scheme. Our proposed scheme provides mutual authentication method to prevent the forged login attack and the forged server attack. In our proposed scheme, authentication of U_i to S represents by $\{DID_i, C_i, T_i\}$ and authentication of S to U_i describes by $\{R_s, T_s\}$. In conclusion, our proposed scheme provides mutual authentication.

B. Support Data Unlinkability

Lee's authentication scheme provides user's login with the dynamic identity $DID_i (= ID_i \oplus b)$, in which $b (= h(M_i \oplus T_i))$ and $C_i (= b^2 \ mod \ n)$ is generated in different runs and is independent due to the one-way property of the hash function. Our proposed scheme's login request message is same as scheme of Lee. Thus, our proposed scheme supports the property of unlinkability.

C. Support User Anonymity

Suppose an adversary U_a intercepts on user U_i's login request message. However, he/she fails to guess the user U_i's identity from $\{DID_i, C_i, T_i\}$. In our proposed scheme, DID_i and C_i implicitly involve the user U_i's identity ID_i. An adversary cannot solve b from C_i where $C_i = b^2 \ mod \ n$ and $b = h(M_i \oplus T_i)$, because of the quadratic residue assumption. Furthermore, we use the timestamp T_i in the login phase, then user U_i's login request message is changed each login time. Thus, our proposed scheme supports user anonymity.

D. Resisting Impersonation Attacks

In the our proposed scheme, only U_i can compute $ID_i = Ni \oplus h(PW_i)$, $b = h(Mi \oplus T_i)$ and $DID_i = ID_i \oplus b$ since only he/she has the secrets N_i, M_i and password PW_i and S can compute $R_s = h(b \oplus h((ID_i||nonce) \oplus x) \oplus T_s)$ since only he/she has the secrets $nonce$ and x. The authentication server S authenticates U_i by checking $h(h(b_i \oplus DID_i \oplus x) \oplus T_i) =? b$ for $i = 1, 2, 3, 4$ and the remote user U_i authenticates S by checking $h(b \oplus M_i \oplus T_s) =? R_s$. Thus, our proposed scheme can resist impersonation attacks.

E. Resisting Smart Card Stolen Attacks

If an adversary U_a steals U_i's smart card, then U_a can extract security parameters $\{h(\cdot), M_i, N_i, R_i, n\}$ from legitimate user U_i's smart card. However, this information does not help them. He/She cannot obtain any information of U_i's ID_i and PW_i because is protected by secret parameters. ID_i in $M_i (= h(ID_i \oplus x))$ is protected by S's long-term secret key x and the collision resistance one-way hash function $h(\cdot)$, and PW_i in $N_i (= ID_i \oplus h(PW_i))$, $R_i (= h(h(ID_i) \oplus PW_i) \oplus M_i)$ is encrypted with ID_i Therefore, the our proposed scheme can resist smart card stolen attack.

F. Resisting Replay Attacks

In the proposed authentication scheme, an adversary cannot correctly modifies $\{DID_i, C_i, T_i\}$ and $\{R_s, T_s\}$ without $ID_i, PW_i, N_i, M_i, R_i$ and x, where $ID_i = Ni \oplus h(PW_i)$, $b = h(M_i \oplus T_i)$, $DID_i = ID_i \oplus b$, $C_i = b^2 \ mod \ n$, and $R_s = h(b \oplus h((ID_i||nonce) \oplus x) \oplus T_s)$. When an adversary tries to use the previous message $\{DID_i, C_i, T_i\}$ to login S or $\{R_s, T_s\}$ to response U_i, a failed adversary will be detected by checking the invalid timestamp T_i and T_s. Thus, the proposed authentication scheme is secure against the replay attacks.

G. Comparison Security Properties

We compare the proposed scheme with the scheme of Lee[18], Lee[19] regarding resistance to possible attacks as depicted by Table 2. Our proposed scheme resists all those attacks to which the previous schemes [18][19] are susceptible.

TABLE II. COMPARISON SECURITY PROPERTIES TABLE

Security Properties	Lee[18]	Lee[19]	Our scheme
User impersonation attacks	No	Yes	Yes
Server impersonation attacks	No	No	Yes
Offline password guessing attacks	No	Yes	Yes
Denial-of-service attacks	No	No	Yes
Smart card stolen attacks	No	Yes	Yes
Modification attacks	No	Yes	Yes
Replay attacks	No	Yes	Yes
Support Mutual Authentication	No	No	Yes
Support User Anonymity	No	Yes	Yes
Support Data Unlinkability	No	Yes	Yes
Wrong password detection by SC	No	No	Yes

VI. CONCLUSION

In 2015, Lee proposed an enhanced scheme of Lee's scheme and demonstrated it is resistance to famous attacks. However, Lee's scheme has no wrong password detection mechanism and may deduce the DoS problem. In this paper, to solve the security vulnerabilities, we proposed an improved protocol for authentication scheme that keeps the similar properties of their scheme and make it more secure. The security analysis explains that our proposed scheme rectifies the weakness of Lee's scheme.

ACKNOWLEDGEMENT

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIP) (No.R0126-15-1111, The Development of Risk-based Authentication·Access Control Platform and Compliance Technique for Cloud Security)

REFERENCES

[1] L. Lamport, "Password authentication with insecure communication," in *Communications of the ACM*, vol. 24, pp. 770-772, 1981

[2] L. Li, I. Lin and M. Hwang, "A remote password authentication scheme for multi-server architecture using neural networks," in *IEEE Transactions on Neural Networks*, vol. 12, pp. 1498-1504, 2001

[3] A. Conklin, G. Dietrich and D. Walz, "Password-based authentication: a system perspective," in *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on*, vol. 50, pp. 629-631, 2004

[4] M. Abdalla, P. Fouque, and D. Pointcheval, "Password-Based Authenticated Key Exchange in the Three-Party Setting," in *On Public Key Cryptography-PKC 2005*, vol. 3386, pp. 65-84, 2005

[5] S. Jiang, and G. Gong, "Password based key exchange with mutual authentication," in *Selected Areas in Cryptography*, vol. 3357, pp. 267-279, 2005

[6] R. Gennaro, and Y. Lindell, "A framework for password-based authenticated key exchange," in *ACM Transactions on Information and System Security (TISSEC)*, vol. 9, pp. 181-234, 2006

[7] Y. Yang, R. Deng, and F. Bao, "A practical password-based two-server authentication and key exchange system," in *Dependable and Secure Computing, IEEE Transactions on*, vol. 3, pp. 105-114, 2006

[8] A. Groce, and J. Katz, "A new framework for efficient password-based authenticated key exchange," in *Proceedings of the 17th ACM conference on Computer and communications security*, pp. 516-525, 2010

[9] I. Jeun, M. Kim, and D. Won, "Enhanced password-based user authentication using smart phone," in *Advances in Grid and Pervasive Computing*, vol. 7296, pp. 350-360, 2012

[10] Y. Lee, and D. Won, "On the Use of a Hash Function in a 3-Party Password-Based Authenticated Key Exchange Protocol," in *Grid and Pervasive Computing*, vol. 7861, pp. 730-736, 2013

[11] Y. Lee, J. Nam, and D. Won, "Security enhancement of a remote user authentication scheme using smart cards," in *On the Move to Meaningful Internet Systems 2006:OTM 2006 Workshops*, Springer Berlin Heidelberg, pp. 508-516, 2006

[12] J. Nam, S. Kim, S. Park, and D. Won, "Security analysis of a nonce-based user authentication scheme using smart cards," in *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 90, pp. 299-302, 2007

[13] W. Yi, S. Kim, and D. Won, "Smart Card Based AKE Protocol Using Biometric Information in Pervasive Computing Environments," in *Computational Science and Its Applications-ICCSA 2009*, vol. 5593, pp. 182-190, 2009

[14] E. Yoon, and K. Yoo, "More efficient and secure remote user authentication scheme using smart cards," in *Proceedings of 11th International Conference on Parallel and Distributed System*, vol. 2, pp. 73-77, 2005

[15] D. He, and S. Wu, "Security flaws in smart card based authentication scheme for multi server environment," in *Wireless Personal Communications*, 2012

[16] J. Mun, Q. Jin, W. Jeon and D. Won, "An Improvement of Secure Remote User Authentication Scheme using Smart Cards," in *International Conference on IT Convergence and Security*, pp. 1-4, 2013

[17] M. Das, A. Saxena and V. Gulati, "A Dynamic ID-based Remote User Authentication Scheme," in *IEEE Transactions on Consumer Electronics*, vol. 50, pp. 629-631, 2004

[18] Y. Lee, "A new dynamic ID-based user authentication scheme to resist smart-card-theft attack," in *Applied Mathematics and Information Sciences*, vol. 6, pp. 355-361, 2012

[19] T. Lee, "An Efficient Dynamic ID-based User Authentication Scheme using Smart Cards without Verifier Tables," in *Applied Mathematics and Information Sciences*, vol. 9, pp. 485-490, 2015

[20] W. Patterson, "Mathematical Cryptology for Computer Scientists and Mathematicians", 1987

[21] K. Rosen, "Elementary Number Theory and its Applications", 1993

[22] Y. Chen, J. Chou and H. Sun, "A Novel Mutual-Authentication Scheme Based on Quadratic Residues for RFID Systems," in *Computer Networks*, vol. 52, pp. 2373-2380, 2008

WARP Net: An Anonymous Whistle-blower and Document-Sharing Network

Sean C. Mondesire
College of Engineering and Computer Science, University of Central Florida, Orlando, FL, USA

Abstract— *Recently, whistle-blowing media articles have revealed wide-scale ethics violations and questionable government and business behavior. These stories are grabbing the headlines and shaking up global political and corporate landscapes. The alarming activities include improper business practices, gross negligence, and legal actions performed by some of the largest and most influential institutions around the globe. Unfortunately, there are many cases of negative repercussions on the whistle-blowers, the individuals who are bringing awareness of these questionable behaviors.*

This work addresses the problem of whistle-blower safety by providing a secure mechanism for information distribution that protects the identity of its sources. We accomplish this goal by presenting WARP Net, an anonymous peer-to-peer overlay network that is centered on data hopping. With WARP Net, we contribute a simple, decentralized network where whistle-blowing documents and memorandums can be anonymously distributed. Through experimentation, we validate the feasibility of WARP Net by comparing its data transfer and routing protocol with FreeNet, an established anonymous peer-to-peer network.

Keywords: Peer-to-Peer, Anonymous Document Sharing, Whistle-blowing.

1. Introduction

Whistle-blower reporting of ethics violations and questionable practices in politics, government, and the corporate world is a recent emerging and popular media trend. This type of reporting has uncovered incidents of improper business practices, employee and animal abuses, and gross negligence . Furthermore, whistle-blowers have uncovered questionable behaviors performed by government agencies, including widespread surveillance programs and corruption of power [1], [2]. Unfortunately, many whistle-blowers receive negative consequences after coming forward, resulting in job termination, harassment, death threats, and legal persecution (ranging from being sued for financial damages to being tried for treason) [3], [4], [5].

This work addresses the problem of whistle-blower safety by providing a secure mechanism for information distribution that protects the identity of its sources. We accomplish this goal by presenting *WARP Net*, an anonymous peer-to-peer (*P2P*) overlay network that is centered on data hopping. With WARP Net, we contribute a simple, decentralized network where whistle-blowing documents and memorandums are anonymously distributed.

The differences between WARP Net and other P2P overlay networks are in the proposed network's *n*-tiered user-structure, data hopping, and ability to self-organize. The combination of each of these features is dedicated to address specific security issues prevalent in popular P2P networks, including the identification of data sources and requesters. Furthermore, this network addresses security issues prevalent in other P2P networks, such as man-in-the-middle attacks and mass malicious user collaboration. Through experimentation, we demonstrate the feasibility of the overlay network and analyze the impact of the data hopping on network traffic. To do so, a data transfer comparison is made between WARP Net and FreeNet, an established anonymous P2P network that serves as the inspiration for many modern peer-based networks.

The paper is organized in the following manner: first, a discussion is made on related P2P networks and the security challenges they face. Second, we define WARP Net, covering the network's topology, message routing, and self-organization. Finally, a demonstration of the network's feasibility is provided with an evaluation of its anonymous message routing.

2. Background

Peer-to-Peer (*P2P*) networking is a type of information distribution where individual networked computers (*nodes*) share data directly with each other. This type of distribution counters client-server based networks, where dedicated servers store and distribute data to nodes. P2P networks have gained wide-spread popularity in the early 2000s with the presence of file-sharing software *Napster*, *Gnutella*, *Kazaa*, *LimeWire*, and *Morpheus* [6], [7], [8]. These and other P2P file-sharing networks have allowed users from across the Internet to share and access media and document files effortlessly. Due to large-scaled copyright infringement and intellectual property and illegal material distribution, many of these early P2P networks have been forced to become inactive or change their distribution method due to government and commercial prosecution [9], [10], [11]. In addition, organizations, such as the *Recording Industry Association of America* (*RIAA*) and the *Motion Picture Association of*

America (*MPAA*) has filed lawsuits against individual P2P users for copyright infringement and the illegal distribution of intellectual property [12]. These lawsuits have encouraged the development of decentralized, anonymous P2P networks that are robust and fault tolerant, hide the identity of its data sources, and protect the activity of their users from third-parties.

2.1 Anonymous P2P Technologies

Freenet is one of the first widely used anonymous P2P networks that aimed at protecting user activity [13]. The network is a decentralized data store which heavily relies on its key-based routing system to protect queries and file retrieval. In essence, Freenet is a network of nodes communicating with one another through the use of encrypted messages. *Freenet nodes* connect to several other users running the Freenet protocol to establish a network of neighbors. When a node wishes to query for a file, encrypted messages are passed from neighbor to neighbor. When a query command is received, the node will search its local data store to detect if portions of the file are stored locally or if the location is known on the network, and return the results to the neighbor who passed the query message. If the file location is unknown to a node, the query message is passed to that node's neighbors. If the file location is not known after reaching a predefined search depth, messages are returned in reverse order, notifying neighbors a search branch has not located the file. The use of relaying messages from node-to-node guarantees the anonymity of the query author. The guarantee is made since nodes can always claim to be relaying another node's query. To further enforce the anonymity of users, Freenet encrypts and fragments shared files to distribute the content across the network. Here, if a *consumer* knows the identity of the file *producer*, the producer cannot be held accountable for the file transmission since it cannot have any idea of what file it is sharing.

Other anonymous P2P systems include GNUnet [14], Free Haven [15], and the popular Tor Project [16]. Similar to Freenet, these systems incorporate message hopping to establish anonymity and claim to be censorship-resistant. In particular, Tor focuses on anonymizing a user's Internet activity by forwarding network requests between nodes until a *time-to-live* (*TTL*) has expired. Upon expiration, the last node to receive the request acts on it on the source's behalf. Then, the request's results are traversed back to the source. Again, this type of routing makes it difficult for attackers to identify request sources, which improves the chances of anonymous web browsing, file transfers, and document sharing.

2.2 Security Issues in P2P Networking

Security exploits are known to have compromise P2P networks in the past, ranging from the spread of malware [17] to the execution of denial of service attacks that cripple the network's servers [18]. Anonymous P2P networks face additional threats, including source identification techniques, such as man-in-the-middle attacks, computer network exploitation, and group collaborating. *Man-in-the-middle* (*MitM*) attacks occur when a third-party interrupts the transfer of data between two, normally directly connected, computers. This attack allows the MitM to intercept-then-forward data transmission, falsify data, compromise encryption keys, and pinpoint the data origins. *Computer network exploitation* (*CNE*) is a sophisticated method of data eavesdropping where a third-party is able to isolate all network information entering and exiting an Internet IP; this type of attack has compromised the identity of data sources on the anonymous network Tor [19]. Finally, *group collaboration* occurs when a trusted P2P group is infiltrated with a large portion of collaborating users with the intention to undermine data hopping. Here, the collaborators share data hopping information to assist in the identification of data sources and queries. Due to the complexity and resource required to execute a successful CNE, the presented work focuses on countering MitM attacks and group compromising.

One generally successful method to counter MitM attacks and group collaboration is the enforcement of the Friend-to-Friend philosophy for P2P networks. The main idea behind *Friend-to-Friend* (*F2F*) networks is that nodes only connect to nodes either they can trust or to nodes a friend of someone in a friend-of-a-friend web can trust. Files are only shared and queries are only made within a *web-of-trust* where each node can be traced to another node through 'friend' associations. Unfortunately, the enforcement of a F2F network does not guarantee complete security as it carries two major disadvantages: 1) malicious "friendly" nodes can infiltrate a network and 2) files outside of the friend web are unreachable.

3. WARP Net

WARP Net is a P2P overlay network that is designed to protect the activity of its users. Particularly, WARP Net anonymizes data sources (whistle-blowers) and distributes their documents to their intended in-network destinations. The network employs a public-key cryptosystem to guard network messages from prying eyes internal and external of the network. Succinctly, the network protects user anonymity by establishing an n-tiered, F2F, encrypted message-hopping network.

3.1 Topology

WARP Net is an n-tiered based hierarchy where each level represents a top-down decomposition of a networked organization. First, the entire network represents an organization with several subparts. Each subsequent level contains a lower structure role in the organization until a level of a collection of individual computers is reached. For simplicity, the remainder of the paper uses a 3-tiered hierarchy, where

each level represents a unique role within the system. Figure 1 shows an example of the 3-tiered network's topology.

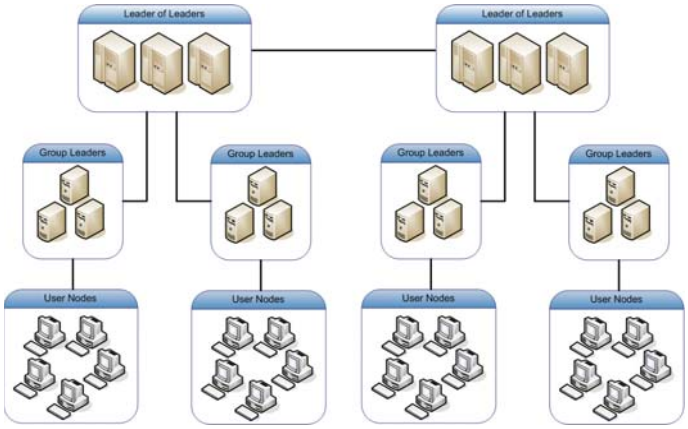


Fig. 1: 3-Tiered WARP Net Hierarchy

In this 3-tier example, all connected computers belong to a fictitious company and have unique organizational roles. The "company" has three departments (administrated by *leaders*). Each leader is responsible for multiple teams (known as *groups*). Each group is comprised of a collection employees (*user-nodes*). A user-node represents an individual employee's connected computer used to query and upload and download documents. Each group has a small set of supervisors (*group-leaders*) that relay information between the users-nodes and leaders. Each leader relays information between leaders of other departments. More details about the responsibilities and capabilities of each role are described below.

User-node: The bottom tier is comprised of user-nodes. User-nodes have the privilege of making, answering, and forwarding queries. They can also introduce and distribute documents and files throughout the network.

Group-Leaders: The middle tiered role is that of group-leaders (*GL*). GLs represent the current hosts of a group where user-nodes connect to in order to join the network and interact with other group members. GLs connect each group member to other groups by establishing links to leader-of-leader nodes. GLs divide their bandwidth and processing power between providing the services user-nodes offer, resolving interaction disputes, validating incoming users, and maintaining the group formation.

Leader-of-Leader: The top tier of the hierarchy is the leader-of-leader (*LL*) role. LLs connect groups to other groups by forwarding messages from one group's GL to a different group's GL. The main job of LLs is to forward messages from one group to all of the other connected groups. LLs act very similar to GLs except they are responsible for GL behavior. The activities of LLs are checked by other LLs of the same leader-group (connected LLs governing the same subset of groups) and their connected GLs. LLs are promoted from the GLs who have fast connections, outstanding reputation ratings, and have managed their groups properly. LLs divide their resources by forwarding messages between groups and other LL's and monitoring group behavior and performance.

Two additional entities that exist in the system are the bootstrap and group servers. *Bootstrap* servers act as an entry point to the network for any node attempting to connect without knowledge of any other nodes. This server constantly receives updates from LLs on the network status and group formations to point newly-arrived nodes to the correct LL to join a desired group. *Group servers* act as dedicated entities that issue group encryption keys for group member validation and malicious node pruning. The network can thrive without both of these entities if addresses of online nodes are available and if a trusted third party is deemed unnecessary for key certification and distribution.

3.2 Group Formation

The network's hierarchy is predetermined and established at the Bootstrap server. This minimal set of network organization requires at least one LL, group, and GL to be online and aware of the organization before accepting any users.

When a node wishes to locate fellow group members through another node, a query can be sent containing a group search message signed using the group's private key. Eventually a node with the group's public key will receive the message, most likely a LL, and retrieve the location of one of the group's active GLs. The query will then be replied along the same path to the verified incoming group member. If a group member joins the network but is the only active one, the closest LL will recognize that group member to be the sole active GL and will redirect any future group inquiries to that active member.

When a group member joins the group or when a GL disconnects, the online GLs must make decisions on who to promote based candidate processing power and bandwidth. Once a new GL is assigned, the other GLs will update the promoted node's status by transmitting collected query data and group status information. In addition the new GL will establish connections to all of the group members it is responsible for. The number of GL to user-node coverage is based on the group's configuration where some groups may want all of their GLs to be connected to all of the members or have each GL be in charge of a section of the group. Groups that allow GLs to connect to all group members allow each GL to act as a backup where if one GL is busy, another GL is easily accessible to provide the same service. Also, messages can be verified easier with majority voting if repetitive messages are transmitted from one group member to all of the GL. Groups that divide the group responsibilities decrease the workload placed on each GL and allow these nodes to focus serving their user-nodes.

Finally, the new GL will connect to the assigned LLs to allow the group to communicate with other groups. Once

the new GL is updated, that node can be used to help the group process network requests and traffic.

LLs are assigned in a similar manner as the GLs. The ratio of LLs to groups should be adequate enough for each LL to process network messages without creating a bottleneck or slow network performance. The performance of LLs is vital to the health of the system since LLs provide the links that connect groups together and manages the communication between different sections of the network. With this reason, LLs should be the most stable and high performing nodes on the network because of the high performance demands query look-ups and data relays necessary to maintain the network.

When a group comes online for the first time, a leader-group will be assigned to it by the pre-configured Bootstrap server. This normally means the leader-group that receives the sole group member will be the leader-group that will service it. If a LL disconnects, the LLs of that leader-group will promote the most outstanding GL or user-node. By promoting user-nodes to LLs, group restructuring is minimized. On the other hand, promoting GLs to LLs allow for faster node comparisons at the cost of forcing the chosen GL's group to restructure to find a replacement GL. Each LL of a leader-group must be connected to all of the same GLs and LLs of other leader-groups for the stability of the network and the ability of LLs to reliably take part in the reputation scheme.

Leader-groups may be disbanded when the number of groups serviced by a leader-group becomes small. In this case, the leader-groups will pass on the responsibility of serving its groups to other leader-groups. This should increase the number of potential query hits since groups are guaranteed to interact with more groups and their members.

3.3 Key Distribution and Use

Several generated keys are exchanged to protect from eavesdropping, user impersonation, transfer verification, and forwarding receipts. It is assumed that every node in WARP Net has its own private-public key pair. Let (Pr_i, Pu_i) be the private-public key pair of a node with ID *i*. This key pair is used for identification verification and signing receipts.

Each node must also keep a group private-public key pair (Pr_{Gi}, Pu_{Gi}) where the node is a member of a group *Gi*. The group key pair is generated by the group creator and is shared among group members. This is done by either the creator exchanging the keys directly with each group member, through invitations, or by posting the keys securely on the group's server.

The bootstrap server receives and stores all group and user identification public keys to issue key certificates by using the bootstrap's private-public key pair (Pr_{Boot}, Pu_{Boot}). The key exchange occurs when a node or group enters the network though this system entry point for the first time or regenerates its key pair.

We assume that each group member stores the public keys of their fellow group members and LL. This is because every node can know the public keys of other nodes in different groups by asking to the bootstrap.

3.4 Routing Protocol

The routing protocol describes how nodes communicate with one another. The passing of messages allow file queries, files transfers, postings, and network status updates to be processed securely throughout the network. All messages exchanged between group members are encrypted using the group's symmetric key. Messages exchanged between non-group members are encrypted in a symmetric key generated between the two nodes' group public keys. All message content remains unchanged while traveling between nodes.

File Broadcasts: All document file names are broadcast to the entire group and to the group's LL once a document file is received at the GL. This broadcast allows users to know when a document has been shared and can be used in future queries.

File Queries: File queries search the network for shared files that satisfy the criteria in the query message.

A user-node who initializes or receives a query sends the query to either another random user-node in the same group or to one of the GLs based on a random roll. If the roll is within the user's forward threshold, the user-node will send the message to another, random user-node. If a user-node received the same query twice, it will forward the query to a GL if a repeated query is not from a GL. If a user-node receives a query from a GL, it searches for the file locally and does not forward the query any further.

As an example, let node A represent a user-node making a query and where node B is the next hop. Both nodes reside in group G1. User-node A will send a message in the following format to node B:

$$E_{SG1}[\text{"}FQ\text{"} \parallel QID \parallel Search\ String]$$

When a query is exchanged between a GL and a LL, the group's symmetric key is used if they both belong to the same group. If not, the contents of the message is encrypted in a symmetric key generated by the GL's and the LL's group public key. "FQ" is the message identifier notifying the message is a file query. The symbol ‖ represents string concatenation. QID is the query identification number. It is generated by taking the hash value of the current time stamp, the sender's user ID, the search string, and a large random number. The label "search string" is the file search criteria each user will use to find files to satisfy the query.

Query Hits: Once a node receives a file query, it will search its local machine for a file that meets the search's criteria and if a match is found, a query hit message is generated. Each node uses the query table to know who to forward a query hit to in order for the response to reach the query author. This query table is a history of all queries

which is stored locally. Once the hit traverses the complete reverse path, the query author should forward the hit to a random group-member to create deniability. Let node *A* be the query author, let node *B* be the node forwarding the hit to *A*, and node *X* be the hit source who resides in group *G2*. If node *A* forwards the hit to another node *C*, *B* cannot determine if *A* is the query author. Node *C* should continue the forwarding chain until a node has received the message twice or the forwarding probability has not been satisfied.

$$ES_{G2}[\text{"}QH\text{"}\| \ QID\| \ HID \ \| \ GroupIDX \ \| \ IP{:}PortMM \ \| \ File \ Info \ \| \ SIGPr_{G2}(\text{H}(File \ Info))]$$

Above is the initial file hit the hit source will send transmit. "*QH*" is the query hit identifier. *QID* is the query ID this hit message is replying to. *HID* is the hit ID which is generated by taking the hash value of the current time stamp, the sender's user ID, file information, and a large random number. *GroupIDX* is the group ID of the hit source. This ID is used to identify where query hits and file transfers originate from. Following the group ID is the IP and port of a middle-man that leads to the transfer of the requested file. Middle-men are nodes that act as a direct link between two groups to exchange files. When a node returns a query hit, he selects a fellow group member to be a liaison between his group and that of the query author. Middle-men allow query authors to bypass a potentially long chain of hops to issue a file request to the hit author. Now a direct connection between the two groups can be established which should result in file requests reaching hit authors faster.

The middle-man address is followed by the file information containing the file name, file description, file offset and length, hash, and other vital information in regard to the file. The message is concluded with the group's signature of the hashed value of same file information. The signature is used to verify that the message came from the specified group. When the message is exchanged between two non-group members, the contents of the message are re-encrypted using the symmetric key generated by both group's public keys.

File Requests: Once a query hit has been returned to the query author *A*, a file request can be performed. To protect *A*'s identity, *A* has the option to connect to the middle-man directly or ask another user-node in the group to connect on its behalf. If the request is forwarded to another group member, that node has the option to connect to the middle-man directly or forward the request again. Similar to forwarding file queries, this decision to connect to the middle-man or forward the request is base on a forward probability. It is essential that the number of forwards is low since long chains from a file query author to the hit author will create noticeable delays in the file transfer. Once a node from *A*'s group connects to the middle-man of *X*'s group specified in the hit message, the middle man will connect to a user-node *W* in its group and request the file. If *W* does not have the requested file in its shared files, it will forward the request to another group member who will perform the same search.

$$ES_{G1}[\text{"}FR\text{"}\| \ HID\| \ FRID \ \| \ IP{:}PortMM \ \| \ File \ Hash \ \| \ File \ Offset\| \ File \ Length]$$

Above is the message for a file request. It represents the message transferred from *A* to the next node in the path to *X* . "*FR*" is a file request identifier. *HID* is the received hit ID. *FRID* is the file request identification used to distinguish file requests. IP:PortMM is the IP and port number of the middle-man of the file source's group. File Hash is the hash of the requested file to be transmitted. File Offset is the beginning offset of the requested file's segment. File Length is the distances from the offset to the requested file segment length.

File forwarding takes place when the hit source receives a file request that matches his query response. Here, the hit source *X* will forward the requested file fragment in the reverse path back for the file request to the query source *A*. An example message is below. "*FF*" is the file forward identifier. *FRID* is the same *FRID* as the file request. File Info represents the hash, and the file offset and length of the fragment. Next is the hash of the file information signed in the hit author's group private key, followed by the fragment data of the actual requested portion of the file.

$$ES_{G2}[\text{"}FF\text{"}\| \ FRID \ \| \ File \ Info \ \| \ SIGPr_{G2}(\ H(File \ Info)) \ \|Fragment \ Data]$$

3.5 Disconnections

Because all user-nodes of a group are connected to each other, the problem of disconnections is concerned with repairing path gaps of queries and file transfers. This repair is done by using receipts forwarded to nodes away to redirect messages over the gap. For instance, if a query traveled from *A* to *B* then to *C* and node *B* disconnects some time after *B* forwarded *C*'s receipt. Node *A* can send *C* *C*'s forwarded receipt to signify that *A* is the next hop after *B*. Now all hit messages will be routed to *A*, repairing the gap. As stated earlier, GL or LL disconnects are handled by promoting outstanding nodes to higher positions. The promoted nodes will receive network updates by their fellow GLs or LLs.

4. Data Analysis

To simulate the unique features of WARP Net, a new P2P simulator was designed and implemented, WARP-Sim. WARP-Sim is a discrete event simulator witten in Java and is used to monitor the network behavior and feasibility of the system. For these feasibility tests, encryption is not performed during the simulation and nodes joining and leaving the network are not simulated. Future research will analyze fault tolerance and encryption delays in WARP Net. The simulator pre-configured the network with node

placement and their web of interconnectivity. Each simulated node possesses characteristics such as available bandwidth and listing of shared files.

4.1 *B. Experiment 1: Initial Performance Evaluation*

WARP Net was simulated using WARP-Sim for 10 runs to determine the average system statistics. 10 runs were performed because there are a number of random variations per simulation run. Each run simulated the P2P network for 3600 simulated seconds with 4,000 nodes processing messages and interacting with each other. Of these 4,000 nodes, 200 groups were formed with 20 group members each. A 5:1 ratio of user-nodes to GL was used. In addition, each leader-group was assigned to oversee 3 groups where each leader-group was comprised of the 3 fastest group-members. LLs are the only nodes prohibited to make file requests and share files to ease the large workload of message forwarding and group-governing expected of these leaders.

The method of assigning bandwidth and number of shared files for each node were comprised of the findings in [20], [21]. 70% of all nodes had high-speed Internet connections. Of those nodes, 90% had broad-band connections and 10% had T1 or T3 connections. Of the 30% dial-up nodes, 60% possessed 33.6-56kbps connections, 15% possessed 64-128kbps connections, and 25% used 14.4-28.8kbps connections.

The number of shared files was determined by the bandwidth, as influenced by the results in [21]. The dial-up nodes had a 20% chance of sharing no files, 60% probability of sharing between 0 and 100 files, 14% chance of sharing between 100 and 1,000 files, and a 6% chance of sharing 1,000 to 10,000 files. The high-speed nodes had a 10% chance of sharing no files, 60% chance of sharing 0 to 100 files, 24% chance of sharing 100 to 1,000 files, and 6% chance of sharing 1,000 to 10,000 files.

2,500 file queries were processed randomly throughout the simulated time. Each node was given an equally-likely chance of making any query at any time. Each query averaged 5 hops. For 20 percent of all query hits returned, the query author produced a reputation query for the hit source's group and waited 60 seconds to receive the group's rating. If the group's rating is greater than or equal to 0 then the query author would issue a file request for the file returned in the hit. Also, all file requests sought a 1 MB fragment of the queried file. The forward probability was 50%.

4.2 Experiment 1 Results

Search Results: Of the 2,500 file requests, approximately 225,422 hits were generated from 404,304 file queries, with about 224,828 of them reaching the file query author. The average time for the first query hit was 7.11 seconds. These query hits spawned 40,260 file requests resulting in 30,935 uploads attempted, 2,729 downloads completed, and 19,175 file fragments transferred among hops and query authors.

File Transfer and Bandwidth Statistics: Even though the average bandwidth for each node was 78.05 KB/s, the average download speed was 7.06 KB/s. This download speed was the average transfer rate from the last hop to the file request author. It took an average of 701.19 seconds for a node to receive a file after issuing a file request. This delay is the time needed for each hop to receive the file plus the time for locating the file's source. Finally, the system used less than 21% of its bandwidth for all interactions.

4.3 Experiment 2: Group-Size Reduction

After noticing the delay the hops place on file transfers, another experiment was executed where the size of groups was decreased from 20 to 10 nodes and assigned a GL for every three user-nodes. All other simulation settings were kept the same as the first experiment.

Experiment 2 produced a faster average file transfer speed of 9.30 KB/s than experiment 1. Also, the average time needed from file request to download was 664.17 seconds, 37.02 seconds faster than experiment 1's.

4.4 Experiment 3: Increased Hop Probability with Group-Size Reduction

Experiment 2 showed that file transfer speeds could be increased with smaller groups since there are less nodes to provide cover. These increased speeds were at the cost of fewer nodes receiving queries due to the average amount of hops set to 5. To remedy this, a third experiment was devised, this time increasing the hop average to 6. All other settings were left the same as experiment 2. Of the 514, 797 file searches from the 2,500 file queries, there were 275, 511 hits. On average, 7.26 seconds were needed for the first query hit to make its way back to the query author. The query authors saw an average download speed of 11.3 KB/s and waited 183.79 seconds for each file.

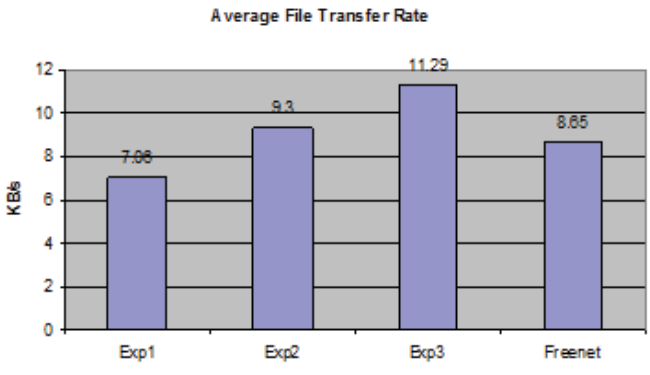


Fig. 2: File Transfer Rates with WARP Net vs. FreeNet

5. Discussion

WARP-Sim was designed to compare WARP Net with Freenet. Because Freenet and WARP Net are different in functionality, file transfers and bandwidth utilization are the most comparable metrics of the two systems. Figure 2 shows the average file transfer speeds between Freenet and WARP Net. Freenet's simulation averaged download speeds to be between 9.7 KB/s for the nodes with fast internet connects and 6.2 KB/s for dial-up nodes. With an average download speed of 7.1 KB/s in experiment 1, 9.3 KB/s in experiment 2, and 11.29 KB/s in experiment 3, WARP Net's file transfer speeds are comparable with Freenet's. With about 70% of all nodes in the simulation of Freenet having high speed Internet connections, one can deduce that an average file transfer speed of both fast and slow nodes is 8.65 KB/s. Compared to this average, WARP Net outperforms the simulated Freenet by 30.5%. Freenet's bandwidth utilization for both download and upload combined averaged 4.98% usage, where WARP Net's nodes used 20% in experiment 1, 14% in experiment 2, and 17% in experiment 3. It is gathered that WARP Net's load on bandwidth utilization is because nodes are processing more system functions such as the addition of interaction receipts than Freenet.

It is concluded that because WARP Net is able to distribute files with a faster transfer rate than the established Freenet and the query routing has resulted in fast responses, WARP Net is a feasible P2P system for exchanging documents. The largest influence on file transfer rates is the number of hops between queries and resulting data transfers.

6. Conclusion

In this paper, WARP Net, an anonymous, friend-to-friend P2P whistle-blowing and document-sharing network was described in detail. The architecture and routing protocol is secure due to the necessity of all nodes exchanging encrypted messages and its use of a web-of-trust to establish peer associations. The anonymity of nodes is enforced through the notion of hop making, using nodes in the network as cover to allow for deniability for network activity. Experiments compared the proposed network to Freenet, an established anonymous P2P network with similar characteristics. Results have determined that data transfer rates perform comparable to the standard Freenet configuration when WARP Net has a high data hop probability. When optimized, WARP Net possesses significantly faster data transfer rates than its competitor.

Future work will validate WARP Net's ability to withstand common security vulnerabilities, incorporate a group-based reputation scheme into the overlay network, and further examine the whistle-blowing dimensions of the system. The security validation will analyze WARP Net's ability to protect user identity and withstand man-in-the-middle attacks, malicious and corrupted data propagation, and computer network exploitation. The reputation method will improve network security by introducing policing within the group to identify malicious files and users. Finally, WARP Net's ability to serve as an anonymous whistle-blowing and document-sharing network will be demonstrated with the presence of realistic security vulnerabilities and malicious users.

References

[1] T. N. Y. T. E. Board, "Edward snowden, whistle-blower," Jan. 2014.

[2] E. M. Glenn Greenwald and L. Poitras, "Edward snowden: the whistleblower behind the nsa surveillance revelations," June 2013.

[3] J. Tate, "Bradley manning sentenced to 35 years in wikileaks case," Aug. 2013.

[4] J. P. Near and M. P. Miceli, "Retaliation against whistle blowers: Predictors and effects.," *Journal of Applied Psychology*, vol. 71, no. 1, p. 137, 1986.

[5] J. Rothschild and T. D. Miethe, "Whistle-blower disclosures and management retaliation the battle to control information about organization corruption," *Work and occupations*, vol. 26, no. 1, pp. 107–128, 1999.

[6] S. Saroiu, K. P. Gummadi, and S. D. Gribble, "Measuring and analyzing the characteristics of napster and gnutella hosts," *Multimedia systems*, vol. 9, no. 2, pp. 170–184, 2003.

[7] N. Leibowitz, M. Ripeanu, and A. Wierzbicki, "Deconstructing the kazaa network," in *Internet Applications. WIAPP 2003. Proceedings. The Third IEEE Workshop on*, pp. 112–120, IEEE, 2003.

[8] J. Liang, R. Kumar, and K. W. Ross, "Understanding kazaa," *Manuscript, Polytechnic Univ*, p. 17, 2004.

[9] G. S. Moohr, "Crime of copyright infringement: An inquiry based on morality, harm, and criminal theory, the," *BUL Rev.*, vol. 83, p. 731, 2003.

[10] R. Stern, "Napster: a walking copyright infringement?," *Micro, IEEE*, vol. 20, no. 6, pp. 4–5, 2000.

[11] D. Lichtman and W. Landes, "Indirect liability for copyright infringement: an economic perspective," *Harv. JL & Tech.*, vol. 16, p. 395, 2002.

[12] S. Goel, P. Miesing, and U. Chandra, "The impact of illegal peer-to-peer file sharing on the media industry," *California Management Review*, vol. 52, no. 3, pp. 6–33, 2010.

[13] I. Clarke, O. Sandberg, B. Wiley, and T. W. Hong, "Freenet: A distributed anonymous information storage and retrieval system," in *Designing Privacy Enhancing Technologies*, pp. 46–66, Springer, 2001.

[14] K. Bennett, C. Grothoff, T. Horozov, I. Patrascu, and T. Stef, "Gnunet-a truly anonymous networking infrastructure," in *In: Proc. Privacy Enhancing Technologies Workshop (PET*, Citeseer, 2002.

[15] R. Dingledine, M. J. Freedman, and D. Molnar, "The free haven project: Distributed anonymous storage service," in *Designing Privacy Enhancing Technologies*, pp. 67–95, Springer, 2001.

[16] T. Project, "Tor project: Anonymity online," Apr. 2015.

[17] A. A. Gostev, A. V. Nikishin, I. I. Soumenkov, and R. V. Rybalko, "System and method for malware detection in peer-to-peer computer networks," July 9 2013. US Patent 8,484,347.

[18] H. Koo, Y. Lee, K. Kim, B.-h. Roh, and C. Lee, "A ddos attack by flooding normal control messages in kad p2p networks," in *Advanced Communication Technology (ICACT), 2012 14th International Conference on*, pp. 213–216, IEEE, 2012.

[19] B. Schneier, "How the nsa attacks tor/firefox users with quantum and foxacid," Oct. 2013.

[20] A. G. H. Skogh, J. Haeggstrom and R. Ayani, "Fast freenet: Improving freenet performance by preferential partition routing and file mesh propagation," Cluster Computing and the Grid Workshops, 2006.

[21] P. G. S. Saroiu and S. Gribble, "A measurement study of peer-to-peer file sharing systems," Multimedia Computing and Networking, 2002.

Vehicle Communication and Infrastructure Security: Initial Thoughts

Mustafa Saed, Muhammad Rizwan
Hyundai-Kia America Technical Center, Inc.
Superior Township, MI 48198, USA
{msaed, mrizwan}@hatci.com

***ABSTRACT*— Demand on providing secure communication among vehicles and Vehicle to Infrastructure networks is constantly growing. Electronics Control Units (ECUs) in vehicles need to authenticate each other and the infrastructure to verify they are whom they claim, and they also need to ensure the integrity of the shared safety critical information. Adversaries can masquerade as real subscribers in vehicle to infrastructure networks and broadcast harmful messages to destroy the system. The intent of this paper is to introduce thoughts on enforcing security at various levels in vehicle architecture to prevent hackers from accessing vehicle ECUs to install malicious software and granting unauthorized access to the vehicle systems.**

Keywords*— *Vehicle Network, CAN Bus, Dispatcher, Wireless Communication, Security, Vulnerabilities

I. INTRODUCTION

Security is one of the most critical issues facing automotive companies developing wireless communication systems for everyday applications demanding interface between the customers and the vehicle. Because of the direct interaction between humans and vehicles, engineers need to provide and implement a secure system in the vehicle to let people take full advantage of the new applications and features in their vehicle. Vehicles that are equipped with Telematics system and internet access, such as BlueLink in Hyundai vehicles and OnStar in GM vehicles, are becoming more vulnerable to cyber-attacks. Providing secure systems in vehicles will deter unauthorized parties from accessing vehicles' communication system and causing hazards in the road. Security concerns are becoming inevitable following the fast development in the technology of the vehicles, especially those equipped with remote access. In addition, allowing physical access to the vehicle's CAN (Controller Area Network) bus with no consideration for security increases the desire to enhance and develop vehicle security system. With the traditional safety telematics services, stolen vehicle tracking, and diagnostics aimed at the physical protection of vehicles, drivers and passengers are becoming main stakeholder. Awareness is growing with respect to the threat of cyber-attacks and their impact on the physical integrity of persons, especially with Vehicle-to-Vehicle communication and autonomous vehicles. Current vehicle architectures are at risk of wireless security break-ins, but future vehicle architectures and systems will only increase the risk because telematics systems have embedded cell phones and wireless protocols containing private information, such as financial records, pin numbers, credit card information, and birth dates. This risk needs to be mitigated. Protecting our customers' private information and the vehicle systems from hackers, unauthorized people, and the infectious viruses should have the highest priority. Viruses will have a direct impact on the trustworthiness and quality from the viewpoint of the consumer as well as the vehicle's safety dynamics, such as the multiple ECU's and its associated systems depending on accurate and uncorrupted information.

To develop and enhance vehicle security, and provide sophisticated safety system for vehicles, two techniques are foreseeable, inter-vehicle Communication Security to provide secure communication/protocol between the vehicle and the infrastructure via wireless network, and intra-vehicle Communication Security to develop and enhance the security communication/protocol between the telematics unit and the ECUs connected via CAN Bus.

Securing the Inter-Vehicle Communication demands the application of cryptography and data security to the packet data session (TCP/IP) and the voice service. A number of attempts have focused on providing two-way communication security between the vehicle and the infrastructure including the vehicle-to-vehicle (V2V) communication as well. However, the security of the V2V communication still needs more development and enhancement.

Duraisamy et al [1] introduced an idea to implement new hardware, which uses Elliptic Curve Cryptography and Digital Signature Algorithm (ECDSA). Their approach allows two parties, a remote agent and a network embedded system, to establish a 128-bit symmetric key, and encrypt all transmitted data via the Advanced Encryption Standard (AES) algorithm.

The Identity-based Batch-Verification (IBV) technique was proposed by Zhang et al [2]. It uses a private key for pseudo identities to avoid the use of certificates. Each received signature should be verified within 300ms intervals based on the Dynamic Short Range Communication (DSRC) protocol used.

Qian et al [3] invested the features of the Medium Access Control (MAC) layer protocol to achieve both Quality of Service (QoS) and security requirement for vehicular networks safety application. Designing an efficient MAC protocol to achieve safety through vehicular networks is essential.

The above attempts/techniques illustrate the critical security concerns arising from vehicle to infrastructure communication. Because the communication between the vehicle and Infrastructure is implemented by using wireless technology, the possibility of vehicle cyber-attack is high. This risk will increase if the security is not considered in the inter-vehicle communication infrastructure. To elucidate this issue and prevent any serious threats to the system, a need for implementing a robust security technique in the inter-vehicle communication and providing a great inter-vehicle communication service to the customers is mandatory [4].

The other security technique that needs consideration in order to provide a robust security system to the vehicle communication, is securing Intra-Vehicle communication. The Intra-Vehicle Communication security compels protecting transmitted data between the vehicle's ECU's through the Controller Area Network (CAN) Bus, which is an open and unsecure automotive protocol. In the past, there was no way for accessing the vehicle remotely. The physical access only allowed getting to the vehicle CAN bus via the On-Board Diagnostic port (OBD). Hence, there were no security concerns related to accessing the vehicle CAN bus and tempering with the vehicle's system. On the other hand, after the enormous changes and development in the wireless technology, the vehicles are impacted by this technology through housing the Telematics unit and allowing it to communicate with the rest of the ECUs in the vehicle via the CAN bus. The CAN bus increases the likelihood for accessing the vehicle CAN bus remotely, which in turn increases the risk of hacking the vehicle and affecting the drivers and vehicles' occupants' safety [5].

This paper proposes initial thoughts for performing remote control vehicle operation with two security mechanisms utilizing the Inter and Intra Vehicle Communication. The rest of the paper is organized as follows: Section 2 focuses on automotive multiplexing methods and CAN protocols. The overview on Next Generation Telematics Pattern (NGTP) is presented in section 3. Section 4 discusses security thoughts for vehicle communication and infrastructure. This is followed by discussion in section 5. Finally, section 6 concludes the paper.

II. Multiplexing Methods and CAN Protocols

A. Multiplexing Methods

Adopting multiplexing methods in automotive technology becomes the greatest achievement in the potential to make vehicles more efficient by reducing the weight in the power distribution system, and increasing the number of the electronic control units. Multiplexing methods are used to connect many ECUs via the CAN bus in a single or dual wire and allows the two-way communication between each other. There are two primary methods used in Multiplexing; time division and frequency division. The time division method inserts a sample of each channel onto the data stream and the channels are selected for a short period of time. This use is the most accurate form of time sharing amongst various channels and is most prevalent method in the automotive industry. The second method, frequency division, uses a different technique which shares the process amongst various channels where information data can be designated with a carrier frequency through each channel to modulate the wave signals.

The Society of Automotive Engineers (SAE) divided the automotive communication sector into three classes. According to Paret et al [6], Class A can support 100 nodes and is categorized to handle data speeds (baud rate) up to 1 kilobit per second (kbps). However, the lag time, which is the time delta between a transmission request and transmission initiation, is 50 ms. Class A baud rate is used in tail light, turn signals, driver convenience features, and entertainment systems. Class B can support 50 nodes and is categorized to support data speeds upwards of 100 Kb/s. For real time events that require urgent speed with high accuracy values, class C is needed. Its data rate is in upwards of 1 Mb/s. Class C baud rate is used mainly in powertrain systems. Class C does not accommodate new systems such as Collision Avoidance System, Global Position System (GPS), and many other related systems.

Network nodes are transmitting and receiving signals via many types of communication, known as protocols. These protocols are created by a set of rules for coding, address structure, transmission sequence, and error detection and handling. These protocols are also referred to as the transmission medium, transmission speed, and electrical signal requirements depending on whether copper wires or optical fiber is used. When associated with automotive networking, protocols cover a majority of functions assigned to the different layers of the Open System Interconnection (OSI) model.

B. CAN Protocols

The communication between the ECUs in the vehicle CAN bus needs relies on a communication protocol called the Controller Area Network (CAN) protocol. A CAN controller acts as mediator to control the communication between the ECUs in the CAN bus. In CAN, disputes between messages are determined on a bit-by-bit basis in a non-destructive arbitration resulting in the highest priority message gaining access to the bus. There are 2,032 different messages supported by CAN protocol with up to 8 bytes of data. Each CAN message data acts differently from other serial communication protocols. The CAN message does not contain information relating to the destination address. The message contains an identifier, which indicates the type of information available. This feature allows convenient addition or deletion of the intelligent nodes in an automotive system. Furthermore, each node decides whether to read or ignore a CAN message [6].

III. NEXT GENERATION TELEMATICS PATTERN (NGTP)

NGTP is a new approach for delivering over-the-air services to in-vehicle devices and handsets alike, with the focus on open interfaces across the entire service delivery chain [7]. There are two versions of NTGP. NTGP version 1 permits the supply of new services faster based on the Telematics technology advances. In addition, varying customer needs can be addressed more quickly by substituting old services with new ones without the stress of introducing technical modifications within the vehicle. NTGP version 2 supports openness and flexibility by splitting the parts of the telematics delivery chain, and launching a 'dispatcher' to offer a single interface between the vehicle's telematics unit and the telematics of the service provider. The open interface generated by NGTP also enables the OEMs (Original Equipment Manufacturer) to constantly introduce new services to both legacy vehicles and new models over the whole vehicle lifecycle.

NGTP's developers has established six objectives dealing with furnishing a technology-neutral pattern and coherent user interface for telematics services, lowering obstacles to collaboration, implementation of new technologies, sustaining legacy systems for connectivity, encouraging innovation, and growing the value for vehicle manufacturers, service providers, content providers, and drivers.

NGTP will enable vehicle manufacturers to use the best offerings from a variety of partners while maintaining a consistent driver experience. The new pattern will also allow service providers (SP) and content providers to sell the same services to multiple vehicle manufacturers. Moreover, NGTP will support legacy systems, allowing older and newer vehicles alike to access new telematics offerings.

In order to foster collaboration and innovation, the specifications that constitute NGTP will be made public under a Creative Commons License. The NGTP group will work with the telematics providers to communicate the specifications, support testing and potential adoption of NGTP.

The NGTP architecture is depicted in Fig. 1. The key component is a technology-neutral intermediary called the Dispatcher (DSPT), which connects the vehicle's Telematics Unit (TU) to the Service Handler (SH) and Service Integrator (SI).

The communication between the vehicle and the call center or the customer is achieved via multiple interfaces. The Service Integrator unit (SI) is responsible for the communication between the Call Center (CC) unit, Public Safety Access Point (PSAP) unit for 911 calls, Content Provider (CP) unit, and other services. The communication between SI, Customer Data Provider (CDP) unit, and Dispatcher is taken care of by the Service Handler (SH). The Dispatcher (DSPT) is in charge of the communication between the SH, Provisioning Data Provider (PDP), and the vehicle. Further details can be found in [7]. The following example should illustrate this organization.

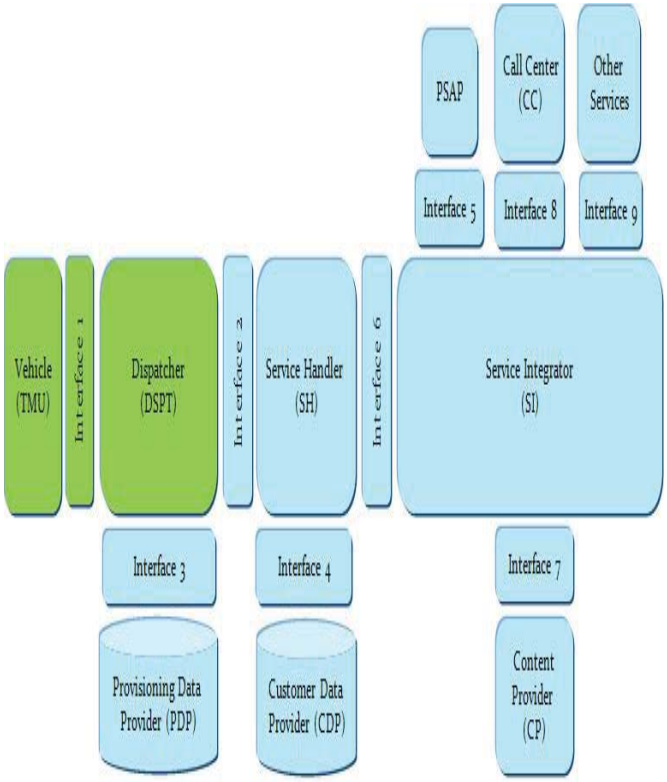


Fig. 1. NGTP version 2 architecture [7]

Assume that the customer is trying to lock his/her vehicle using a mobile application or the web portal. The following steps will take place in this scenario:

a. Customer logs in with his/her account using mobile app/web portal.
b. Customer sends request to lock the vehicle's door. This Short Message Service (SMS) is encrypted by the wireless carrier provider, such as Verizon Wireless.
c. The SI will receive the request and forward it to SH after verifying the customer information with the Content Provider (CP).
d. The SH forwards the request to Dispatcher after identifying the customer's subscription
e. The Dispatcher will decrypt the SMS message to direct the request to the right vehicle after validating the VIN# and Mobile Dialup Number (MDN).
f. The vehicle will receive the request from the Telematics Units and send it to the Body Control Module (BCM) to execute the request.

IV. Security in Vehicle Communication and Infrastructure

To achieve the highest possible security level, a security approach is suggested. As shown in Fig. 2, this approach enforces security within the Telematics Module Unit (TMU) and the Body Control Module (BCM). The reason behind using two-level security is to have two defense lines. If the dispatcher is hacked (first-level), the hacker needs to attack the second level of security to gain control of the vehicle's remote service. This paper will only deal with the security between the dispatcher and the vehicle and not the end to end security (not all the units of Fig. 1).

Encrypted messages are received and decrypted by the TMU. Contained within the TMU protocol is either a seed or a regenerated key which will be exchanged /verified with the BCM at each ignition cycle. OEMs will decide whether to use the seed or regenerated key. There is also a mandatory shared symmetric key (K). The TMU sends an encrypted CAN signal to the BCM, which needs to decrypt and verify the integrity of the CAN signal. Upon successful completion, the CAN message will be sent to the respective ECU for the activation of a remote service.

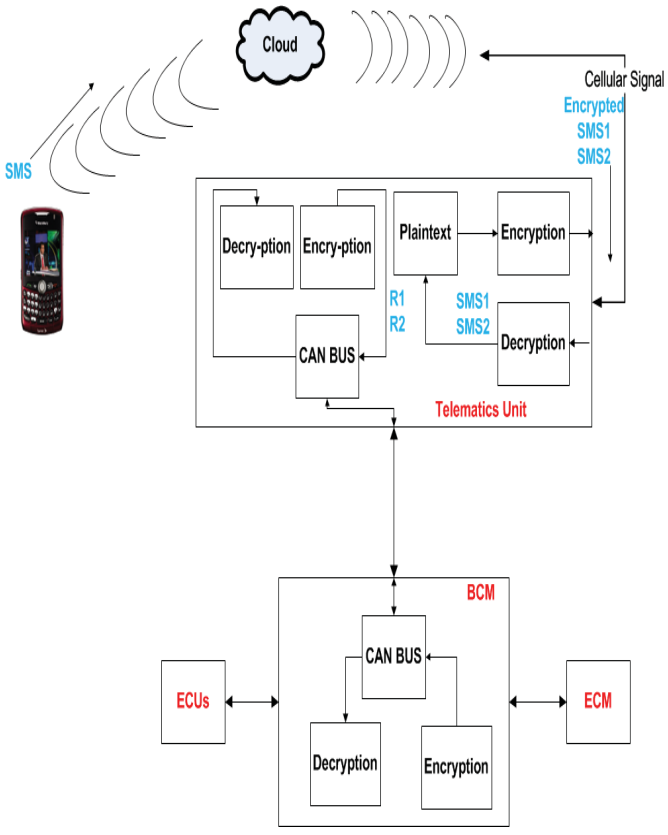


Fig. 2 Security architecture for vehicle communication

A. First Level of Security (Dispatcher to Vehicle)

Customers will only send one SMS for their remote service request, such as remote door lock (RDL). This SMS will be delivered to the Dispatcher, who will transform it into two SMS messages using the specific Original Equipment Manufacturer (OEM) encryption policy. The automotive company will decide the encryption method used based on the cost and their ECUs specifications. This encryption is accomplished within the dispatcher and the TMU embedded in the vehicle. Therefore, other partners, such as the Cellular Provider and the TSP, will not have any knowledge of this encryption.

The key exchange will be completed between the dispatcher and the TMU so that encrypted and decrypted communication will only occur between the dispatcher and TMU. SMS1 and SMS2 will be sent at two different times. If SMS1 is sent at t_0, then SMS2 will be sent at t_0+30 sec with a tolerance of ± 5sec. When the TMU receives the encrypted SMS1 and then SMS2 from the dispatcher, it decrypts SMS1 and SMS2 to obtain the plaintexts R1 and R2 respectively.

B. Second Level of Security (Intra vehicle communication)

In order to prevent hackers from accessing the embedded vehicle computers, a second level of security in the vehicle communication between Module to Module (M2M) communications is proposed. Remote service request requires reaching the second-level security layer to complete the request.

The plaintext R1 and R2 will be employed when decrypting the remote request via the CAN bus. The seed and the shared key have been injected at the manufacturer site and will be used to perform the CAN encryption and authentication. The security approach is described as follows:

1) The TMU stores the new seed (S_{i+1}) and the one before it (S_i). The initial seed (S_0) will be injected in the vehicle (TMU and BCM) at the manufacturing time. At each ignition cycle, the BCM generates a new seed (S_{i+1}) and sends it to the TMU after encrypting it with symmetric key K. Ultimately, TMU decrypts the received message using K to get S_{i+1}.
2) The TMU combines S_{i+1}, R_1 and R_2, and encrypts $Y = S_{i+1} \| R_1 \| R_2$ using the symmetric key K to obtain $M_1 = E_K(Y)$.
3) The TMU finds H(Y) and appends it to M_1 to obtain $[H(Y) \| M_1]$ before sending it to BCM.
4) The BCM decrypts the message received in (3) above to get Y, verifies the hash value, and extracts $S_{i+1} \| R_1 \| R_2$. The S_i (BCN already has S_i) and S_{i+1} are then forwarded to the validation circuit in the BCM.
5) If the validation is successful, the BCM uses R1 and R2 to operate the customer's remote request. Successful

notification will be sent to the telematics server and the customer.

6) If the validation process fails, the BCM will send a failure notification to the TMU to notify the telematics server and the customer of the failure.

V. DISCUSSION

The future of vehicle security is very critical. To improve security enhancement, many issues need to be taken into consideration. The first needed enhancement concentrates on embracing the Internet Protocol Version six (IPV6) in the vehicle communication, NASPInet, anonymization, behavioral economics/privacy, and cross-domain security involving IT. This approach will definitely enhance the vehicle security approach [8].

Another enhancement to security of the vehicle to infrastructure communication involves using the public key infrastructure (PKI) to address all the related requirements of the operation and devices of the vehicle communication [9]. This could be augmented by implementing and developing the vehicle security certificate lifetime and securing the trusted device profile [10].

An area that seems neglected is the customer's privacy. Customer privacy and the privacy of the information should be protected in all inter and intra vehicle communications [11]. Furthermore, critical data, such as business location, should also be protected [12].

Authentication in vehicle communication, such as the module to module communication, should be enriched. This could be achieved by implementing a robust security approach for the vehicle communication as a future priority [13]. Finally, reporting and updating any newly created vulnerability related to the vehicle communication by monitoring and tracking the communication and the data flow through the vehicle and keeping the customer's privacy in consideration is a must [13].

VI. CONCLUSION

To enhance the security of telematics system, initial thoughts of In-Vehicle Communication and Infrastructure has been proposed. Future approaches, techniques, and methods needed to improve and enhance this security are discussed. The security features required for the vehicle have been addressed. The paper provided important and practical ideas to make the telematics system a reliable secure system so that customers can take full advantage of all its features. More work will be done as our thoughts are part of a work in progress. Vehicle communication security requirements related to Inter vehicle and Intra vehicle communications will be the focus of our work in progress. Implementing powerful cryptographic protocols will be our main focus to achieve a robust vehicle security system.

REFERENCES

[1] Roshan Duraisamy, Zoran Salcic, Maurizio Adriano, and Miguel Morales-Sandoval, "Supporting Symmetric 128-bit AES in Networked Embedded Systems: An Elliptic Curve Key Establishment Protocol-on-Chip," University of Auckland, University of Rome, National Institute for Astrophysics, Optics and Electronics, 2006.

[2] Chenxi Zhang, Rongxing Lu, Xiaodong Lin, Pin-Han Ho, and Xuemin (Sherman) Shen, "An Efficient Identity-based Batch Verification Scheme for Vehicular Sensor Networks," University of Waterloo, 2008.

[3] Yi Qian, Kejie Lu, and Nader Moayeri introduced paper, "Performance Evaluation of Secure MAC protocol For Vehicular Networks," National Institute of Standards and Technology, University of Puerto Rico, 2008.

[4] U.S Department of Transportation, National Highway Traffic Safety Administration, "Vehicle Safety Communications Project; Task 3 Final Report; Identify Intelligent Vehicle Safety Applications Enabled by DSRC," Notional Technical information service (22161), Virginia, March 2005.

[5] William Stallings, "Cryptography and Network Security Principle and practices," New Jersey, NJ: Pearson Prentice Hall, 2010.

[6] Paret, D. and Riesco, R., "Multiplexed Networks for Embedded Systems: CAN, LIN, FlexRay, Safe-by-Wire," SAE International, June 20, 2007.

[7] The NGTP, "Next Generation Telematics Pattern," October 2010. Available: http://telematicsnews.info/2010/10/24/bmw-announces-ngtp-20-next-generation-telematics-pattern_o1241/

[8] Zagar, D. and Grgic, K., "IPv6 security threats and possible solutions," WAC, July, 2006, pp. 1-7.

[9] Zhao, M., Smith, S., and Nicol, D., "Evaluating the Performance Impact of PKI on BGP Security," PKI Research and Development Workshop, Gaithersburg, 2005.

[10] Wolf M. and Gendrullis, T., "Design, Implementation, and Evaluation of a Vehicular Hardware Security Module," ICISC, 2011.

[11] Hersteller Initiative Software, "SHE Secure Hardware Extension V1.1," 2009. Available: http://www.automotive-his.de

[12] Hoppe, T., Kiltz, S., and Dittmann, J., "Security Threats to Automotive CAN Networks Practical Examples and Selected Short-Term Countermeasures," Computer Safety, Reliability, and Security, 2008.

[13] B. A. Forouzan, Cryptography and Network Security. New York, NY: Mc Graw Hill, 2008

SESSION

CRYPTOGRAPHIC TECHNOLOGIES + HARDWARE SECURITY

Chair(s)

Dr. Levent Ertaul
Dr. Jiann-Shiun Yuan

A Comparison of HMAC-based and AES-based FFX mode of Operation for Format-Preserving Encryption

Levent Ertaul, Jalaj Neelesh Shah, Sofiane Ammar
California State University, East Bay, Hayward, CA, USA
levent.ertaul@csueastbay.edu, jshah22@horizon.csueastbay.edu, sammar@horizon.csueastbay.edu

***Abstract* — As usage and importance of smart phones and tablets grow, apps have come to dominate digital media. With limited computation capacity of mobile devices, performance plays a vital role in providing good user experience to the apps. This in conjunction with the recent security breaches leading to millions of stolen credit cards, makes it essential to ensure confidentiality while maintaining high performance. This paper presents performance comparison of AES (CBC) and HMAC (SHA-1) based PRFs for FFX mode of Format Preserving Encryption for a mobile app that functions as a credit card wallet.**

I. Introduction

Recent security breaches into various US retailers like Target [1], Home Depot and 7-11[2] not only indicate financial losses but also highlight the vulnerability of financial-information systems.

According to The Nilson Report 2013,[3] Credit Card Frauds around the world have grown from $2.5 billion to $7.5 billion in the last decade (as we can see in figure 1.1). While the growing trend continues this decade too, it has sharpened. Between 2010 and 2012 alone, there was a growth of $3.5 billion. It is only expected to grow in the coming years. This makes secure storage of Credit Cards or all cards for that matter even more important.

Figure 1.1 Global Card Fraud.[3]

Credit cards are stored in encrypted form. The encryption technique used i.e. Format-Preserving Encryption (FPE) [14] is slightly different than the regular encryption techniques. FPE encrypts plaintext of a particular length and format into ciphertext of the exact same length and format. For instance, encrypting a 16-digit Credit Card Number (CCN) using FPE would give a 16-digit number. FPE is a rapidly emerging cryptographic tool in applications like financial- information security in legacy databases. It becomes vital for structured data such as CCNs and Social Security numbers as the databases expect them to be in the exact same format and of exact same length for data-level encryption. As shown in figure 1.2, a regular encryption scheme like AES [15] would result in ciphertext of characters and varied length, FPE would give us ciphertext that would seem to look like a genuine CCN.

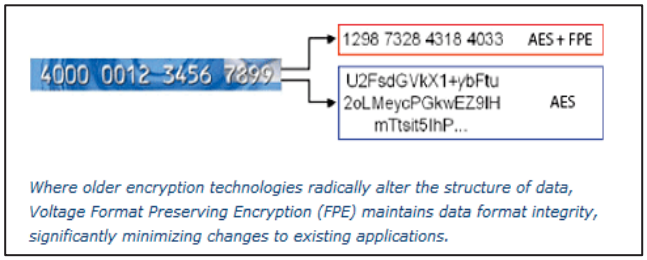


Figure 1.2 Format Preserving Encryption against Regular AES [4].

This also serves as a means to disguise intruders as it becomes difficult to distinguish between real CCNs and encrypted CCNs.

Wherever money is involved, security must be high and rightly so. Confidentiality is the most important thing while dealing with credit cards. In encryption mechanisms, it is generally true that increasing number of rounds increases quantitative security. While storing Credit Cards, we would ideally want as many numbers of rounds as possible, but, increasing the number of rounds would make the encryption process slower. Earlier, we said that performance is very important from a user point of view. Thus, a right balance has to be attained so that none is compromised.

Figure 1.3 shows that Mobile devices have out taken Desktops in terms of numbers globally. Smart phones and Tablets account for 60% of time spent on digital media in the US. The same report also suggests that it is 'usage of apps' that leads to this trend as 52% of this time is spent on apps alone. This also marks the decline of web dominance.

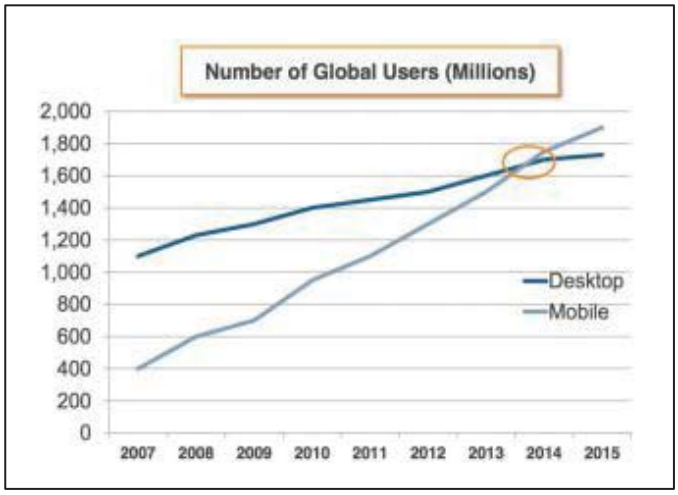


Figure 1.3 Number of Global Users of Digital Media.[5]

Smart phones and Tablets still do not match up with the computation capacity of desktops. For users to hang on to apps, good user experience is essential which can be provided if the app is fast enough. There are various apps like 'Google Wallet'[16] for storing Credit Cards and making transactions.

A lot of resources are put into such apps and as momentum shifts towards such e-payment systems, we reckon it is essential to have a good balance between security and performance. While there are no known weaknesses of FPE [13] (if parameters chosen correctly), not much work has been done to test performance of FPE. This paper compares various round functions for FFX mode[6] of FPE and presents quantitative results. These results would help in choosing the right round function so that the overall algorithm is fast.

In the section II, we look at the basic algorithm in which various parameters are given and the Encryption process is explained. Section III gives specifications of the implementation followed by the tests and results in section IV. Finally we lay out the conclusion.

II. ALGORITHM

A. *Mode of Operation*

We chose the FFX mode for FPE given by Bellare, Rogaway and Spies [6] as it is an extension to FFSEM [18] and supports tweaks that prevent dictionary attacks. FFX is defined as Format Preserving Feistel-based Encryption. The 'X' stands for parameter profile, which in our case is A10.

B. *Tweak*

Tweak's literal meaning is to alter or to modify. Tweak is defined as a set of unrelated mappings by the authors of FFX.[6] The idea behind tweaking is that while Issuer Identification Number (IIN)[12] for different Credit Card issuers ensures that the first few digits for each of them are different, the remaining digits can still be identical. This could lead to Dictionary attacks [17]. Thus, it is recommended to tweak some of the middle digits with the remaining ones.

In our algorithm, we tweak the middle eight digits with the starting four and the last four. We, however, do not use unrelated mappings to tweak. We use arithmetic and logical operations over the middle eight digits with the combination of first and last four digits. In this way, we believe we are bringing in more variation. For instance, for a Visa [22] card starting with 4, the original paper would only have one tweaked output per mapping, while with our tweak it could be anything from 0 to 9 depending on the fifth digit.

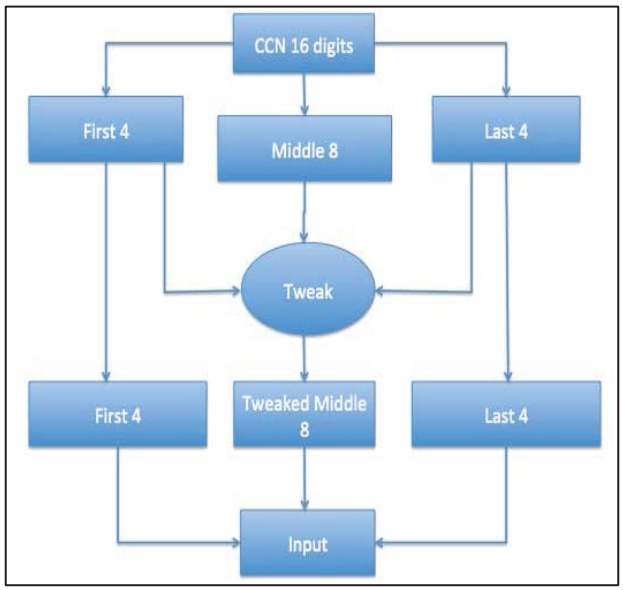


Figure 2.1 Working of Tweak

As we can see in figure 2.1, the tweaked middle eight digits are then combined together with the original first four and the last four. This together goes through the FFX.Encrypt (shown in figure 2.2).

C. *Round Function*

The Round Function that is basically a Pseudo Random Function (PRF) can be constructed from a Block cipher or a Hash Function. AES and HMAC [19] are recommended for Block cipher and hash function respectively [6]. We use CBC mode[20] for AES-based round function while SHA-1[21] for HMAC-based round function.

D. *Parameter Choices*

We use Parameter collection A10 as our implementation is based on 16 digit decimal numbers. Another set of parameters known as Parameter collection A2 is to be used for binary inputs.[6] Parameter Choices for A10 are given in Table I.

TABLE I. PARAMETER COLLECTION A10

Parameter	Choice
Radix	10
Key	128, 192, 256-bit keys
Addition	Blockwise
Method	1
Split	8
Rounds	12

E. FFX.Encrypt

The tweakedCCN, Tweak and the Key are then passed on to the main encryption function (figure 2.2). The tweaked CCN is split into two halves. The right half is hashed with SHA-1 based HMAC using a secret key. The hashed right half is then added (blockwise) to the left half. This becomes the right half for the next round while the right half of the last round becomes the left half of the next. The process goes on for twelve rounds until the two halves are finally merged.

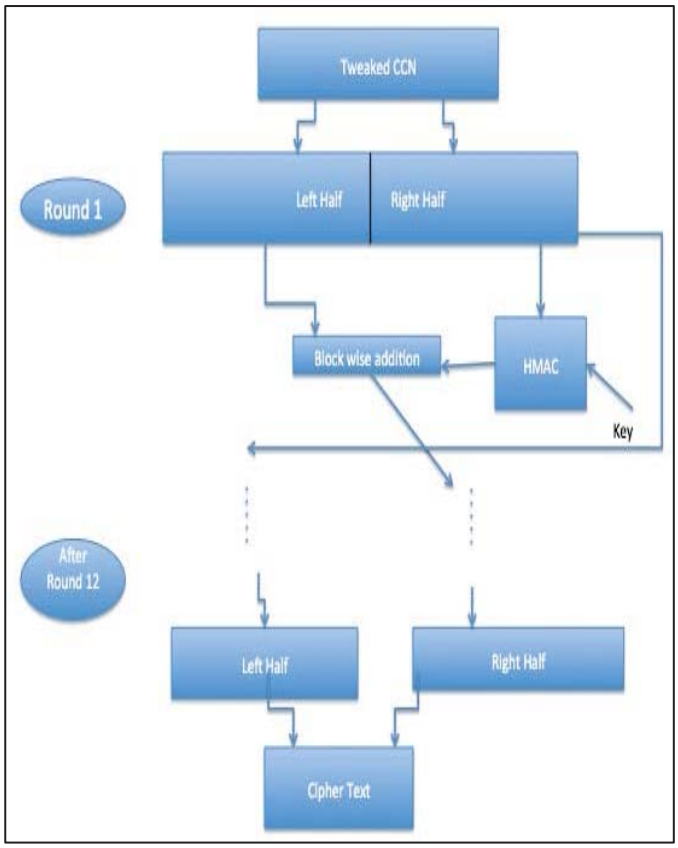


Figure 2.2 One complete cycle of unbalanced Feistel-based FFX.Encrypt

F. Cycle Walking

Cycle Walking is essential to FFX as it ensures that the ciphertext from FFX is of the desired format. With respect to our algorithm, this essentially means that if we assume that we want to encrypt an American Express [23] card. We know that the IIN for American Express is either '34' or '37'. In order to maintain its format, the ciphertext should also start with '34' or '37'. FFX alone cannot guarantee this. It has to be used in conjunction with Cycle Walking or Dense Encoding [6]. We choose Cycle Walking. As soon as FFX.Encrypt terminates, it is checked if the ciphertext falls within the set of VALIDCCN(X) that specifies the validity predicate. (For which American Express would be 34XXXX and 37XXXX). If yes, the algorithm terminates, otherwise FFX.Encrypt is called upon the result of the first cycle.

III. IMPLEMENTATION

A. Specification

The detailed Hardware and Software specifications are given in Table II and Table III respectively.

TABLE II. HARDWARE SPECIFICATION

Type	Specification
Type of System	64-bit Operating System
Processor	Intel® i5 Quad-Core 2.5GHz
Memory	4GB RAM

TABLE III. SOFTWARE SPECIFICATION

Type	Specification
Operating System	Windows 8
IDE	NetBeans 8.01[27], Android Studio[26]
Programming language	Java
Runtime Environment	JRE 6
Development Kit	JDK 1.7.0.45
Database	MySQL 6.1
Network Model	TCP/IP Client/Server Model
Crypto Library	Java.Security

B. Screenshots

We implemented a mobile wallet that can store encrypted credit cards. The screenshots are taken on Android Studio.

On launching the app, it would ask for a four-digit access pin (figure 3.1). It is done to avoid unauthorized access to the wallet. This four-digit pin can be set up at the time of installing the app.

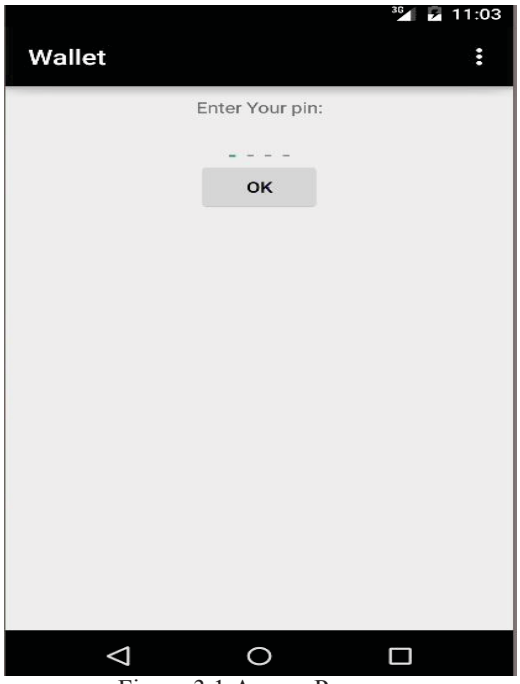


Figure 3.1 Access Page

If the pin is verified, the user is logged in (figure 3.2). The user can now see current credit cards that the wallet holds or the user can add a new card.

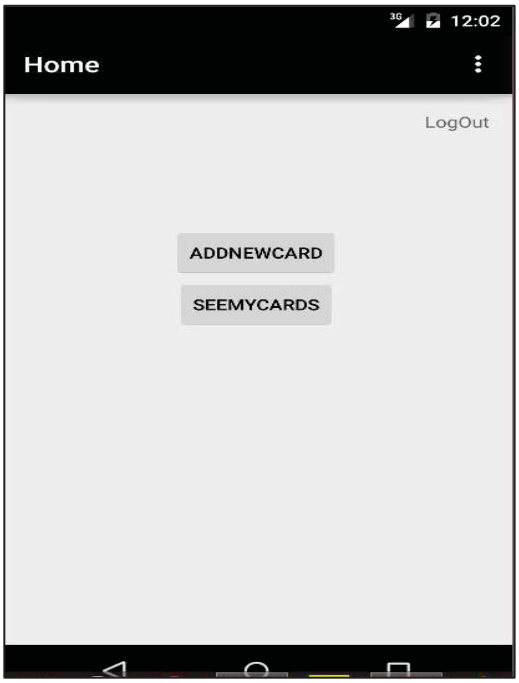


Figure 3.2 Home Page

Let's assume that the user taps on 'ADDNEWCARD.' The next screen will take the card details in the manner shown in figure 3.3.

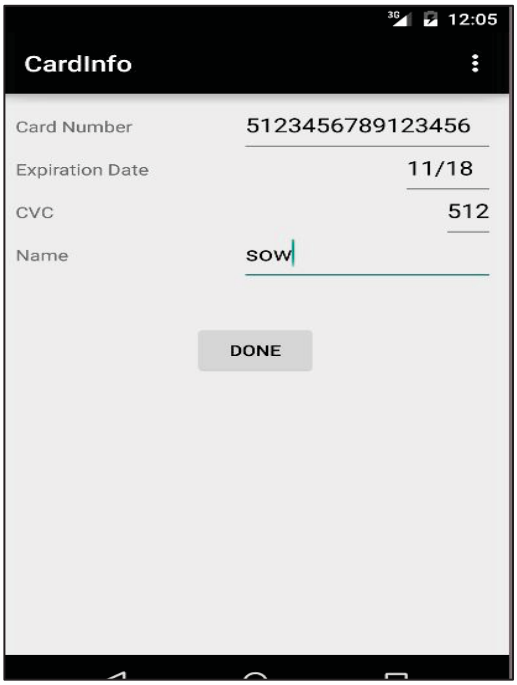


Figure 3.3 Add Card Details Page

For user convenience and ease in remembering, the user can nick name the newly entered card (figure 3.4).

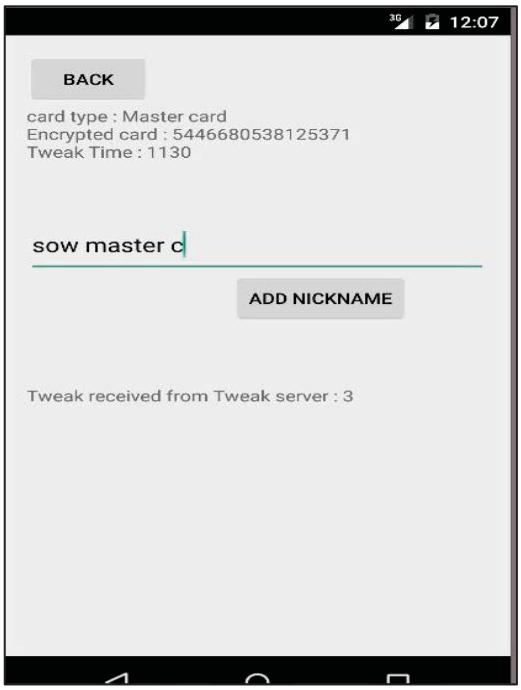


Figure 3.4 Nick Name page

The user is now taken back to the Home page from where current cards can be seen by tapping 'SEEMYCARDS' as shown in figure 3.5.

Figure 3.5 Stored Card Display Page

IV. TESTS AND RESULTS

For testing purpose, simulation of a sample size of 1000 or 4000 on android based mobile phone was not possible due to limited memory on mobile devices. The file containing log of CCNs could not be processed. Thus, we used CPU clock to time the performance of various combinations on the system specified in the above tables. We timed FPE only so as to get a precise measure of the performance of the algorithm itself by removing anomalies due to lag in Client-Server model and Database connections.

While we ran tests for big samples on Windows system having much more computation capabilities, we also ran test for very small sample sizes on Android Studio as well. Simulation on mobile emulator, showed no notable deviation from the performance seen on computer system. This could be due to the fact that the mobile device configured on the emulator did not have any other resources taken by the system.

A. *Comparison of AES CBC v. HMAC (SHA-1)*

The basic motivation of the paper was to find out that among the two round function candidates i.e. CBC mode of AES and HMAC, which one performs better. We used 256-bit key on a sample of 4000 credit cards. HMAC SHA-1 shows 67% better performance than AES CBC (Figure 4.1)

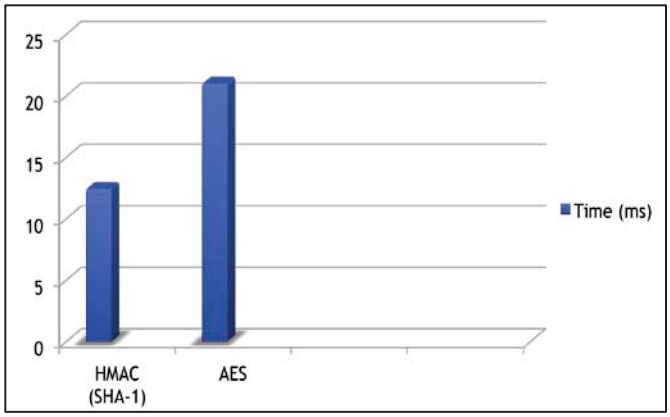


Figure 4.1 AES CBC v. HMAC (SHA-1)

B. *Comparison between different key sizes: 128-bit v. 192-bit v. 256-bit*

We can see in figure 4.2 that there is little difference in performance of SHA-1 when different key sizes are used. It is because of the change in number of cycles that each run took. On the first look of it, it gives an idea that changing key size affects the number of cycles. After several runs, we can conclude that variation in key size does not affect performance and that the number of cycles was completely random and independent of key size.

Performance figures according to benchmarks[9] suggest that as we increase key size for AES CBC, the performance deteriorates. However, in our tests the results (figure 4.3) were surprising. 192-bit key size showed dramatically good results. The random number of cycles again played a role in this.

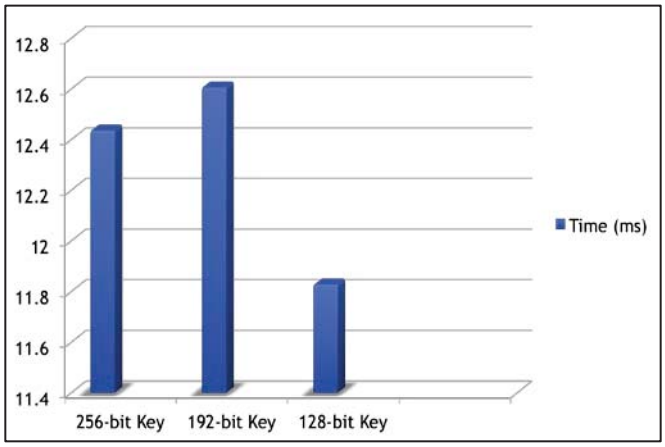


Figure 4.2 128-bit v. 192-bit v. 256-bit keys

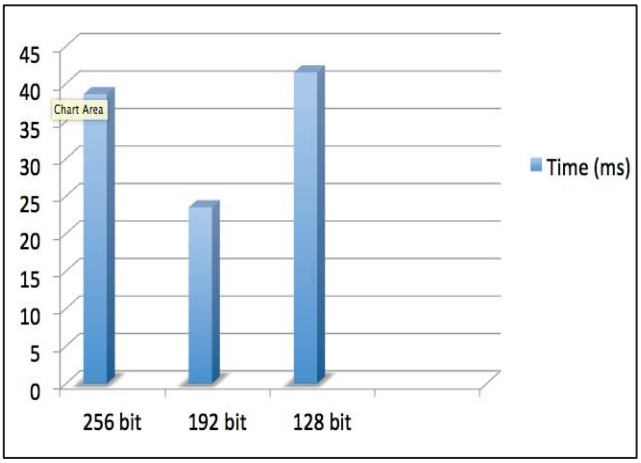


Figure 4.3 128-bit v. 192-bit v. 256-bit keys

C. Comparison based on Credit Card Issuers: Visa v. MasterCard[24] v. Discover[25] v. American Express

In order to look at the practical aspect of the implementation, we ran tests on different samples each limited to credit cards issued by a particular company. Typically one would assume that, the greater the fixed number of digits for a credit card, the higher the constraint on Cycle Walking, thus, the algorithm would go through more number of cycles. As a result, the number of fixed digits at the beginning of a CCN that vary as the issuer varies, alters the time that the encryption would take. We ran the tests through both round functions i.e. AES CBC and HMAC SHA-1.

As we can see in figure 4.4 and 4.5, the results are no different than expected. Visa shows the best performance as its Issuer Identification Number (IIN) is 4.[13] Just one condition has to be satisfied, thus, fewer cycles. With MasterCard the IIN is 51-55. With American Express, the IIN is '34' and '37.'[10] Thus, American Express takes more time and cycles as despite the equal number of conditions on MasterCard and American Express, the latter has a stricter choice between two digits only. While the IIN for Discover is '6011' [11], the four conditions take a toll on the performance of the algorithm that it practically crashed most times. Thus, we relaxed the Cycle Waking constraint to '60'.

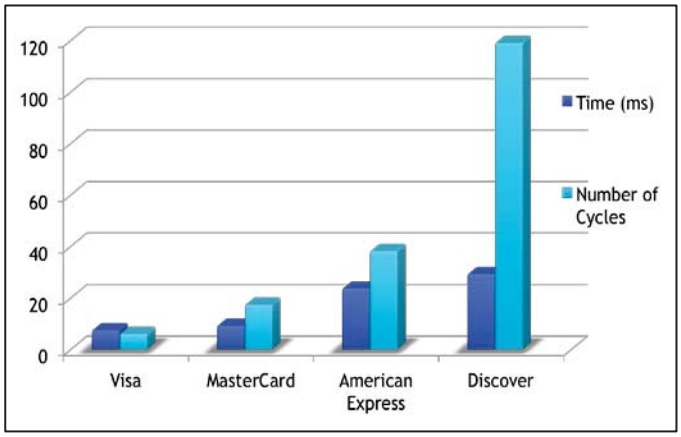


Figure 4.4 Visa v. MasterCard v. Discover v. American Express

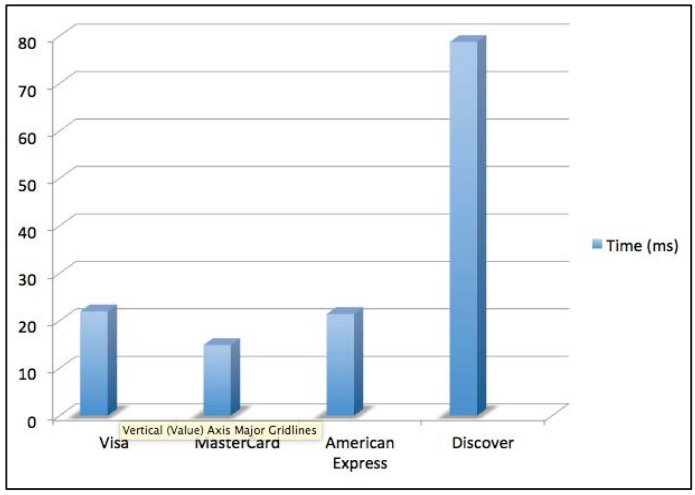


Figure 4.5 Visa v. MasterCard v. Discover v. American Express

D. Comparison of SHA-1 v. SHA-256[21] v. SHA-512[21]

It is proven that it takes a complexity of less than 80 to find collisions in SHA-1[7]. If Moore's law [8] holds still until mid-2020s, the computation power would be 2 times from what it is now. Thus, there is ample evidence why we need to migrate from SHA-1 to SHA-2 [21]. In alignment with this, we extended our tests to SHA-256 and SHA-512. As we can see in figure 4.6, SHA-512 performs better. According to the performance benchmarks [9], one would expect SHA-1 to be the fastest. However, SHA-1 based HMAC as a round function takes more number of cycles. While the difference between the number of cycles taken by SHA-256 and SHA-512 is not much, SHA-512 is much faster [9]. Although SHA-256 is slower (per round) than the rest, it takes lesser number of rounds and thus shows better performance than SHA-1.

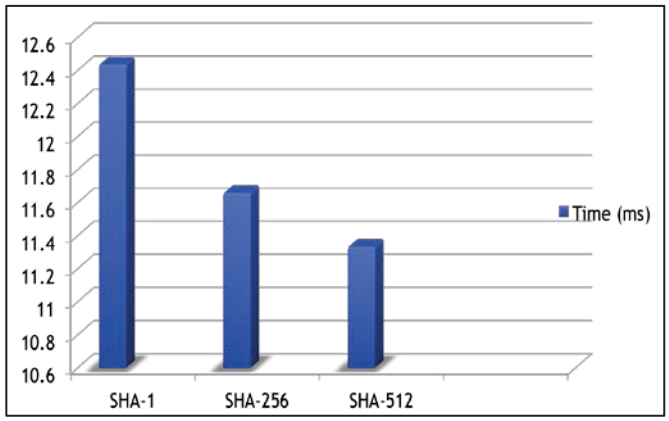


Figure 4.6 SHA-1 v. SHA-256 v. SHA-512

V. CONCLUSION

In this paper, we compared how different round functions for FFX line up in terms of performance. We also tested for different key sizes as migration from 128-bit keys to 256- bit keys has already initiated.

We also show how using different credit card companies

affect performance. Discover cards took a toll on performance as it took more cycles to give the ciphertext because of the longer IIN. Visa took the least number of cycles and thus least time.

While we use SHA-1 for HMAC-based round function, we also extend our implementation for SHA-256 and SHA-512.

We come to the conclusion that HMAC is a good candidate for FFX in terms of performance. It outruns AES by almost 67%. We recommend using SHA-512 for implementing HMAC as it shows promising performance and has fewer collisions as well.

We hope that more enhanced mobile wallets are launched in the future and the results presented in this paper help the designers.

VI. ACKNOWLEDGEMENT

We gratefully acknowledge the assistance and participation of Jil Trivedi, Sowjanya Kosaraju: Math and Computer Science, CSU East Bay. Jil and Sowjanya contributed towards implementation and testing of AES (CBC) based design.

VII. REFERENCES

[1] Sara Germano, Robin Sidel, Danny Yadron. "Target Faces Backlash After 20-Day Security Breach." The Wall Street Journal. 19 Dec 2013. Web. 19 Mar 2015.

[2] Dan Goodin. "TJX suspect indicted in Heartland, Hannaford breaches." The Register. 17 Aug 2009. Web. 18 March 2015.

[3] "Global Card Fraud." 2013 Nilson Report. Aug 2013. 18 March 2015.

[4] "Preserving Critical Business Functions by Maintaining Data Format" Voltage security.

[5] Adam Lella, Andrew Lipsman. "The U.S. Mobile App Report" comScore. 21 Aug 2014. Web. 18 March 2015.

[6] Mihir Bellare, Phillip Rogaway, Terence Spies. "The FFX Mode of Operation for Format-Preserving Encryption." NIST submission. 20 Feb 2010.

[7] Xiao Yun Wang, Yiqun Lisa Yin, Hongbo Yu. "Finding Collisions in Full SHA-1." NSFC Grant No. 90304009.

[8] Moore, Gordon E. "Cramming more components onto integrated circuits" (PDF). Electronics Magazine. 1965.

[9] "Crypto++ 5.6.0 Benchmarks". 1 April 2009. Web. 18 March 2015

[10] "Card Security Features" (PDF). American Express. January 2001.

[11] "Discover Network - IIN Range Update, 8.2" (PDF). September 2008.

[12] "Identification cards -- Identification of issuers." Part 1: Numbering system. ISO/IEC 7812-1:2006.

[13] Phillip Rogaway. "A Synopsis of Format-Preserving Encryption". 27 March 2010. University of California, Davis, CA, USA.

[14] John Black and Philip Rogaway, "Ciphers with Arbitrary Domains". Proceedings RSA-CT, 2002, pp. 114–130.

[15] Joan Daemen, Vincent Rijmen, "The Design of Rijndael: AES – The Advanced Encryption Standard". Springer, 2002. ISBN 3-540-42580-2.

[16] "Coming soon: make your phone your wallet". Official Google Blog. May 26, 2011. Retrieved April 22, 2015.

[17] R. Shirey. "Internet Security Glossary". May 2000. RFC 2828.

[18] Terence Spies. "Feistel Finite Set Encryption Mode". NIST.

[19] H. Krawczyk, M. Bellare, R. Canetti. "HMAC: Keyed-Hashing for Message Authentication". February 1997. RFC 2104.

[20] NIST Computer Security Division's (CSD) Security Technology Group (STG) (2013). "Block cipher modes". Retrieved April 22, 2015.

[21] NIST. "Secure Hash Standard (SHS)". FIPS PUB 180-4.

[22] "Visa Inc.". visa.com. Retrieved April 22, 2015.

[23] "American Express". Americanexpress.com. Retrieved April 22, 2015.

[24] "MasterCard". Mastercard.com. Retrieved April 22. 2015.

[25] "Discover Financial". discover.com. Retrieved April 22, 2015.

[26] "Android Studio". developer.android.com. Retrieved April 22, 2015.

[27] "Net Beans". netbeans.org. Retrieved April 22, 2015.

Block-DCT Based Secret Image Sharing over GF(2^8)

Rosemary Koikara[1], Mausumi Goswami[1], Pyung-Han Kim[2], Gil-Je Lee[2], Kee-Young Yoo[2]

[1]*Computer Science and Engineering, Christ University, Faculty of Engineering*
Bangalore 560074, Karnataka, India

[2]*School of Computer Science and Engineering, Kyungpook National University*
80 Daehakro, Bukgu, Daegu 702-701, Republic of Korea

Abstract— *In this paper, we are concerned with securing secret information in the form of a secret image in such a way that the secret image is shared among the participants and no one share gives information about the secret hidden. Secret image sharing is a method of sharing secret message among multiple cover images, making it difficult to trace the message. We concentrate on performing secret image sharing in the frequency domain. Here, secret image sharing is done on grayscale images using Block-DCT (Discrete Cosine Transform). This method has the advantage of DCT-based data hiding schemes, i.e., since the secret data is embedded into DCT coefficients of the cover images. In the proposed scheme we improve Koikara et al.'s scheme by using operations over the GF(2^8) to share the secret image and also to reconstruct it back. In our method the security of the secret information is maintained and the quality of the stego image and secret image is improved.*

Keywords: Secret Image Sharing, Block-DCT, Image Security**.**

1 Introduction

Security threats have always been a concern when it comes to transferring information over the Internet. Over the past 3 decades several approaches have been proposed to protect secret messages being passed in the network [1]. Cryptographic algorithms and protocols have been developed to secure data. Most of these algorithms involve transformation of information into unrecognizable data. Some examples of these transformation algorithms are RSA [2], DES [3], etc. and are called encryption algorithms. Encryption algorithms generally encrypt data with a key which a receiver will not be able to transform the data back into recognizable form without it. Though cryptographic keys can be used for protecting data we need efficient key management schemes to protect the key [4]. The issue with key management scheme is that most of them keep the key in one location. Hence, the key may become inaccessible due to some misfortune. Therefore, threshold schemes are needed due to the mentioned reason.

In 1979, Shamir [4] and Blakley [5] introduced the idea of secret sharing. Shamir proposed a (k, n)-threshold scheme which divides the secret data into n independent shares such that the secret data can be reconstructed using k or more shares. His scheme is based on polynomial arithmetic operation and Lagrange's interpolation. The goal was to take k points, and it is guaranteed that a unique polynomial $f(x)$ with those k points would exist such that $f(x) = y$. Blakley's scheme was based on hyperplane intersections instead of polynomial interpolation. The proposed scheme uses Shamir's (k, n)-threshold scheme as it is more efficient than the one developed by Blakley [4].

There have been many methods proposed for secret image sharing based on Shamir's scheme. In 2002, Thien and Lin [6] first proposed the use of Shamir's (k, n)-threshold sharing scheme for sharing images. In this scheme the size of the generated shares is only 1/k of original image. But, the disadvantage is that there is a loss of information as the grayscale values were limited to a range of {0, 250}. So, a process was included to remove this loss, but this in turn increased the shadow size. Later in 2006, Bai [7] proposed a method similar to Thien and Lin's method which uses a combination of matrix projection and Shamir's method. Though the share size in this case was significantly less than the size of the secret image, there was still an increase in the size of the shares. In 2010, Alharthi and Atrey [8] proposed an *improvement* over Thien and Lin by reducing the computing time. Though this scheme reduces the time complexity and the security, it still has Thien and Lin's disadvantage of the share size being larger. In 2015, Koikara et al. [9] proposed a scheme that uses Shamir's scheme [4] by sharing images in cover images that have been transformed to the frequency domain using block-DCT.

In this paper, we improve Koikara et al.'s scheme [9] using a Galois field (GF) polynomial arithmetic operation over GF(2^8). The proposed scheme is based on Shamir's (k, n)-threshold scheme. Shamir's secret sharing scheme has been used for images by researchers because of the utility and high security it provides. But we carry out the sharing and reconstruction procedures in the frequency domain using block-DCT based transformation to embed the secret data because performing the data hiding in the frequency domain increases the security of the secret information. As a result, we can conclude that the quality of the stego images are superior to the previous work.

This paper is organized as follows. Section 2 explains some related works in the field of secret image sharing. Section 3 describes the proposed scheme. The experimental results are shown in Section 4. Section 5 gives the conclusion of this paper.

2 BACKGROUND

There are different techniques that have been proposed for sharing a secret image over a set of cover images. Some techniques perform it in the spatial domain while others in the frequency domain. A simple way of embedding the shadow image into the respective cover image is the use of least significant bit (LSB) substitution. In certain cases the LSBs are randomly visited and in certain cases the pixel values are incremented or decremented.

In this section, we briefly review the related works and introduce the basic concepts.

2.1 Discrete Cosine Transform

Discrete Cosine Transform transforms an image into the frequency domain and removes the sine component of the image. For 2-dimensional DCT we first divide the image into blocks of size 8 × 8 pixels each. Then we perform the 2-D DCT as given in Eq. (1) on each of these blocks.

$$F(u,v) = \frac{C(u)C(v)}{4}\sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y) \cos\left[\frac{\pi(2y+1)v}{16}\right]\cos\left[\frac{\pi(2y+1)v}{16}\right] \quad (1)$$

,

$$\text{where } C(e) = \begin{cases} \frac{1}{\sqrt{2}}, & if\ e = 0 \\ 1, & if\ e \neq 0 \end{cases}$$

Here, $F(u, v)$ and $f(x, y)$ represent a DCT coefficient at coordinate (u, v) and pixel value at (x, y) respectively.

Eq. (2) gives the mathematical expression of inverse DCT.

$$f(x,y) = \frac{1}{4}\sum_{u=0}^{7}\sum_{v=0}^{7} C(u)C(v)F(u,v) \cos\left[\frac{\pi(2x+1)u}{16}\right]\cos\left[\frac{\pi(2x+1)v}{16}\right] \quad (2)$$

,

$$\text{where } x = 0,\ldots,7 \text{ and } y = 0,\ldots,7$$

An 8 × 8 block is used to apply DCT because of basically the following major reasons:

1) *Many experiments were performed for various block sizes and 8 × 8 gave the best results.*

2) *It is more complex to perform DCT on matrices of sizes greater than 8 × 8.*

3) *Matrices of size less than 8 × 8 do not retain enough information to continue along the pipeline.*

2.2 Galois Field

We need to perform modular arithmetic for the sharing of shares instead of real arithmetic so that we have a field in which interpolation is possible. Hence, we use the modular operation with a prime number. We can also use Galois Field arithmetic operation for this.

From [10] we come to know that the order of a finite field must be a power of a prime p^n, where n is a positive integer. The finite field of order p^n is generally written as GF(p^n) and it stands for Galois field. A characteristic 2 finite field with 256 elements is called Galois Field GF(2^8). In GF(2^8), we use an irreducible polynomial $m(x)$ as Eq. (3) to compute the modular operation.

$$m(x) = x^8 + x^4 + x^3 + x + 1 \quad (3)$$

2.3 Shamir's (k, n)-threshold Secret Sharing Scheme

Adi Shamir [4] introduced a (k, n)-threshold secret sharing scheme and devised a scheme to divide a secret data D into n shares, $D_1, D_2, \ldots, D_n$ in such a way that:

1) *Any k or more D_i pieces makes D easily computable.*

2) *Any $k-1$ or fewer D_i pieces leaves D completely undetermined.*

Suppose, (x_1, y_1), (x_2, y_2), …, (x_k, y_k) be k points in the 2-dimensional plane such that all the x_i's are distinct and data D is a number; and we need to divide it into n shares. We can do it using Eq. (4).

$$q(x) = s + a_1 x + \cdots + a_{k-1}x^{k-1} \pmod p \quad (4)$$

where, $s = D$ and $a_1, a_2, \ldots, a_{k-1}$ are $k-1$ randomly chosen integers from a uniform distribution over the integers in $[0, p)$.

Now we need a minimum of $k-1$ of these n shares for the secret to be extracted. The coefficients of $q(x)$ can be found out using interpolation, and $q(0)$ will be the secret, D.

2.4 Thien and Lin's Secret Image Sharing

Thien and Lin [6] proposed a secret image sharing scheme based on Shamir's (k, n)-threshold sharing scheme. Unlike Shamir's scheme, a random coefficient is not used in the polynomial equation for creating the secret shares. The coefficients are pixels from the secret image. Since grayscale values of pixels have the range {0, 255}, the prime number p is taken as 251, as it is the greatest prime between 0 and 255.

The polynomial for each share can be represented by Eq. (5).

$$q(x) = a_0 + a_1 x + \cdots + a_{k-1}x^{k-1} \pmod{251} \quad (5)$$

Where, $a_0, a_1, \ldots, a_{r-1}$ are r pixels from the secret image that have not been shared yet. This is a form of multi-secret sharing. The size of each shadow image is 1/k of the secret image.

For the reconstruction of the secret image they use k of the n shares in Lagrange's interpolation as given in Eq. (6).

$$f(x) = \sum_{i=1}^{k} y_i \prod_{\substack{1<i<k \\ i\neq j}} \frac{x - x_j}{x_i - x_j} \pmod p \quad (6)$$

Where $y_i = q(x)$ the participant's share value. Their scheme is a lossy method because the grayscale value is limited to the range {0, 250}. So, a process can be used to

handle grayscale values larger than 250. But this increases the size of the shadow image.

2.5 Review of Koikara et al.'s scheme

In 2015, Koikara et al. [9] proposed a novel secret sharing algorithm in the frequency domain. It is based on the (k , n)-threshold scheme and has two phases: embedding and reconstruction phases.

Embedding of the secret is done in the frequency domain by using Block-DCT. The polynomial equation used in this scheme is:

$$q_j(x) = a_0 + a_1 x + \cdots + a_{k-1}x^{k-1} \pmod{256} \tag{8}$$

$$q_j(x) = \left\lfloor \frac{a_0 + a_1 x + \cdots + a_{k-1}x^{k-1}}{256} \right\rfloor \tag{7}$$

where $\lfloor \quad \rfloor$ indicates the floor operation

In Eq. (7) and (8), j is the section of secret image for which the polynomial is generated. Each section, j has k pixels. The coefficients a_0 to a_{k-1} are k pixel's intensity values from the secret image. The value of x is changed according to the share. Here, 256 is used so as to prevent loss of information. When we use p = 251 in Eq. (5) a loss in information is there for pixels greater than intensity value 251. But as 256 is not a prime number, both the remainder value and the quotient value has to be embedded. A parity check is performed to reduce the round-off error that occurs. But this parity check does not eliminate the round-off error, it only reduces it. The round-off error occurs during the processing of the cover images.

Their scheme used the floor and modular operation on integers. Hence it has a few disadvantages. Since both the floor values and the modular values are stored in the share there is an increase in the size of each share. Also there will be additional complexity during the secret sharing and the secret restoration phases.

3 PROPOSED SCHEME

In this section, we propose an improved scheme for secret sharing in the frequency domain using modular operation over GF(2^8). This has various advantages. There is a significant decrease in the share size of each shadow image. Since there are fewer number of operations there will also be a reduction in the complexity of the algorithm.

In our scheme we use a (k, n) threshold scheme similar to Shamir's. The polynomial equation can be mathematically expressed by Eq. (9) with irreducible polynomial $m(x)$ as Eq. (3).

$$f(x) = s_0 + s_1 x + \ldots + s_{k-1}x^{k-1} \left(\mathrm{mod\ GF}(m(x))\right) \tag{9}$$

In Eq. (9) $s_0, s_1 \ldots s_{k-1}$ are k pixels from the secret image that are yet to be shared. The value of x determines the share. Hence, we use a multi-secret sharing scheme as in Thein and Lin's scheme in which the polynomial equation has more than one secret. All the coefficients in the polynomial equation represent a secret pixel. Using Eq. (9) we will obtain n shares that will be used to embed into the cover image. As with the other threshold schemes even our scheme does not require all the n shares for extraction of the secret image. A minimum of k shares are required for reconstructing the secret image. The k shares have to be valid shares. Once the secret information is extracted from each of the shares, the Lagrange's interpolation as given by Eq. (10) is used on the secret image.

$$f(x) = \sum_{i=1}^{k} y_i \prod_{\substack{1<i<k \\ i\neq j}} (x \oplus x_j)(x \oplus x_j)^{-1} \left(\mathrm{mod\ GF}(m(x))\right) \tag{10}$$

In Eq. (10), $\oplus$ indicates the bitxor operation and $(a)^{-1}$ implies the multiplicative inverse of a. Fig. 1(a) shows the basic block diagram of the secret sharing module and Fig. 1(b) shows the basic block diagram of the secret restoration module.

In the following subsections we describe the algorithms used in our proposed scheme. Let $C_1, C_2 \ldots C_n$ be n $M \times N$ cover images and S be a $P \times Q$ secret image such that we have n participants. Let $S_1, S_2, \ldots, S_n$ be n $M \times N$ stego images. Also, let k be the threshold and S' be the extracted secret image.

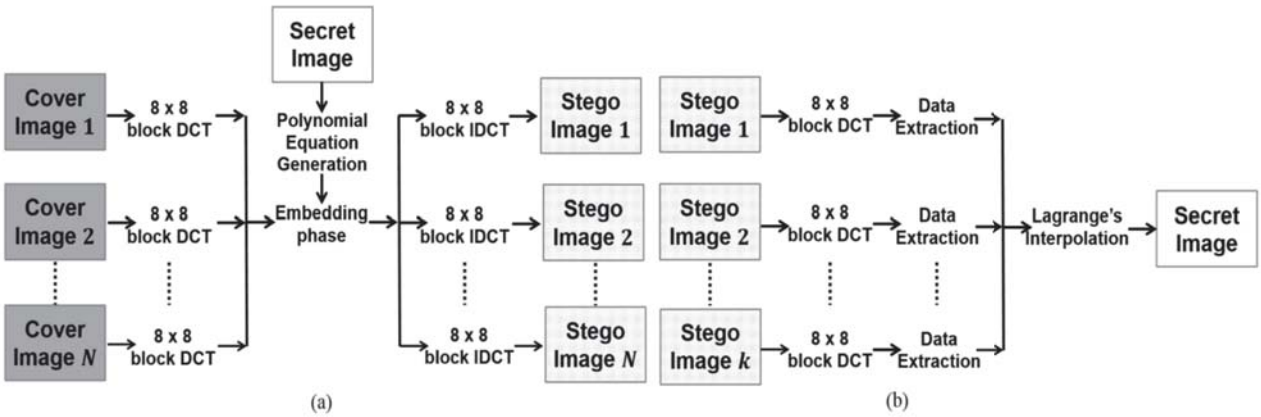


Fig. 1 Schematic Diagram: (a) Secret Sharing Module and (b) Secret Reconstruction Module

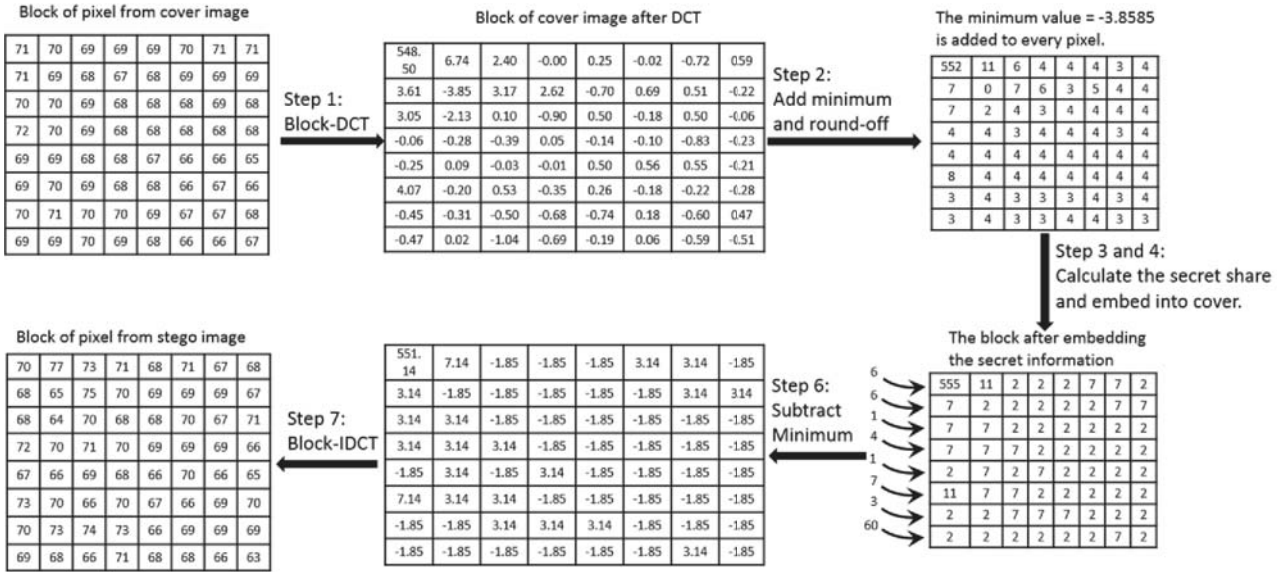


Fig. 2 An example of a secret sharing phase for a 8 × 8 block of one cover image

3.1 Sharing Algorithm

In this subsection, the sharing algorithm is described.

Input: n cover images $C_1, C_2,..., C_n$ of size $M \times N$ and a secret image SI of size $P \times Q$.

Output: n stego images $S_1, S_2, ..., S_n$ of size $M \times N$.

Step 1: Divide each cover image $C_1, C_2,..., C_n$ into non-overlapping blocks of size 8 × 8 pixels and apply 2-D DCT to each of these blocks. This is done by using the formula given in Eq. (1) in each one of the 8 × 8 blocks.

Step 2: Calculate the minimum value in each cover image and add it to each pixel of the cover images. This is done to avoid numbers less than 0. Now, round off-the pixels.

Step 3: Take k number of not yet shared pixels from SI and use the polynomial equation given in Eq. (9) to find n shares.

Step 4: Embed the n shares found in Step 2 into the 3rd LSBs of corresponding DCT transformed $C_1, C_2, ..., C_n$.

Step 5: Add the parity bit information to reduce the error due to round-off. When we perform DCT on a block of pixels we get floating point numbers which are truncated. Hence, there is a loss of information. Parity check reduces this loss of information to some extent. Let x be the pixel to which parity information must be added. Now we use the following substeps:

5.1: Calculate the even parity of x and embed it into the 1st LSB position.

5.2: Calculate the odd parity of x and embed it into the 2nd LSB position.

Step 6: Repeat Step 3, Step 4 and Step 5 until all pixels of SI are embedded into $C_1, C_2, ..., C_n$.

Step 7: Obtain n stego images $S_1, S_2, ..., S_n$ by performing inverse DCT on the new $C_1, C_2, ..., C_n$. IDCT is also done by dividing the images into non-overlapping blocks of size 8 × 8 pixels. The equation for IDCT is given in Eq. (2).

Fig. 2 shows the example of secret sharing for a 8 × 8 block for cover image 1. Suppose, for a (4, 5) threshold scheme, $k = 4$ and $n = 5$. If in Step 3 the 4 not yet shared pixels are a_0=156, a_1=159, a_2=158, a_3=155. Then, the polynomial equation is $f(x) = 156 + 159x + 158x^2 + 155x^3 (mod\ \mathrm{GF}(2^8))$. The 5 shares are calculated as follows, $f(1) = 6, f(2) = 67, f(3) = 174, f(4) = 34$ and $f(5) = 145$. From Fig. 2 after Step 2 we have the 1st pixel of the cover as 552. Now we need to insert the 1st bit of $f(1)$ to the 3rd LSB of 552. After embedding it remains the same. For the parity check in Step 5, $x = (552)_{10} = (1000101000)_2$. Then x can be changed as follows, $x = (11000011)_2 = (555)_{10}$.

3.2 Reconstruction Algorithm

In this subsection, the data extraction algorithm for reconstructing the secret image is given.

Input: k stego images, $S_1, S_2, ..., S_k$ of size $M \times N$

Output: Extracted secret image S'.

Step 1: Divide the k stego images, $S_1, S_2, ..., S_k$ into non-overlapping blocks of size 8 × 8 pixels and apply 2-D DCT on each of these blocks. This is done by using the formula given in Eq. (1).

Step 2: Calculate minimum value from the stego image and add it to all the pixels of the stego image and round-off the pixels.

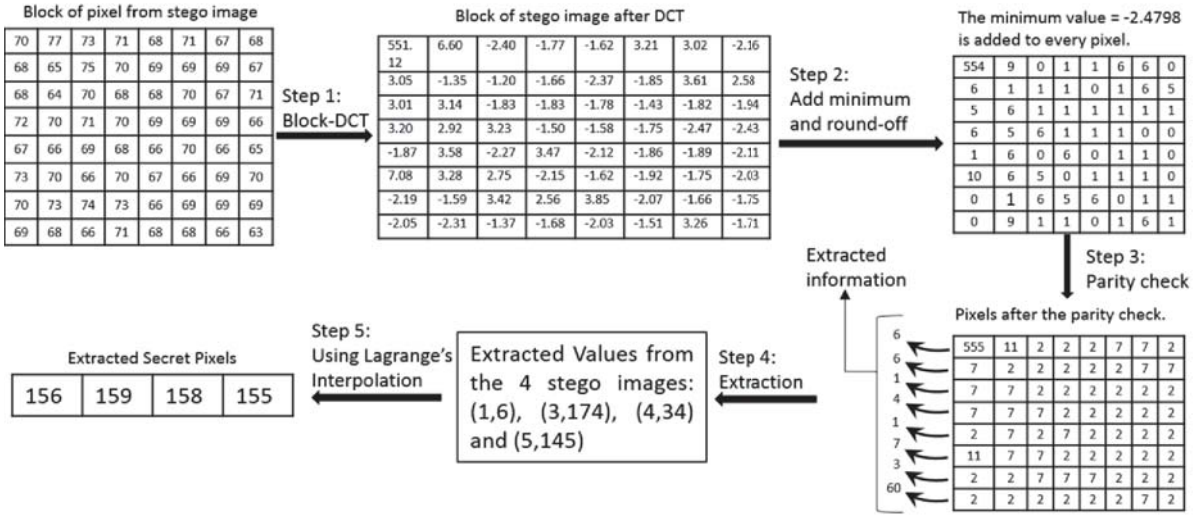


Fig. 3 An example of a secret reconstruction phase for an 8 × 8 block of one stego image.

Step 3: Take eight pixels from each of $S_1, S_2, \ldots, S_k$ and perform parity check for each of the 8 pixels. Let be one pixel, then parity is performed the following sub-steps:

- 3.1 Calculate the even parity of *x* and embed it in *x's* 1st LSB
- 3.2 Calculate the odd parity of *x* and embed it in *x's* 2nd LSB.
- 3.3 The computed *x* is called *x'*.
- 3.4 If $x = x'$ then it is correct else increment or decrement the *x* and go to Substep 3.1.

Step 4: Extract the 3rd LSB of each of the parity checked pixels to get *k* shares of the *k* pixels of the secret image.

Step 5: Use above *k* pixels in Lagrange's interpolation to retrieve *k* pixels of the secret image. The equation of Lagrange's interpolation is given in Eq. (10).

Step 6: Repeat Step 3, Step 4 and Step 5 until all the pixels of the secret image S' are processed.

Fig. 3 shows the reconstruction phase for 8 × 8 pixels block of stego image 1. Suppose, after Step 2 we have the first pixel of the block as 554. So, for Step 3 we have the first pixel of block as 554. So, for Step 3 we have if $x = (554)_{10} = (1000101010)_2$, then we have $x' = (1000101010)_2 = (555)_{10}$. Hence, we get back the secret bit from the 3rd LSB.

4 EXPERIMENTAL RESULTS

In this section some of the results of experiments carried out are given in order to evaluate our scheme. For the evaluation we measured the quality of the stego images when compared to the respective cover images.

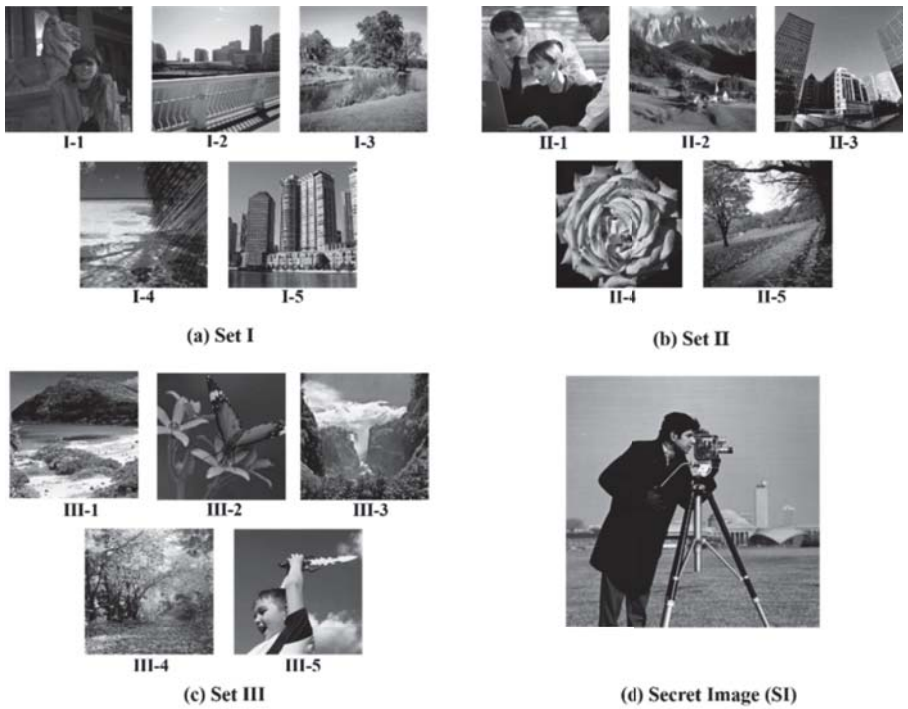


Fig. 4 Cover Image (a) Set I (b) Set II (c) Set III and (d) Secret Image SI

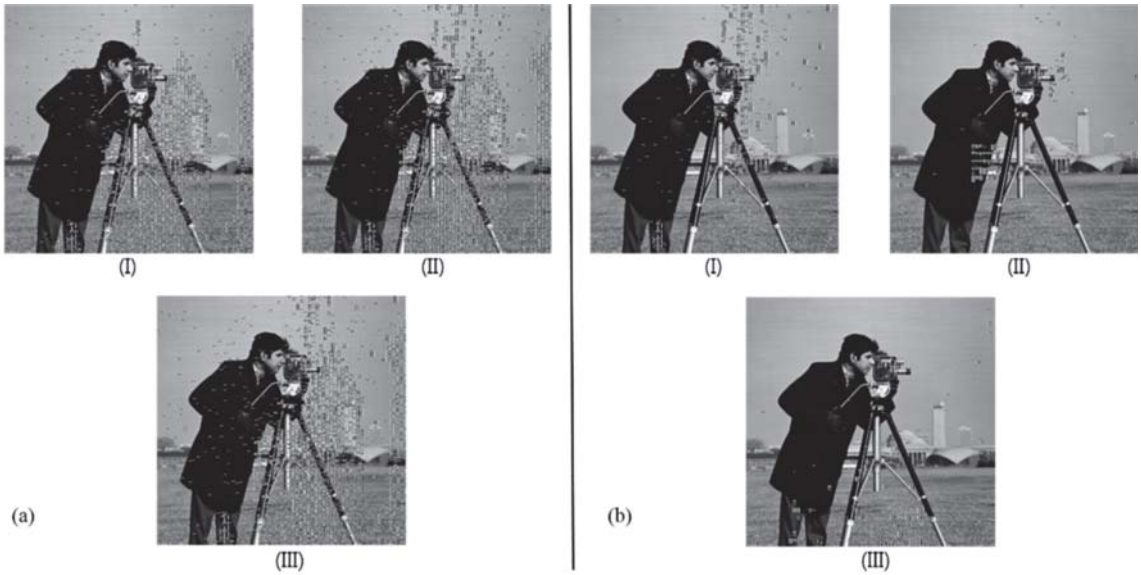


Fig. 5 Reconstructed Secret Image for set I, set II and set III using (a) Koikara et al.'s Scheme and (b) Proposed Scheme.

This comparison was carried out using the peak-signal-to-noise rate/ratio which we further refer to as *PSNR*. The

PSNR is defined as follows,

$$PSNR = 10 \times \log\left(\frac{255^2}{MSE}\right) \tag{11}$$

The mean square error (MSE) of an image of size $M \times N$ pixels is defined as follows,

$$MSE = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\frac{(f(x,y) - g(x,y))^2}{M-N} \tag{12}$$

where $f(x, y)$ is the pixel intensity value at (x, y) of the cover image, i.e., the original pixel value of $g(x, y)$ is the pixel intensity value at (x, y) of the stego image. The *PSNR* value is expressed in dB. The higher the *PSNR* value, the higher is the quality of the stego image when compared to the cover image. But, a low value of *PSNR* indicates higher distortion. When the *PSNR* value is less than 30dB the image undergoes a lot and losses significant information.

We evaluate the scheme by using 3 sets of images. The three sets of cover images used are shown in Fig. 3. The results for (4, 5) threshold scheme is evaluated in this section. The resulting secret image for Koikara et al.'s scheme and our proposed scheme is given in Fig. 5(a) and Fig. 5(b) respectively. We compared the results of Koikara et al.'s scheme in the frequency domain with the new one. In Table I, we compare the *PSNR* values of the stego images. The *PSNR* value of stego images using both Koikara et al.'s [9] and the proposed scheme is given. By observing Table I we notice that there is an increase in the *PSNR* values of the stego images. This increase implies that there is an improvement in the quality of the stego images. The *PSNR* values of all stego images are more than 35dB hence they are acceptable. We also have Table II which gives us the comparison of *PSNR* values of the extracted secret image. From Table II, we notice that there is a significant increase in the quality of the secret images when compared to Koikara et al.'s scheme. This is due to the fact that in Koikara et al.'s field there was a greater margin for error since both the floor as well as mod values had to be embedded. But in the proposed scheme just one value i.e., the mod value is stored. To make sure that the amount of data embedded in both the schemes are same we embed random bits in the pixels of the stego images into which secret has not been embedded. This is done because the secret shares are not big enough to be embedded into the entire cover images. As the data hiding is done in the frequency domain the choice of the cover image used plays a huge role in this scheme.

TABLE I
PSNR COMPARISON OF STEGO IMAGE FOR (4,5) -THRESHOLD SCHEME

Cover Image Set	Stego Image -1		Stego Image -2		Stego Image -3		Stego Image -4		Stego Image -5	
	Previous	Proposed	Previous	Proposed	Previous	Proposed	Previous	Proposed	Previous	Proposed
Set I	48.07	50.97	44.69	47.51	50.20	51.75	47.52	49.67	47.23	49.89
Set II	46.23	48.99	48.44	50.43	49.87	51.48	45.16	47.77	46.87	49.28
Set III	47.22	49.72	47.17	49.14	46.79	48.78	47.78	50.07	46.24	48.84

TABLE II
PSNR COMPARISON OF EXTRACTED SECRET IMAGE

PSNR (dB) Extracted Secret Image		
Secret Image Set	Previous Scheme	Proposed Scheme
(I)	34.61	40.86
(II)	34.02	42.77
(III)	34.04	43.78

TABLE III
COMPARISON OF EMBEDDING CAPACITY OF COVER IMAGE

Embedding Capacity (No. of bits) per Stego Image		
Cover Image Size	Previous Scheme	Proposed Scheme
256 × 256	8192	8192
512 × 512	32768	32768
1024 × 1024	131072	131072

5 CONCLUSION

We have proposed the (t, n)-threshold secret sharing in the frequency domain using 2D-DCT and performing operations over GF(2^8). Koikara et al.'s scheme was done using integer arithmetic. The main aim of the proposed scheme is to improve the quality of the stego image when compared to Koikara et al.'s scheme as well as to improve the extracted secret image. From Table II we can conclude that the *PSNR* value of the extracted secret image has increased by roughly 6dB to 8dB. As we observe in Table III the embedding capacity of the stego images have been maintained. All the operations were done using GF(2^8) to prevent any loss of information that may take place. When we perform secret sharing in the frequency domain the security of the information hidden increases as the embedding is done into the DCT coefficients of the cover image and not directly into the pixels of the cover image.

While using this particular scheme we must be vary about the type of cover image used. The future works in secret sharing in the frequency domain would be to make it more generalized, such that a set of any cover images may be used for sharing the secret.

6 ACKNOWLEDGEMENTS

This research work was supported by an academic student exchange program between Christ University, India and Kyungpook National University, Republic of Korea signed on 21st February 2013 and the IT R&D program of MSIP/IITP. [10041145, Self-Organized Software platform (SoSp) for Welfare Devices]].

7 REFERENCES

[1] Imai, Hideki, et al. "Cryptography with information theoretic security." *Information Theory Workshop, 2002. Proceedings of the 2002 IEEE*. IEEE, 2002.

[2] R. L. Rivest, A. Shamir and L. Adelman. "A Method for Obtaining Digital Signatures and Public-key Cryptosystem." *Communications of the ACM*, col. 21, no. 2, pp. 120-126, Feb. 1978

[3] R. L. Rivest, A. Shamir and L. Adelman. "A Method for Obtaining Digital Signatures and Public-key Cryptosystem." *Communications of the ACM*, col. 21, no. 2, pp. 120-126, Feb. 1978.

[4] W. Diffie and M. E. Hellman. "Special feature Exhaustive Cryptanalysis of the NBS Data Encryption Standard." *IEEE computers*, vol. 10, pp. 74-84, 1977.

[5] A. Shamir. "How to share a secret." *Communications of the ACM*, vol. 22, no. 11, p p. 612-613, 1979.

[6] G. R. Blakeley. "Safeguarding cryptographic keys", in Proc. AFIPS National Computer Conf., vol. 48, pp. 313-317, 1979.

[7] C. C. Thien, J. C. Lin. "Secret image sharing." *Computer & Graphics*, vol. 26, no. 1, pp. 765-770, 2002.

[8] L. Bai, S. Biswas and P. E. Blasch, "An Estimation Approach to Extract Multimedia Information in Distributed Steganographyic Images." in *Proc. 10th International Conference on Information Fusion*, IEEE, 2007.

[9] S. Alharthi and P. K. Atrey. "An improved scheme for secret image sharing." In *IEEE ICME Workshop on Content Protection and Forensics*, July 2010.

[10] R. Koikara, D. J. Deka, M. Gogoi and R. Das. "A Novel Distributed Image Steganography Method Based on Block-DCT." in *Advanced Computer and Communication Engineering Technology, Lecture Notes in Electrical Engineering 315*. Springer International Publishing, 2015, pp. 423-435.

[11] F. N. Johnson and S. Jajodia. "Exploring Steganography: Seeing the Unseen." *Computer*, vol.31, pp. 26-34, Feb. 1998.

[12] R. Das, T. Tuithung. "A Review on "A Novel Technique for Image Steganography Based on Block-DCT and Huffman Encoding"." in *Proc. of the 4th International Conference on Computer Graphics and Image Processing*, ICGIP-1012, SPIE, 2012.

[13] A. Cheddad, J. Condell, K. Curran, M. P. Kevitt. "Digital Image Steganography: Survey and Analysis of Current Methods." *Signal processing*, vol. 90, pp. 727-752, 2010

[14] B. Ki, J. He, J. Huang, Y. Q. Shi. "A Survey on Image Steganography and Steganalysis." *Journal of Information Hiding and Multimedia Signal Processing*, vol. 2, no. 2, 2011.

[15] N. Provos, P. Honeyman. "Hide and seek: An introduction to steganography." *IEEE Security and Privacy*, vol. 1, no. 3, pp. 32-44, 2003.

[16] William Stalling. Cryptography and Network Security Principles and Practices, 4th ed., Prentice Hall, 2005.

[17] J. Fridrich. Steganography in Digital Media: Principles, Algorithms, and Applications. Cambridge University Press, 2009.

[18] A. Nag, S. Biswas, D. Sarkar, P. P. Sarkar, "A Novel Technique for Image Steganography based on Block-DCT and Huffman Encoding", *International Journal of Computer Science and Information Technology*, Vol. 2, No. 3, June 2010.

[19] Rajarathnam Chandramouli, Mehdi Kharrazi, Nasir Menon. "Image Steganography and Steganalysis: Concepts and Practice." *Digital Watermarking*. Springer-Verlag Berlin Heidelberg, 2004.

[20] C. Y. Yang, W.C. Hu and C. H. Lin, "Reversible Data Hiding by Coefficient-bias Algorithm", *Journal of Information Hiding and Multimedia Signal Processing*, vol. 1, no. 2, Apr. 2010.

[21] S. S. Alharthi and P. K. Atrey. "Further Improvements on Secret Image Sharing Scheme.", *Proceedings of the 2nd ACM workshop on Multimedia in forensics, security and intelligence*. ACM. October 29, 2010.

[22] L. Bai. "A reliable (k, n) image secret sharing scheme." *IEEE International Symposium on Dependable, Autonomic and Secure Computing*, pp. 31–36, 2008.

Logic Design and Comparison of Arithmetic Structures for AES Cryptographic Systems

Mostafa Abd-El-Barr
Department of Information Science
College of Computing Sciences and Engineering
Kuwait University

Aisha Al-Noori
Department of Computer Engineering
College of Computing Sciences and Engineering
Kuwait University

Abstract - *The Advanced Encryption Standard (AES) is a symmetric key block cipher cryptosystem adopted by the NIST as the world standard for data encryption/decryption. Multi-valued logic (MVL), as opposed to two-valued logic (TVL), is a propositional calculus in which there are more than two truth values. Multiple-valued logic offers opportunities for overcoming a number of difficulties facing TVL such as chip area, power consumption, and delay. A number of researchers have conducted intensive research work in attempts to improve and enhance the performance of cryptographic systems in terms of speed, area, and power consumption. In this paper, we present a number of arithmetic operations required by the AES cryptosystem using both TVL and MVL. We also cover the Galois field arithmetic operations performed in cryptographic systems. A comparison among different realizations is provided using the AT^2 performance measure where A is the area and T is the delay.*

Keywords: Multi-valued logic (MVL), Cryptography, AES Cryptosystem, MVL Arithmetic Operations, Galois field arithmetic.

1 Introduction

Efficient multiple-valued logic (MVL) employing conventional binary CMOS (Complementary Metal Oxide Semiconductor) circuits have been reported in the literature [1]-[2]. Examples of the MVL reported hardware realizations of circuits include adders [3], multipliers [4][5][6], memory [7][8][9], digital signal processing (DSP) [2], and cryptography [10][11].

Computation in Galois Field $GF(2^k)$ plays a crucial role in cryptographic applications. The high speed and security requirements of cryptographic applications depend primarily on the speed and security of the arithmetic computations performed in Galois Fields. Examples of $GF(2^k)$ operations include addition, multiplication, and exponentiation [12].

In this paper, we cover both TVL and MVL gate level realizations for the AES cryptography. The paper is organized as follows. In Section 2, we provide some background. In Section 3, we provide the TVL and MVL arithmetic operations. In Section 4, we introduce alternate realization of MVL operations. In Section 5, we cover the Galois field arithmetic operations. In Section 6, we introduce alternate realization of some operations in GF(4). In Section 7, we provide a comparison among the proposed realization of MVL and GF(4) operations. In Section 8, we provide a number of concluding remarks.

2 Background material

In this section, we provide some background material.

Definition 1: An n-variable r-valued function, $f(X)$, is defined as a mapping $f: R^n \rightarrow R$, where $R=\{0,1,\cdots,r-1\}$ is a set of r logic values, $r \geq 2$ & $X=\{x_1,x_2,\cdots,x_n\}$ is a set of r-value n variables. □

Examples one-variable and two-variable 4-valued functions $f_1(x)$ and $f_2(x,y)$ are shown in Fig. 1, while Fig. 2 shows definition of a number of unary and two-variable r-valued operators.

x	0	1	2	3
$f_1(x)$	3	0	1	2

(a) One-variable

x	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3
y	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
$f_2(x,y)$	0	2	2	3	2	2	3	2	3	3	1	1	1	2	3	0

(b) Two-variable

Figure 1. Example 4-valued functions

Operator	Definition
Cycle (Cyclic)	$x^{\rightarrow k}=(x+k) \bmod r$
Successor	$x^{\rightarrow}=(x+1) \bmod r$
Predecessor	$x^{\leftarrow}=(x-1) \bmod r$
Window Literal	$x^{a,b}=\begin{cases}(r-1) & \text{if } a \leq x \leq b \\ 0 & \text{otherwise}\end{cases}$
Min	$x \bullet y=\begin{cases}x & \text{if } x<y; \\ 0 & \text{otherwise}\end{cases}$
Truncated sum	$x \oplus y=\min((r-1),\ (x+y))$

Figure 2. Definition of sample unary and two-variable r-valued operators.

3 Binary and multi-valued logic arithmetic operations

In this section, we present a number of binary and MVL arithmetic structures.

3.1 Binary-based modulo-4 addition

The Modulo-4 addition of two numbers X and Y is given by A = (X+Y) mod 4, where A, X, and Y are 4-valued variables. Fig. 3(a) shows the addition in a tabular form.

The mod-4 addition can be realized by decomposing the 4-valued output A into two binary values a_0 and a_1 as shown in Fig. 3(b). The 2-level NAND-NAND logic equations for a_0 and a_1 are:

$$a_0 = \overline{(\overline{y_0 y_1 x_0}) \cdot (\overline{y_0 x_0 x_1}) \cdot (\overline{y_0 y_1 x_0 x_1}) \cdot (\overline{y_0 y_1 x_0 x_1}) \cdot (\overline{y_0 x_0 x_1}) \cdot (\overline{y_0 y_1 x_0})}$$
$$a_1 = \overline{(\overline{y_1 x_1}) \cdot (\overline{y_1 x_1})}$$

It should be noted that the two last rows and columns have been flipped for binary function realization to represent the K-map while it is left as is in MVL. The NAND realization of the mode-4 addition requires three NAND2, four NAND3, two NAND4 and one NAND6.

X / Y	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

(a) Tabular form.

	00	01	11	10
00	0	0	1	1
01	0	1	0	1
11	1	0	1	0
10	1	1	0	0

a_0

	00	01	11	10
00	0	1	1	0
01	1	0	0	1
11	1	0	0	1
10	0	1	1	0

a_1

(b) Binary decomposition.

Figure 3. The mod-4 addition.

3.2 Binary-based modulo-4 subtraction

A tabular representation of modulo-4 subtraction is shown in Fig. 4(a). The 4-valued output S can be decomposed into two binary values s_0 and s_1 as shown in Fig. 4(b). The 2-level NAND-NAND logic equations for s_0 and s_1 are:

$$s_0 = \overline{(\overline{y_0 y_1 x_0}) \cdot (\overline{y_0 x_0 x_1}) \cdot (\overline{y_0 y_1 x_0 x_1}) \cdot (\overline{y_0 y_1 x_0 x_1}) \cdot (\overline{y_0 x_0 x_1}) \cdot (\overline{y_0 y_1 x_0})}$$
$$s_1 = \overline{(\overline{y_1 x_1}) \cdot (\overline{y_1 x_1})}$$

The NAND realization of the mode-4 subtraction requires three NAND2, four NAND3, two NAND4 and one NAND6.

3.3 Binary-based modulo-4 multiplication

Modular multiplication is given by M = (X*Y) mod 4, Where X, Y, and M are 4-valued variables. A tabular representation of the Modulo-4 multiplication is shown in Fig. 5(a). The 4-valued output M can be decomposed into two binary values m_0 and m_1 as shown in Fig. 5(b). The 2-level NAND-NAND logic equations for m_0 and m_1 are:

$$m_0 = \overline{(\overline{y_0 y_1 x_0}) \cdot (\overline{y_1 x_0 x_1}) \cdot (\overline{y_0 x_0 x_1}) \cdot (\overline{y_0 y_1 x_1})}$$
$$m_1 = \overline{(\overline{y_1 x_1})}$$

The NAND realization of the mode-4 multiplication requires two NAND2, four NAND3 and one NAND4.

X / Y	0	1	2	3
0	0	1	2	3
1	3	0	1	2
2	2	3	0	1
3	1	2	3	0

(a) Tabular form.

	00	01	11	10
00	0	0	1	1
01	1	0	1	0
11	0	1	0	1
10	1	1	0	0

s_0

	00	01	11	10
00	0	1	1	0
01	1	0	0	1
11	1	0	0	1
10	0	1	1	0

s_1

(b) Binary decomposition.

Figure 4. The mod-4 subtraction.

X / Y	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	0	2
3	0	3	2	1

(a) Tabular form.

	00	01	11	10
00	0	0	0	0
01	0	0	1	1
11	0	1	0	1
10	0	1	1	0

m_0

	00	01	11	10
00	0	0	0	0
01	0	1	1	0
11	0	1	1	0
10	0	0	0	0

m_1

(b) Binary decomposition.

Figure 5. The mod-4 multiplication.

3.4 Binary-based modulo-4 division

Modular division is given by Z = (X/Y) mod 4, where X, Y, and Z are 4-valued variables. A tabular representation of the Modulo-4 division is shown in Fig. 6(a). It should be noted that "D" in the table represents undefined (don't care) value. The 4-valued output Z can be decomposed into two binary values z_0 and z_1 as shown in Fig. 6(b). The 2-level NAND-NAND logic equations for z_0 and z_1 are:

$$z_0 = \overline{(\overline{y_0 x_0}) \cdot (\overline{y_1 x_1}) \cdot (\overline{y_1 x_0 x_1}) \cdot (\overline{y_0 x_0 x_1})}$$
$$z_1 = \overline{(\overline{y_0 x_1}) \cdot (\overline{y_1 x_0 x_1}) \cdot (\overline{y_0 y_1 x_0}) \cdot (\overline{y_0 x_0 x_1})}$$

The NAND realization of the mode-4 division requires three NAND2, five NAND3 and two NAND4.

X / Y	0	1	2	3
0	D	D	D	D
1	0	1	2	3
2	0	3	1	2
3	0	2	3	1

(a) Tabular form.

	00	01	11	10
00	D	D	D	D
01	0	0	1	1
11	0	1	0	1
10	0	1	1	0

z_0

	00	01	11	10
00	D	D	D	D
01	0	1	1	0
11	0	0	1	1
10	0	1	0	1

z_1

(b) Binary decomposition.

Figure 6. The mod-4 division.

4 Alternate realization of MVL arithmetic operations

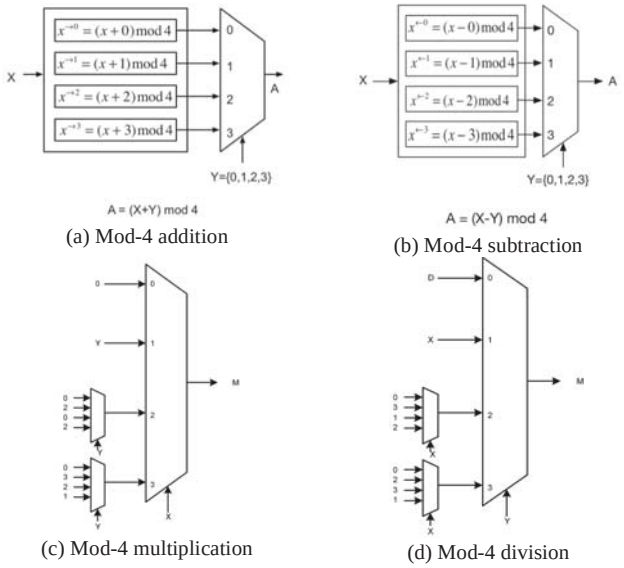
In this section, we introduce alternate realizations of the four mod-4 operations using the Unary *r*-valued functions shown in Fig. 2(a). These are shown in Fig. 7.

The mod-4 addition can also be implemented using the successor unary operation $x^{\rightarrow} = (x+1) \bmod 4$ (see Fig. 2(a)). This realization is shown in Fig. 7(a).

The realization consists of four successor operations over one of the inputs (X in this case) plus a 4-to-1 multiplexer controlled by the other input (Y in this case) such that if Y = 0, then A = (X+0) mod 4; otherwise if Y =1, then A = (X+1) mod 4; otherwise if Y =2, then A = (X+2) mod 4; otherwise if Y =3 then A = (X+3) mod 4.

The mod-4 subtraction can also be realized using the predecessor operation $x^{\leftarrow} = (x-1) \bmod 4$ (see Fig. 2(a)). This realization is shown in Fig. 7(b).

A Multiplexer-based realization of the modulo-4 multiplication is shown in Fig. 7(c). A multiplexer-based realization of the modulo-4 division operation is shown 7(d).



(a) Mod-4 addition (b) Mod-4 subtraction

(c) Mod-4 multiplication (d) Mod-4 division

Figure 7. Multiplexer-based realization of the four mod-4 operations.

TABLE I. GF(4) REPRESENTATIONS

MVL	2-tuple representation	ω (Primitive element)	Polynomial Representation
0	00	ω^{∞}	0
1	01	ω^{0}	1
2	10	ω^{1}	x
3	11	ω^{2}	$x+1$

5 Galois field arithmetic operations

The elements of GF (4) are 0, 1, 2, and 3. We assume that 0 denotes the additive identity and that 1 denotes the multiplicative identity. A useful representation that clarifies the results of the Galois Field operations is to use ω to represent the primitive element and $x^2 + x + 1$ as the irreducible polynomial. Under these assumptions, Table I shows relationship among different $GF(4)$ representations.

5.1 Galois field addition

The addition operation can be understood in terms of the following $GF(2^k)$ property $(\alpha \pm \omega)^{\mu} = \alpha^{\mu} \pm \omega^{\mu}$ where $\mu = 2^k$ and $\alpha, \omega \in GF(2^k)$. The binary-based realization of GF(4) addition can be defined as shown in Fig. 8. The GF addition is performed by adding the 2-tuple representations of the two arguments bit-wise mod 2. For example adding 2 (10) plus 3 (11) results in 1 (01). Similarly, adding 2 (10) plus 2 (10) results in 0 (00), etc. The 2-level NAND-NAND logic equations for ag_0 and ag_1 are:

$$ag_0 = \overline{(\overline{y_0 x_0}) \cdot (\overline{y_0 x_0})}$$
$$ag_1 = \overline{(\overline{y_1 x_1}) \cdot (\overline{y_1 x_1})}$$

X / Y	0	1	2	3
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

(a) Tabular form.

	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	1	1	0	0
10	1	1	0	0

ag_0

	00	01	11	10
00	0	1	1	0
01	1	0	0	1
11	1	0	0	1
10	0	1	1	0

ag_1

(b) Binary decomposition.

Figure 8. GF(4) Addition

The NAND realization of the GF(4) addition requires six NAND2.

5.2 Galois field multiplication

The multiplication operation can be explained in terms of the ω shown in Table 1 as follows:

$$\omega^i \times \omega^j = \omega^{(i+j) \bmod (2^2-1)} = \omega^{(i+j) \bmod 3}$$

The binary-based realization of GF(4) multiplication can be defined as shown in Fig. 9. The 2-level NAND-NAND logic equations for mg_0 and mg_1 are:

$$mg_0 = \overline{(\overline{y_0 y_1 x_0}) \cdot (\overline{y_0 y_1 x_1}) \cdot (\overline{y_0 x_0 x_1}) \cdot (\overline{y_0 y_1 x_0 x_1})}$$
$$mg_1 = \overline{(\overline{y_0 y_1 x_1}) \cdot (\overline{y_1 x_0 x_1}) \cdot (\overline{y_0 x_0 x_1}) \cdot (\overline{y_0 y_1 x_0})}$$

The NAND realization of the GF(4) multiplication requires seven NAND3 and three NAND4.

X / Y	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	3	1
3	0	3	1	2

(a) Tabular form.

	00	01	11	10
00	0	0	0	0
01	0	0	1	1
11	0	1	1	0
10	0	1	0	1

mg_0

	00	01	11	10
00	0	0	0	0
01	0	1	1	0
11	0	1	0	1
10	0	0	1	1

mg_1

(b) Binary decomposition.

Figure 9. GF(4) Multiplication

5.3 Galois field division

The binary-based realization of the division operation in GF(4) X/Y is shown in Fig. 10.

The 2-level NAND-NAND logic equations for zg_0 and zg_1 are:

$$zg_o = \overline{(\overline{y_o x_o}) \cdot (\overline{y_1 x_1}) \cdot (\overline{y_1 x_o x_1}) \cdot (\overline{y_o x_o x_1})}$$
$$zg_1 = \overline{(\overline{y_o x_1}) \cdot (\overline{y_1 x_o x_1}) \cdot (\overline{y_o y_1 x_o}) \cdot (\overline{y_o x_o x_1})}$$

The NAND realization of the GF(4) division requires three NAND2, five NAND3 and two NAND4.

6 Alternate realization of GF(4) operations

We also introduce alternate realizations of the four operations in GF(4) in the following subsections.

6.1 Alternate GF(4) addition realization

A possible realization of the GF (4) addition can be made by decomposing the addition into two components as

X / Y	0	1	2	3
0	D	D	D	D
1	0	1	2	3
2	0	3	1	2
3	0	2	3	1

(a) Tabular form.

	00	01	11	10
00	D	D	D	D
01	0	0	1	1
11	0	1	0	1
10	0	1	1	0

zg_0

	00	01	11	10
00	D	D	D	D
01	0	1	1	0
11	0	0	1	1
10	0	1	0	1

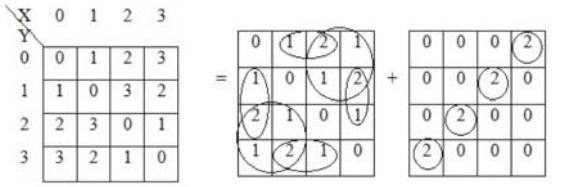
zg_1

(b) Binary decomposition.

Figure 10. GF(4) Division

per the technique we have proposed in [13]. This is shown in Fig. 11.

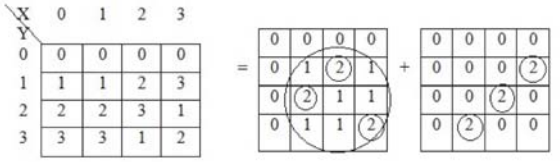
6.2 Alternate GF(4) multiplication realization



$$f(x,y) = 1^1x^2 \bullet 1^0y^0 \oplus 1^2x^3 \bullet 1^0y^1 \oplus 1^3x^3 \bullet 1^1y^2 \oplus 1^0x^0 \bullet 1^1y^2 \oplus 1^0x^1 \bullet 1^2y^3 \oplus 1^1x^2 \bullet 1^3y^3 \oplus 2^3x^3 \bullet 2^0y^0 \oplus 2^2x^2 \bullet 2^1y^1 \oplus 2^1x^1 \bullet 2^2y^2 \oplus 2^0x^0 \bullet 2^3y^3$$

Figure 11. Decomposition of the GF(4) addition using the technique in [13].

A possible realization is based on splitting the GF(4) map as per the technique we introduced in [13]. This is shown in Fig. 12. A multiplexer-based realization of the GF(4) multiplication operation is introduced in Fig. 14(a).



$$f(x,y) = 1^1x^3 \bullet 1^1y^3 \oplus 1^2x^2 \bullet 1^1y^1 \oplus 1^1x^1 \bullet 1^2y^2 \oplus 1^3x^3 \bullet 1^3y^3 \oplus 2^3x^3 \bullet 2^1y^1 \oplus 2^2x^2 \bullet 2^2y^2 \oplus 2^1x^1 \bullet 2^3y^3$$

Figure 12. Decomposition of the GF(4) multiplication using the technique in [13] .

6.3 Alternate GF(4) division realization

Fig. 14(b) shows a multiplexer-based realization of the GF(4) division operation. Yet a third realization of the GF(4) division operation is possible using the technique introduced in [13]. This is shown in Fig. 13.

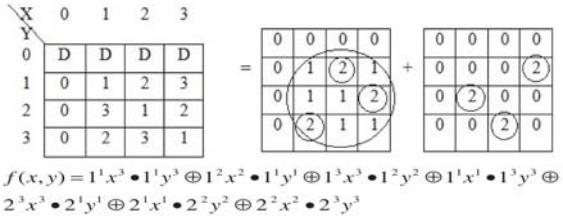
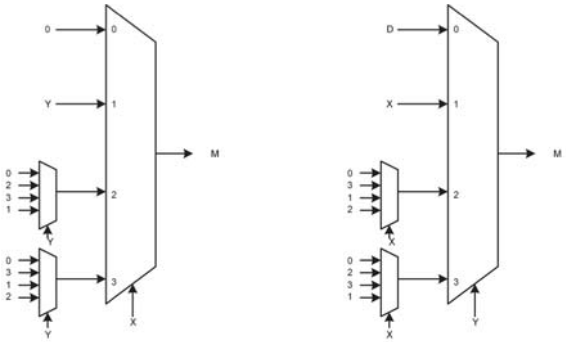


Figure 13. Decomposition of the GF(4) division using the technique in [13] .



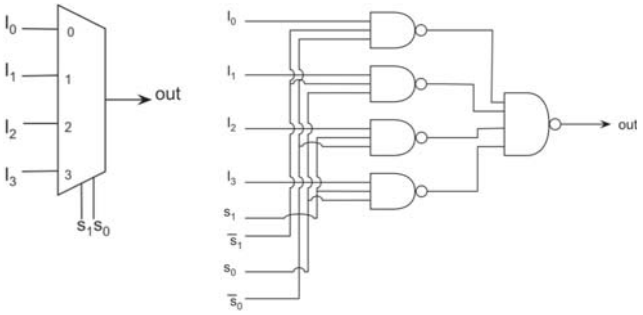
(a) GF(4) multiplication. (b) GF(4) division.

Figure 14. Multiplexer-based realization of GF(4) multiplication and division operations.

7 Comparison

In order to compare the binary-based realization of the MVL arithmetic operations with the multiplexer-based realizations, we provide the gate level of the multiplexer-based realization of the four operations depicted in Fig. 7. Fig. 15 shows the symbol and the NAND realization of 1-bit 4-to-1 multiplexer. For simplicity, we use the multiplexer symbol in our realization rather than the NAND gates. Fig. 16 shows the gate-level of the multiplexer-based realizations of the mod-4 addition and multiplication introduced in section 4. The gate-level of the mod-4 subtractions is similar to that of the mod-4 addition and the gate-level of the mod-4 division is similar to that of the mod-4 multiplication. The successor unary operator used to realize the mod-4 addition is represented in Fig. 16 (a) using 1-bit 4-to-1 multiplexers. The multiplexer-based realization of the mod-4 multiplication is decomposed using 1-bit 4-to-1 multiplexers. This is shown in Fig. 16 (b).

We compare different realizations based on the area (A), the delay (T) and the area delay squared (AT^2) product. All delay results presented in this section are based on using 0.35 µm CMOS technology of AMS Corp. [14]. According to the CMOS library used one NAND2 gate has 0.1 ns delay. If we assume that the delay of one transistor is 0.05 ns, then the delay of x-input NAND gate is 0.05x ns. The area is measured using the number of equivalent NAND2 gates. We use the term GE (gate equivalent), where one gate equivalent corresponds to one NAND2

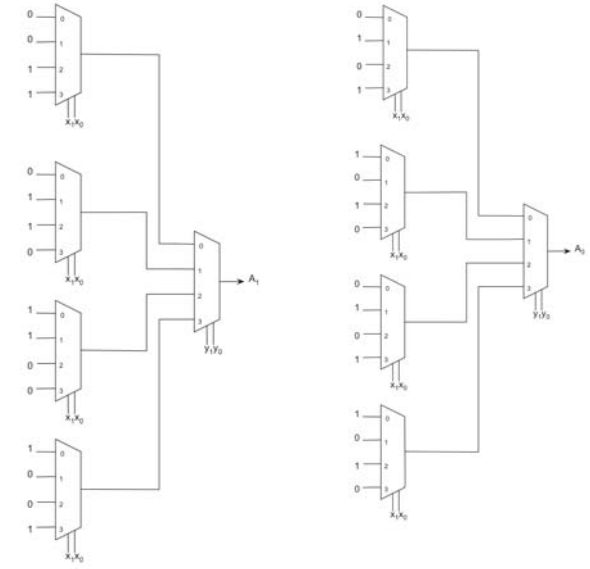


(a) Symbol (b) NAND Realization

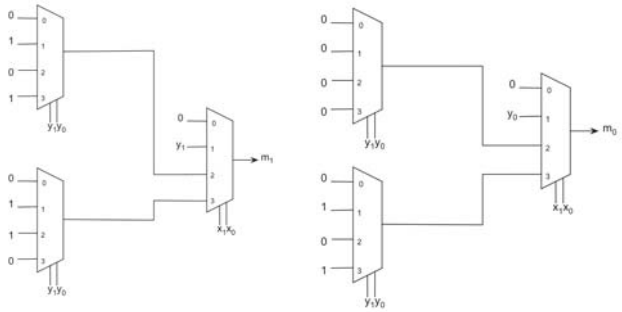
Figure 15. 1-bit 4-to-1 Multiplexer

gate. In our area calculations, we use the evaluation model presented in [15]. The binary-based realizations of the four mod-4 operations consume less area and delay than the multiplexer-based realizations of the same operations. The area (A), delay (T) and area delay squared (AT^2) product of the four mod-4 operations are summarized in table II and III.

In section 6, we introduce multiplexer-based realizations of the GF(4) multiplication and division operations. Fig. 17 shows the gate level of the multiplexer-



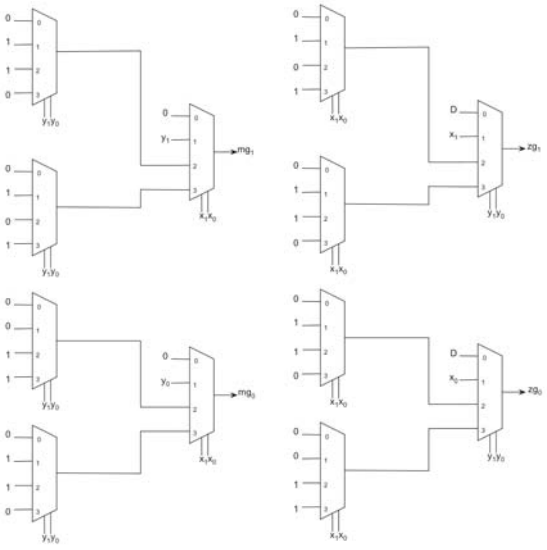
(a) The mod-4 addition.



(b) The mod-4 multiplication.

Figure 16. Gate-level of the multiplexer-based realizations of mod-4 addition and multiplication using 1-bit 4-to-1 multiplexer

based realization of GF(4) multiplication and division using the 1-bit 4-to-1 multiplexer symbol.



(a) GF(4) multiplication. (b) GF(4) division.

Figure 17. Gate-level of the multiplexer-based realizations of GF(4) multiplication and division operations

The binary-based realizations for both GF(4) multiplication and division consume less area and delay than the multiplexer-based realizations of the same operations. The area (A), delay (T) and area delay squared (AT^2) product of the GF(4) multiplication and division are summarized in table IV.

Further comparison are made between the binary-based realizations of the Mod-4 and GF(4) multiplication operations. We observe that the binary-based realization of Mod-4 multiplication consumes less area and delay than the realization of the GF(4) multiplication.

8 Concluding remarks

In this paper, we considered the realization of a number of arithmetic operations required by the AES cryptosystem using MVL. Our coverage includes MVL basic modular structures, algorithms, and circuits, which can be used for the AES cryptography. We presented the binary decomposition and the gate-level realization of Modulo-4 Addition, Subtraction, Multiplication and Division. Furthermore, we introduced alternate realizations of the MVL arithmetic operations. We have also covered the Galois field arithmetic operations required for cryptographic systems. In each case we have considered alternative possible realizations for a given operation and showed the different possible gate-level realizations. Different realizations of the MVL arithmetic operations and GF(4) operations are compared based on the area, delay and area delay squared product. The results show that the binary-based realizations of the Modulo-4 and the GF(4) operations outperform the multiplexer-based realizations of the same operations in terms of the area delay squared (AT^2). The authors are currently working on the CMOS implementation of the proposed circuits.

TABLE II. AREA AND DELAY OF MOD-4 ADDITION AND SUBTRACTION

	Mod-4 addition		Mod-4 subtraction	
	BB[(1)]	**MB[(2)]**	**BB[(1)]**	**MB[(2)]**
Area (GE)	22.5	105	22.5	105
Delay (ns)	0.6	0.7	0.6	0.7
AT^2	8.1	51.45	8.1	51.45

TABLE III. AREA AND DELAY OF MOD-4 MULTIPLICATION AND DIVISION

	Mod-4 multiplication		Mod-4 division	
	BB[(1)]	**MB[(2)]**	**BB[(1)]**	**MB[(2)]**
Area (GE)	12.5	63	18.375	63
Delay (ns)	0.35	0.7	0.35	0.7
AT^2	1.53	30.87	2.25	30.87

TABLE IV. AREA AND DELAY OF GF(4) MULTIPLICATION AND DIVISION

	GF(4) multiplication		GF(4) division	
	BB[(1)]	**MB[(2)]**	**BB[(1)]**	**MB[(2)]**
Area (GE)	22.125	63	18.375	63
Delay (ns)	0.4	0.7	0.35	0.7
AT^2	3.54	30.87	2.25	30.87

[(1)] BB stands for binary-based realization of the operation
[(2)] MB stands for multiplexer-based realization of the operation

9 Acknowledgment

The authors would like to acknowledge the financial support of Kuwait University in the form of a Funded Research Grant # WI 04/10.

10 References

[1] E. Dubrova, "Multiple-Valued Logic in VLSI", Multiple-Valued Logic: An International Journal, 2002, pp. 1-17.

[2] M. Khan, "Synthesis of quaternary reversible/quantum comparators", Journal of System Architectures, vol. 54, no. 10, October 2008, pp. 977-982.

[3] M. Kameyama, S. Kawahito, T. Higuchi, "A Multiplier chip with Multiple-Valued Bidirectional Current-Mode Logic Circuits", IEEE Computer, April, 1988, pp. 43-56.

[4] H. Razavi, S. Bou-Ghazale, "Design of a Fast CMOS Ternary Adder", Proceedings 17th ISMVL, May 1997, pp. 20–23.

[5] S. Kawahito, Kameyama, M., Higuchi, T., and Yamada, H., "A 32×32-bit multiplier using multiple-valued MOS current-mode circuits", IEEE Journal of Solid-State Circuits, vol. 23(1), Feb. 1988, pp. 124-132.

[6] J. Kim and Ahn, S., "High-speed CMOS de-multiplexer with redundant multi-valued logic", International Journal of Electronics, 94, 2007, pp. 915-924.

[7] K. Naiff, D. Rich and K. Smalley, "A Four-State ROM Using Multilevel Process Technology", IEEE JSSC, Vol. 19, No. 2, April 1984, pp. 174–179.

[8] Intel Strata TM Flash. Available at http://www.intel.com/design/flash/isf/overview.pdf.

[9] T. Okuda and Murotani, T., "A four-level storage 4GB DRAM", IEEE JSS Cts., vol. 32, no. 11, 1997, pp. 1743-1747.

[10] Homma, N., Saito, K., and Aoki, T., "Formal Design of Multiple-valued Arithmetic Algorithms over Galois Fields and its Application to Cryptographic Processor", Proceedings 2012 IEEE 42nd International Symposium on MVL, May 2012, pp. 110-115.

[11] Qin, H., Sasao, T., and Iguchi, Y., "A Design of AES Encryption Circuit with 128-bit Keys using look-Up Table Ring on FPGA", IEICE Transactions on Information Systems, Vol. E89-D, No. 3, March 2006, pp. 1139-1147.

[12] A. Odlyzko, "Discrete logarithms in finite fields and their cryptographic significance", Proceedings of the Eurocrypt on Advances in Cryptology' 84, France, 1984, pp. 224-314.

[13] Abd-El-Barr, M., Vranesic, Z., and Zaky, S., "Algorithmic Synthesis of MVL Functions for CCD Implementation ", IEEE Transactions on Computers (TC), vol. 40, no. 8, August 1991, pp. 977-986.

[14] "Speciality Process 0.35μm CMOS Application Notes," AMS Corp., Austri, [Online]. Available: http://ams.com/eng/Products/Full-Service-Foundry/Process-Technology/CMOS/0.35-m-CMOS Application-Notes.

[15] Vázquez, Alvaro, and Elisardo Antelo., "Area and Delay Evaluation Model for CMOS Circuits", Internal Report, Univ. of Santiago de Compostela, [Online]. Available: http://www. ac. usc. es/node/1607, 2012.

An Authentication Scheme Based on Elliptic Curve Cryptosystem and OpenID in the Internet of Things

Jong Jin Lee[1], Youn-Sik Hong[2], and Ki Young Lee[1]
[1]Dept. of Information and Telecommunications Engineering,
[2]Dept. of Computer Science and Engineering
Incheon National University, Incheon 406-772, Korea

Abstract - *Authentication is a communication protocol processing procedure. In the Internet of Things, secure communication should be constructed between one "thing" and another by such a procedure. The identity that the second "thing" or object claims should be consistent with what the first one claims. Claimed identity information becomes a single message. Based on this message, we verify the identity of the "things". The purpose for both communication partners to implement authentication protocol is to have solid communication in the high layer (e.g., application layer). In order to do that, usually the authentication protocol has several sub-tasks such as identification key establishment, or key switching and consultation. In an authentication process, identity of the claimer can be acquired through message identification. In authenticated key establishment protocol, key establishment materials are also important protocol messages, which is part of entity authentication. In this paper, we focus on simple and efficient secure key establishment based on ECC (Elliptic Curve Cryptosystem). And we proposed ECC and OpenID based user authentication scheme. Our analysis shows that our approach can prevent attacks like eavesdropping, the man-in-the middle, key control attack, and replay attacks.*

Keywords: Internet of Things, User Authentication, Elliptic Curve Cryptosystem (ECC), OpenID

1 Introduction

Nowadays, through the communication between various smart devices including a smart phone, it may be provided to the user after be generated a secondary data. Because a series of information can be gathered, processed, handled and controlled. In these environments, it may be exposed to the attack by sending the information to users which was not justified. Therefore, execution process of authentication for the user is required. However, due to constrained environment such as a low-power, ultra-small objects in the Internet of Things, there are omitted case for necessary authentication phases and process. Accordingly, security damage incidents and accidents are increasing, due to the exposure of the transmitted information to device that it does not authentication and authorization though secure authentication phases. At this time, man-in-the-middle attacks such as an information gathering, imitation, blocking and an invasion of privacy can occur.

In order to solve these security vulnerabilities and problems, various user authentication methods are proposed in earlier studies. In earlier user authentication and identification technologies, there are divided such as ID-based, certification-based and SIM-based methods. And first, ID-based as a traditional authentication method can be lightweight and fast operation, however, there are problems for a relatively low safety and key management[1]. The certification-based method has the problem of the certification management, because it is how to authenticate using by issued certification. Finally, the SIM-based method is a how to perform the authentication by storing and managing authentication information in file system of a SIM card, thus, it is physically strong. However, it is necessary such as a separate software, a how to manage. Likewise, there is still problem on the aspect of safety and efficiency. In this paper, we analyze the problems and limitations on applying earlier user authentication methods in Internet of Things. And we also propose the method and architecture for user authentication in the Internet of Things. We also propose the hybrid authentication method to apply using both OpenID-based scheme and public key based algorithms.

2 Related works

In this section, we introduce the basic concepts of ECC and OpenID.

2.1 Elliptic Curve Cryptosystem

Elliptic curve cryptography (ECC) can be categorized as public cryptography with its many advantages over the other public cryptography. After it was first proposed by Koblitz [2] and Miller [3] independently in the nearly same year, it has been extensively studied and implemented by mathematicians, cryptographers and computer scientists over the world. Till now, the best algorithm needs full exponential time to solve the underlying mathematical problem of ECC, which is referred to as the elliptic curve discrete logarithm problem (ECDLP). Contrasted with ECC, there are sub-exponential-time algorithms to tackle the integer factorization and the discrete logarithm problems on which RSA and DSA is relied on, respectively. It is believed that ECC with the key length of 162 bits is at the same secure level as RSA with the key length of 1024 bits.

Table 1. Comparison of length of key for RSA and ECC[4]

Security Level	RSA key length	ECC key length
80	1024	160-223
112	2048	224-225
128	3072	256-283
192	7680	384-511
256	15360	512-571

ECC offers a security level equivalent to RSA while using a far smaller key size; therefore it leads to the better performance in limited environments like cellular phones, PDA, sensor networking, etc. The standard organizations such as IEEE, NIST, IETF and ISO have accepted ECC as an alternative and efficient public key cryptosystem. It can provide various security services such as key exchange, privacy through encryption, and sender authentication and message integrity through digital signature.

Typical Elliptic Curve can be defined in Finite Field GF(p) as follows, where p is a large prime.

$$y^2 = x^3 + ax + b \quad (1)$$

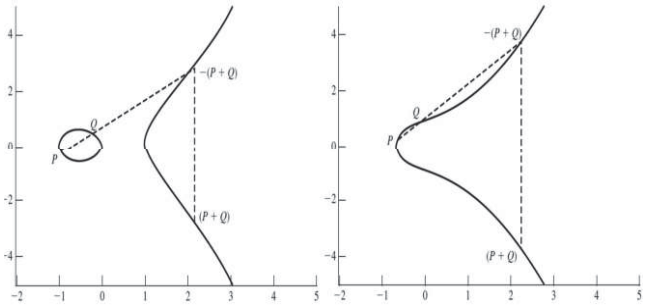


Figure 1. Elliptic Curves of $y^2 = x^3 - x$, $y^2 = x^3 + \mathrm{x} + 1$

As Elliptic Curve Cryptosystem on GF(p) is safer than counterpart in GF(2^m), we only discuss the curve defined on GF(p) where $\Delta = 4a^3 + 27b^2 \neq 0$.

$\Delta \neq 0$ means that there is only one tangent line at every point on the elliptic curve. If a pair (x, y) meets the equation (1), it is called one point on the curve.

The addition of two points on the elliptic curve is very simple. Suppose P and Q are on the elliptic curve, where $\mathrm{P} \neq \mathrm{Q}$ and $\mathrm{P} \neq -\mathrm{Q}$, we draw a line which goes through these two points firstly. As the x′s order in the curve is 3, the line will have the third intersection point N with the curve. Then we draw a line which parallels to y axis through Point N. Thus, this line will have the second point R on the curve; and the Point R is the point we want to compute.

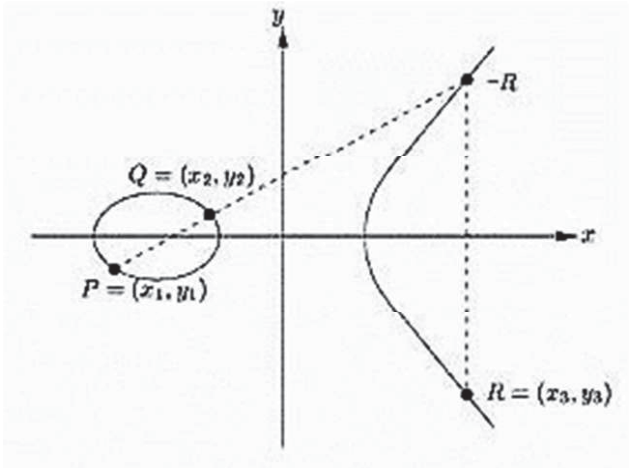


Figure 2. Point Addition on an Elliptic Curve

$$R = P + Q \quad (2)$$

If P = -Q, the line through P and Q will parallel to y axis; then we regard that the infinite point Θ is the result.

$$\Theta = P + Q \quad (3)$$

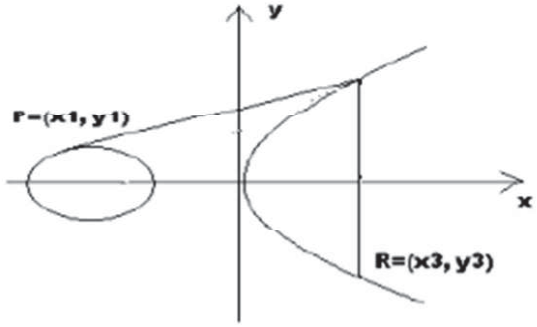


Figure 3. Point Double on an Elliptic Curve

If P = Q, we will draw a tangent line at P, and this line will have the third point N on the curve because the x's order of the curve is 3. Then we draw a line which parallels to y axis through N point. This line will have the second point R on the curve, which is the point we want to compute.

$$R = 2P \quad (4)$$

With these three operations, we can compute kP, where k is a large integer. First of all, k can be represented in the binary form; then we can use point operations to compute kP. National Institute of Standards and Technology (NIST) recommends four random elliptic curves which can be used in real elliptic curve cryptosystem. To improve the computing efficiency of ECC, it recommends that the coefficient a can be -3.

ECDLP refers to that we try to find an integer d so that Q = dP, where P and Q are two points on the elliptic Curve. We can compute dP easily if we know d and P, but it is a difficult problem for us to find out d if we only know P and Q.

Parameters for Elliptic Curve Cryptosystem include F, a, b, P, n, and h. F is the finite field; a and b are the coefficients of the curve; P is the base point; n is the order of P and a large prime; h = #E(K)/n ; and #E(K) is the number of the points on the curve.

We suppose that Bob wants to send a message M to Alice. Firs of all, Alice chooses an elliptic curve, private key d, and public key Q where Q = dP. She distributes her public key and the parameters on an authenticable channel. Bob gets all these parameters on the channel.

The encryption process is displayed as follows.

(1) Bob represents M as an element m in GF(p).
(2) Bob selects a random number k ∈[1, n-1].
(3) Bob computes $P_1 = (x_1, y_1) = kP$.
(4) Bob computes $P_2 = (x_2, y_2) = kQ$. If $x_2 = 0$, go to step 2.
(5) Bob computes $c = m * x_2$.
(6) Bob sends (P_1, c) to Alice.

Alice gets the cipher text from Bob. The decryption process is displayed as follows.

(1) Alice computes the point $P_N = dP_1 = (x_2, y_2)$, then she gets x_2.
(2) Alice computes $m = c * x_2^{-1}$, then she gets the message M.

We suppose that Alice wants to send a message M to Bob. Bob can verify whether Alice really sends this message with Alice's public key. The signature process is displayed as follows.

(1) Alice represents M as a bit string.
(2) Hash function is used to compute the hash value of m : e = H(M).
(3) A random integer k is randomly selected: k ∈[1, n −1] .
(4) Alice computes the point (x_1, y_1) = kP, and let $r = x_1 (mod\ n)$. If r = 0, go to step 3.
(5) Alice computes $s = k^{-1}(e + rd)(mod\ n)$. If s = 0, go to step 3.
(6) Alice sends the message M and the signature (r, s) to Bob.

After Bob receives the message M and Alice's Signature (r, s), he verifies the signature as follows.

(1) Bob gets Alice's public ECC key.
(2) Bob computes (x_1, y_1) = sP + rQ.
(3) Bob computes HASH value e = H(M).
(4) Bob computes $r' = x_1 + e$.

The message is really from Alice if $r = r'$, otherwise it is not.

2.2 OpenID

OpenID mechanism is a decentralized authentication scheme for the SSO mechanism [5]. OpenID users can chose a trustworthy OpenID server to register their OpenID. They are identified by a URL like: http://yourname.openidserver.com. In OpenID mechanism, three parties are involved: the OpenID provider(OP),the service provider which is also called Relying Party(RP) and the user. We assume that the OP and the RP trust each other in advance, OP has a trusted list of RPs. In OpenID mechanism, users only need to have a pair of identity and password. The typical communication flow of this mechanism is described as followed:

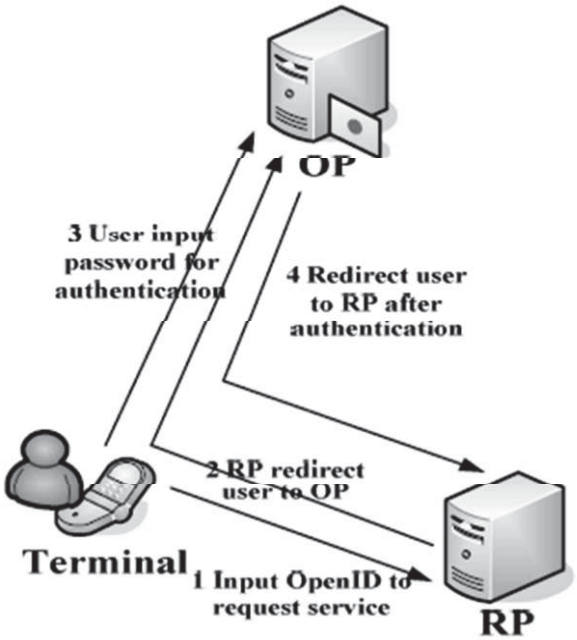


Figure 4. OpenID authentication flow

1) Smart terminal user input OpenID and submit it to the RP;
2) RP normalize the user's OpenID and identify the OP, then RP redirect OP to the smart terminal;
3) User input the corresponding password;
4) OP authenticates the user by OpenID and password pair;
5) If the authentication is successful, the RP page will be redirected to the user

When user submit his/her OpenID like http://myname.openidserver.com to the RP, RP will parse the URL and get two things: one is the OP address "openidserver.com"; another is the user's identity "myname". The RP associate with OP using redirection according to the OP's address and ask OP to authenticate this user's identity. Then OP show user the password login screen and get the password, after authentication, the service page will be redirected to the user. In this way, the RP don't know the user's password, and RP is trusted by OP ahead of this flow. So, users can use one pair of OpenID and password to login onto many service website.

3 Proposed scheme

3.1 Architecture

Based on what we have learned from current literatures of Internet of Things, we may reasonably draw

an abstract architecture for it (as shown in Fig. 5). "Things" or objects become end nodes in the Internet environment. They have unique global addresses (e.g., IPv6 address) and are capable of communicating with each other over the Internet. In order to organize and manage massive resources, every object will pre-register on a nearby trustworthy access point or gateway (denoted as Registration Authority, or RA). This assumption has another advantage that the RA can expend computing and storage capacity of the "things" or objects for authentication purpose. Meanwhile, RA is also able to maintain a history record of all access requests for auditing purpose.

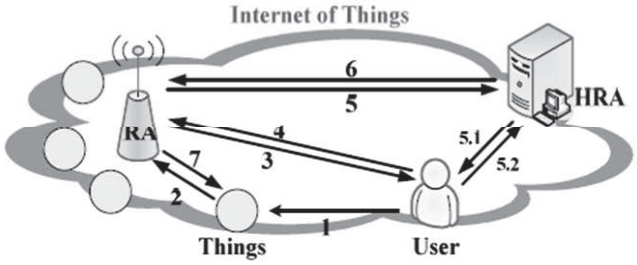


Figure 5. The architecture of proposed scheme

3.2 Authentication Protocol

As shown in Figure 5, a complete request procedure for accessing a "Thing" involves seven steps.

- Step 1: User request to access a "Thing";
- Step 2: "Thing" sends an authentication request to its RA for verification purpose;
- Step 3: RA request User ID;
- Step 4: User response with HRA information;
- Step 5: RA verifies the user HRA information and sends ID verification request to the HRA;
- Step 5.1: HRA challenge the user with a question;
- Step 5.2: User response the challenge with an answer;
- Step 6: HRA response ID OK or not;
- Step 7: RA response the "Thing" about the user ID and issue a session key with the user as we described.

To better describe our protocol, we first introduce some relevant terms here.

Table 2. Notations used in the proposed scheme

Symbol	Description
F_p	a finite field
E	an elliptic curve defined on F_p with a large order
P	a point on E
G	the group of elliptic curve points on E
h()	One-way hash function
s	the RA's private key
$\boldsymbol{ID}_u$	the identity of the user
$\boldsymbol{ID}_t$	the identity of the "thing"

As we known, key establishments and distribution are the fundamental tasks for entity authentication. We can use either SKC or PKC for their implementations, but we have to know the pros and the cons of each algorithm. SKC based schemes suffer the following problems: they require a large memory to store key materials, provide low scalability due to distribution of the keys, add and revoke keys, and require complicated key pre-distribution. On the other hand, PKC-based schemes suffer from high energy consumption and considerable time delay. PKC provides a more flexible and simple interface compared to SKC, which does not require key pre-distribution, pair-wise key sharing, or complicated one-way key chain schemes. For our situation, it is a wise choice if we adopt a PKC-based solution and at the meantime, we also address the aforementioned constraint problems. Based on current research achievements, we believe ECC-based solution is a solid one to be considered.

To establish a session key for two entities, taking a user and an object as an example, only three steps are required as follows.

- Firstly, the RA who is responsible for the object will produce a random P∈G and compute P_s= sP in F_p. Note that, the s is a secret key that is assumed to be assigned before the RA has joined the IoT. For each user with $\boldsymbol{ID}_u$, RA will generate P_u = h($\boldsymbol{ID}_u$) and the private key of the thing S_u = s P_u
- Secondly, the user generate an ephemeral private key a and compute Q_u = a S_u and Q_u' = a P. Then the user will send an authentication message {$\boldsymbol{ID}_u$, Q_u, h($\boldsymbol{ID}_u||\boldsymbol{ID}_t||Q_u||Q_u'$)} to the RA. Once receive the message, RA will compute $Q_u'' = s^{-1}Q_u$ and check whether h($\boldsymbol{ID}_u||\boldsymbol{ID}_t||Q_u||Q_u''$) equal to h($\boldsymbol{ID}_u||\boldsymbol{ID}_t||Q_u||Q_u'$) or not. If not, authentication fails. Otherwise go to step 3.
- The third step is session key establishment. Similarly, the RA will choose a random ephemeral key b and compute Qt = bP for the desired "thing". The session key will be h(abP) based on ECC algorithm.

The next question is how to authenticate a legitimate user in the IoT. "Things" and users are in different domains. They could locate in different hierarchy level of the network. Central authentication method is only valid if a wide accepted KDC (key distribution center) is available. In industry, OpenID technology solves this problem. OpenID enables users to have a single account that allows them to log on to many different sites by authenticating a single identity provider [6]. One approach to identity management is federated identity management, in which participating sites form a circle of trust. Therefore, if the

user is authenticated to one site, the other sites will automatically log the user in if the user visits them [6]. This lightweight idea should be adopted into our design. As such, user authentication is performed in the user domain or registered OpenID service provider. We denote it as home registration authority (HRA). Note that, peer-to-peer authentication method is another solution that can be utilized for further research. However, without solving the mutual-trust problem between two entities, this approach cannot be success.

The IoT needs to authenticate entities that are accessing the pervasive network in order to provide service to only registered members. The entity may be an IoT user or adevice. The IoT is able to support a wide range of ages of users and reflect their own characteristics and needs. As a result, we can selectively use our favorite authentication method among existing authentication methods. The authentication mechanisms are safe and reliable. Our proposed authentication mechanism satisfies these requirements. The RA verifies the certificate contents and the identity of the "thing". Two RA models exist in general PKI. In the first model, the RA collects and verifies the necessary information for the requesting entity before a request for a certificate is submitted to the HRA. The HRA trusts the information in the request because the RA already verified it. In the second model, the HRA provides the RA with information regarding a certificate request that it has already received. The RA reviews the contents and determines if the information accurately describes the user. The RA provides the HRA with a "yes" or "no" answer. It is a device of the kind that has the same or more computing power, memory, and data protection module. Therefore, the RA generates key pairs and requests and receives certificates for all "thing".

4 Analysis of proposed scheme

4.1 Eavesdropping Attack

Each run produces a different session key, and knowledge of past session keys does not allow deduction of future session keys. In our scheme, the session key is calculated by one way hash and session secrets. Know that only the user and RA know the abP, which is computed from the random ephemeral key. That is, even if the previous session secrets are revealed, the other secrets will remain unknown to the adversary.

4.2 Man-in-the-middle Attack

Compromising of a long term secret key, such as SA' at some point in the future, does not lead to compromise of communications in the past. Note that in our scheme, even if the adversary compromises the RA's secret key, it cannot compromise the previous session key because the adversary cannot know the ephemeral key a or b such that it cannot compute the session key. Also, our protocols satisfy both partial forward secrecy and perfect forward secrecy since it is hard to compute the session key without knowing the ephemeral key a or b .

4.3 Key Control Attack

Both communication entities select a random number to generate the session key, which would be discarded after the session expired. Neither one can control the outcome of the session by, for example, restricting it to lie in some predetermined small set. In other words, neither entity can force the session key to a pre-selected value. Hence, our proposed protocol can resist any key control attack.

4.4 Replay Attack

In case a malicious one gained a valid session key or captured network traffic in the IoT, the protocol should resist replay attack by introducing a nonce in every transmitted message. However, it is an optional choice that could vary on different applications. Besides, the session key could be used for identification. Therefore, replayed message from unidentified person will be discarded.

5 CONCLUSION

With the rapid development of IoT, it has penetrated into every aspect of our lives and works, but information security has become the bottle neck of its further development. In this paper, we have proposed Elliptic Curve Cryptosystem and OpenID based user authentication in Internet of Things. Analysis results show that our approach can prevent attacks like eavesdropping, the man-in-the middle, key control attack, and replay attacks.

Acknowledgments

This work was supported by the National Research Foundation of Korea(NRF).

References.

[1] Toan-Thinh TRUONG, Minh-Triet TRAN, and Anh-Due DUONG, "Robust mobile device integration of a fingerprint biometric remote authentication scheme," Advanced Information Networking and Applications(AINA), pp.678-685, 2012.

[2] Weis S, Sarma S, Rivest R, et al, "Security and privacy aspects of low-cost radio frequency identification systems", International Conference on Security in Pervasive Computing, Berlin:Springer, pp.454-469, 2003.

[3] Ohkubo M, Suzuki K, Kinoshita S, "Efficient hash-chain based RFID privacy protection scheme", International Conference on Ubiquitous Computing (Ubicomp), Workshop Privacy: Current Status and Future Directions , September 2004.

[4] V. Gayoso Martinez, L.Hernandez Encinas, "Implementing ECC with Java Standard Edition 7", International Journal of Computer Science and

Artificial Intelligence, Vol.3, Iss.4, pp.134-142, (2013)

[5] Ryu Watanabe and Toshiaki Tanaka. "Federated Authentication Mechanism using Cellular Phone-Collaboration with OpenID", KDDI R&D Laboratories, Inc.ryu@kddilabs.jp.

[6] L. Xiong, X. Zhou, and W. Liu, "Research on the architecture of trusted security system based on the Internet of things," in: Proceedings of the 4th International Conference on Intelligent Computation Technology and Automation, 2011, pp. 1172- 1175.

[7] A. Sarma and J. Girao, "Identities in the future Internet of things,"Wireless Personal Communications: An International Journal, vol.49, issue 3, May 2009, pp. 353-363.

[8] A. Vapen, D. Byers, and N. Shahmehri, "2-clickAuth – optical challenge-response authentication," in: Proceedings of 2010 International Conference on Availability, Reliability and Security,2010, pp. 79-86.

[9] K. Wong, Y. Zheng, J. Cao, and S. Wang, "A dynamic user authentication scheme for wireless sensor networks." IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006.

[10] H. Tseng, R. Jan, and W. Yang, "An improved dynamic user authentication scheme for wireless sensor networks." IEEE Global Communications Conference, 2007.

[11] H. Wang and Q. Li, "Distributed user access control in sensor networks," Distributed Computing in Sensor Systems, pp. 305-320.

[12] J. Zheng, J. Li, M. J. Lee, and M. Anshel, "A lightweight encryption and authentication scheme for wireless sensor networks," International Journal of Security and Networks, Vol. 1, No.3/4 pp.138 - 146, 2006.

An improved NTRU Cryptosystem via Commutative Quaternions Algebra

Nadia Alsaidi[1], Mustafa Saed[2], Ahmad Sadiq[3], Ali A. Majeed[1]
[1]Department of Applied Sciences, University of Technology, Iraq
[2]Hyundai-Kia America Technical Center, USA
[3]Department of Computer Science, University of Technology, Iraq
nadiamg08@gmail.com, msaed@hatci.com, drahmad_tark@uotechnology.edu.iq, ali_alany_91@gmail.com

***Abstract*—NTRU is a public key cryptosystem operating on the ring $Z[X]/(X^N - 1)$, which is known as the ring of convolution polynomials of rank *N*, where *N* is a prime. Reducing the decryption failure probability is a big challenge associated with such type of cryptosystem and is related to the ring that *NTRU* is based on. In this paper, a new multidimensional public key cryptosystem is proposed using commutative ring of quaternions that is not fully fit within Circular and Convolutional Modular Lattice. The decryption failure of this new algebraic structure is reduced. Furthermore, its complexity is four times the complexity of the classical *NTRU*. This results in high secured system resistance to some well-known attacks. Despite this advantage, the computational time analysis shows that the proposed system is slower than the original *NTRU*.**

***Keywords*— public key cryptography; NTRU; lattice; quaternion algebra; non-associative cryptosystem.**

I. INTRODUCTION

CRYPTOGRAPHY is the science of protecting the privacy of information during communication under hostile conditions. Modern telecommunication networks, especially, the Internet and mobile-phone networks have tremendously extended the limits and possibilities of communications and information transmissions. Associated with this rapid development, there is a growing demand for cryptographic techniques, which have spurred a great deal of intensive research activities in the study of cryptography

In mid-1990, a software company needed a cryptosystem that deals with a few bits processors and small numbers. Three mathematicians, Jeffry Hoffstein, Jill Pipher and Joseph Silverman [1] suggested a new cryptosystem, *NTRU* (Number Theory Research Unit). This system is a public–key cryptosystem. The computational and space complexity problems motivated them to propose this system that was fully presented in 1998. It is not based on integer factorization and discrete logarithm problem, but, it is based on a class of arithmetic operations that are efficiently performed with insignificant storage and time complexity [2]. This property made NTRU very suitable choice for a large number of applications, such as mobile phones, portable devices, low-cost smart cards, and RFID devices [3].

Since the introduction of NTRU cryptosystem, many researchers tried to improve its performance during the past fifteen years. This was done through the development of its algebraic structure to some Dedekind domain and Euclidean rings such as $Z[i]$, and $GF(2^k)[x]$. The first generalization of NTRU to Euclidean integer was proposed by Gaborit, et al. [4]. Through his initiative suggestion of replacing NTRU algebraic structure with other rings, he referred to it CTRU. In 2005, Coglianese et al. [5] improved the *NTRU* cryptosystem by replacing its original ring with a $k \times k$ matrices ring of polynomial with order *n*, known as *MaTRU*. It has improved speed by a factor of $O(k)$ over NTRU. In 2009, Malekian et al. [6] presented the *QTRU* cryptosystem. It was a multi–dimensional public key using quaternion algebra extended ring, which is broader than Dedekind domain and Euclidean algebra. Their underlying algebraic structure was non–commutative. This implied keeping the positive points of NTRU, and making it more resistant to some lattice-based attacks [7]. Another framework based on the Eisenstein integers Z [w], was presented by Jarvis [8] in 2011. This ring is defined as a cube root of unity and the coefficients are integers from Z. They called it *ETRU*, and showed that *ETRU* had improved the *NTRU* security [9].

In this paper a new *NTRU* cryptosystem is proposed using commutative ring of quaternions *CQ*. It has the same structure of *QTRU* but depends on the polynomial algebra with coefficients in *CQ*. It will be referred to as *CQTRU*. Some conditions on the parameter selection are placed to alllow the proposed system high chance for successful decryption.

The text of this paper is organized in the following way: a brief summarization of the *NTRU* cryptosystem is presented in Section 2. Some mathematical description of the alternative *CQ*, as a base ring for the proposed system, is discussed in Section 3. In Section 4, the proposed *CQTRU* is introduced, whereas the implementation of *CQTRU* with the improvement of the decryption failure probability is presented in Section 5. The performance analysis is discussed in Section 6, and the conclusions are presented in Section 7.

II. THE NTRU CRYPTOSYSTEM

A simple description of the NTRU cryptosystem is summarized in this section. For more details, the reader is referred to [1, 10-14]. The NTRU system is principally based on the ring of the convolution polynomials of degree *N-1* denoted by $R=Z[x]/(x^N-1)$. It depends on three integer parameters *N, p* and *q*, such that, $(p, q)=1$. Before going through *NTRU* phases, there are four sets used for choosing *NTRU* polynomials with small positive integers denoted by L_m, L_f, L_g and $L_r \subset R$. It is like any other public key cryptosystem constructed through three phases: key generation, encryption and decryption.

A. Key Generation phase

To generate the keys, two polynomials *f and g* are chosen randomly from L_f and L_g respectively. The function *f* must be invertible. The inverses are denoted by F_p , $F_q \in R$, such that:

$$F_p * f \equiv 1 \ (mod\ p) \quad \text{and } F_q * f \equiv 1 (mod\ q)$$

The above parameters are private. The public key *h* is calculated by,

$$h = p\, F_q * g \ mod\ q \qquad (1)$$

Therefore; the public key is {*h, p, q*}, and the private key is: {*f* , F_p}.

B. Encryption phase

The encryption is done by converting the input message to a polynomial $m \in L_m$ and the coefficient of *m* is reduced *modulo p*. A random polynomial *r* is initially selected by the system, and the cipher text is calculated as follows,

$$e = r * h + m \quad mod\ q. \qquad (2)$$

C. Decryption phase

The decryption phase is performed as follows: the private key, *f,* is multiplied by the cipher text *e* such that,

$$f * e \ \ mod\ q = f*(p.h*r+m\ )\ m\ od\ q$$
$$= p.f*h*r+f*m \ \ mod\ q$$
$$= p.f*F_p^{-1}*g*r+f*m \ \ mod\ q$$
$$= p.g*r+f*m \quad mod\ q$$

The last polynomial has coefficients most probably within the interval (*-q/2,q/2*], which eliminates the need for reduction *mod q*. This equation is reduced also by *mod p* to give a term *f*m mod p,* after diminishing of the first term *p.g*r*. Finally, the message *m* is extracted after multiplying by $Fp^{-}1$, as well as adjusting the resulting coefficients via the interval [*-p/2, p/2*).

III. Algebraic structure of CQTRU

The suggestion of replacing the original ring of *NTRU* with other rings Gaborit et al. [4], and based on *NTRU* structure, a new scheme for *NTRU* cryptosystem that depends on polynomial algebra with coefficients in the commutative ring of quaternions *CQ* is proposed to introduce a new cryptosystem called *CQTRU*. Prior to establishing the validity of the proposed system, The *CQ* ring should be defined with its addition and multiplication operations, and the existence of the multiplicative inverses [15-19].

A. Commutative Quaternions (CQ)

In a four-dimension vector space, a commutative quaternions set is denoted by *CQ*, and defined as:

$$CQ = \{\ a = t + xi + yj + zk : t,x,y,z \in R \text{ and } i,j,k \notin R\}.$$

Where; *i, j, k* satisfy the following multiplication rules: $i^2=k^2 = -1$, $j^2= 1$ and $ij= k$.

In this paper, *i, j* and *k* are defined as $i^2= a$, $j^2=b$, $k^2 = ab$ and $ij=k$. By this definition, a general commutative algebraic system is defined. Assuming *F* is an arbitrary field, the commutative quaternion algebra *A* can be defined over *F* as:

$$A = \{a + bi + cj + dk \mid a, b, c, d \in F, i^2 = a, j^2 = b, i\, j = k\}.$$

Clearly, if we assume that $a = -1$, $b = 1$ and *F* be the field of real numbers *R*, then, based on the choices of *a* and *b* and the nature of the field *F*, the original definition of commutative quaternion is obtained.

Let A_0 and A_1 be two commutative quaternion algebras such that:

$A_0=\{f_0 + f_1.i + f_2\, j + f_3.k \mid f_0, f_1, f_2, f_3 \in R_p, i^2= -1, j^2=1, ij=k\}$ and
$A_1=\{g_0 + g_1.i + g_2.j + g_3.k \mid g_0, g_1, g_2, g_3 \in R_q\ , i^2= -1, j^2=1, ij= k\}.$

Assume that $a_0, a_1 \in A_0$ *(or* A_1 *)*, such that, $a_0 = t_0 + x_0.i + y_0.j + z_0.k$ and $a_1 = t_1 + x_1.i + y_1.j + z_1k$. Then, the operation on these two commutative quaternions; i.e. addition, multiplication and multiplicative inverse, will be given as:

$$a_0 + a_1 = (t_0 + t_1) + (x_0 + x_1)i + (y_0 + y_1)j + (z_0+z_1)k$$

$$a_0 \cdot a_1 = (t_0\, t_1 - x_0\, x_1 + y_0\, y_1 - z_0\, z_1) + (x_0\, t_1 + t_0\, x_1 + z_0\, y_1 + y_0\, z_1)i + (t_0\, y_1 + y_0\, t_1 - x_0\, z_1 - z_0\, x_1)j + (z_0\, t_1 + t_0\, z_1 + x_0\, y_1 + y_0\, x_1)k.$$

B. Multiplicative invers in CQ Algebra

In *NTRU* public key cryptosystem scheme, the most important factor is the existence of the multiplicative inverse. For any element *a* in *CQ* to be used in *CQTRU*, the existence of its multiplicative inverse module *p* and *q* has to be checked.

For each $a \in CQ$, *a* can be represented by a 2×2 complex matrix, such that, if $a=a_0+b_0i+c_0j+d_0k \in CQ$, then *a* can be uniquely represented as $a=c_1+jc_2$, where $c_1=a_0+b_0i$, and $c_2=c_0+d_0i$, $c_1, c_2 \in C$. Here C is the set of complex numbers [10].

Hence, for $a=c_1+jc_2$, $\phi(a) = \begin{pmatrix} c_1 & c_2 \\ c_2 & c_1 \end{pmatrix}$, where ϕ is a bijective map.

Knowing $\phi(a)^{-1}$, the multiplicative inverse a^{-1} of $a \in CQ$ is calculated as follows:

If $(\alpha)^2+(\beta)^2 \neq 0$, then $a^{-1} = \delta_0+\delta_1 i+\delta_2 j+\delta_3 k$,, where $\alpha=[\ a_0^2 + b_0^2 + c_0^2 + d_0^2\ ]$, $\beta= [2a_0*b_0-2c_0*d_0]$.

Let $(\alpha^2+\beta^2)^{-1} = \partial$, then we have $[\delta_0= \partial(\alpha*a_0-\beta*b_0)$, $\delta_1= \partial(\alpha*b_0+\beta*a_0)$, $\delta_2=\partial(\beta*d_0-\alpha*c_0)$, *and* $\delta_3=\partial(\alpha*d_0+\beta*c_0)]$.

IV. THE PROPOSED CQTRU CRYPTOSYSTEM

In order to obtain a full understanding of how the *CQTRU* cryptosystem works, the algebraic structure for key generation, encryption and decryption, is designed as follows.

At the beginning, the parameters *N, p, q* have the property that *N* is an integer, *p* and *q* are relatively prime, and in all the algorithms, the parameter *m* represents either *p* or *q* depending upon which one is passed into the function.

A. Key Generation phase

To generate the public key, two small commutative quaternion *F* and *G* are randomly generated, such that

$$F = f_0 + f_1 i + f_2 j + f_3 k \in L_f .$$
$$G = g_0 + g_1 i + g_2 j + g_3 k \in L_g .$$

As it was mentioned above, *F* is invertible over A_0 and A_1 if $(\alpha^2+\beta^2)$ is invertible in Z_{cp} and Z_{cq}. Otherwise; a new commutative quaternion is generated. The inverses of *F* over Z_{cp} and Z_{cq} are denoted by F_p and F_q respectively.

Now, the public key is calculated as follows:

$$H = F_q \cdot G \mod q$$
$$= (f_{q0} * g_0 - f_{q1} * g_1 - f_{q2} * g_2 - f_{q3} * g_3) +$$
$$(f_{q1} * g_0 + f_{q0} * g_0 + f_{q2} * g_3 + f_{q3} * g_2) i +$$
$$(f_{q0} * g_2 + f_{q2} * g_0 - f_{q1} * g_3 - f_{q3} * g_1) j +$$
$$(f_{q0} * g_3 + f_{q3} * g_0 + f_{q2} * g_1 + f_{q1} * g_2) k .$$

The commutative quaternions *F*, F_p and F_q will be kept secret in order to be used in the decryption phase. It is obvious that the estimated time to generate a key for the proposed scheme is 16 times slower than that of *NTRU*, when the same parameters (*N, p and q*) are selected for both cryptosystems. However, with a lower dimension *N*, we can achieve the original *NTRU* speed.

As mentioned previously, the new system is a 4-dimension space. Hence, if one chooses the coefficients of *i, j* and *k* to be zeros in the commutative quaternions *F* and *G*, then the system will be completely similar to *NTRU*. Moreover, this choice of zero coefficients for *j* and *k* will yield a cryptosystem based on complex numbers. Finally, if one of the coefficients of *I*, *j* or *k* is equal to zero, we obtain a tridimensional scheme.

B. Encryption phase

At the beginning of the encryption process, the cryptosystem must generate a random commutative quaternion called the blinding quaternion. The input message should be converted into a commutative quaternion. The cipher text will be computed and sent in the following way:

Let $M = m_0 + m_1 i + m_2 j + m_3 k$

where, $m_0, m_1, m_2, m_3 \in L_m$, generate a random quaternion $R = r_0 + r_1 i + r_2 j + r_3 k$, and $r_0, r_1, r_2, r_3 \in L_r$.

Hence, the encryption function used is:

$$E = p.H \cdot R + M \mod q \quad (3)$$

In this phase, a total of four data vectors are encrypted at the same time.

C. Decryption phase.

After receiving the cipher text *E*, the original message is constructed as follows.

The private key *F* is used to find *B*:

$$B = F \cdot e \mod q \quad (4)$$

The coefficient of *B* should be reduced mod *q* into the interval $(-q/2, q/2]$.

The next step in the decryption process is to calculate the commutative quaternion *D*.

$$D = F_p \cdot B \mod p. \quad (5)$$

The original message is obtained by reducing *D* in the interval $[-p/2, p/2]$.

D. How Decryption Works :

Since $B = F \cdot E \mod q$
$= (F \cdot (p.H \cdot R + M)) \mod q$
$= (F \cdot p.H \cdot R + F \cdot M) \mod q,$

the value of *H* is substituted to get,

$$B = (pF \cdot F_q \cdot G \cdot R + F \cdot M) \mod q$$
$$= (pG \cdot R + F \cdot M) \mod q .$$

Since $D = F_p \cdot B \mod p$, then

$$D = F_p \cdot (pG \cdot R + F \cdot M) \mod p$$
$$= (F_p \cdot pG \cdot R + F_p \cdot F \cdot M) \mod p$$

The term $(F_p \cdot pG \cdot R)$ will be disappear after reducing mod *p*, to obtain the term $(F_p \cdot F \cdot M)$.

Since $F_p \cdot F = 1 \mod p$, normalizing the result into the interval $(-p/2, +p/2]$ yields the original message *M*. Therefore, the decryption speed is half the encryption speed because decryption includes 32 convolutions product. This is clearly analogous to the *NTRU* cryptosystem.

V. IMPLEMENTATION AND EXPREMENTS

Both CQRTU and NTRU are implemented in Matlab. The experiments were performed on a PC with 2.4 GHZ Intel Core 3, Quad processor and 4 MB Ram under windows 7, 32 bit operating system. For $p=3$, key generation, encryption and decryption speed with the probability of successful decryption are shown in Table 1. The probability of decryption failure depends on the choice of public parameters.

However, when N is fixed and the other parameters take larger values, the probability of decryption failure is decreased.

Table 1. Speed & probability of successful decryption, $p=3$

					Time in (ms)			
N	q	d_f	d_g	d_r	Gen.	Encr.	Decr.	Pro(failure)
73	128	10	8	5	65.3	10.9	18	0.000051782
73	128	12	10	6	67	12	18	0.0000003176
107	192	15	12	5	115	28	52	0.000288
107	192	20	12	10	116	27	50	0.0000028248
149	256	20	12	10	142	32	63	0.0000001192
149	256	35	25	20	145	33	60	0.0005515
167	256	40	20	18	186	36	68	0.00083229
167	256	50	21	19	186	39	70	0.00002532
211	256	40	20	18	275	53	93	0.000021775
211	256	30	24	22	278	53	94	0.000005822
257	256	40	20	18	350	71	126	0.0000004112
257	256	30	24	24	356	72	124	0.0000076072

A. *Decryption failure*

The probability of decryption failure is decreased if all commutative quaternion coefficients of $F \bullet E=(pG \bullet R+F \bullet M)$ lie in the interval $(\frac{-q}{2},\frac{q}{2}]$. For the *CQTRU*, this probability is computed as follows:

To calculate $var[a_{i,\ j}]$, it is sufficient to assume that $E[f_{i,k}]\approx 0$, $E[g_{i,k}]=E[r_{i,k}]=E[m_{i,k}]=0$, $E[a_{i,k}]=0$ where $i=0,\ 1,\ 2,\ 3$ and $k=0,\ldots,N-1$, and E is the mean function. Since each coefficient of quaternion element is a polynomial of degree N, then we have

$$Var\left[r_{i,k}, g_{j,i}\right]=\frac{4d_r \cdot d_g}{N^2}$$

$$Var\left[f_{i,k}, m_{j,i}\right]=\frac{d_f(p-1)(p+1)}{6N}$$

$$Var\left[a_{0,k}\right]=\frac{16p^2 d_r \cdot d_g}{N}+\frac{4d_f(p-1)(p+1)}{6}$$

$$\Pr\left(\left|a_{i,k}\right|<\frac{q}{2}\right)=2\phi\left(\frac{q-1}{2\sigma}\right)-1$$

Where; ϕ denotes the distribution of the standard normal variable, and

$$\sigma=\sqrt{\frac{16p^2 d_r d_g}{N}+\frac{4d_f(p-1)(p+1)}{6}}\ .$$

$a_{i,k}$'s are assumed to be independent random variables. The successful decryption probability in *CQRTU* can be calculated by the following two observations:

$$\left(2\phi\left(\frac{q-1}{2\sigma}\right)-1\right)^N,\ \left(2\phi\left(\frac{q-1}{2\sigma}\right)-1\right)^{4N} \qquad (6)$$

VI. PERFORMANCE ANALYSIS

After comparing *NTRU* to other cryptosystems, such as *RSA* and *ECC*, which are based on the number theoretic problem (e.g., factorization and discrete logarithm) [20], *NTRU* was found to have an advantage over them due to its fast and low space storage arithmetic operations. This turned *NTRU* into a very suitable choice for a large number of applications.

A. *Computational complexity*

For encryption, one commutative quaternion multiplication is needed in addition to 16 convolution multiplication and 4 polynomial addition; both with *O(N)* complexity. In the encryption phase, any incoming data is converted into polynomial with coefficients between $-p/2$ and $p/2$. In other words, m_0, m_1, m_2 and m_3 are small polynomials modulo q.

B. *Security Attacks*

1- Alternate keys analysis in *CQTRU*

Compared to *NTRU*, any alternate of the private key f can be used to encrypt and decrypt the same messages as f. The attacker needs only to find one polynomial having the same properties of f. In *CQTRU*, to find the alternate private key F, the attacker needs to find four polynomials of the same properties of the privet key F. Hence, *CQTRU* is more robust to this attack than *NTRU*. Accordingly, it is considered to be more secure than *NTRU*.

2- Brute Force Attacks

Compared to *NTRU*, to recover the privet key f; an attacker has to try using all possible $f' \in L_f$ in an attempt to check if $f * h \bmod q$ has small polynomial coefficients or not. Another way is to try all possible $g' \in L_g$ and check if $g * h^{-1} \bmod q$ has small coefficients. In *CQTRU*, the attacker uses the same procedure, where he/she knows all the public parameters and constant d_r, d_g, d_f, q, p, and N. The attacker needs to look in the space of large order to be able to look in the spaces L_f and L_g, as follows:

$$\left|L_f\right|=\binom{N}{d_f}^N\binom{N-d_f+1}{d_f}=\frac{(N!)^4}{(d_f!)^8(N-2d_f)!^4}$$

$$\left|L_g\right|=\binom{N}{d_g}^N\binom{N-d_g+1}{d_g}=\frac{(N!)^4}{(d_g!)^8(N-2d_g)!^4}$$

The space of L_f is a bigger than the space of L_g. For this reason, it is easier for the attacker to search in L_g. By using the brute force attack, an attacker can break a message encrypted by *CQTRU*. This can be done by searching in the space L_r because $E=H \bullet R+M \pmod q$ is known. If the attacker has an ability to find the random commutative quaternion R then he/she will be able to find the original message by calculating

$M=E-H\bullet R$ (*mod q*). It is obvious that in a brute force attack, the security of any message depends on how hard it is to find R. The order of the space L_r is calculated using the same approach of calculating the order of L_f and L_g,

$$|L_r| = \binom{N}{d_r}^N \binom{N-d_r+1}{d_r} = \frac{(N!)^4}{(d_r!)^8 (N-2d_r)!^4}$$

This comparison shows that *CQTRU* is more robust to this attack than *NTRU*.

3- Lattice based attacks

It is known that every commutative quaternion is isomorphism to a matrix called the fundamental matrix given in (7):

$$q = q_0 + q_1 i + q_2 j + q_3 k \cong \begin{pmatrix} q_0 & -q_1 & q_2 & -q_3 \\ q_1 & q_0 & q_3 & q_2 \\ q_2 & -q_3 & q_0 & -q_1 \\ q_3 & q_2 & q_1 & q_0 \end{pmatrix} \quad (7)$$

The system parameters (d_f, d_g, d_r, p, q, N) are known to the attacker as well as the public key $H=F_q\bullet G=h_0+h_1 i+h_2 j+h_3 k$. When the attacker manages to find one of the commutative quaternions F or G, the *CQTRU* cryptosystem is broken. Note that, h_0, h_1, h_2 and h_3 are polynomials of order N over Z. These polynomials can be represented as vectors over Z^N as follows:

$H= h_0+h_1 i+h_2 j+h_3 k \cong [h_0\ h_1\ h_2\ h_3]$, where

$h_0 = h_{0,0} + h_{0,1} x + \ldots\ldots\ldots + h_{0,N-1} x^{N-1}$

$\cong [h_{0,0}\ h_{0,1} \ldots\ldots\ldots\ldots\ldots h_{0,N-1}]$,

$h_1 = h_{1,0} + h_{1,1} x + \ldots\ldots\ldots + h_{1,N-1} x^{N-1}$

$\cong [h_{1,0}\ h_{1,1} \ldots\ldots\ldots\ldots\ldots h_{1,N-1}]$,

$h_2 = h_{2,0} + h_{2,1} x + \ldots\ldots\ldots + h_{2,N-1} x^{N-1}$

$\cong [h_{2,0}\ h_{2,1} \ldots\ldots\ldots\ldots\ldots h_{2,N-1}]$,

$h_3 = h_{3,0} + h_{3,1} x + \ldots\ldots\ldots + h_{3,N-1} x^{N-1}$

$\cong [h_{3,0}\ h_{3,1} \ldots\ldots\ldots\ldots\ldots h_{3,N-1}]$

Since the polynomial ring Z is isomorphic to the circulant matrix ring of order N over Z, the polynomials h_0, h_1, h_2 and h_3 can be represented in their isomorphic representation for lattice analysis as:

$$h(i)_{N\times N} = \begin{pmatrix} h_{i,0} & \cdots & h_{i,N-1} \\ h_{i,N-1} & \cdots & h_{i,N-2} \\ \vdots & \ddots & \vdots \\ h_{i,2} & \cdots & h_{i,1} \\ h_{i,1} & \cdots & h_{i,0} \end{pmatrix} \quad (8)$$

where $i=0, 1, 2, 3$.

With respect to the above assumptions, to describe the partial lattice attack first, let the commutative quaternions F and G be represented by $F=[f_0\ f_1\ f_2\ f_3]$, and $G=[g_0\ g_1\ g_2\ g_3]$ where $f_0, f_1, f_2, f_3, g_0, g_1, g_2, g_3 \in Z[x]/(x^N-1)$. In order to form the lattice, the vectors $[u_0\ u_1\ u_2\ u_3\ v_0\ v_1\ v_2\ v_3]$ must belong to Z^{8N}. This lattice is denoted by $L_{partial}$ and defined by:

$$L_{partial} = \begin{pmatrix} I_{4N\times 4N} & 0_{4N\times 4N} \\ H_{4N\times 4N} & q_{4N\times 4N} \end{pmatrix} \in Z^{8N} \quad (9)$$

where, I refers to the identity matrix, 0 is the zero matrix, and H is the fundamental matrix of h_i's. $L_{partial}$ contains a vector in the form $[u_0\ u1\ u2\ u_3\ v_0\ v_1\ v_2\ v_3] \in Z^{8N}$, that satisfies $F\bullet H=G$. However, there is a major difference between *NTRU* and *CQTRU* lattices, such that all points spanned by the *CQTRU* lattice merely includes a partial subset of the total set of vectors satisfying $F \bullet H=G$. To see this, let $[u_0\ u_1\ u_2\ u_3\ v_0\ v_1\ v_2\ v_3]$ denote the vector satisfying $F\bullet H=G$, then $[-u_1\ u_0\ -u_3\ u_2\ -v_1\ v_0\ -v_3\ v_2]$ is the answer. Also, since $iF \bullet H=iG$, therefore, $L_{partial}$ will not necessarily contain such vector. The attacker may manage to use the lattice reduction algorithm [21-22] to find a short vector satisfying $F\bullet H=G$. However, even with such promising assumption, $L_{partial}$ has a dimension that is four times larger than the lattice dimension of *NTRU* with the same order N. Hence, the *CQTRU* with the parameters ($N=107$, p, q) offers the same level of security as *NTRU* with the parameters ($N=428$, p, q). Therefore, for any chosen parameters (N, p, q) to be used in *CQTRU*, the system will be four times slower than *NTRU* with the same parameters as it is shown by Tables (2) - (4), which demonstrate that for the three phases; key generating, encryption and decryption, *CQTRU* is also slower than *NTRU* under the same environments. However, the *CQTRU* security is four times as that offered by *NTRU* with the same parameters. On the other hand, *NTRU* with $4N$ dimensions is sixteen times slower with respect to computational time than *NTRU* with N dimensions. Therefore, *CQTRU* has a security advantage over *NTRU*.

Table 2. Key generating time in ms for *NTRU* and *CQTRU*

N	q	d_f	d_g	d_r	*NTRU*	*CQTRU*
73	128	10	8	5	20	67
107	128	15	12	5	40	116
149	192	20	15	10	52	142
167	192	25	22	18	56	186
211	256	28	25	22	76	278
257	256	33	30	28	98	356

Table 3. Encryption time in ms for NTRU and CQTRU

N	q	d_f	d_g	d_r	*NTRU*	*CQTRU*
73	128	10	8	5	3.1	10.9
107	128	15	12	5	8	28
149	192	20	15	10	9.5	32
167	192	25	22	18	11	39
211	256	28	25	22	14	53
257	256	33	30	28	19	71

Table 4. Decryption time in ms for NTRU and CQTRU

N	q	d_f	d_g	d_r	*NTRU*	*CQTRU*
73	128	10	8	5	5.2	18
107	128	15	12	5	14	52
149	192	20	15	10	17	63
167	192	25	22	18	19	70
211	256	28	25	22	24	93
257	256	33	30	28	33	126

The partial lattice attacks do not always give successful results because $L_{partial}$ does not necessarily contain all solutions of $F \bullet H=G$ in such a way that f_0, f_1, f_2, f_3, g_0, g_1, g_2, or g_3 would be short vectors. Therefore, the attacker must find a lattice that contains all vectors which satisfy the congruence $F \bullet H=G$.

VII. Conclusions

By changing the underlying ring of *NTRU*, the *NTRU* cryptosystem has been improved through the introduction of a new *NTRU* like public key cryptosystem. This is constructed by replacing the base ring of *NTRU* with a commutative quaternions ring that resulted in the emergence of *CQTRU* cryptosystem. Despite the apparent increase in computational time, it is considered to be reasonable with consideration to its higher complexity. This generalization of the algebraic structure of the *NTRU* resulted also in an improved security level over *NTRU*, and a significant improvement in the reduction of the decryption failure probability.

References

[1] J. Hoffstein, J. Pipher, and J. H. Silverman, "NTRU: A ring-based public key cryptosystem," in *Lecture Notes in Computer Science*. Springer-Verlag, vol. 1423, 1998, pp. 267–288.

[2] N. Smart, F. Vercauteren, J. H. Silverman, "An algebraic approach to NTRU (q = 2^n) via Witt vectors and overdetermined systems of nonlinear equations", SCN 2004, Amalfi, Italy, LNCS vol. 3352, Springer, 2004.

[3] J. Hoffstein and J. Silverman, "Optimizations for ntru," in *In Public Key Cryptography and Computational Number Theory*, 2000, pp. 11–15.

[4] P. Gaborit, J. Ohler, P. Sole, "CTRU, a polynomial analogue of NTRU", INRIA, Rapport de recherche 4621, INRIA 2002, ftp://ftp.inria.fr/INRIA/publication/publi-pdf/RR/RR-4621.pdf

[5] M. Coglianese and B.-M. Goi, "MaTRU: A new NTRU-based cryptosystem," in *INDOCRYPT*, 2005, pp. 232–243.

[6] E. Malekian, A. Zakerolhosseini, "NTRU-Like Public Key Cryptosystems beyond Dedekind Domain up to Alternative Algebra", Transactions on Computational Science X Lecture Notes in Computer Science Volume 6340, 2010, pp 25-41

[7] Malekian E., Zakerolhosseini A., Mashatan A.: QTRU: a lattice attack resistant version of NTRU PKCS based on quaternion algebra (preprint). Available from the Cryptology ePrint Archive: http://eprint.iacr. org/2009/386.pdf. Accessed Sep. 2012.

[8] K. Jarvis, "NTRU over the Eisenstein integers", Masters Thesis, University of Ottawa, 2011.

[9] K. Jarvis, M. Nevins, "ETRU: NTRU over the Eisenstein integers", Designs, Codes and Cryptography, vol.74, No.1, 2015, pp 219-242.

[10] D. Coppersmith and A. Shamir, "Lattice attacks on NTRU", in *EUROCRYPT*, 1997, pp. 52–61.

[11] M. Nevins, C. Karimianpour, and A. Miri, "Ntru over rings beyond z," Designs, Codes and Cryptography, vol. 56, No1, 2010, pp.65–78.

[12] R. Kouzmenko, "Generalizations of the NTRU cryptosystem," Master's thesis, Polytechnique, Montreal, Canada, 2006.

[13] J. Hoffstein, J. Pipher, and J. H. Silverman, *An Introduction to Mathematical Cryptography*, ser. Science+Business Media, LLC.2ed edition, Springer, 2014.

[14] J. Pipher, "Lectures on the NTRU encryption algorithm and digital signature scheme", Brown University, Providence RI 02912Grenoble, June 2002.

[15] F. Catoni, R. Cannata and P. Zampetti, "An Introduction to Commutative Quaternions", Adv. appl. Clifford alg. vol.16, No.1, 2006, pp.1–28.

[16] R. D. Schafer, *An introduction to non-associative algebras*. New York: Dover Publications Inc., 1996, corrected reprint of the 1966 original.

[17] W.D. Banks, I.E. Shparlinski, "A Variant of NTRU with Non-Invertible Polynomials", In: Menezes, A., Sarkar, P. (eds.) INDOCRYPT 2002, LNCS, Springer, Heidelberg vol. 2551, 2002, pp. 62–70.

[18] J. C. Baez, "The octonions," Bulletin of the American Mathematical Society, vol. 39, No. 2, 2002, pp. 145–205.

[19] J. H. Conway and D. A. Smith, *On Quaternions and Octonions: Their Geometry, Arithmetic, and Symmetry*. A. K. Peters, Ltd., 2003.

[20] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of applied cryptography*. Boca Raton, Florida: CRC Press, 1996.

[21] J. Hoffstein, J. H. Silverman, and W. Whyte, "On estimating the lattice security of NTRU," Technical Report 104, Cryptology ePrint Archive (2005), http://eprint.iacr.org/2005/104/ 20, 2005.

[22] J. H. Silverman, "Dimension-reduced lattices, zero-forced lattices, and the NTRU public key cryptosystem," Technical Report No. 13 (1999). Available at (http://www.ntru.com), 1999.

Split Manufacturing in Radio-Frequency Designs

Yu Bi, *Student Member, IEEE,* Jiann-Shiun Yuan, *Senior Member, IEEE,* and Yier Jin, *Member, IEEE*
Department of Electrical Engineering and Computer Science, University of Central Florida

***Abstract*—With the globalization of integrated circuit (IC) design flow and the outsourcing of chip fabrication service, intellectual property (IP) piracy and malicious logic insertion become main security threats to tamper hardware infrastructures. While most of the protection methods are dedicated for digital circuits, we try to protect radio-frequency (RF) designs which are more likely to be IP piracy victims. For the first time, we apply the split manufacturing method in RF circuit protection. Three different implementation cases are introduced for security and design overhead tradeoffs, i.e., the removal of the top metal layer, the removal of the top two metal layers, and the design obfuscation dedicated for RF circuits. We also develop a quantitative security evaluation method to measure the protection level of RF designs under split manufacturing. Finally, a class-AB power amplifier is used for demonstration through which we prove that: 1) the removal of top metal layer or top two metal layers can provide high-level protection for RF circuits with lower request to the domestic foundries; 2) design obfuscation method provides highest level of circuit protection, though at the cost of design overhead; 3) split manufacturing is more suitable to RF designs than to the digital circuits and it can effectively improve hardware tamper resistance and reduce IP piracy in the untrusted off-shore foundries.**

***Keywords*-Hardware Tamper Resistance, Hardware Trust, IP Piracy, Power Amplifier, RF Circuits, Split Manufacturing**

I. Introduction

The globalization of integrated circuits (IC) supply chain, especially the outsourcing of chip fabrication and the integration of third-party intellectual property (IP) cores, breeds security concerns and makes it easier to compromise the once trusted IC development process [1], [2]. Among all security threats, malicious logic insertions (aka hardware Trojan attacks) and IC piracy are of the most critical security threats that the US government is facing after more and more domestic IC companies go fabless. Following the trend of a growth of merchant foundry industry, fabless IC design houses can have access to reasonably-priced advanced-process capacity without the need for huge capital expenditure (the cost of developing a semiconductor foundry will be over $5.0 billion by 2015 [3]). The reduced fabrication cost, at the same time, sacrifices the design security and leaves all IC designs in the hands of foundry. The International Chamber of Commerce (ICC) stated in their 2011 report that the total global economic and social impacts of counterfeit and pirated products are as much as $775 billion every year.

For this reason, both governmental agencies and industrial companies are looking for a balance between fabrication cost and design security to prevent foundry from learning the design details of the submitted design layout. In order to address such threats, various hardware Trojan detection methods and hardware metering methods have been developed [4]–[7]. Among all these approaches, design obfuscation and camouflaging are candidates but both methods require the modification to the original circuits which may cause performance overhead. Intelligence Advanced Research Projects Activity (IARPA) proposed a new methodology, called split manufacturing, which only adds trivial efforts to IC designers but can effectively prevent IC piracy [8]. The key idea of split manufacturing is to protect circuit/system designs by dividing the manufacturing chips into Front-End-of-Line (FEOL) consisting of transistor layers to be fabricated by offshore foundries and Back-End-of-Line (BEOL) consisting of metallizations to be fabricated by trusted domestic facilities. Through this divided fabrication procedure, the design intention is not fully disclosed to the FEOL foundry. Even though the concept is straightforward, a successful implementation requires further research on various aspects, especially the balance between cost and security when the designer splits the layout into FEOL and BEOL. Analytical and experimental results have already been presented in digital circuits [9]–[13]. However, the analog/RF designs are rarely discussed using split manufacturing even though analog/RF circuits are more likely to be IP piracy victims than their digital counterparts.

In fact, the fundamental difference between digital design flow and RF design process has already raised the concern whether it is still applicable to apply split manufacturing in RF design. A deep look into both design flows proves us that it would be more suitable to apply split manufacturing in RF circuits than in digital circuits because of the unique functionality metal layers play in RF designs: 1) Metal layers are solely used as interconnections between gates and modules in digital circuits while in RF circuits, metal layers are also used to build functional blocks (e.g., inductors are often located on top metal layer; capacitors are built in upper level metal layers); 2) While metal layers are abstracted as wire connections in digital designs, wire length and wire direction are both functional parameters in RF designs. Therefore, a foundry fabricating the FEOL part of digital circuits may face a mathematical problem with finite solutions in order to recover the whole functionality of the design[1]. On the other hand, the foundry of RF FEOL needs to explore an infinite solution space to recover the RF design.

Based on the above discussion, it becomes obvious that the split manufacturing should be more effective to protect RF circuits from IP piracy. To assess our claim, analytical calculation and experimental demonstration are performed in this paper to solidify our findings and to push the territory of split manufacturing to cover all kinds of circuit designs.The rest of the paper is organized as follows: Section II introduces

[1]Note that the possible solution space could be large given large amount of standard cells in digital circuits. In fact, this is the key criterion to evaluate the security level of split manufacturing method in digital circuits.

the state-of-the-art split manufacturing practices. Section III presents the RF design flow. A detailed analytical analysis of applying split manufacturing in RF designs is presented in Section IV. Finally, the conclusions are drawn in Section V.

II. Split Manufacturing in Digital Domain

The concept of split manufacturing was officially proposed by IARPA through the Trusted Integrated Chips (TIC) program. The new program aims to develop and demonstrate new split manufacturing to chip fabrication where security and intellectual property protection can be assured [8]. Since then, a few embodiments of split manufacturing in digital circuits have been proposed. Imeson et al. proposed a method by applying 3D integration technology in split fabrication. Using a through silicon via (TSV), they came up with a security algorithm from graph problem to obfuscate the circuit by lifting certain wires to a trusted tier [10]. Rajendran et al. examined a split fabrication after metal3 layer, where digital benchmark circuits are separated into several partitions without interconnections [9]. Since the connections within each gate are mostly located in metal1 and metal2 layers which are known to the FEOL foundry, they further proposed a fault-analysis based pin swapping algorithm to defend the common proximity attacks. More recently, Vaidyanathan et al. investigated feasibility of split fabrication after metal1 layer so that untrusted foundries only have the information of basic gate-level blocks [11]. A similar technique is then applied to digital/analog IP designs [12]. A defense strategy against recognition attacks on IPs and an obfuscation method were both proposed as well as experimental demonstrations on a 1KB SRAM and a 14-bit current steering digital-to-analog converter (DAC). Hill et al. [13] described a comparative study of an asynchronous FPGA manufactured in both a standard process and a split manufacturing process. Compared to the standard process, split manufacturing process suffers penalties either on operating frequency or on the energy consumption.

III. RF Design Flow

Thanks to the advanced EDA tools for RF circuit designs and the development of RF design kits, RF engineers become more productive than ever before. Nevertheless, a typical RF design still involves heavy work of design fine-tuning and designers' experience plays a critical role here [14], [15]. Figure 1 shows the steps among a modern RF design flow.

From Figure 1 we can learn that steps I-III are the preparation of the RF circuit specification. Taking a power amplifier as an example, the defined specification will include design information such as the delivered output power, the amount of circuit stages, the operation class, etc. Different from digital designs where the specification will be strictly followed, however, the specification for RF circuits only serves as a guideline as it often happens that the performance of the final design deviates from the original settings (experienced RF engineers may be able to narrow the performance gap).

Guided by the specification, the circuit schematic will be designed, simulated and optimized. The optimized schematic will then guide the work of layout design and post-layout

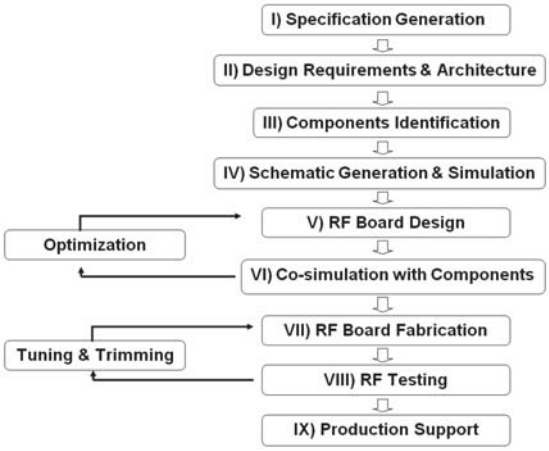


Fig. 1: Standard RF circuit design flow

simulation. All physical-level parameters come into the map during the layout design and post-layout simulation such as parasitic capacitors, wire resistance, etc. For RF circuits, the parasitic components can significantly affect the design performance and significantly deviate the circuit performance from the schematic level simulation. Therefore, the large portion of design time will be spent in layout optimization and circuit fine-tuning, even for experienced designers. If the circuit passes the post-layout simulation, it will be sent to foundry for fabrication and for post-fabrication testing. Even though current foundries embrace advanced technology and delicate equipments, the parasitics introduced by fabrication process remain a problem, i.e., unpredictable parasitic resistance and capacitance during the fabrication would both affect circuit functionality and performance. A fabricated RFIC circuit may not work properly which raises the demand of further tuning and trimming. To lower the fabrication cost and to increase the yield rate, techniques of post-fabrication calibration are used in modern RF designs, e.g., knob adjustments and Transverse Electro-Magnetic (TEM) cell.

IV. Split Manufacturing in RF Circuits

As we mentioned earlier, the removal of metal layers in RF circuits will not just hide the interconnections between circuit components but also eliminate the passive components which are built in metal layers. Since a typical RF circuit only includes very few transistors and other passive components, the recovery of interconnections between these components will not be a difficult task. Rather, to derive the missing passive components and their sizes would be the main advantage to apply split manufacturing in RF designs. For the same reason, the difficulty level for attackers with the FEOL at hand to recover the passive components and to guess the sizes of these passive components will be the key criteria to assess the effectiveness of split manufacturing application in RF designs.

Compared to digital split fabrication [9] where the proximity attack dominates the security analysis, routing and mapping are no longer an issue for RF circuits. Furthermore, the recognition attack mechanism used in [12] cannot explain accurately the issue with RF split fabrication. To better guide

the implementation of split manufacturing in RF circuits and to balance between the security level and design efforts, we propose three approaches/scenarios to perform the RF split fabrication:

- Scenario I: Remove only the top metal layer from the layers to generate FEOL. Since the inductors are often located in the top layer, the FEOL foundry does not have the information of interconnections through top metal layer as well as the inductor locations and sizes.
- Scenario II: Remove the top and the second to the top metal layers. In this scenario, two upper metal layers are removed so that both inductors and capacitors are missing from the FEOL layout because the capacitors are often built through the top two metal layers.
- Scenario III: Design obfuscation. For RF designs, inductors are always located in metal rings and lower metal layers will be removed inside the rings for performance optimization. Therefore, the rings themselves, which contain multiple metal layers, would indicate positions and approximate sizes of inductors. Similarly, the lower metal layers will not be used where capacitors are located. Therefore, attackers in both scenarios I and II may learn the precise positions of the removed inductors/capacitors and may even further estimate their sizes. To further increase the security level but still to avoid performance overhead, we propose an obfuscation technique during the design phase to insert non-functional rings and to create empty zones in the original design. Using this method, it becomes more difficult for attackers to pin down the location, the count, and the sizes of passive components.

For the demonstration purpose, the TSMC 0.18 μm technology supporting six metal layers is used. In experimental demonstrations, the scenario I indicates the removal of metal6 layers. Similarly, scenario II means the removal of metal5 and metal6 layers. Scenario III follows the same rules that new rings and empty zones are removed from the metal layers metal1 to metal4. Note that the proposed three scenarios can be applied to any other process technology with the adjustment of available metal layers.

A. Split Manufacturing on Class-AB Power Amplifier

To demonstrate all three application scenarios as well as their security levels, an one-stage single-transistor class-AB power amplifier is investigated as our first example where we assume that the inductor is using metal6 layer and the capacitors are using metal5 and metal6 layers [16].

The class-AB power amplifier (see Figure 2 for detailed schematic) works at 5.8 GHz with a low supply voltage of 1.9 V. It is designed to deliver 19.8 dBm output power and 28.1% power-added efficiency.

1) Scenario I: Removal of Metal6 (Inductors): Since metal6 is removed from the FEOL, the schematic of the class-AB power amplifier, showed in Figure 3, is missing all inductor information. Although the attackers can easily recover the count and the locations of all inductors, they do not know the exact sizes and the values of the inductors. More precisely, the attackers can learn that 3 inductors are used in the design

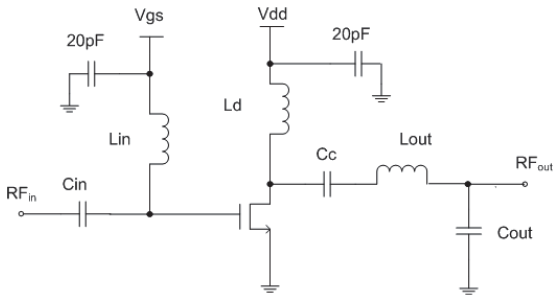


Fig. 2: Schematic of a class-AB power amplifier

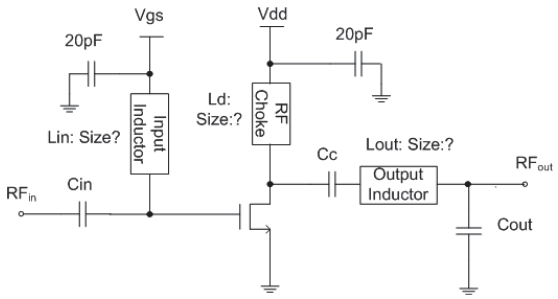
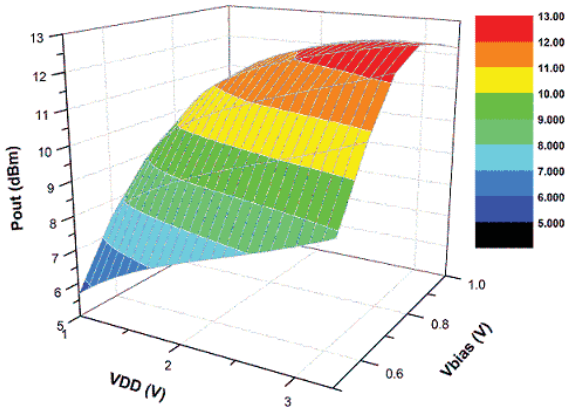


Fig. 3: A class-AB power amplifier with metal6 removed (missing inductors)

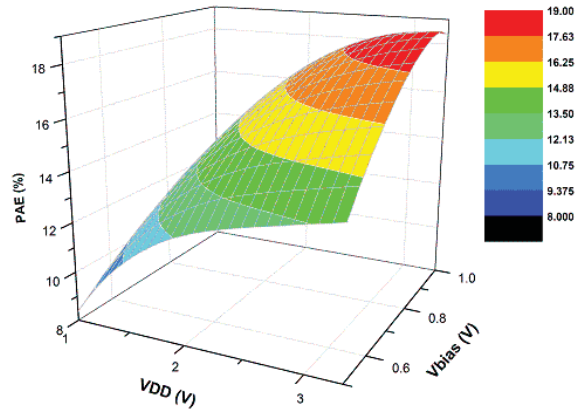
through the inductor rings. They can also extract the values for all other components. Therefore, the attackers with the FEOL of the power amplifier at hand can easily guess the general functionality of the entire design. But a detailed specification including the the supply voltage and the operating frequency remains unknown. As a result, the task for attackers to recover the entire circuit is not as simple as sweeping all possible inductor values. As we emphasized earlier, we assume that the attackers are also experienced RF designers so they would also apply the analytical calculation and other parameters from the known components in order to derive the inductor values. The procedure to recover the whole circuit from the known FEOL by attackers is described in the following steps (Note that the IP piracy cost is directly related to the complexity of the these steps):

Step 1: In the first step, the attackers will try to find out the operating conditions such as bias voltage, supply voltage and operating frequency, which can significantly shift the power amplifier performance. Since the untrusted foundry is also the provider of the fabrication process (in our case, we are using the 0.18 μm technology), the attackers should be aware of the available supply voltage for this technology (from 1 to 3.3 V). The attackers should at least try 23 different supply voltages if a step size of 0.1 V is chosen[2]. In terms of gate biasing, the reasonable range for a power amplifier varies from 0.4 to 1 V but it is not necessary that all designs follow this setting (e.g., an exception would be presented in the experimentation section). Hence, using 0.05 V as a voltage sweeping step, the gate biasing can have at least 13 different cases for attackers to choose. Meanwhile, the operating frequency still remains a puzzle to attackers, which acts as an imperative role in

[2]They may try more supply voltages with smaller voltage step size in order to get more accurate simulation results.

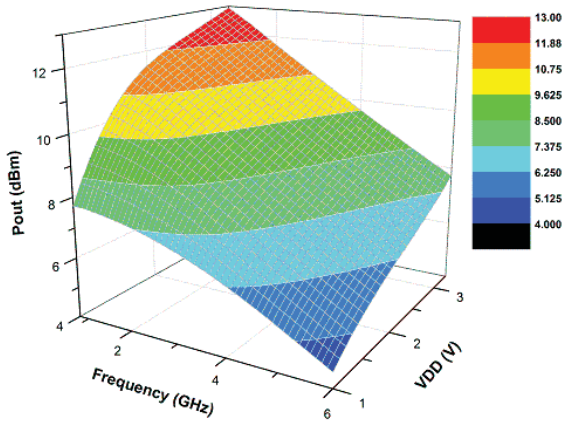


(a)

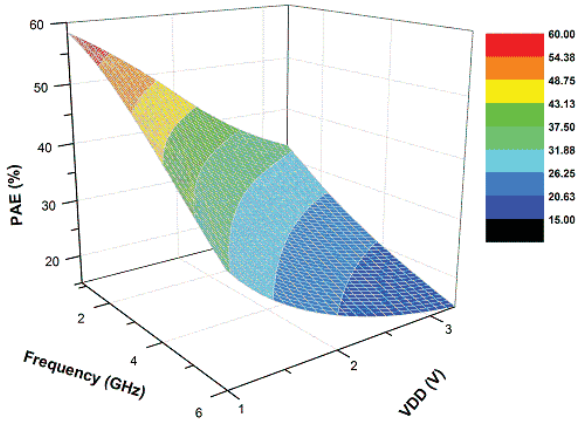


(b)

Fig. 4: (a) Supply voltage and gate biasing versus output power (b) Supply voltage and gate biasing versus power-added efficiency



(a)



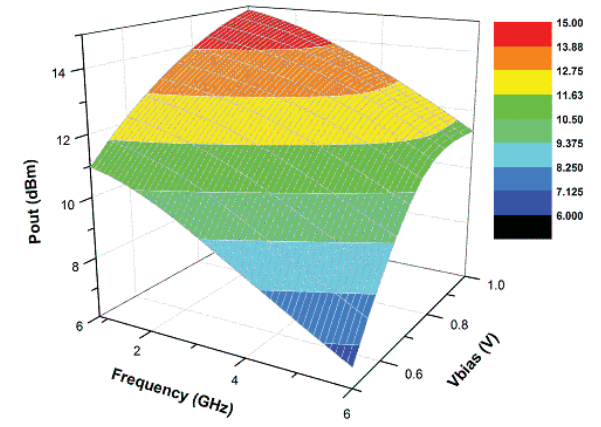
(b)

Fig. 5: (a) Supply voltage and frequency versus output power (b) Supply voltage and frequency versus power-added efficiency

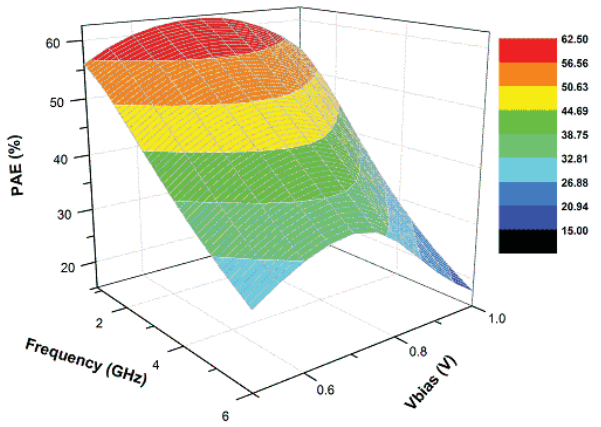
RF design. The attackers may narrow down the spectrum by assuming this example design works in the commercial communication protocol range, which is basically from 0.8 to 6 GHz. Again, the design may or may not take the communication frequency as its operating frequency, because the attackers are not aware if this layout works for some specific applications, either military or scientific confidentiality. Under this assumption, it comes to a group of 53 possible values if a step of 0.1 GHz is selected.

With all these possible cases available, the attackers will then run simulation to recover the original design by choosing the best performance of output power and power-added efficiency. For example, Figures 4 (a) and (b) show the case that the actual supply voltage and gate bias, namely 1.9 and 1 V, do not deliver the best output yields. Similarly, Figures 5 (a) and (b) show that the maximum output power is not coincident with the maximum power-added efficiency. Since this power amplifier is designed for the low-power application, the specification defines the operating frequency to be 5.8 GHz; however, Figure 5 shows that the defined operating frequency is located in the middle level of the overall performance. Clearly attackers cannot recover the original design if the optimized parameter settings are chosen. Figures 6 (a) and (b) reflects the relationship of circuit performance versus frequency and gate bias. As you can find from the figure, the actual values for frequency and gate bias, 5.8 GHz and 1 V, are located in the low performance area. Therefore, If the attackers follow any of the recovery process through Figures 4, 5 and 6, they cannot find the correct settings. Note that these sample testing process only represents a small fraction of the overall testing space meaning that it will take significant amount of time for attackers to fully simulate the design and collect the original design parameters even for a simple RF circuit.

Step 2: In the second step, we assume that the attackers have chosen the correct operating conditions for the power amplifier, they then need to set the biasing conditions to precisely recover the inductor values. Following a general RF design methodology, the experienced attackers will sweep the RF choke L_d and the input inductor L_{in} by a reasonable range,

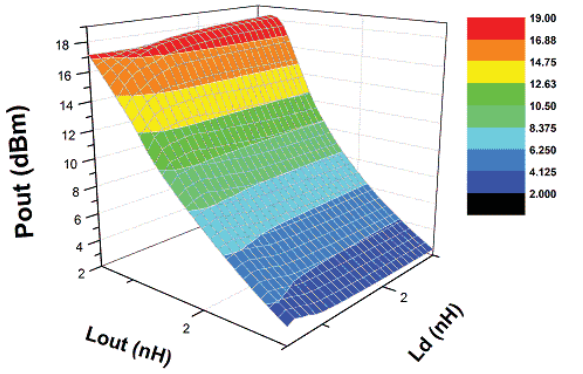


(a)

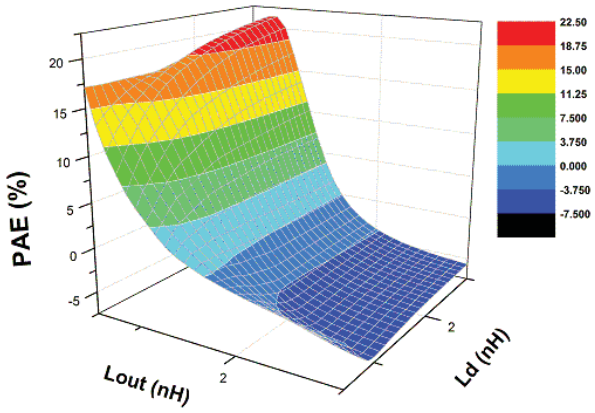


(b)

Fig. 6: (a) Gate biasing and frequency versus output power (b) Gate biasing and frequency versus power-added efficiency



(a)



(b)

Fig. 7: (a) Output inductor and RF choke versus output power (b) Output inductor and RF choke versus power-added efficiency

which is from 0.5 to 3 nH in the 0.18 μm technology, to check the input reflection coefficient S11 and to further guess the frequency range, rather than a random sweeping on different frequencies. Based on the simulation results, the attackers will probably learn the circuit working frequency between 4 and 7 GHz. The derived frequency range helps to narrow the possible range of the input inductor but, still, the attackers need to select the inductor value from 4 to 7 GHz for the overall performance simulation. The attackers will then sweep the RF choke L_d and the output inductor L_{out} to optimize the output performance and the matching network. The simulation results will be meaningless if a wrong input inductor value is chosen.

Figure 7 illustrates the output results that vary with respect to the RF choke and the output inductor. The actual values for the RF choke and the output inductor are 963 and 670 pH, respectively. However, from Figure 7 we can see that both values produce good but not the best performance. It is possible that the attackers only aim to the best performance so they may choose inductor values from the wrong rage.

2) Scenario II: Removal of Metal5 and Metal6 (Capacitors and Inductors): In this case, both inductors and capacitors are not available to the untrusted foundry because of the removal of metal5 and metal6 layers from the FEOL. The missing capacitors add additional uncertainty for attackers to recover the whole design. That is, the unknown capacitors add more freedom in the simulation though parameter sweepings and will produces large amounts of combinations of inductors and capacitors. In this case, it is much easier for an experienced attacker to follow the typical power amplifier design procedure to retrieve the missing components.

Step 1: The first step of circuit testing is exactly the same as that in Scenario I.

Step 2: After selecting the operating point, the attacker needs to decide the RF choke inductor and output coupling capacitor. The 0.18 μm technology indicates that the reasonable ranges for inductor and capacitor are 0.5 to 5 nH and 1 to 10 pF, respectively. Using a sweeping step of 0.1 nH and 0.1 pF for inductors and capacitors, respectively, the attackers will come up with a total of 45 possible values for inductors and

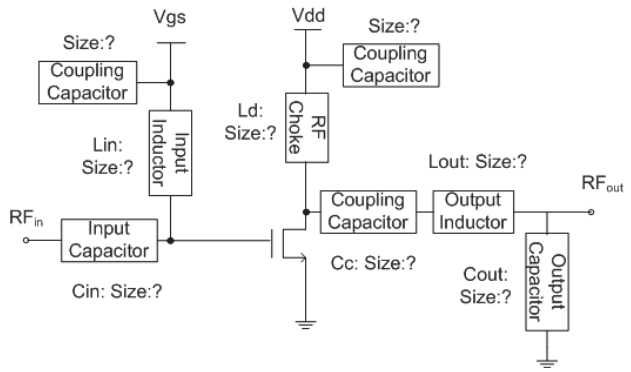


Fig. 8: Schematic of class-AB power amplifier without top two metal layers (missing inductors and capacitors)

90 possible values for capacitors[3]. Figure 9 shows the overall circuit performance when the values of the choke inductor and the output capacitor are changing. The figure helps attackers to recover the correct values of both components.

Step 3: After selecting the RF choke and coupling capacitor from various combinations, the attackers need to do the output matching to achieve a matched 50 Ω output. The RF designers often perform output matching through load pull simulation, which provides the designers a bunch of matching combinations to choose from. Advanced EDA tools can help synthesize the maximum output power and power-added efficiency as well as further reflect the impedance of the optimal points. After choosing the impedance, the designers can use the Smith chart to recover the output matching network. Because of the simple structure of the single transistor power amplifier, the output matching network only includes one inductor and one capacitor. Relying on the load pull simulation, the attackers can retrieve four possible matching networks as showed in Figure 10.

The possible topologies cover L-type (Figures 10(a) and (b)), II-type (Figure 10(c)) and T-type (Figure 10(d)), which are all basic network topology in RF design. All component values for each topology are located in reasonable design ranges; however, only the first two networks are possible given the number of passive components.

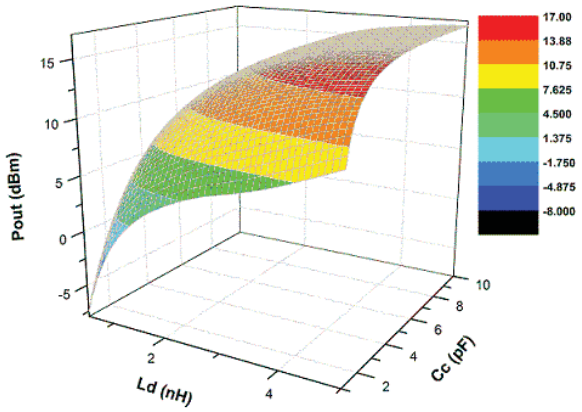
Step 4: After the load pull simulation, the attackers need to use the source pull simulation to recover the input matching network, which follows a similar procedure to the load pull simulation.

Step 5: The final tuning is necessary for attackers to adjust the performance before all circuit parameters are recovered.

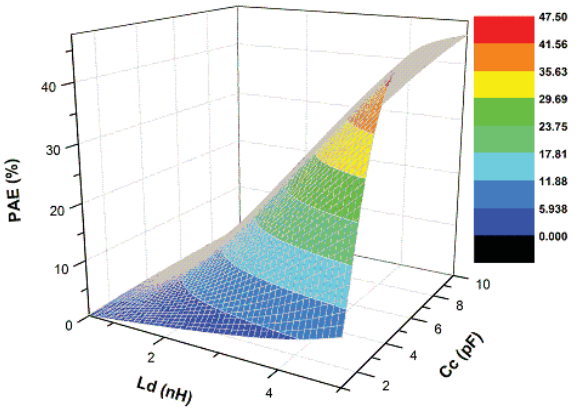
3) Scenario III: Obfuscation Techniques: Although various obfuscation techniques can be applied that increase the difficulty for attackers to recover the original circuit, in order to balance the performance impact and lower the design cost only two obfuscation methods are demonstrated in this paper. Those two methods add 1) extra block space where the capacitors/inductors are located and 2) dummy cells to mislead the attackers into incorrect simulations.

To avoid high frequency signals interfering with each

[3]Note that the range of inductor shifts from 0.5 to 5 nH rather than from 0.5 to 3 nH due to the fact that capacitor values are unknown in Scenario II.



(a)



(b)

Fig. 9: (a) RF choke and output coupling versus output power (b) RF choke and output coupling versus power-added efficiency

other, the lower level metals are not used where the inductors/capacitors are located. The existence of these empty areas may reveal to the attackers the approximate sizes of the inductors/capacitors which can lead to the recovery of the original design. To address this issue and to further increase the difficulty of RF IP piracy, we propose an obfuscation technique to deliberately increase passive component area. This will have the effect of lowering the correlation between the area of each inductor/capacitor and their value.

A second method will also be applied which includes unused empty blocks in the original design so that the attackers cannot find the correct circuit structure. Those extra blocks can be located either in the input or the output side. For example, the attackers will only select L-types output matching networks from Figures 10(a) and (b), but they will also consider other topologies if two empty blocks are inserted.

Different from the IP protection scenarios I and II, the obfuscation technique in scenario III requires modifying the original layout. The RF design performance will be affected due to the sensitivity of layout modifications. To address

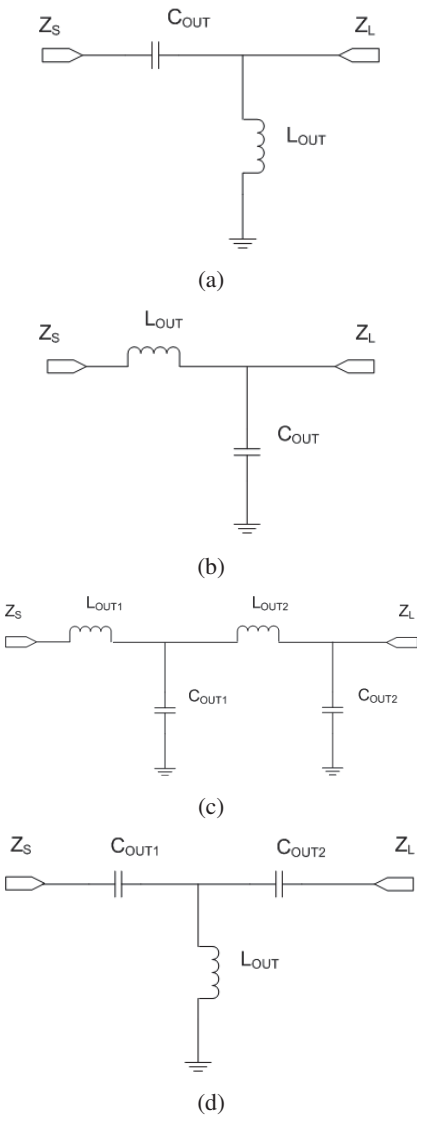


Fig. 10: Four possible output matching network for the class-AB power amplifier

this issue, we suggest a new RF design methodology, called security co-design, which considers security at the early stage of the RF designs by altering some design rules to integrate the obfuscation technique in the design flow.

V. Conclusion

Split manufacturing has presented a new solution against reverse engineering and IP piracy when the IC design flow becomes more globalized. Different from all previous work to apply the split manufacturing in digital circuits, we introduced the first attempt to implement a similar method in RF designs. Quantitative analysis was presented to assess the security protection level for RF designs when the untrusted foundries would like to recover the circuit designs based on part of the circuit layout. To further guide the application of split manufacturing in RF circuits, three different FEOL and BEOL separation and obfuscation methods were introduced. All these methods were demonstrated on one RF circuit: a class-AB power amplifier. The simulation results confirm that the unknown passive components, either inductors or capacitors, along with the missing DC biasing conditions, can raise significant uncertainty for the attacker to recover the RF circuits. In conclusion, split manufacturing is more effective in RF IC trust than in digital circuit security.

References

[1] "Defense science board (dsb) study on high performance microchip supply," *http://www.cra.org/govaffairs/images/2005-02-HPMS_Report_Final.pdf*, 2005.

[2] S. Adee, "The hunt for the kill switch," *IEEE Spectrum*, vol. 45, no. 5, pp. 34–39, 2008.

[3] Age Yeh, *Trends in the global IC design service market*, DIGITIMES Research, 2012.

[4] D. Agrawal, S. Baktir, D. Karakoyunlu, P. Rohatgi, and B. Sunar, "Trojan detection using IC fingerprinting," in *IEEE Symposium on Security and Privacy*, 2007, pp. 296–310.

[5] Yousra Alkabani and Farinaz Koushanfar, "Active hardware metering for intellectual property protection and security," in *USENIX Security*, 2007, pp. 291–306.

[6] Yier Jin, Bo Yang, and Yiorgos Makris, "Cycle-accurate information assurance by proof-carrying based signal sensitivity tracing," in *IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, 2013, pp. 99–106.

[7] Yu Bi, P.-E. Gaillardon, X.S. Hu, M. Niemier, Jiann-Shiun Yuan, and Yier Jin, "Leveraging emerging technology for hardware security - case study on silicon nanowire fets and graphene symfets," in *Test Symposium (ATS), 2014 IEEE 23rd Asian*, Nov 2014, pp. 342–347.

[8] Intelligence Advanced Research Projects Activity, "Trusted integrated chips (TIC) program," 2011.

[9] Jeyavijayan Rajendran, Ozgur Sinanoglu, and Ramesh Karri, "Is split manufacturing secure?," in *Design, Automation Test in Europe Conference Exhibition (DATE), 2013*, March 2013, pp. 1259–1264.

[10] Frank Imeson, Ariq Emtenan, Siddharth Garg, and Mahesh Tripunitara, "Securing computer hardware using 3d integrated circuit (ic) technology and split manufacturing for obfuscation," in *Presented as part of the 22nd USENIX Security Symposium (USENIX Security 13)*, Washington, D.C., 2013, pp. 495–510, USENIX.

[11] Kaushik Vaidyanathan, Bishnu P Das, Ekin Sumbul, Renzhi Liu, and Larry Pileggi, "Building trusted ics using split fabrication," in *Hardware-Oriented Security and Trust (HOST), 2014*, May 2014.

[12] Kaushik Vaidyanathan, Renzhi Liu, Ekin Sumbul, Qiuling Zhu, Franz Franchetti, and Larry Pileggi, "Efficient and secure intellectual property (ip) design with split fabrication," in *Hardware-Oriented Security and Trust (HOST), 2014*, May 2014.

[13] B. Hill, R. Karmazin, C.T.O. Otero, J. Tse, and R. Manohar, "A split-foundry asynchronous fpga," in *Custom Integrated Circuits Conference (CICC), 2013 IEEE*, Sept 2013, pp. 1–4.

[14] J.S. Yuan and Y. Bi, "Process and temperature robust voltage multiplier design for rf energy harvesting," *Microelectronics Reliability*, vol. 55, pp. 107–113, 2015.

[15] J.S. Yuan, Y. Xu, S.D. Yen, Y. Bi, and G.W. Hwang, "Hot carrier injection stress effect on a 65 nm lna at 70 ghz," *Device and Materials Reliability, IEEE Transactions on*, vol. 14, no. 3, pp. 931–934, Sept 2014.

[16] J. Carls, R. Eickhoff, P. Sakalas, S. von der Mark, and S. Wehrli, "Design of a c-band cmos class ab power amplifier for an ultra low supply voltage of 1.9 v," in *Microwave and Optoelectronics Conference, 2007. IMOC 2007. SBMO/IEEE MTT-S International*, Oct 2007, pp. 786–789.

Robust PUF Circuit Design against Temperature Variations and Aging Effect

Georgiy Brussenskiy, *Student Member, IEEE*, and J.S. Yuan, *Senior Member, IEEE*
School of Electrical Engineering and Computer Science
University of Central Florida, Orlando, Florida 32816, USA
Email: gbrussenskiy@knights.ucf.edu, Jiann-Shiun.Yuan@ucf.edu

***Abstract* – Physical unclonable functions have proven to be great candidates for solving many important security issues in the last few decades. In the effort to design reliable PUF circuits, it is critical for PUF to be robust against temperature variations and aging effect. Device aging takes place mainly due to Hot Carrier Injection (HCI) and Negative Bias Temperature Instability (NBTI) while temperature variations are caused by various environmental conditions. Both temperature variations and device aging degrade circuit performance to a large extent. In this paper, we propose a robust Ring-Oscillator PUF designed to significantly reduce the impact of temperature variations and aging effect. Frequency degradation due to temperature fluctuations in our model is 21.7% as opposed to 57.5% for a conventional RO. Our robust PUF has an average inter-chip HD of 43.5% (close to ideal value of 50%).**

Index Terms – robust PUF, RO-PUF, reliable PUF, temperature variations PUF, aging effect PUF

I. INTRODUCTION

The number of electronic devices has grown substantially over the last several decades. As a result of such proliferation of electronics, the security aspect has gained more significance. The foundation of any security-related application is based on the ability to verify device's identity. It used to be the case that non-volatile memories (NVM) were the primary means to store devices' IDs. But due to the fact that IDs can be duplicated as well as high cost associated with maintenance of using NVM, security experts started to search for an alternative[7].

Therefore, Physical Unclonable Functions have emerged to address these issues. A Physical Unclonable Function (PUF) is a special circuit that maps a set of challenges to responses by exploiting variations that come from IC manufacturing. Different types of PUFs have been put forward such as Ring-Oscillator PUF [1], Arbiter PUF[2], SRAM-PUF[3], and Butterfly PUF[4]. Ring-Oscillator PUF from MIT has been reported to have great advantages of being easily implemented in FPGA and ASIC as well as having more reliable performance and relatively easy fabrication process [1, 5, 6].

Reliability is one of the biggest concerns when designing PUF. The ID of a chip must not depend on time or operating environment[7]. Temperature variations and device aging result in decreased stability and hence reduced reliability[8-10].

II. BACKGROUND

Aging effect is primarily due to negative bias temperature instability (NBTI) and hot carrier injection (HCI). Discussion of each phenomenon is presented in next subsections.

A. Negative Bias Temperature Instability (NBTI)

NBTI that occurs in PMOSs is one of the most important concerns among circuit designers when it comes to assessing the reliability. NBTI is caused by interface traps under high temperature and negative gate voltage bias when the operating temperature increases[11]. Threshold voltage of PMOS device increases when the gate voltage is driven below its source voltage. Both formation of interface states as well as trapping of holes in oxide defects have been identified to be the reasons behind increase in threshold voltage[12].

Since threshold voltage increases, the drain current and transconductance gm decrease, and hence may result in IC failures. Either high temperature or negative gate voltage can result in NBTI, but its effect is amplified when they are combined[13]. It has been reported that Vth of PMOS transistors can be changed by up to 50mv after ten years which implies over 20% rise in circuit delay [13, 14**]**. NBTI degradation can be partially recovered by annealing at high temperature provided that stress voltage is removed[13]. As Vt degrades, the gate overdrive (Vg – Vt) reduces resulting in smaller current and frequency degradation of ring oscillators[13].

B. Hot Carrier Injection (HCI)

HCI occurs when some of the carriers that drift near the drain acquire energies that are much higher than the thermal energy of carriers. This may result into undesired tunneling of hot carriers through the gate oxide and hence leading to formation of gate current or they could also become trapped. As a consequence, threshold voltage is changed[12]. HCI is an important reliability issue for

nMOS rather than pMOS since electrons have higher mobility than holes. HCI leads to decreased drain current in nMOS, but increased drain current in pMOS. The primary causes of susceptibility to HCI-related aging are high temperature levels, large Vdd, and high switching activity[7].

C. Ring-Oscillator PUF

Regular RO-PUF is illustrated in Figure 1. It consists of multiple ring oscillators, two multiplexers, two counters, and a comparator. First, one pair of ring oscillators is selected using multiplexers. Then, the number of oscillations are counted using counters. Next, random bits are generated after frequencies of ring oscillators are compared. One of the oscillators will be faster than the other due to process variations.

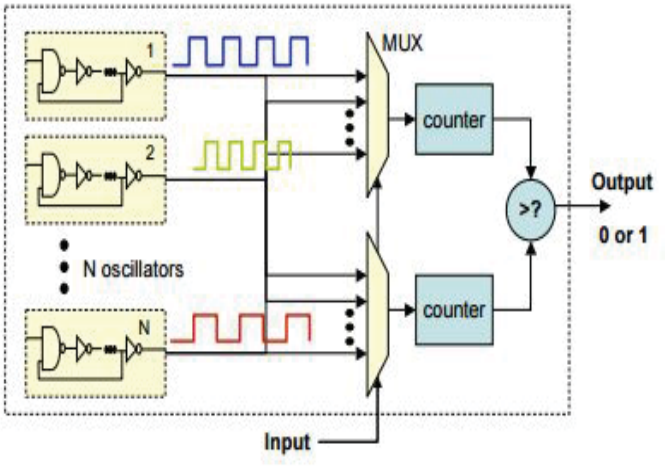


Figure 1: Regular RO-PUF

D. PUF Quality Metrics

Three main metrics of evaluating PUF performance are randomness, reliability, and uniqueness. Randomness determines PUF's ability to produce random responses[15]. Uniqueness can be defined as the independence of PUF responses to the same challenge[16] and its ideal values is 50%. The reliability measures the consistency of PUF challenge-response pairs at different operating conditions[16].

III. RELATED WORK

Both temperature variations and aging effect have been studied in several works. With regards to latter one, the impact of aging was reported for SRAM PUF in [17]. Also, in [18], Feige-Fiat-Shamir identification scheme was used to identify the onset of aging's impact on stability[18]. Furthermore, the study of aging effect on fpga-based PUF was done in [19]. In addition, Aging-Resistant Ring Oscillator PUF has been proposed to mitigate NBTI and HCI in [15].

Temperature variations' impact on PUF was analyzed in few studies. Two methods for improving reliability were presented in [20]. The first one concentrates on improving the gate overdrive (Vgs – Vt(T)) through optimization of supply voltage while second one utilizes negative temperature coefficient characteristic of n^+ and p^+ used as source feedback resistors. In [6], temperature-aware cooperative (TAC) RO-PUF is presented to improve the efficiency by pairing up the ring oscillator pairs in order for them to cooperate.

Our model is based on ARO-PUF work introduced in [15], but modified for significantly minimizing temperature's impact on frequency degradation and reducing aging effect.

IV. CIRCUIT IMPLEMENTATION

The purpose of this section is to discuss and evaluate low-power addition-based current source topology[21] that was used in our design. Current source topology and its temperature dependence are investigated in following subsection. Moreover, current source operation and performance are outlined and analyzed as well.

A. Current Source Topology

Our model is based on low-power temperature compensated ring oscillator with addition based current source topology presented in [21].

Current source acts as the bias current source since it has the same loading effect as a single transistor that drives the same amount of current. The circuit diagram is illustrated in Figure 2. As can be seen from the schematic, NFETs M1 and M3 are designed to obtain good local matching in order for drain currents I1 and I3 in both transistors to change in the same way as process conditions[21].

The circuit is designed in such a way that if I1 increases because of process variation, gate voltage of M2 is pulled down which causes I2 to decrease. Likewise, if the magnitude of I1 decreases, the gate voltage of M2 increase and hence causes I2 to go up as well. Therefore, in both cases, the total result is a stable output current which consists of I1 and I2 that's less dependent on process conditions.

Parameter alpha is utilized for current expressions in equations (1) and (2) for modeling the degree of velocity saturation[21].

$$I_1 = k_1(Vgs_1 - Vth_1)^{alpha} \quad (1)$$

$$I_2 = k_2(Vgs_2 - Vth_2)^{alpha} \quad (2)$$

$$I_3 = I_1 + I_2 \quad (3)$$

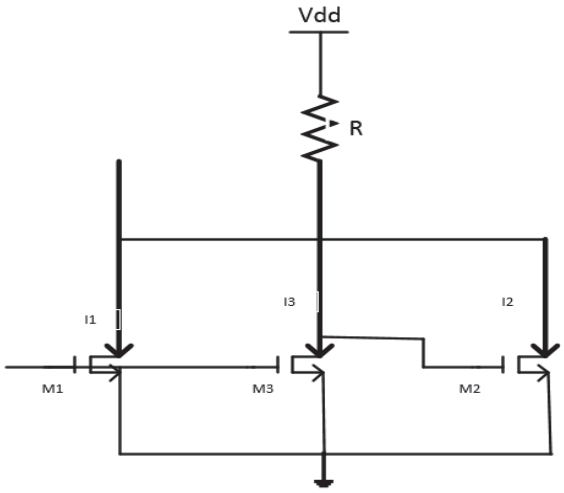


Figure 2. Addition-based current source

The process-varying parameters in (1,2,3) are *k1*, *k2*, *Vth1*, and *Vth2*. ΔI can be calculated by taking partial derivatives with respect to the process-varying parameters such as *k1*, *k2*, *Vth1*, and *Vth2*. Taking into account local matching conditions *Vth* = *Vth1* = *Vth2*, ΔI is equal to:

$$\Delta I = (1+k2/k1)\,\Delta I1 + alphak2(Vgs2 - Vth2)^{alpha-1}\,\Delta Vgs2 \quad (4)$$

By setting equation (4) to zero, the amount of feedback to Vgs2 is

$$\Delta Vgs2 = -\Delta I1*(1+k2/k1)/(alphak2*(Vgs2-Vth2)\text{^}(alpha-1)) \approx -\Delta I1R \quad (5)$$

The resistance value, R, which meets process compensation condition is calculated as follows:

$$R^0 = \frac{1+k2/k1}{alphak2(Vgs2-Vth2)\text{^}(alpha-1)} = (1+\frac{k2}{k1})/gm2 \quad (6)$$

$k1^0$, $k2^0$ are found by looking at Vgs2 DC bias condition:

$$Vgs2 = Vdd - I1^0R^0 \quad (7)$$

Assuming Vgs2 = Vgs1 since M2 needs to be biased at the same gate voltage as M1, and plugging in equations for currents (1), (2), (3), and (7) for Vgs2, result in

$$\frac{k2^0}{k1^0} = (Vgs1 - Vth0)/(alpha*(Vdd - Vgs1)-(Vgs1 - Vth0)) \quad (8)$$

Equations (6) and (8) allow to reduce variation term ΔI and decrease the standard deviation of the total current compared to a transistor current[21].

B. Current Source Temperature Dependence

Temperature fluctuations are also compensated due to the fact that critical parameters that are dependent on temperature such as resistance, threshold voltage, and mobility carriers can be used as same variables that we used for process variations[21]. The relationship between temperature and aforementioned parameters are displayed below:

$$\mu_n \propto T\text{^}(-\alpha_k) \quad (9)$$

$$V_{th}(T) = Vth(T_0)\,(1+alpha\_k*V_{th}(T-T0)) \quad (10)$$

$$R(T) = R(T_0)(1+alpha_R(T-T_0)) \quad (11)$$

Similar to derivation for process variation, the variation term of ΔT can be calculated, which is caused by a shift in temperature in the addition-based current source ΔI_{ADD} and in the single transistor ΔI_{TRAN}

$$\Delta I_{TRAN}/ I_{TRAN}(T_0) = \Delta T((alpha_{Vth}V_{th0})/(V_{gs0}- V_{th0}) - alpha\_k/T) \quad (12)$$

In (12), the first-order ΔT term is cancelled, which only leaves higher order terms. As a result, it indicates that the current source compensates for temperature variation.

C. Architecture and Operation

The proposed architecture is based on ARO-PUF architecture[15], but modified by using addition-based current source and reducing the number of stages. Fig. 3a displays the first three stages of one of ring oscillators in proposed model. A total of 50 ROs are used for implementation of Robust PUF and every ring oscillator consists of 33 stages. Fig. 3b and 3c illustrate Oscillating and Non-Oscillating modes, respectively.

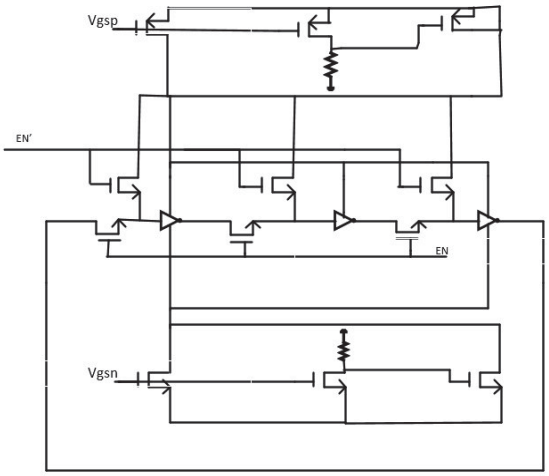


Figure 3a. Proposed RO-PUF

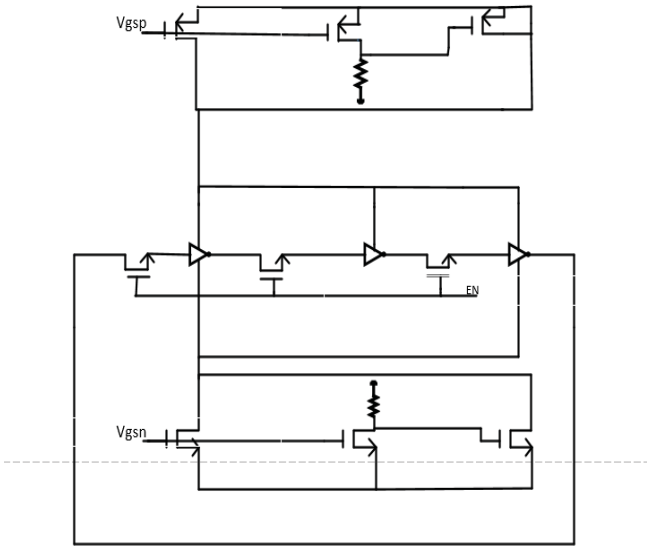


Figure 3b. Oscillating Mode

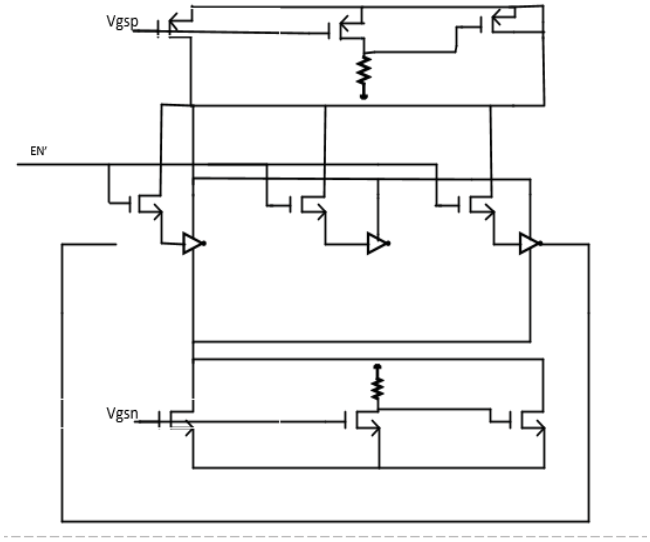


Figure 3c. Non-Oscillating Mode

In oscillating mode, the circuit works as ring-oscillator while in non-oscillating mode, transistors that connect inverter's input with current source cause the signal of inverter's input to Vdd – Vt which diminishes the impact of HCI and NBTI[15].

V. SIMULATION RESULTS

Simulation analysis using HSPICE was performed for proposed robust PUF. Monte Carlo simulation for 1000 chip instances was performed as well. The impact of temperature and aging is analyzed in subsequent sections. Furthermore, PUF quality metrics are evaluated.

Fig. 4 displays the relationship between frequency and temperature. As can be seen from the graph, the frequency degradation due to temperature variations is 21.79% in proposed robust PUF as opposed to 57.47% for conventional RO-PUF.

A. Temperature Variations

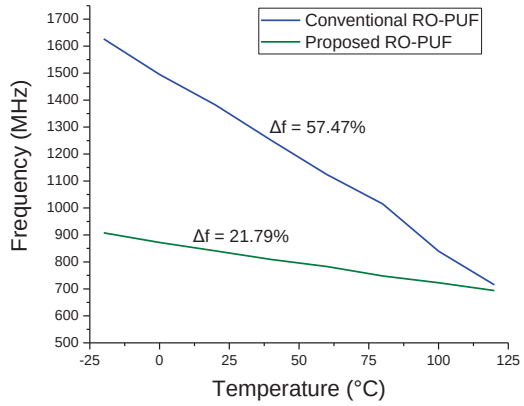


Figure 4. Frequency vs Temperature

B. Aging Effect

Fig. 5 illustrates the impact of aging on frequency degradation. As can be seen from the plot, aging effect was improved by 11.71% with respect to conventional RO-PUF.

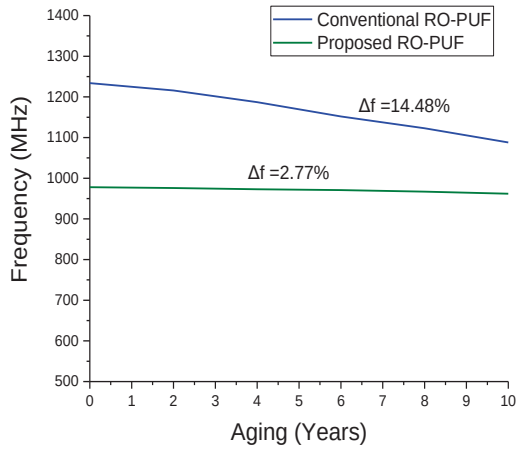


Figure 5. Frequency vs Aging

Figure 6 represents % of error by calculating the occurrence frequency of how many times bits were flipped. In our model, 8.93 bits flipped on average in contrast with 10.35 bits flipped of conventional RO-PUF.

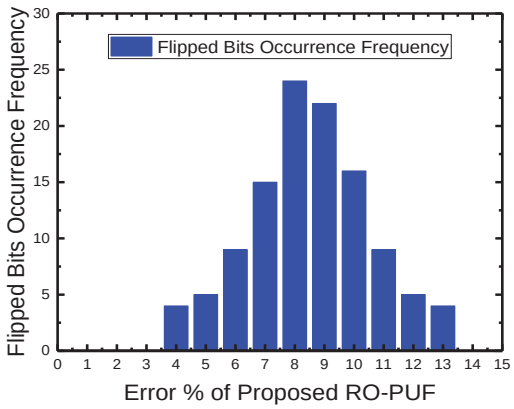


Figure 6. Error rate in % for Robust PUF due to Aging

C. Uniqueness

Figure 7 represents the uniqueness of Robust PUF which is .43 (close to ideal value of 0.50)

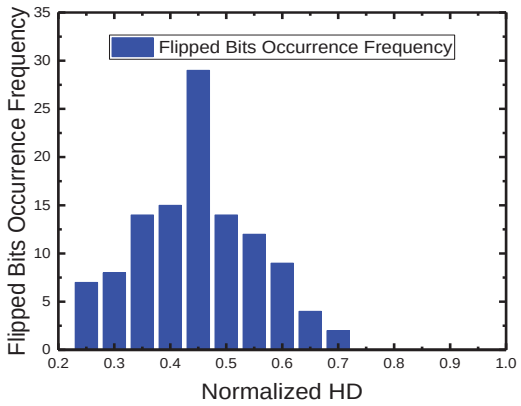


Figure 7. Uniqueness of Robust PUF

D. Randomness

Temperature(°C)	Robust PUF ('1' in response)
-25	68
0	63
25	59
50	64
75	64
100	65
125	67

Table 1. Response to various challenges at different temperatures expressed in % of '1'

The response of our robust PUF, illustrated in Table 1, at normal temperature (25 °C) has 59% '1' and 41% '0' (close to ideal value of 50% for '1' and 50% for '0'). The response is also 64.29% for '1' (and 35.71% for '0') on average at various temperatures (from -25°C to 125°C).

VI. CONCLUSION

In conclusion, our proposed robust PUF circuit illustrates that the impact of temperature variations was reduced by 35.68% and the aging effect was decreased by 11.71% with respect to conventional PUF.

References

[1] G. E. Suh and S. Devadas, "Physical Unclonable Functions for Device Authentication and Secret Key Generation," in Proc. 44th ACM/IEEE Design Automation Conf. DAC 07, pp. 9-14, June 2007.

[2] B. Gassend et al., "Silicon physical random functions," in CCS 02: Pro- ceedings of the 9th ACM conference on Computer and communications security. New York, NY, USA: ACM, pp. 148-160, 2002.

[3] J. Guajardo, S.S. Kumar, GJ. Schrijen, and P. Tuyls, "FPGA Intrinsic Pills and Their Use for IP Protection," Workshop on Cryptographic Hardware and Embedded Systems (CHES), Sep. 2007.

[4] S. Kumar, J. Guajardo, R. Maes, GJ. Schrijen and P. Tuyls, "The Buttefly PUF: Protecting IP on every FPGA," In IEEE International Workshop on Hardware Oriented Security and Trust, 2008.

[5] A. Maiti et al., "A large scale characterization of RO-PUF," in Proc. IEEE Int. Hardware-Oriented Security and Trust (HOST) Symp, pp. 94-99, 2010.

[6] C. Yin and G. Qu, "Temperature-aware cooperative ring oscillator PUF," Hardware-Oriented Security and Trust, 2009. HOST '09. IEEE International Workshop on, pp. 36-42, July 2009.

[7] Dinesh Ganta, Leyla Nazhandali, "Study of IC Aging on Ring Oscillator Physical Unclonable Functions," Quality Electronic Design(ISQED), 2014 15th International Symposium, pp. 461-466, 2014

[8] D. Lorenz, G. Georgakos, and U. Schlichtmann, "Aging analysis of circuit timing considering nbti and hci," in On- Line Testing Symposium, (IOLTS). 15th IEEE International, pp. 3-8, 2009.

[9] F. Catthoor, P.Raghavan, H. Kukner, S. Khan, and S. Hamdioui, "Incorporating parameter variations in BTI impact on nano-scale logical gates analysis," IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), pp. 158-163, 2012.

[10] I.M. Filanovsky and A. Allam, "Mutual Compensation of Mobility and Threshold Voltage Temperature Effects with Applications in CMOS Circuits," IEEE Transactions on Circuits and Systems - 1: Fundamental Theory and Applications, vol. 48, no. 7, pp. 876-884, Jul. 2001.

[11] Yi Liu, "Study of Oxide Breakdown, Hot Carrier, and NBTI Effects on MOS Device and Circuit Reliability," 2005

[12] R. Jacob Baker, "CMOS Circuit Design and Layout," IEEE Series on Microelectronic Systems, IEEE Press, 2010.

[13] D. Schroder et al., "Negative bias temperature instability: Road to cross in deep submicron silicon semiconductor manufacturing," Journal of Applied Physics, vol. 94, no. 1, pp. 1–18, 2003.

[14] S. Borkar, "Electronics beyond nano-scale cmos," in DAC '06. 43rd ACM/IEEE, pp. 807–808.

[15] M. T. Rahman, D. Forte, M. Tehranipoor, "ARO-PUF: An aging-resistant ring oscillator PUF design," Design, Automation, and Test in Europe Conference and Exhibition, pp. 1-6, 2014.

[16] Lang Lin, Dan Holcomb, Dilip Kumar Krishnappa, Prasad Shabadi, Wayne Burleson, "Low Power Sub-Threshold Design of Secure Physical Unclonable Functions Presentation," Low-Power Electronics and Design (ISLPED), 2010 ACM/IEEE International Symposium, pp. 43-48, 2010

[17] D. Lim, et al., "Extracting secret keys from integrated circuits," IEEE TVLSI, vol. 13, no. 10, pp. 1200–1205, 2005.

[18] M. S. Kirkpatrick et al., "Software techniques to combat drift in pufbased authentication systems," in SECSI 2010, p. 9.

[19] A. Maiti et al., "The impact of aging on an FPGA-based physical unclonable function," in FPL 2011, pp. 151–156.

[20] R. Kumar, H.K. Chandrikakutty, S. Kundu, "On Improving Reliability of Delay Based PUF under temperature variations," Hardware Oriented Security and Trust (HOST), 2011 IEEE International Symposium, pp. 142-147, 2011

[21] Xuan Zhang, A.B. Apsel, "A Low-Power, Process-and-Temperature-Compensated Ring Osscillator with Addition-Based Current Source," Circuits and Systems I: Regular Papers, IEEE, pp. 868-878, 2011.

SESSION

SECURITY MANAGEMENT II

Chair(s)

Dr. Rob Byrd
Dr. Mohammed Misbahuddin

Securing Organizations: A Framework For Implementing Optimized Security Measures

A. Peters, R. Byrd
School of IT and Computing, Abilene Christian University, Abilene, Texas, U. S. A.

Abstract - *Security breaches are a constant fear because of their tremendous impact. Breaches consume vast amounts of resources, damage an organization's reputation and can directly affect a business's customers. With companies such as Target and Home Depot having devastating security breaches, how will other businesses manage to stay secure? We developed a solution to the expensive or inaccessible security tools for each unique company. Our framework allows an organization, or business, of any size and industry to determine the most effective and cost-efficient security measures. A few key organizational parameters determine which security actions will best help that particular organization. Once each characteristic is defined, a prioritized list of security actions can be implemented that will optimize security based on an organization's specific needs. All types of businesses will be able to use this framework as a guide to securing their organizations.*

1 Introduction

As technology evolves, security is becoming increasingly important to an organization's survival. Security breaches are a constant fear for all because of their tremendous impact. Breaches not only consume vast amounts of resources, but they also damage an organization's reputation and directly affect their customers. The very publicized breach of Target in 2013 affected 70 to 110 million people and cost the organization about $148 million to repair the damage caused by the hackers [14]. This sum excludes the loss of revenue due to the breach [14]. As company budgets continue to get tighter, the amount of money required to maintain a secure organization continues to increase. Organizations must find the most effective and cost efficient actions to secure their vital information.

1.1 Overview

This paper will begin with a literature review. This section will contain summaries of the major sources that were especially important in the development of the framework. Within the literature review is a section with summaries of the ten domains of information security with a description of each domain from various sources. The method portion of this paper describes the process by which the framework was created. The paper culminates with the conclusion, future work, and a list of references.

2 Literature Review

2.1 Case Study

In order to get the best information on commonly effective security practices, we analyzed a case study of two global organizations that have continually been successful in their security. Singh et al. (2013) interviewed employees from each company and organized their efforts by the different subdivisions within security [13]. These components include Information Security Requirements, Top Management Support, Information Security Culture, Information Security Audit, ISM Best Practices, Asset Management, Information Security Incident Management, Information Security Regulations Compliance, and ISM Effectiveness [13]. Information is very important for both companies, so security was key to their success. However, for one company, information was more critical to their success than the other. With this in mind, we were able to distinguish which actions are best for organizations with varying information sensitivity. Singh et al. distinguished the net revenue, employee size, and business sector of the two companies. We compared the known facts of the two businesses to the analysis done by the case study. One was an "information delivery platform" with "5200+ employees" while the other was a "global management consulting, technology services and outsourcing company" with "257,000 employees [13]." We were able to see how these factors affect the security of these businesses. One prevalent aspect of the study was the difference in size and number of employees. While both businesses were very concerned with information security, the larger business was able

to devote more resources into putting security measures into action.

2.2 BSIMM

There are many different ways to measure the security level of an organization. However, there is an existing framework that assesses software companies' maturity in security measures. The BSIMM, Building Security In Maturity Model, is a tool for companies to measure their level of software security [9]. The authors studied 67 firms in order to create a list of 112 activities for companies to initiate [9]. These 112 activities are divided into four main categories (Governance, Intelligence, SSDL Touchpoints, Deployment) [9]. The Governance category in the BSIMM includes strategy and metrics, compliance and policy, and training. Intelligence is another subdivision identified in the BSIMM. This includes attack models, security features and design, and standards and requirements. Actions within attack models include identifying possible attackers and recording previous attacks [9]. SSDL Touchpoints includes architecture analysis, code review, and security testing. This subdivision is specifically for software developing companies.

The last subdivision is Deployment, which includes penetration testing, software environment, configuration management, and vulnerability management. We used components from each of the four categories to understand the areas where security is needed and ways to successfully address those specific areas.

2.3 CISSP

The *All in One CISSP Exam Guide Sixth Edition* is a study tool used by many professionals to prepare for the Certified Information Systems Security Professional exam. The book discusses the ten domains of information security in depth. This allowed for a better understanding of the ten domains of security and find solutions to common problems that would be within the domains. Below are summaries of each of the domains of information security.

Domains

Information Security Governance and Risk Management.

Information Security Governance and Risk Management is the first domain of information security. It includes risk analysis and management, policies, procedures, and the layers of responsibility [4]. Implementing these aspects is "necessary for organizations to practice security in a holistic manner" [4]. One way for businesses to fully move toward optimal security is for the CEO or president of the company or organization to understand what needs to change for security to improve. This can be accomplished by having regular meetings with the head of the company, several board members, and the top IT employees [12]. Having a time set aside to discuss important security topics will help the extremely common communication gap found in large companies between the boardroom and IT department [12]. Another very important aspect is training. Training is where employees learn what not to do and how to prevent others from doing something that would put the business' security at risk. Simply by educating employees to be suspicious of malware, unsolicited phone calls or e-mails, and not discussing confidential information will help the business maintain security [2].

Access Control.

Access Control includes identification, authorization, and the many aspects involved in access control. The CISSP defines access controls as "security features that control how users and systems communicate and interact with other systems and resources" [4]. "Access controls make it harder for attackers to break in" as well as "limit damage if a system is attacked" [11]. One approach is to define authorization by having each employee assigned to a very specific level of information access. This will prevent granting too much access to employees and thus reduce the information risk [11]. With access controls in place, having someone monitor and modify these controls as needed, will keep the system running smoothly [11].

Security Architecture and Design.

Security Architecture and Design includes not only computer architecture (processing and memory), but also security models and systems evaluation methods [4]. Time-of-check/time-of-use attacks take "advantage of the dependency on the timing of events that take place in a multitasking operating system" [4]. A way to prevent this type of attack would be to "apply software locks to the items" the operating system "will use when it is carrying out its checking tasks" [4].

Physical and Environmental Security.

Physical and Environmental Security includes the planning process, internal support systems, and perimeter security [4]. A case study researching a university found that many overlook physical security [10]. The actions to physically secure are dependent on

the location and type of technology used for each business and organization. However, having locks on every important door, hiring guards, and installing security systems are general ways to keep businesses secure.

Telecommunications and Network Security.
Telecommunications and Network Security involves the many layers and the hardware associated with networking. Implementing a firewall and anti-virus software should be among the first actions for a business. "Firewalls are the network security systems that control incoming and outgoing data traffic by analyzing the packets and determining whether they should be let 'through' or blocked" [6]. Mobile phone and instant messaging (texting) are also included in this domain. Precautions, such as requiring all employees' phones to have a passcode and encrypting all messages and calls, should be taken with business mobile phones.

Cryptography.
Cryptography focuses on the methods of encryption, key management, e-mail standards, and Internet security. Following the Internet Protocol Security protocols will set up proper channels for data exchanges [4]. E-mail standards are also important for businesses that share important information via e-mail. PGP is adequate for a company that only needs to encrypt some e-mail messages while link encryption implementation is optimal for a company that needs all data encrypted [4]. Encrypting all hard drives is also an effective preventative measure [13].

Business Continuity and Disaster Recovery Planning.
This domain embraces preventive measures, insurance, and recovery strategies. The process of developing a plan, implementing the plan, and maintaining the plan is the fundamental method in this domain. Utilizing a disaster recovery plan, the business should at a minimum conduct semiannual emergency drills [13)] Backing up critical information at a server in a different location is also ideal in case of disaster [13].

Legal, Regulations, Investigations, and Compliance.
This domain includes intellectual property laws, compliance, privacy, and cybercrime investigations. Businesses should form an incident response team to follow procedures for incident response. These include triage, investigation, containment, analysis, tracking, and recovery [4].

Software Development Security.
Software Development Security includes the software development life cycle, web security, database management, and malware. This domain is discussed in depth in the BSIMM. One preventative measure is for software developers to integrate security into new software from the start [4, 9]. This process saves time when compared to constant patching, and is ultimately better from a security standpoint [4].

Security Operations.
Security Operations includes operational responsibilities, configuration management, data leakage, network and resource availability, and vulnerability testing. This domain is the broadest and involves the preservation of security in all aspects of technology in a business [4]. "Resource protection, change control, hardware and software controls, trusted system recovery, separation of duties, and least privilege" are the central aspects of operations security [4].

2.4 The Bureau of Labor Statistics
Certain characteristics and metrics should be defined to customize the security measures. The first metric is the industry type of the organization/company. The Bureau of Labor Statistics defines all industries into two categories: good-producing and service-providing. Good-producing industries include natural resources and mining, agriculture, manufacturing, and construction [6]. Service-providing industries include trade, transportation, utilities, information, finance, insurance, real estate, health care, education, food services, leisure, and hospitality [6]. By differentiating between these two types of industry, the ability to customize cyber security needs is achievable. For example, a clothing retailer has extremely different security needs when compared to an airline company. This is largely due to the fact that a clothing manufacturer is a good-producing business, while an airline company is a service-providing business. According to the Bureau of Justice Statistics, the two business sectors with the highest prevalence of cyber crime were both in the service-providing industry—Telecommunications and Software [15]. The two business sectors with the lowest prevalence of cyber crime were goods-producing industries—Agriculture and Hunting/Fishing/Forestry, as stated in the graph below [15].

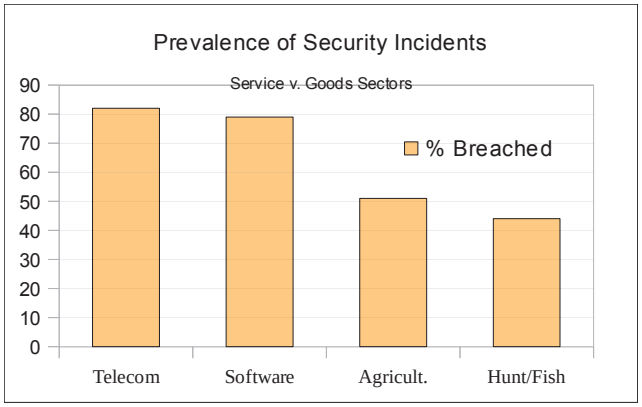


Figure 1. Prevalence of Security Incidents

2.5 MIL-STD-882E

The MIL-STD-882E is a military based document that describes the US Department of Defense Systems Engineering method of eliminating and reducing vulnerabilities. The Department of Defense uses severity and probability to define risk. A scale of 1 to 4 is used to define severity. The scale begins at 1, which designates catastrophic [1]. Catastrophic is defined as resulting in "death, permanent total disability, irreversible significant environmental impact, or monetary loss equal to or exceeding $10 million" [1]. The next two options are 2 meaning critical and 3 meaning marginal [1]. The last option is 4, meaning negligible [1]. Negligible is resulting in "injury or occupational illness not resulting in a lost work day, minimal environmental impact, or monetary loss less than $100 thousand" [1]. Probability of the item being stolen is another aspect to assessing risk. Probability is on a scale of A to F [1]. The levels of probability include: A. Frequent ("continuously experienced"), B. Probable ("will occur frequently"), C. Occasional ("will occur several times"), D. Remote ("unlikely, but can reasonably be expected to occur"), E. Improbable ("unlikely to occur, but possible"), and F. Eliminated ("incapable of occurrence") [1]. The severity categories and probability levels are then combined to create a Risk Assessment Code [1]. The Risk Assessment Matrix allocates a "risk level of High, Serious, Medium, or Low for each Risk Assessment Code" [1].

3 Procedure/Methods

We began this process by identifying a problem that we could solve. After researching various newspapers and academic articles, we found that there is not an accessible and affordable way for businesses and organizations to learn how to further secure their information.

After researching methods of securing companies as shown in the literature review, we organized a framework to identify areas needed to be secured and ways to secure them. The BSIMM and case studies were especially beneficial in determining these areas. As a part of the framework we developed metrics that must be defined to determine the best actions for a specific organization. The user of the framework will identify the business sector of the company/organization, number of IT employees, the total number of employees, the total number of IP addresses, the annual net income, and the information risk.

We divided the business sector metric into two choices: goods-producing and service-providing. The Bureau of Labor Statistics provides this concept of organizing each industry into two categories. The goods-producing sector includes Natural Resources and Mining (including Agriculture, Fishing, and Hunting), Gas and Oil Extraction, Construction, and Manufacturing. The service-providing sector includes Trade, Transportation, Utilities, Warehousing, Information, Finance, Insurance, Real Estate (including Renting and Leasing), Professional and Business Services, Management, Waste Management, Education, Health Services, Health Care, Social Assistance, Leisure and Hospitality, Entertainment and Recreation, Accommodation, and Food Services. Distinguishing which sector the organization represents will allow for more accurate responses due to the major differences in the types of security needed for companies in the different sectors.

The next metric is the number of employees. The user states the total number of employees and the number of Information Technology employees. This is a way to relate the size of the company with a plan for hiring more IT employees. These two numbers will also give a ratio of IT employees to total employees, which is beneficial.

The user of the framework will next determine the number of IP addresses used within the company. We chose for them to define IP addresses in order to include all devices. Simply determining the number of computers would not be an accurate representation, as it would exclude smart phones and tablets, which are becoming increasingly popular in businesses.

Annual net income of the organization will then be requested in the framework. Net income is the expenses

subtracted from the revenue. This value will help determine which actions the organization can afford to implement, such as buying new software or hiring employees. The net income will then go into a ratio with the total number of IP addresses. Dividing the number of IP addresses by the net income, results in a ratio of IP/$. This calculation is crucial in the determining of beneficial security actions.

The user of the framework will determine the two subcategories defined in the MIL-STD-882E to measure the information risk. The two components of risk are severity and probability. The severity of information is measured on a scale of 1 to 4. 1 meaning catastrophic, 2 meaning critical, 3 meaning marginal, and 4 meaning negligible. The definition of each of these was previously defined in the Literature Review. The user will then define the probability of the information being stolen on an A to F scale. A meaning frequent, B meaning probable, C meaning occasional, D meaning remote, E meaning improbable, and F meaning eliminated. The framework will then use the two numbers to determine the overall risk. This process is demonstrated in the table below [1].

RISK ASSESSMENT MATRIX				
SEVERITY / PROBABILITY	Catastrophic (1)	Critical (2)	Marginal (3)	Negligible (4)
Frequent (A)	High	High	Serious	Medium
Probable (B)	High	High	Serious	Medium
Occasional (C)	High	Serious	Medium	Low
Remote (D)	Serious	Medium	Medium	Low
Improbable (E)	Medium	Medium	Medium	Low
Eliminated (F)	Eliminated			

Table 1. Risk Assessment Matrix

The actions that the framework will suggest are collected from various sources including the BSIMM, academic articles, case studies, and the CISSP. They are then organized by the order in which they should be implemented. This organization will be unique for each business based on the metrics defined by the user.

4 Summary and Future Work

Each metric (Industry sector, Total Number of Employees, Number of IT Employees, Number of IP Addresses, Annual Net Income, Severity, and Probability) of this framework is crucial in determining what actions will be most beneficial for an organization. By implementing this framework, businesses and organizations of all industries and sizes will be able to increase their current level of security.

This framework is significant because it is comprehensive and comprehendible. The framework encompasses the necessities of all ten domains of security while it is also easy for all to understand. One does not have to be an information security analysist to use this framework. Anyone will be able to implement the framework's suggested security measures. The framework is also cost-effective and efficient. A business will be able to use this framework to increase their security instead of hiring a security analysist, which costs over $90,000 a year according to Bureau of Labor Statistics [7]. This framework is also efficient. It is quick to use and the suggested security measures can be applied immediately or can be used as a checklist to confirm a business' current level of security. This framework will revolutionize how businesses go about security.

This paper began with an introduction describing today's world of information technology. The literature review then examined the sources that were valuable to the creation of the framework. The method gave a detailed account of the process by which the framework was developed. The conclusion told the influence that this framework may have in the security industry.

In the future we would like to automate this process with software. The software would allow the user of the program to enter their organization's metrics. The software would then return a prioritized list of actions that would be best for that specific organization. Each action would include a description and an updated estimate cost of implementing it.

5 References

1. United States of America, Department of Defense, *Standard Practice System Safety,* MIL-STD-882E, 2012.
2. "Avoiding Social Engineering and Phishing Attacks Security Tip (ST04-014). *US-CERT,* United States Computer Emergency Readiness Team, Feb. 2013.
3. Y. Chen and K. Ramamurthy and K. Wen, "Organizations' Information Security Policy Compliance: Stick Or Carrot Approach?", *Journal Of Management Information Systems* 29.3, pp. 157-188, 2012.

4. Harris, Shon. *CISSP All-in-One Exam Guide,* Sixth ed. New York: McGraw-Hill Education, 2013.
5. "Home Depot Confirms Breach," *CNBC,* Sept. 2014.
6. P. Hunter, "Cyber Security's New Hard Line," *Engineering & Technology (17509637)* 8.8: pp. 68-71, 2013.
7. "Industries at a Glance: NAICS Code Index," *U.S. Bureau of Labor Statistics,* U.S. Bureau of Labor Statistics, n.d.
8. J. Lanz, "Cybersecurity Governance: The Role Of The Audit Committee And The CPA," *CPA Journal* 84.11: pp. 6-10, 2014.
9. G. McGraw and S. Migues and J. West, "The Software Security Framework (SSF)," *BSIMM-V*. N.p., n.d.
10. H. Saini and T.C. Panda, "Extended Cyber Defense Architecture For A University: A Case Study," *IUP Journal Of Science & Technology* 6.2: *Applied Science & Technology Source,* pp. 33-47, 2010.
11. G. Sampemane, "Internal Access Controls." *Communications Of The ACM* 58.1: *Applied Science & Technology Source, pp. 62-65,* 2015.
12. T. Scully, "The Cyber Security Threat Stops In The Boardroom," *Journal Of Business Continuity & Emergency Planning* 7.2: pp. 138-148, 2013.
13. A.N. Singh, et al, "Information Security Management (ISM) Practices: Lessons From Select Cases From India And Germany," *Global Journal Of Flexible Systems Management* 14.4: pp. 225-239, 2013.
14. "Target Shares Tumble As Retailer Reveals Cost Of Data Breach," *Forbes,* Forbes Magazine, August 2014.
15. R. R. Rantala, United States of America, Bureau of Justice Statistics. Office of Justice Programs, *Cybercrime against Businesses, 2005*. N.p.: n.p., 2008.

Securing the Internet of Things: Developing a Security Standard

A. Clark, R. Byrd
School of IT and Computing, Abilene Christian University, Abilene, Texas, U. S. A.

Abstract – *The Internet of Things (IoT) and security are seldom found in the same phrase. Consumers and producers are often concentrated on improving the consumer aspects of technology that they do not think about the necessary related security. The research goal was to develop a framework for establishing security standards for the IoT. That is accomplished through developing a standardized set of IoT categories (e.g., personal health, household appliances, business world) intersected with inter-networking media types (e.g., wired, wifi, bluetooth, 4G) that allow the exploration of possible attack vectors and their preventative measures. Next in the process is to establish requirements for each category and inter-networking subset. Finally, a cataloging system making the requirements accessible to manufacturers must be created. The IoT is a concept that is quickly taking over our world today and only with proper IoT security standards in place will it have the chance of being secure.*

1. Introduction

The Internet of Things (IoT) and security are seldom found in the same phrase. Our world is studiously focused on creating new and innovative technology that it does not stop to think about how the security of our lives determines the level of success.

1.1. Background

The Internet of Things is, and has been, a part of society starting in the late 1960s, when the first successful prototype of the Internet was created. The IoT is the idea that every single thing that we come into contact with throughout our day to day lives is connected, in some way, to the Internet. According to ARM, IoT is, "from a technology perspective, the IoT is being defined as smart machines interacting and communicating with other machines, objects, environments and infrastructures, resulting in volumes of data generated and processing of that data into useful actions that can "command and control" things and make life much easier for human beings … similar to the world envisioned in the 1970s cartoon *The Jetsons*, only better."[6]

1.2. Research Goal

We already live this concept today, with smart televisions and cell phones. However, not everything is connected to the Internet yet, so as technology advances in cell phones and house appliances, there are going to be more and more 'things' connected to the Internet. This poses the problem of security related to those items connected. As IoT grows, are we going to take the time to make each item is secure in its connection? That is the problem statement that we reached – to implement security into the Internet of Things.

1.3. Overview

In the following sections, we will review the information that is already on the Internet concerning IoT, analyze the results to discover what we can do about the problem (fixing the lack of security), and finally explain a possible solution to the problem.

2. Literature Review

In this section, we are going to identify areas of the Internet of Things that already exist in articles that we found, as well as detail the main connection methods that we decided to focus on for the purpose of this paper.

In our background research, we discovered the list of categories and sections that were developed by the Bureau of Labor Statistics (BLS). The result of investigating those sections was realizing that the creators implemented the categories for employees that have an occupation in every study area throughout the world. Through the findings, we decided to build categories of the IoT based on the BLS standards. In addition to the findings from the BLS, we did find several applications of use for IoT. One such article listed some example use cases under the IoT: machine-to-machine communication; machine-to-infrastructure communication; telehealth (remote or real-time pervasive monitoring of patients, diagnosis and drug delivery); continuous monitoring of, and firmware upgrades for, vehicles; asset tracking of goods on the move; automatic traffic management; remote security and control; environmental monitoring and control; home and industrial building automation; and "smart" applications, including cities, water, agriculture, buildings, grid, meters, broadband, cars, appliances, tags, animal farming and the environment, to name a few.[5] Another article found on the ARM website says that "IoT capabilities can be added to just about any

physical object including clothing, jewelry, thermostats, medical devices, household appliances, home automation, industrial controls, and even light bulbs."[6] Yet despite these and various other articles that describe similar categories, we could not find a complete list that puts all of them together in one article. Fortunately, through these, we can still see that the Internet of Things is already very broad in the world today.

3. Inter-Networking Media

Secondly, we need to investigate the various connectivity types of inter-networking media that can be used for IoT. Essentially, there is an infinite selection of inter-networking media that could be used, some not even found yet. For research purposes, we decided to focus on seven major ones that could easily be applied to the Internet of Things, ordered by convenience based on required assets of the specified connection type. They are as follows: a wired connection, an infrared (IR) connection, a blue-tooth (BT) connection, a radio frequency identification (RFID) connection, a WiFi connection, a 3G connection, and lastly, a 4G connection.

3.1. Wired Connections

A wired connection is one that uses a cable, specifically an Ethernet cable, to connect two or more devices together as long as there is an Ethernet adapter on each one of those devices[4]. The speed of this connection is 10 Mb/sec to 100 Mb/sec[4]. An example of this is connecting a computer to an Internet port on the wall with an Ethernet/LAN cable.

3.2. Infrared Connections

An IR connection is one that uses an LED to transmit the infrared signal into bursts of light that can then be processed as information as long as the device(s) is within a short distance and a direct line of sight. The speed of this connection is 1 Gb/sec up to 3 Gb/sec, but it is easily blocked by common objects, such as cars and people[8]. Some examples of an IR connection are computers (its mouse, floppy disk drives, printers, and keyboards), car locking systems, and home security systems.

3.3. Bluetooth Connections

Third on the list is a BT connection, which is one that connects a device to another paired device to transfer data (photos, music, calls, etc) within a distance of up to 100 meters, or 328 feet[3]. BT uses radio transmissions to do this, and the devices need to have a wireless capability built into them. The speed is measured by hops, 1600 hops/sec at 2.4-2.5 Ghz/sec[3]. An example of a blue-tooth connection is hooking up a cell phone with a car radio system that both have their blue-tooth capability turned on to play music while the person is driving.

3.4. Radio Frequency Identification Connections

A fourth connection type is radio frequency identification, or RFID. This particular connection method uses a tag, similar to security tags found on expensive technology in stores today, to store data about the object it is currently placed on. To transfer the data from the object to the server requesting it, radio waves are used to transmit the data. A large advantage with RFID is that the object does not have to in a direct line of sight to transfer, instead being able to transmit data several of hundreds of feet away using a variation of about 50 radio channels (or frequencies) with obstacles in its way[1]. RFID can be useful in the Internet of Things by connecting clothes to the Internet, since the little tag isn't much of an inconvenience if it is placed conspicuously.

3.5. WiFi Connections

The fifth connection type that we looked at is WiFi, perhaps one of the most common connections so far. WiFi is a wireless connection between a device(s) and a server using radio waves to connect, similar to walkie-talkies[9]. WiFi transmits data at frequencies of 2.4 GHz and/or 5GHz, which is higher than cell phones, walkie-talkies, televisions, etc[9]. The speed of WiFi is up to 450 Mb/s of data, depending on the frequency of the connection[9]. A wireless adapter is needed for the device to be able to connect to the hotspot, the main reason for WiFi not being the most convenient for IoT. Despite that, WiFi is a very convenient connection method because it does not require any tags or wires and the hotspot can support multiple devices at once.

3.6. 3G Connections

Sixth is 3G (third generation) – a data connection that enables users to access advanced services (such as sharing images, text messaging, e-mail, etc) anytime, anywhere, as long as the device with 3G has service[2]. The speed of this connection is 144 Kb/s in high mobility traffic (e.g., riding in a vehicle), 384 Kb/s in pedestrian traffic (e.g., walking), and 2 Mb/s or higher for indoor traffic (e.g., standing/sitting inside a building)[2]. This type of inter-networking media would be the most convenient because it is the most popular, if 4G didn't exist.

3.7. 4G Connections

A 4G (fourth generation) data connection can do everything a 3G connection can, except it connects the device and transmits data with a broadband

width ten times faster than 3G[7]. The download speed of 4G is 5-12 Mb/s, sometimes even reaching 50 Mb/s, and the upload speed is 2-5 Mb/s[7]. The high mobility communication speed is 100 Mb/s, and a speed of 1 Gb/s in low mobility communication[7]. 4G is quickly becoming the most common, popular, and convenient connection because it is currently the newest one available and usable. We feel 4G would be a valuable connection method for the Internet of Things because of it being as strong and popular as it is.

A	B	C	D	E	F
	Personal Health	Household Appliances	Personal Accessories	Electrical Appliances	Business World
Wired	Y	N	N	Y	Y
IR	N	Y	N	Y	Y
BT	N	Y	Y	Y	Y
RFID	Y	Y	Y	Y	Y
WiFi	Y	N	N	Y	Y
3G	Y	N	Y	N	Y
4G	Y	N	Y	N	Y

Fig 1. Security of the IoT Table

4. Methodology and Procedures

Given that we can categorize IoT, we need to create a complete categorization system, determining security procedures for each media in those categories. Figure 1.1 displays our idea of a finished system.

Using well-developed knowledge from our previous literature and some common sense, we have discovered that there are other areas that we need to make room for. As you can see in Figure 1.1, we have organized and completed the categories of the Internet of Things, as well as specifying which connection methods will most likely be used for the corresponding category. It could change depending on the object, personal preference, individual availability, and developments of new technology, so this is not final – but it's a start.

Because technology today is advancing, BT and WiFi are soon going to be the most popular connection methods for apparel. Furthermore, based on the Certified Information Systems Security Professional (CISSP) domains, we have reason to include physical security and encryption in our solution. That reason being we need to have a way to let the server and owner of the apparel know when a piece of clothing is not where it's supposed to be. We do have home security companies already in business so that won't be too hard, but regardless; it is still an issue that needs to be taken care of.

In Table 1. below, the horizontal axis represents the categories of the IoT, and the vertical axis consists of the connection media, organized by least convenient to most convenient, as explained in section 2.2. The cells between the two axes have either an N (no) or a Y (yes) in it. The N represents an inconvenient relationship between the category and the connection method, whereas the Y shows a convenient relationship between them. The ten CISSP domains helped us determine the two different relationships since they need to be consulted in every aspect. Our resulting conclusion is that for every category, medium, and domain, there is a solution to make the pairing of devices and the Internet together secure.

5. Explanation of Categories

The five main categories we developed, based on our research, cover all of the things that are in the IoT. In the description, there might be a few items that are missed, but most will be covered. When we took the BLS categories into consideration with the IoT, we decided to sub-sect two of their major headers, goods-producing services and service-producing services. There was one problem with this: each of the occupations in the two headers focused on the people-aspect of them. So in order to relate the sections to the IoT we changed their focus to the actual products of the occupations with the different sub-categories: personal health, household appliances, personal accessories, electrical appliances, and the business world.

5.1. Service-Producing Services

These services include all of the companies that produce products to better the lives of individuals throughout the world. Their products are part of the personal health and the electrical appliances categories. Some items in the other categories could be considered service-producing as well, however the companies that produce those items are not solely focused on assisting people live their lives. The CISSP domains cover all of these items in the categories by providing a section for each to fit into with their respective security.

5.2. Goods-Producing Services

These services include all of the companies that produce physical objects that we, as consumers, incorporate into our lives with the influence of varying motives. The sub-categories in this header include household appliances, personal accessories, and the business world. Most, if not all, of the

companies that produce the items that belong in these categories are focused on providing material possessions for our lives. Once again, the CISSP domains also cover every single one of these material objects by providing security for all of the ranges of these items, especially when they are connected to the Internet.

5.3. Service-Producing Services: Personal Health

The personal health category starts in the cell B1, with a wired connection, RFID, WiFi, 3G, and 4G methods found to be convenient for most, if not all, of what it includes. Toiletries, medical appliances (such as breathing machines, epi-pens, dialysis treatments, etc.), medicine, shower items, hair products, and any other items you might consider to be part of personal health is what this category is made up of. Security for these necessities is very important because if one or more of them were to go missing or be damaged, the owner's health could be in severe danger. Connecting them to the Internet, making them a part of the Internet of Things, would make them more secure because the owner would be able to trace the location of the stolen or damaged health necessity, then being able to either get it back or have a valid reason for asking for another one from the doctor or specialist. However, if the connection on the item was secure, there is a lower chance of that item being damaged or stolen in the beginning. Security for breathing machines or dialysis treatments, for example, could require a fingerprint, a retina scan, a passcode, or voice recognition in order to be used and/or taken from the place where it's kept. If someone is sick, they probably use a toothbrush to brush their teeth, and now their virus is on that toothbrush. If, for some reason, a person doesn't know where their own toothbrush is and would like to use that one, the security measures would prevent that person from using the brush with the virus on it, protecting them from the sickness in that aspect. This shows that security in the personal health category of the Internet of Things can protect both the owner and those around him/her.

5.4. Goods-Producing Services: Household Appliances

The household appliances category starts in the C1, with an IR, BT, and RFID connection being preferred for most, if not all, of what this particular category includes. Kitchen appliances that aren't electric (gas stoves and ovens, dishwashers, gas dryers, etc.), living/family room accessories (such as couches, recliners, coffee tables, etc.), dining room accessories (such as dinner tables, silverware, good china, etc.), and bedroom furniture (such as beds and dressers) all make up the household appliances category. Security for these items is obviously necessary, and a home security system is already in use today. There are multiple alarm systems by companies such as ATP that place cameras around the outside and inside of your house to keep watch over the valuable appliances. In addition to that, connecting all of these items to the Internet would also double, if not triple, the security on your household because we could create a library system that records the dates and times when a certain item was used and when it was put back in its place, so that if there is a question about anything, the owner can look at that, and then search the cameras for that specific date and time to see what happened. This would also help prevent mischief between siblings and parents, because the mystery of who did what would be gone just by looking at the library record. The security to access this record would be high, of course, with ways mentioned in previous sections, like the retina scan, fingerprint, voice recognition, etc.

5.5. Goods-Producing Services: Personal Accessories

The personal accessories category starts in cell D1 with BT, RFID, 3G, and 4G connection methods being preferred for all the items included in this particular category. Apparel, shoes, televisions, PCs, Apple technology (such as iPods, iPads, and Macs), bedroom accessories (excluding furniture), cars, and phones are included in this category. Keeping security on these items, when connected to the Internet, is vital to the owner since they are what makes the individual successful and happy in their personal life. To keep these items secure in the IoT, the library system mentioned in 3.2.2 will be used for them while they aren't being used, maintaining a record of when the owner used a particular accessory and quit using it. In addition to that, the home security system that is already in use in homes today would also apply. When the items are being used, 3G and 4G would conveniently work for the object because the data connection would keep it connected to the Internet wherever it went, including outside the home, transmitting the location and simple phrase of what exactly it's being used for. That information is transmitted to the person who requested it, who has passed the security measures protecting the log (fingerprint, retina scanner, password, etc.). This measure is efficient because if that particular accessory is stolen, the owner and/or police are able to track where it is and who has it, as well as what it's being used for at the time. To further this security on the object, the connection, whether it's BT, RFID, 3G, or 4G, won't have the option to disconnect. All of these security measures put together prove a high

secure level for all the individual's personal accessories.

5.6. Service-Producing Services: Electrical Appliances

The electrical appliances category begins in cell E1. The most efficient connection methods that would be used with this category are Wired, IR, BT, RFID, and WiFi. These are the preferred connection types because since this category consists of appliances that either stay in one place or stay under one roof. Kitchen appliances (such as beaters, refrigerators, electric stoves and ovens, microwaves, electric washers and dryers, etc.), wiring throughout the house, and the boiler room if it's electric are most of what makes up the electrical appliances category. Connecting these items to the Internet increases the security in the house itself as well as decreasing the probability of any intruder hacking into the house wiring system to either throw the inhabitants for a loop and/or leak a poisonous substance into the house. A connection to the Internet also makes it more beneficial for the owner because if he/she wants to leave even though the oven is on, or they forgot to turn something off before leaving for a long vacation, all that's needed is to go on the Internet, access the library that all of these appliances are kept on, and perform the action needed. Furthermore, the library system can let the owners who are on vacation know the status of all the appliances in case an intruder tried to mess with it while the owners were gone. This layered security gives a satisfied feeling to the owner so they don't have to worry about anything as well as keeping their home and all of its appliances secure through a connection to the Internet.

5.7. Goods-Producing Services: The Business World

The business world category starts in cell F1, with all seven connection methods chosen as preferred and convenient. Now this category includes all the business devices used exclusively for business (computers, phones, cars, etc.), servers, furniture in business buildings, and any other appliances found in the business world. Because this category consists of a large range of items, all seven of the types fit. Some of the items leave the building, some of them stay. In today's world, mostly every business requires an Internet connection in their building to successfully complete the day's work. This makes it even easier to extend that connection to everything in the building, heavily increasing the security of it all. This increased security assures the CEO and the employees that their prized success is safe, and if something does happen to any part of it, the fact that everything in the business is connected to the Internet makes the process of finding it efficient. All that's needed to do this is to have the right person pass the security measures that protect the log in the library system, locate the particular object in question, and see where it is at that time. This convenient process means that an action that would normally take down the business would not do that because it would be discovered and prevented before it could do much, if not any, harm. Because of this security assurance, connecting everything in yours and others business worlds to the Internet secures all that is in the building and a part of each employee's job at the office and at home.

6. Future Work

The IoT Security Chart is only one step in the process of fully developing security for the IoT – there are several steps after this. The following sub-sections describe our next two plans for this growing concept.

6.1. Set of Requirements

Just as the FDA has requirements that each product has to pass, the IoT also needs a set of requirements that each thing will have to pass. Creating this would provide guidelines to successfully add security to the Internet of Things which achieves our goal. These guidelines are now possible to develop because this paper has given us the background for them, leaving the set of requirements as our next step in this process, in the future.

6.2. Chart of Accounts

After the set of requirements has been established, a chart of accounts needs to be developed. This system was briefly explained in the previous sections, yet to be satisfactory, this system has to be researched and laid out. It means another paper describing the workings of the chart – how to secure the wide variety of information stored inside it, how to keep everything up to date, and how to have it work with the different connection media types. Once all of this is done, and the chart of accounts is working properly, then the IoT can officially be a part of our lives.

7. Conclusion

We have now finished our first step, starting the long process into furbishing the IoT. The foundation was laid by using the framework in the BLS for the basis of the categories in the IoT through background research, and implemented security through creating that needed framework. Next, we built the IoT Security Chart, taking care of the basic need for the security to even begin to apply, having the categories correspond to the connecting media. This chart

provides the means to generalize requirements by inter-networking media and IoT category pairs. These generalizations will allow for simplified, yet effective measures in the securing of the IoT. The Internet of Things is a concept that is quickly taking over our world today, and only with proper IoT Security standards in place, will it have the chance of being secure.

8. References

1. Moscatiello, Richard. "Basic Concepts in RFID Technology." *SlideShare*. LinkedIn Corporation, 2007. Web. Dec. 2014. <http://www.slideshare.net/PeterSam67/basic-concepts-in-rfid-technology>.

2. "3G Technology." *EngineersGarage*. EngineersGarage, 2012. Web. Dec. 2014. <http://www.engineersgarage.com/articles/what-is-3g-technology-specifications>.

3. "A Look at the Basics of Bluetooth Technology." *Bluetooth Basics*. Bluetooth, 2015. Web. Dec. 2014. <http://www.bluetooth.com/Pages/Basics.aspx>.

4. Muscatello, Joshua, and Joshua Martin. *Wireless Networks Security*. *Institute for Information Assurance*. Indiana University of Pennsylvania, 20 Apr. 2005. Web. Dec. 2014. <https://www.iup.edu/WorkArea/DownloadAsset.aspx?id=61283>.

5. "What the Internet of Things (IoT) Needs to Become a Reality." *What the Internet of Things (IoT) Needs to Become a Reality - White Paper* (2014): 2-11. *Freescale*. Freescale, May 2014. Web. Dec. 2014. <http://www.freescale.com/files/32bit/doc/white_paper/INTOTHNGSWP.pdf>.

6. "From Sensor to Server." *Internet of Things (IoT)*. ARM, 2014. Web. Dec. 2014. <http://www.arm.com/markets/internet-of-things-iot.php>.

7. Segan, Sascha. "3G vs. 4G: What's the Difference?" *PCMAG*. Ziff Davis, 10 Feb. 2015. Web. 15 Mar. 2015. <http://www.pcmag.com/article2/0,2817,2399984,00.asp>.

8. Kaine-Krolak, Maureen, and Mark Novak. "An Introduction to Infrared Technology: Applications in the Home, Classroom, Workplace, and Beyond ..." *An Introduction to Infrared Technology: Applications in the Home, Classroom, Workplace, and Beyond ...* Trace R&D Center, 1995. Web. Dec. 2014. <http://trace.wisc.edu/docs/ir_intro/ir_intro.htm>.

9. Brain, Marshall, Tracy V. Wilson and Bernadette Johnson. "How WiFi Works." *HowStuffWorks*. HowStuffWorks.com, 30 April 2001. Web. Dec. 2014. <http://computer.howstuffworks.com/wireless-network.htm>.

Device-based Secure Data Management Scheme in a Smart Home

Ho-Seok Ryu[1], and Jin Kwak[2]
[1]ISAA Lab., Department of Computer Engineering, Ajou University, Suwon, Korea
[2]Department of Information and Computer Engineering, Ajou University, Suwon, Korea

Abstract - *Due to the developments in IT, smart home services using network-based smart devices are becoming more diverse. A smart home provides users with numerous services, regardless of time and place, through interactions among users, objects, and services. However, there are security concerns such as data leakage, data forgery, and unidentified access. In case of smart home data is exposure at threats, smart home exist very danger into characteristic of smart home. This paper will examine smart home communication and analyze the security problems and security requirements. Based on this information, we will propose a device-based secure data management scheme for a smart home.*

Keywords: Smart home, Smart devices, Data management, Mobile.

1. Introduction

The use of smart devices is increasing as information communication technology continues to develop. There is an increase in the types of available smart devices, smart home devises, and smart health devices. Accordingly, the ubiquitous society has become a part of our lives and is still developing.

A smart home is an intelligent environment where users and home appliances send/receive information and data in real-time. The smart home can be divided into home platform technology, wired or wireless network technology, smart device technology, and green home technology. Users can control devices in the home in real time through wired or wireless network technology[1]. Through smart home communication, users can access the smart home's meter reading system, boiler control, lighting control, appliance control, and various services in external[2,3].

However, smart home communication is not immune to security threats because it is equipped with network functionality. Security threats such as data forgery, illegal access, and privacy invasion are a real possibility if the smart home is accessed by a malignant device. In addition, new security threats are arising with technology convergence. In addition, smart home exist second danger into characteristic of smart home.

In this paper, we propose a data management scheme that is secure and efficient for a smart home environment, overall reducing security concerns. This scheme can upload and download data to authenticated devices. We will analyze smart home security and propose a device-based, secure, data management scheme suitable for a smart home environment.

This paper is organized as follows. Section 2 describes a smart home. Section 3 analyzes the security requirements of a smart home communication network, and the security issues such a network faces. Section 4 proposes a device-based, secure, data management scheme suitable for a smart home. Section 5 presents a security analysis of our proposed scheme, and Section 6 concludes our findings.

2. The smart home

With the development in information communication technology, mounted wireless devices have become an integral part of many appliances and electronic devices, creating a class of devices called smart devices. . With the appearance of these smart devices, came the concept of the "smart home." Technology and services for smart homes are developing rapidly and are diverse. A smart home makes tasks in the user life more convenient and easy to perform. In addition, smart devices are becoming increasingly automated. The communication system of a smart home is composed of a wired or wireless network connected to smart home devices. It provides various services allowing the user to supervise the smart home, regardless of the time and the location of the user. Therefore, a smart home is the collection of a set of automated, smart devices, connected and communicating on a common network[4].

Smart home technology can be divided into home platform technology, wired or wireless network technology, smart device technology, and green home technology. Because home platform technology links home technology to external networks, it includes home-server, gateway, and home middle ware technology. Green home technology provides comfortable and economic life, including green management technology, green home-network technology, and smart grid interlock technology. In addition, smart device technology can be described as making use of existing appliances and sensors. The most important technology among smart home technology is the networking technology[5].

The networking technology of a smart home provides the connection between smart devices. Among various networking technologies, wireless network technology is drawing more attention as it continues to evolve at a rapid

pace and requires relatively low power. Some examples of such technologies are Wi-Fi, WPAN, 3G/4G/LTE, Bluetooth, Microwave, and Ethernet. Smart home devices provide remote control services by connecting the existing home appliances to a CPU and a wired, or wireless, network technology. Users can then be provided with smart home services by using a mobile device away from home.

3. Analysis of security problems and security requirements

In this section, we will analyze the problems that arise from data management in a smart home environment. Based on these issues, we will analyze the security requirements for such an environment[6].

3.1 Analysis of Security Problems

3.1.1. Data leakage

A user can download sharing so they can access their smart home externally through a wireless network. This makes it possible for an attacker to gain access to the home through an unauthenticated smart device. If the attacker leaks important information gained through access to the home, this is a breach of privacy and can lead to issues regarding confidentiality.

3.1.2. Data falsification

A smart home transfers data to a user through a wireless network. Accordingly, an attacker can gain access to the home through the network, and falsify data before it reaches the user. In addition, the attacker can intercept user commands to the smart home and can control the smart home system instead. Therefore, the integrity of important data stored in the smart home cannot be ensured.

3.1.3. Unauthorized access

When transmitted data is received via a wireless network in a smart home environment, an attacker is able to insert malignant code into smart devices, giving the attacker access to the home through an unauthenticated device. Smart devices that contain malignant code become zombies and can be used to send malignant mail and execute distributed denial of service (DDOS) attacks. In addition, cameras can be installed or activated in smart devices through malignant code, invading the smart home owner's privacy. These types of cyberattack are mounting continually, and pose serious security threats to users of smart homes.

3.2 Analysis of Security Requirements

3.2.1. Data confidentiality

Smart home data contains sensitive information such as private information, control messages, and confidential data, which is controlled through the network. Through unauthorized access, an attacker can obtain this information, leak private information and sensitive messages, and remotely control smart devices. To prevent these sorts of attacks, the device through which the user accesses the smart home should be authenticated and malicious the attacker hasn't to access to smart home.

3.2.2. Data integrity

The data of smart devices can be falsified via malicious devices that gained access through the wireless network. Thus, transferred data and messages should not be prone to falsification from illegal smart devices in a smart home environment.

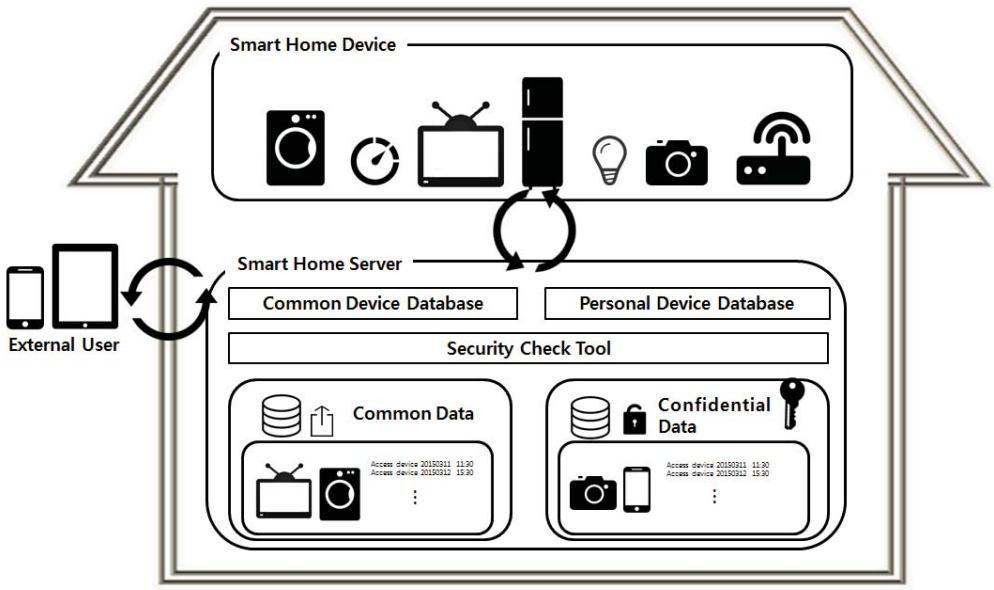


Fig. 1. Proposed scheme

3.2.3. Device authentication

Many smart devices can be accessed by devices without regard for security, allowing unauthenticated smart devices to be accessed through the smart home's wireless network. Disposable and cloned smart home devices can access the smart home, allowing malignant code to be inserted into the smart device. This compromises the smart home communication and creates zombie smart devices. Also, a smart home system can become dangerous if the attacker can disguise the attack as though it is from a smart device within the home. Thus, the authentication of smart devices is essential to the smart home environment.

4. Proposed scheme

In this section, a server safely stores and manages the data of the smart home.

We proposed a data management scheme, in which this secure smart home server manages the data of all smart devices registered in the home. The server stores data that is divided by importance into public data and confidential data. This allows for secure and convenient data management. Confidential data can only be accessed through use of a password. Additionally, a security check tool scans the integrity of the data before it is saved to the server. Also, data be saved and download through an authenticated device, enhancing the safety and reliability of the data. However, even if the authentication device it that have not access authority can't download data.

The proposed scheme is composed of three phases: the registration phase in which some rules need to be met by a smart device in order to register with the server; the data storage phase, in which a smart device saves data to the server; and the download phase, in which a user's smart device downloads data from the server.

4.1 Notations

Table 1 shows the notations used to explain the process of the proposed scheme.

TABLE I. NOTATIONS

Notation	Description
DeviceInfo	Smart home device's information
DeviceInfo'	Smart home device's information authentication requested
PK_D	Public key of a smart device
PK_S	Public key of a smart server
N	Random number
T_S	Time stamp from smart home server
T_D	Time stamp from a smart home device
ΔT	Valid time interval for transmission delay
V_C	Value access to confidential data
V_P	Value access to public data

4.2 Registration Phase

In the registration phase, new smart devices are registered to the smart home server and are divided into separate groups in order to separate public data and confidential data. The procedure is as follows.

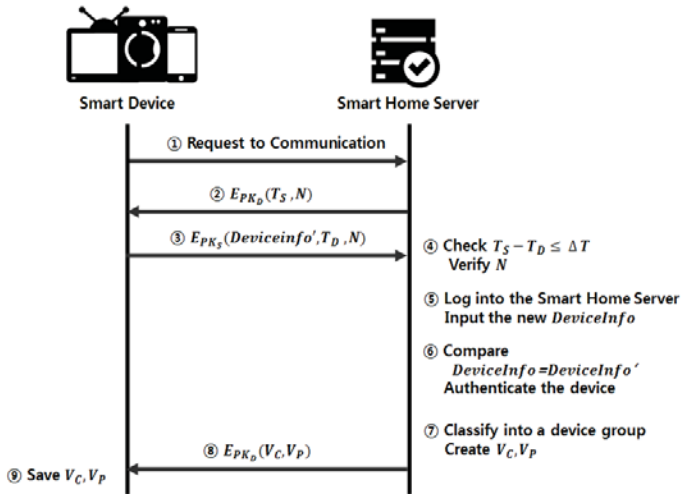


Fig. 2. Registration phase

1) A smart device requests to communicate with the smart home server.

2) The smart home server encrypts its time stamp and a random number into public key for the smart device in order to prevent reply attacks, and transfers this key to the device.

3) The smart device encrypts its information, its time stamp, and random number into the public key for the smart home server, and transfers this key to the server.

4) The smart home server validates the time interval for the transmission delay by comparing the differential between the time stamp of the smart home server and the time stamp of the smart device.

$$T_S - T_D \leq \Delta T$$

5) A user logs into the smart home server using their ID and password, and inputs the serial number and information of the device.

6) The smart home server authenticates that the smart device information received and user-input, smart device information are the same.

7) The authenticated smart device is classified into a device group and is granted access to the data, where it creates a value access to the data. This value consists of two things: a value access to public data, and a value access to confidential data. The smart home server creates values appropriate to the smart devices.

8) The smart home server encrypts the value access into the public key of the smart device and transfers it to the smart device.

9) The smart device saves the value access to the data, and communicates with the server that it is ready to exit.

4.3 Data Storage Phase

This section describes the procedure for data generating or data acquired smart device connecting to the smart home server, verification data. In addition, we will discuss the rules used to store the data security level.

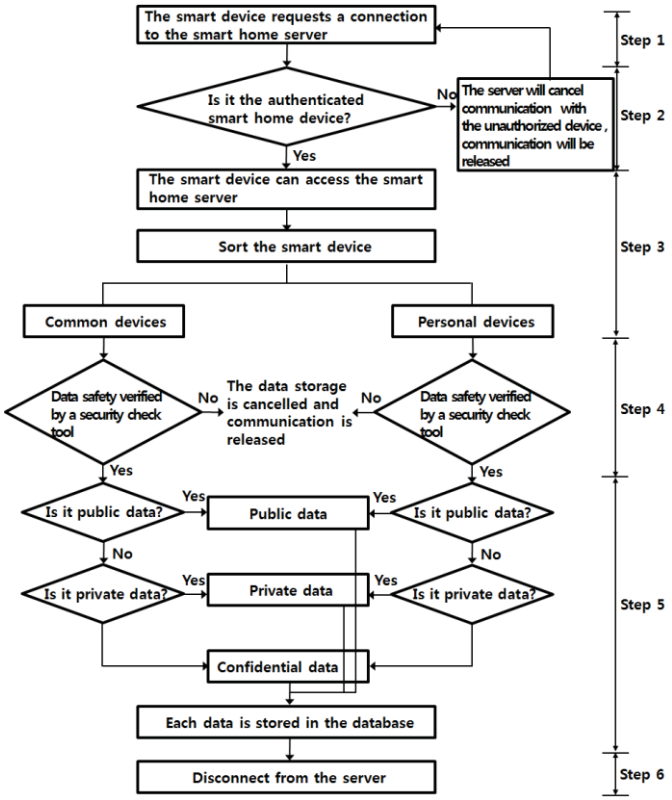


Fig. 3. Data storage phase

1) The smart device requests a connection to the smart home server in order to generate/acquire synchronized data.

2) The smart home server authenticates the device by comparing the smart device information registered during the registration phase to the information of the requesting smart device. If this smart device is not an authenticated device, the server will cancel communication with the unauthorized device and communication will be released.

3) Authenticated smart devices can access the smart home server. Smart device are sorted into either common devices, which are used together, or personal devices, which are personally used.

4) Data safety is verified by a security check tool in the assorted smart device. If a virus is found, the data storage is cancelled and communication is released.

5) Data verified by the security check tool as fit for storage, is divided into either public data or confidential data for secure and convenient data management. When storing public data, the smart home server stores the hash of the value access to public data and the data itself.

$$D_{store} = H(D||V_p)$$

When storing confidential data, the smart home server stores the hash of the value access to confidential data and the data itself.

$$D_{store} = H(D||V_S)$$

6) The data is stored in the database, disconnects from the server.

4.4 Data Download Phase

This section describes the download procedure using a user's smart device to request necessary data. The user connects to the smart home server through a smart device and can download data if they have appropriate authorization.

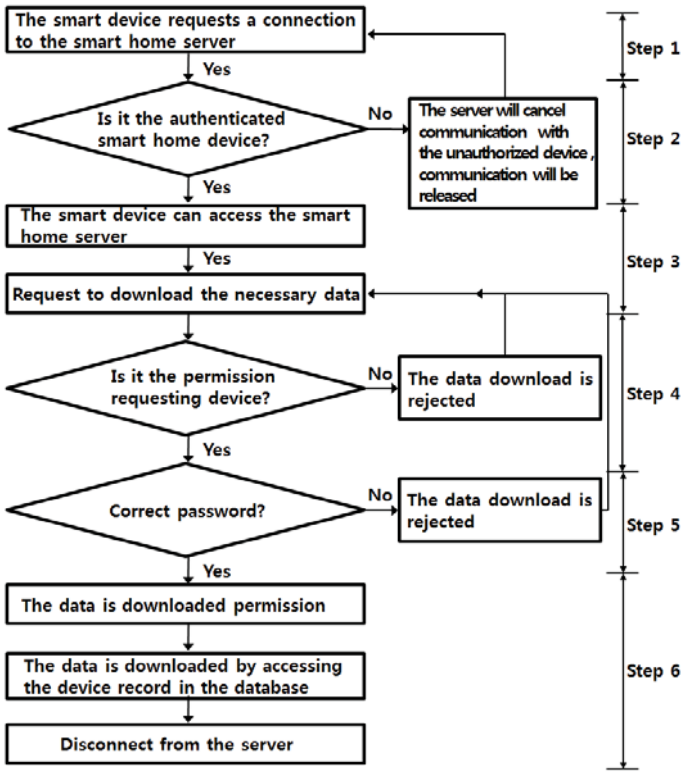


Fig. 4. Data download phase

1) The smart device requests to connect to the smart home server in order to download the necessary data.

2) The smart home server authenticates the device by comparing the smart device information registered during the registration phase to the information of the requesting smart device. If this smart device is not an authenticated device, the server will cancel communication with the unauthorized device, and communication will be released.

3) Authenticated smart devices can access the smart home server. The user is now able to request to download the necessary data through the smart device.

4) The smart home server confirms that the requesting smart device has download permission. If the device does not have permission, then the server rejects the data download and returns to step 3).

5) If the smart device has download permission, then user authentication is required through a request for the user to enter their password. Hashed data will be downloaded by decrypting the value access to the data. If the user authentication fails, the data download is rejected and returns to step 3).

6) If the user authentication succeeds, then the smart home server has permission to download the data. The data is downloaded by accessing the device record in the database, and then disconnects from the server.

5. Security analysis of the proposed scheme

In this section, we analyze the security of our proposed device-based, secure, data management scheme in smart home environment.

5.1 Confidentiality

A smart device must ensure confidentiality because it has important data such as private information, control messages, and sense information. This paper's proposed scheme is to authentication smart device before allowing access to the smart home server. Unauthenticated smart devices are not allowed to store and download data because they do not have access to the smart home server. Even if a user loses a device, or device information is leaked, an attacker cannot access the data on the smart home server because they must have password. In addition, if the smart device was authenticated, the user cannot access the data on the smart home server if they do not know password because the data is divided into encrypted public data and encrypted confidential data.

5.2 Integrity

Data is prone to risks such as data and message falsification by the access of malicious smart devices through the wireless network in a smart home communication environment. This paper proposes a scheme in which data is stored in a hash with a value access when the data is stored to the smart home server. When smart devices download the data, value access that has authority with data will be encrypt. Therefore, this proposed scheme prevents data falsification.

5.3 Device Authentication

Smart devices can insert malignant code through unauthenticated device access. In this situation, the smart device will become a zombie device. It is able to send malignant mail and execute distributed denial of service (DDOS) attack. Our proposed scheme prevents the change of smart device information because the smart home server saves the information of each smart device during the initial registration phase, saving hash values of this information. By using the hash values for communication, the information of the smart devices cannot be changed. In addition, because the smart home server supervises all of the smart devices of the home, access of unauthorized devices can be prevented and authentication of smart devices can be provided.

6. Conclusions

Smart home technology continues to develop and provides various services through open network communication among smart devices. However, there are still security concerns such as data forgery, unidentified access, and invasion of privacy, and new security threats continue to arise. In order to address this, we need a safe data-management method to prevent these security threats.

In this paper, we analyzed the security concerns and security requirements and suggested a safe data management method based on the devices in the smart home environment. This suggested method can block unauthorized access through device verification.

Research regarding smart homes is currently booming, both nationally and worldwide. Safe data management is very important because the smart home contains sensitive data. Finally, we expect that the suggestions made in this paper will be helpful to future studies and developments regarding a safe smart home environment.

7. Acknowledgment

This work was supported by the ICT R&D program of MSIP/IITP, Republic of Korea. [13-912-06-003, Development of Mobile S/W Security Testing Tools for Detecting New Vulnerabilities of Android]

8. References

[1] Gao Chong, Ling Zhihao, Yuan Yifeng, "The research and implement of smart home system based on Internet of Things," pp.2944-2947, Sept. 2011

[2] Hwa-jeong Suh, Dong-gun Lee, Jong-seok Choe, Ho-won Kim, "IoT security technology trends" The Korea Institute of Electromagnetic Engineering and Science, Vol. 24, No. 4, pp. 27–35, July. 2013

[3] Tae-woong Lee, Cheol-su Son, Won-jung Kim, "The Implement of Intelligent Home Network System on Smart Phone," The Korea Institute of Electronic Communication Sciences, Vol. 6, No. 4, pp 505-509, Aug. 2011

[4] Ji-Yean Son, Ji-Hyun Lee, Jeu-Young Kim, Jun-Hee Park, Young-Hee Lee, "RAFD: Resource-aware fault diagnosis system for home environment with smart devices," Consumer Electronics, IEEE Transactions on, Vol. 58, No. 4, pp. 1185-1193, Jan. 2013

[5] Seong-gu Sim, Ho-jin Park, Jun-hee Park, "Smart home standardization construction and strategy," The Korea Institute of Information Scientists & Engineers, Vol. 30, No. 8, pp. 19–25, Aug. 2012

[6] A. Wright, "Cyber security for the power grid: cyber security issues & Securing control systems," ACMCCS, Nov. 2009

Group management and security management on the open horizontal integration model for IoE

Enrique Festijo and Younchan Jung
School of Information, Communications and Electronics Engineering,
Catholic University of Korea, Bucheon-si, Gyeonggi-do, Korea 420-743

Abstract— *The advent of IoE calls for an effective group management and reliable security for numerous groups of heterogeneous devices that are interconnected. This paper proposes group management and security management for IoE based on the open horizontal integration model with SDN and NFV concept. The overlay network controller acts as an SDN controller capable of handling network functions such as group management and security management. For group management, we propose a group-based key management architecture based on Identity-based cryptography and Diffie-Hellman key exchange. For security management, we use the packet key scheme that is based on the idea that relatively small key size can be used as long as it provides high level of security. This kind of technique is practically applicable especially in a resource constrained environment such as in IoE, which demands for 'light weight' encryption/decryption technique as well as high level of security.*

Keywords: Group key management, Security management, IoE security, SDN, NFV

1. Introduction

The fifth generation (5G) of wireless world is projected towards a smart environment. Today, it is realized as Internet of Things (IoT) where heterogeneous things can sense and can communicate mostly wireless. Almost as soon as the IoT has taken root conceptually, the idea evolved into a more broadly conceived Internet of Everything (IoE). Internet of Everything (IoE) is the network connection of people, process, data and things. With these current technological trends, the wireless world is increasingly calling for the effective management and reliable security of heterogeneous infrastructures and devices to meet the future Internet (FI) requirements [1]. Current networks are vertically integrated. In a vertical scenario, propriety protocols and technologies have their own domain specific devices and applications [2]. For example, specific domain applications such as manufacturing, healthcare, logistics, and etc., have their own particular protocols and technologies to meet specific requirements. In a network point of view, this means that the control plane (that decides how to handle network traffic) and the data plane (that forwards the traffic according to the decisions made by the control plane) are bundled inside the networking devices, reducing flexibility and hindering innovation and evolution of the networking infrastructure [3]. These are the main reasons why transition from a closed vertical solutions to open horizontal solution is now a trend. In a horizontal scenario, easy support of wide diversity of Internet of Everything (IoE) applications can be achieved. From an architectural point of view, and as far as wireless communication technologies are concerned, horizontal integration model addresses the main issue of interoperability.

This paper aims to address the issue of group management difficulty and complexity as well as security management issues on a conventional network, which is based on a close vertical model. We propose group management and security management for IoE based on the open horizontal integration model with SDN and NFV concept. Our group-based key management aims to ensure high level of security among group members in the IoE environment by using an exclusive group key and packet keys. It is based on Identity-based cryptography (IBC), which offers a very interesting property that the device's public key is related directly to its identity (ID). Moreover, in this paper, the device's private key is generated by a private key generator (PKG) in which in this paper is also referred to the overlay network controller (ONC). ONC, which is located at the control layer and acts as an SDN controller capable of handling network functions such as group management and security management. Furthermore, since conventional group key generation and enryption/decryption techniques usually require long computational times and high computational load, which is not completely acceptable when applied in resource constrained IoE environment, this paper proposes a "light-weight" group key and packet key generation for security management.

In Section II, we discuss the background and related works for our proposed scheme. In Section III, we provide a detailed explanation for our proposed secure and private group-based IoE. Section IV shows the security strength and latency analysis of our scheme followed by conclusion in Section V.

2. Background and Related Works

2.1 Current technology trends for enabling intelligent toward 5G (SDN and NFV)

The introduction of intelligence toward 5G is the current research trend that can address the complexity of the heterogeneous networks and devices. The main emerging technologies for applying intelligence toward 5G communications are Software Defined Networking (SDN) and Network Function Virtualization (NFV) [1].

2.1.1 Software-defined networking

SDN is an emerging networking paradigm that gives hope to change the limitations of current vertical network infrastructures [3]. First, it breaks the vertical integration by separating the network's control logic (the control plane) from the underlying routers and switches that forward the traffic (the data plane). Second, with the separation of the control and data planes, network switches become simple forwarding devices and the control logic is implemented in a logically centralized controller (or network operating system), simplifying policy enforcement and network (re)configuration and evolution. SDN network architecture has a centralized network controller in the control plane, which is responsible for allocating traffic to network elements in the separated data plane of the network. The network controller centrally maintains the intelligence and state of the entire network.

2.1.2 Network Function Virtualisation

NFV also known as virtual network function (VNF) is a new way to build an end-to-end network infrastructure with evolving standard IT virtualisation technology so as to enable the consolidation of many heterogeneous network devices onto industry standard high-volume servers, switches, and storage [4], [5]. The network function of a network device is implemented in a software package, which is running in virtual machine(s). Therefore, it would become easier to introduce or test a new network function by simply installing or upgrading a software package, that is run by the servers. While SDN separates the control and forwarding planes to offer a centralized view of the network, NFV primarily focuses on optimizing the network services themselves. It offers a new way to design, deploy and manage networking services. It decouples the network functions, such as network address translation (NAT), firewalling, intrusion detection, domain name service (DNS), and caching, to name a few, from proprietary hardware appliances so they can run in a software.

2.2 Overlay Peer-to-peer Network

For an efficient group management, each member(node) of the group must maintain a reliable network connection. In this paper we used the concept of an overlay network. An overlay network is a computer network which is built on top of another network [6]. Overlay networks are becoming widely used for content delivery and file sharing services because they provide effective and reliable services by creating a virtual topology on top of existing networks. Examples of overlay networks include cloud provider networks, peer-to-peer (P2P) networks, virtual private networks (VPNs), content delivery networks (CDNs), experimental networks, and voice over IP (VoIP) services such as Skype. Skype uses an overlay peer-to-peer network [7]. There are two types of nodes in this overlay network, ordinary hosts and super nodes. An ordinary host is a Skype application that can be used to place voice calls and send text messages. A super node is an ordinary host's end-point on the Skype network. An ordinary host must connect to a super node and must authenticate itself with the Skype login server.

3. Secure and private group-based IoE

In this section, we discuss our proposed group management and security management for IoE, that is based on the open horizontal integration model approach. This paper aims to address the issue of management difficulty and complexity of a conventional network, which is based on a close vertical model. As shown in Fig. 1, our approach is based on an open horizontal integration model.

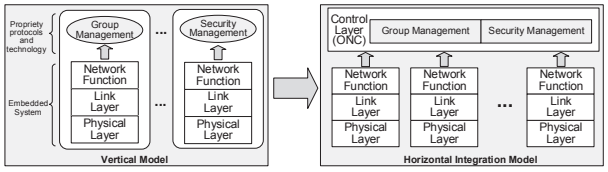


Fig. 1: Group key management and security management based on horizontal integration model

3.1 Group-based key management

Our group-based key management aims to ensure high level of security among group members in the IoE environment. This means that only authorized member of the same group have the ability to communicate securely. Secure communication among the group members is achieved by using an exclusive group key and packet keys. As illustrated in Fig. 2, our group-based key management is based on Identity-based cryptography (IBC),which offers a very interesting property that the device's public key is related directly to its identity (ID). Moreover, in this paper, the device's private key is generated by a private key generator (PKG), which is also referred to overlay network controller (ONC). ONC, which is located at the control layer, acts as an SDN controller capable of handling network functions such as group management and security management.

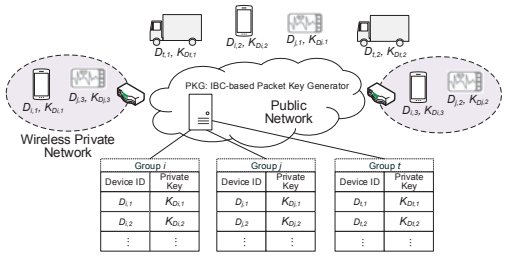


Fig. 2: Group management based on IBC

3.1.1 Joining a group and obtaining the group key

Each mobile node in the group has its own public key, which is related to its unique ID. The ONC (PKG), which is located in the control layer, has the private keys of all the authorized group members. In the initial setup, the group agrees with the global parameters q (prime number) and α (primitive root of q). These parameters are used for Diffie-Hellman key exchange method to create group key and packet keys. As shown in Fig. 3, D_5 sends a JOIN packet via UDP to ONC with state "J" implying that it wants to join the group. To fully understand the joining procedure, we itemized below the procedures that need to be followed. Based on Fig. 3, the following are the step-by-step procedures for group joining a secure group and obtaining the group key.

- D_5 sends JOIN packet to ONC via UDP, $(D_5 \| E(D_5 \| N_{D_5} \| IP_{D_5} \| PN_{D_5} \| IP_{ONC} \| K_{G,prv}, D_5))$.
- ONC decrypts the encrypted message in the JOIN packet and performs authentication.
- If D_5 is found to be an authorized node, ONC sends JOIN_OK packet, $(E(N_{D_5} + 1, K_{G,n}))$.
- D_5 decrypts the encrypted message in the JOIN_OK packet. To decrypt the encrypted message in the JOIN_OK packet, D_5 computes first for the intermediate key based on the Leaf Order in the key tree and blind key information obtained from the JOIN_OK packet. D_5 intermediate key computation is as follow.

$$K_{3,4} = BK_{L4}(K_{D_2})^{K_{D_5}} \bmod q. \qquad (1)$$

- After D_5 computes for the intermediate key $K_{3,4}$, it can now decrypt the encrypted message $(E(N_{D_5} + 1 \| K_{G,n}, K_{3,4}))$ in the JOIN_OK packet. Hence, it can obtain the group key $K_{G,n}$.
- ONC and D_5 established an overlay TCP connection. ONC then updates its table with the information it obtains from the encrypted message of JOIN packet. At the same time, D_5 also updates its table with the information it obtains from the encrypted message of the JOIN_OK packet (e.g. leaf order and $K_{G,n}$).

Because of the group key lifetime expiration, as illustrated in Fig. 4, group key rekeying is done periodically. ONC sends

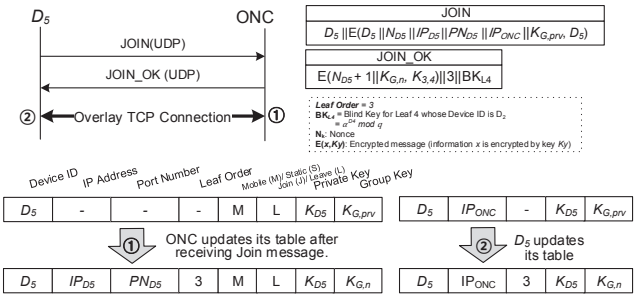


Fig. 3: Mobile node joins the secure group and obtains the group key

RekeyingDistribute message every rekeying period (T_{RK}). After receiving RekeyingDistribute message, each member of the group looks first for its leaf order in the key tree and for the corresponding blind key information of its adjacent pair. For example, D_1 looks for its corresponding blind key, that is $BK_{L2}(K_{D_3})$. Next, D_1 computes for the intermediate key, $K_{1,2} = BK_{L2}(K_{D_3})^{K_{D_1}} \bmod q$. Finally, D_1 uses the intermediate key $K_{1,2}$ to decrypt $E(K_{1,8}, K_{1,2})$ and obtain the group key, that is, $K_{1,8} = D[E(K_{1,8}, K_{1,2})]$.

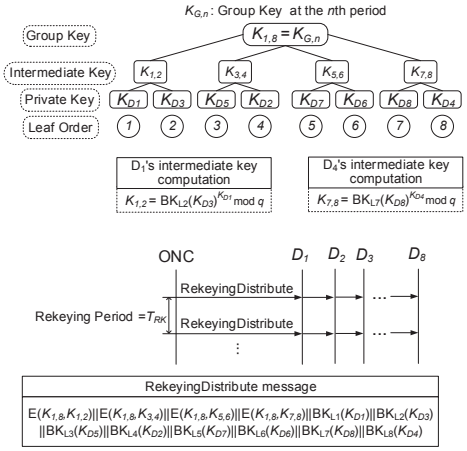


Fig. 4: Intermediate key computation and group key rekeying

3.2 Security management

For our security management, we propose a packet key scheme, that is based on the idea that relatively small key size can be used as long as it provides high level of security. The use of smaller key means less computational load and encryption latency. This kind of technique is practically applicable especially in a resource constrain environment such as in IoE, which demands for 'light weight' encryption/decryption technique as well as high level of security.

3.2.1 Packet key generation

Fig. 5 shows how packet key generation and message exchange is done in order to achieve secure peer-to-peer communication between D_i and D_j. In this scenario, they already have overlay TCP connection to ONC.

- D_i sends a BVV (Blind Value Vector) Request to ONC via TCP. The BVV Request contains the device ID of D_i and D_j concatenate by a nonce value and is encrypted by the group key, $(E((D_i, D_j)\|N_{D_i}, K_{G,n}))$. ONC forwards this request to D_j.
- D_i also sends a Binding Request via UDP to ONC. Binding request contains nonce value encrypted by the group key, $(E(N_{D_i} + 1, K_{G,n}))$. ONC informs D_j about this binding request by sending Binding Inform message via TCP. Binding Inform, which is encrypted by the group key, contains the UDP IP address and port number of D_j concatenated with the nonce and Diffie-Hellman global parameters (α, q), $((E((UN_{i_{IP}}, UN_{i_P})\|N_{D_i} + 1\|(\alpha, q), K_{G,n}))$.
- D_j sends a Binding Request to ONC via UDP. This Binding Request contains nonce encrypted by the group key, $(E(N_{D_i} + 2, K_{G,n}))$. ONC informs D_i about this binding request by sending Binding Inform message via TCP. Binding Inform, which is encrypted by the group key, contains the UDP IP address and port number of D_i concatenated with the nonce and Diffie-Hellman global parameters (α, q), $((E((UN_{j_{IP}}, UN_{j_P})\|N_{D_i} + 2\|(\alpha, q), K_{G,n}))$.
- After D_i and D_j receives the Binding Inform from ONC, a UDP connection is now established between them.
- D_i generates its SVV (SVV_{D_i})and computes its BVV (BVV_{D_i}). D_i directly sends BVV Send message to D_j via UDP. BVV Send contains the nonce value concatenated with the BVV_{D_i} encrypted by the group key, $(E(N_{D_i} + 3\|BVV_{D_i}, K_{G,n}))$.
- D_j generates its SVV (SVV_{D_j}) and computes its BVV (BVV_{D_j}). D_j directly sends BVV Send message to D_i via UDP. BVV Send contains the nonce value concatenated with the BVV_{D_j} encrypted by the group key, $(E(N_{D_j} + 4\|BVV_{D_j}, K_{G,n}))$.
- After the BVV Send exchange, D_i and D_j now has the BVV of each other. The BVV is use for packet key computation for encryption and decryption.
- For packet key verification, D_i sends PacketKey Verify to D_j via UDP. PacketKey Verify contains the index of the BVV used concatenated with the nonce encrypted by packet key, $I_n\|(E(N_{D_i} + 5\|, K^p_{D_{i_n}}))$.
- D_j also sends PacketKey Verify to D_i. PacketKey Verify contains the index of the BVV used, concatenated with the nonce encrypted by packet key, $I_m\|(E(N_{D_i} + 6\|, K^p_{D_{j_m}}))$.

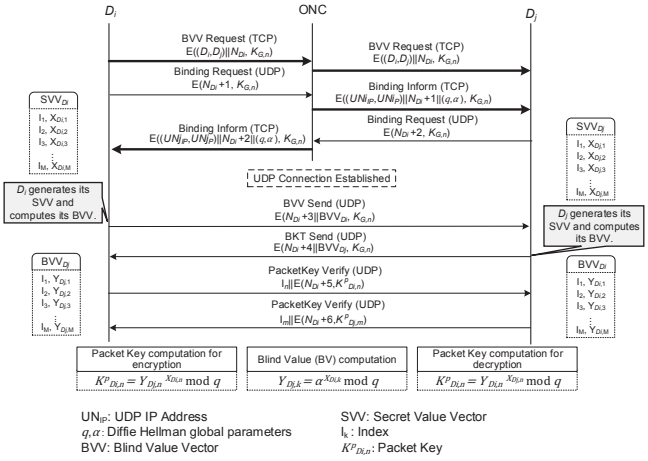


Fig. 5: Packet key generation for security management

3.2.2 Use of packet key for reliable UDP traffic

As illustrated in the Fig. 7, to achieve a peer-to-peer reliable UDP traffic in IoE, packet key is used for encryption and decryption of all the messages. In this scenario, BVV exchanges between D_i and D_j are already done. D_i has BVV of D_j (BVV_{D_j}) and its own SVV (SVV_{D_i}). On the other hand, D_j also has BVV of D_i (BVV_{D_i}) and its own SVV (SVV_{D_j}). D_i and D_j uses the selected blind key from the BVV, which is chosen in a round robin manner, to generate a packet key for encryption and decryption. As shown in Fig. 6, the D_j generates the packet key $K^P_{D_i,n}$ depending on the blind key $Y_{D_i,n}$ selected from the BVV and the secret value $X_{D_i,n}$ from the SVV.

$$K^P_{D_i,n} = Y_{D_i,n}{}^{X_{D_i,n}} \bmod q. \tag{2}$$

The packet key $K^P_{D_i,n}$ is used to derive the keystream $KS^P_{D_i,n}$ via RC4 algorithm.

$$KS_{A,i} = \text{ RC4 } (K_{A,i}). \tag{3}$$

D_i sends IoT Message to D_j via UDP. IoT Message

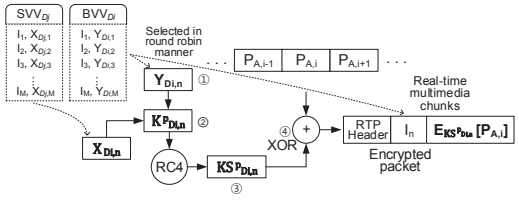


Fig. 6: Use of different blind keys selected from the BVV for real-time packet key generation and encryption

contains the index value of BVV used for packet key

computation. It also contains the message of D_i concatenated with the nonce value encrypted by the packet key, $I_g \| (E(Msg_{D_i} \|, K^p_{D_{i_g}}))$. After receiving the IoT message from D_i, Dj first look for the index value of the BVV (I_g), then computes the packet key for decryption. Next, D_j sends IoT Message Ack to D_i via UDP. IoT Message Ack contains the index value of BVV used for packet key computation. It also contains the nonce value encrypted by the packet key, $I_h \| (E(N_{D_i} + 1 \|, K^p_{D_{i_h}}))$.

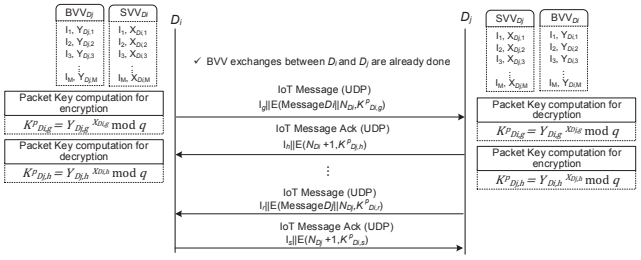


Fig. 7: Use of packet key for reliable UDP traffic

3.2.3 Use of packet key for real-time voice communication with QoS satisfaction

One of the applications of our packet key scheme is for secure real-time voice communication with QoS satisfaction. As shown in Fig. 8, given the scenario that BVV exchanges between D_i and D_j are already done. D_i has BVV of D_j (BVV_{D_j}) and its own SVV (SVV_{D_i}) and D_j also has BVV of D_i (BVV_{D_i}) and its own SVV (SVV_{D_j}), secure real-time voice communication is achieved by using the real-time packet key generation for encryption and decryption of the voice chunks.The voice stream exchanges happen between D_i and D_i with voice chunk encrypted by packet key.

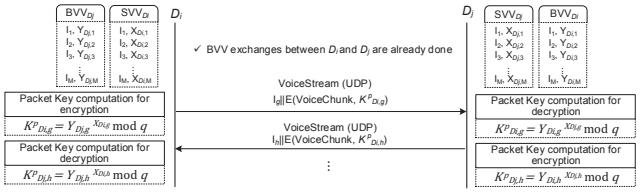


Fig. 8: Use of packet key for real-time voice communication with QoS satisfaction

4. Latency and security strength analysis

4.1 Group key computational load and network load for group key management

As shown in Fig. 4, the group key agreement process requires computation of intermediate key and the time it takes to decrypt the RekeyingDistribute message. For example, as illustrated in Fig 4, to obtain the group key $K_{1,8}$, D_4 needs to compute first the intermediate key $K_{7,8}$ then decrypt the RekeyingDistribute message, that is, $D[E(K_{1,8}, K_{7,8})]$. The intermediate key computation requires different computation load depending on the key size. We define l^k_{unit} as the time it takes for a device to compute the intermediate key as function of key size k. We also define D_{time} as the decryption latency for the computation $K_{1,8} = D[E(K_{1,8}, K_{7,8})]$. For the purpose of measuring the D_{time}, we found out through our simulation that it takes around 84 milliseconds to execute RC4 encryption/decryption for 1000-byte payload regardless of the key size.

Table 1, shows the intermediate key computation latency as a function of key size. We obtained those results from simulation using an android tablet (LG G Pad 7) with Quad-core 1.2 GHz Cortex-A7 CPU and internal memory of 1GB RAM. So, for the performance of our group key management, we performed the following analysis.

Table 1: Intermediate key computation latency as a function of key size

Key size in digit (bits)	Latency (l^k_{unit}) in msec
50 (166)	1.0529
60 (200)	3.1975
70 (233)	4.6682
80 (266)	7.1141
90 (299)	7.8359
100 (332)	8.9977

- The group computational load $GK_{C_{load}}$ in milliseconds, which is the computational load required for a device to obtain each group key. $GK_{C_{load}}$ depends on the parameters: l^k_{unit}, and D_{time}.

$$GK_{C_{load}} = l^k_{unit} + D_{time} \quad (4)$$

- The network load NET_{load} in bits per second, which is the per-link network load required between the device and the network. NET_{load} depends on the parameters: BK_{size}, N and $\lambda_{T_{RK}}$.

$$NET_{load} = BK_{size} \times (N + \frac{N}{2}) \times \lambda_{T_{RK}} \quad (5)$$

where BK_{size} is blind key size in bits, N is the number of nodes in the group and $\lambda_{T_{RK}}$ as the rekeying rate, that is, $\frac{1}{T_{RK}}$.

Fig. 9 shows the group key computational load for different group key sizes. $GK_{C_{load}}$ increases from 86 to 94 milliseconds as the group key size varies from 50 to 100 digits. It is shown that an 80-digit key causes $GK_{C_{load}}$ of 92 milliseconds. Such amount of latency is tolerable for a resource constrained environment such as in IoE.

Fig. 10 shows the per-link network load in bits per second (bps) versus the number of nodes in the group for different $\lambda_{T_{RK}}$. For $\lambda_{T_{RK}} = \frac{1}{30}$ (rekeying happens every 30 seconds),

the NET_{load} increases from 531 to 8499 bps as the number of nodes varies from 32 to 512. For the case that $\lambda_{T_{RK}} = \frac{1}{240}$ (rekeying happens every 4 minutes), the NET_{load} increases from 66 to 1062 bps as the key size varies from 50 to 100 digits.

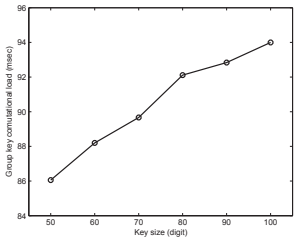


Fig. 9: Group key computational load for different key sizes

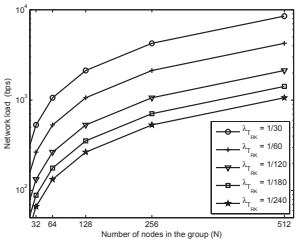


Fig. 10: Per-link network load versus number of nodes in a group

4.2 Packet key computation for security management

For the performance of our packet key computation, we performed the following analysis.

- The encryption latency, which is the latency for packet key generation and the latency for encrypting the payload using RC4. In this paper, we assume that the latency for encrypting the payload using RC4 is the same to the latency for decrypting it.
- The BVV creation latency, which is the average latency to create M-sized BVV for different packet key sizes.
- The security strength, which depends on the BVV size and key size.

4.2.1 Encryption latency

In our packet key scheme we define the encryption latency as [① + ② + ③ + ④] in Fig. 6. The time it takes for [① + ②] corresponds to the latency for packet key generation. Moreover, the time it takes for [③ + ④] corresponds to the latency for encrypting the payload using RC4. The latency for packet key generation depends on the key size while the latency for encrypting the payload using RC4 takes around 84 milliseconds 1000-byte payload regardless of the key size. Fig. 11 shows the encryption latency for different key sizes. It shows that encryption latency increases from 87 to 103 milliseconds as the key size varies from 50 to 100 digits. Considering that the real-time audio/video applications need to be time sensitive, as the key size becomes smaller, the better quality of real-time audio/video can be obtained. Then, the key size is required to be reduced as much as possible as long as it will provide strong security. It is shown that the 80-digit key causes encryption latency of 100 milliseconds. Such amount of latency can be tolerable for the secure real-time audio/video applications.

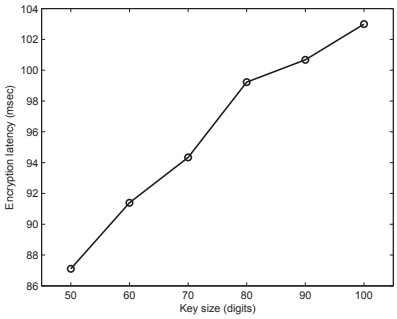


Fig. 11: Encryption latency for different key sizes

4.2.2 BVV creation latency

The proposed packet key scheme solves the real-time issue by means of making the node send an M-sized BVV. This means that the blind key exchange procedure does not occur on a packet-by-packet basis. The blind key selection is done in a round robin manner. In order to adapt to real-time packet key creation, smaller value of M and key size is desirable as long as it will provide strong security. To find the average latency needed for M-sized BVV computations, we do simulation from the mobile node's viewpoint by using android java programming using an android tablet (LG G Pad 7) with Quad-core 1.2 GHz Cortex-A7 CPU and internal memory of 1GB RAM. Each blind key computation corresponds to the following equation.

$$Y_{D,i} = \alpha^{X_{D,i}} \bmod q, \text{ for } i = 1, 2, 3, \cdots, M \qquad (6)$$

Fig. 12 shows the average latency to create M-sized BVV as the packet key size varies. For the packet key size of 80 digits, the average latency remains below 700 milliseconds as long as the BVV size M does not exceed 100.

4.2.3 Security strength depending on BVV size and key size

It is intuitively clear that the security strength is proportional to size M of the BVV as well as the key size. When the M-sized BVV is used for encryption, the exhaustive key search will require to try every value among all possible candidate keys of which the number resides anywhere between $M \times 10^{\text{`}key\ size\text{'}}$ and $10^{M \times \text{`}key\ size\text{'}}$. From brute-force attacker's viewpoint, $M \times 10^{\text{`}key\ size\text{'}}$ and $10^{M \times \text{`}key\ size\text{'}}$ are the number of exhaustive key search trials for the best case and the worst case scenarios, respectively. This paper considers the best case scenario for the brute-force attackers.

Fig. 13, shows the years it will take to break the M-sized BVV. With the use of a massive parallel microprocessor, it may be possible to achieve processing rates in many order of magnitude, thus, we assume a system that can process 10^{60} keys per second. For an 80-digit packet key size under the condition that $M = 60$, it will take around

10^{12} years to break the key, which is of equivalent security level comparing to 1024-bit RSA and 160-bit ECC (see Table 2). Based from the Table 2, our packet key scheme, that is based on DH and RC4 encryption, with 80-digit key size and $M = 60$ yields to a lesser encryption time as compared to the existing algorithms such as RSA/AES and ECC/AES. The use of smaller key means less computational load and shorter encryption latency. This kind of technology is practically applicable especially in a resource constrained environment such as in IoE, which demands for 'light weight' encryption/decryption technique as well as high level of security.

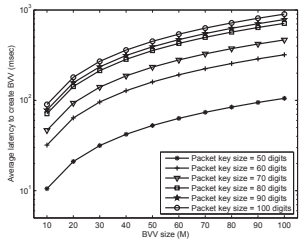


Fig. 12: Average latency to create M-sized BVV

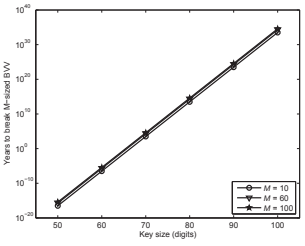


Fig. 13: Years to break M-sized BVV

Table 2: Encryption latency and security strength comparison for DH/RC4-based packet key, RSA/AES and ECC/AES

	Key size (bits)	Encryption latency (msec)	Years to break
RSA and AES Encryption	1024	$\approx$ 297 [8]	$\approx 10^{12}$[9], [10]
ECC and AES Encryption	160	$\approx$ 153 [8]	$\approx 10^{12}$ [9], [10]
DH and RC4 Encryption	266 (80 digits)	$\approx$ 100	$\approx 10^{12}$ ($M = 60$)

5. Conclusion

In this paper, we addressed the issue of group management difficulty and complexity as well as security management issues on a conventional network, which is based on a close vertical model. We proposed group management and security management for IoE based on the open horizontal integration model with SDN and NFV concept. The ONC, which is located at the control layer is capable of handling network functions such as group management and security management. Secure and private group-based IoE is achieved by using exclusive group key and packet keys. This paper solved the issue related to the exchange of the DH public (blind) key values on a real-time basis by means of making the node send an M-sized BVV. The M-sized BVV is used for real-time packet key generation for encryption/decyption. Our analysis showed that a 80-digit key causes the group key computational load of 92 milliseconds. Such amount of latency is tolerable for a resource constrained environment such as in IoE. Moreover, for group key exchange, the network load increases from 531 to 8499 bps for $\lambda_{T_{RK}} = \frac{1}{30}$ (rekeying happens every 30 seconds) and increases from 66 to 1062 bps for $\lambda_{T_{RK}} = \frac{1}{240}$ (rekeying happens every 4 minutes) as the number of nodes varies from 32 to 512. Finally, our simulations showed that our packet key scheme for security management performs better in terms encryption latency compared to the existing algorithms such as RSA/AES and ECC/AES. At equal security level, our DH/RC4 encryption only requires approxiamately 100 milliseconds encryption latency for the case that key size is 80 digits and the size of BVV is 60 (M = 60). Given the resource constrained environment such as in IoE, this kind of 'light weight' encryption/decryption technique with high level of security is desirable.

Acknowledgment

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0015200).

References

[1] P. Demestichas, A. Georgakopoulos, D. Karvounas, K. Tsagkaris, V. Stavroulaki, J. Lu, C. Xiong, and J. Yao, "5g on the horizon: Key challenges for the radio-access network," *Vehicular Technology Magazine, IEEE*, vol. 8, no. 3, pp. 47–53, Sept 2013.

[2] S. Severi, F. Sottile, G. Abreu, C. Pastrone, M. Spirito, and F. Berens, "M2m technologies: Enablers for a pervasive internet of things," in *Networks and Communications (EuCNC), 2014 European Conference on*, June 2014, pp. 1–5.

[3] D. Kreutz, F. Ramos, P. Esteves Verissimo, C. Esteve Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, Jan 2015.

[4] B. Han, V. Gopalakrishnan, L. Ji, and S. Lee, "Network function virtualization: Challenges and opportunities for innovations," *Communications Magazine, IEEE*, vol. 53, no. 2, pp. 90–97, Feb 2015.

[5] L. Battula, "Network security function virtualization(nsfv) towards cloud computing with nfv over openflow infrastructure: Challenges and novel approaches," in *Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on*, Sept 2014, pp. 1622–1628.

[6] N. M. K. Chowdhury and R. Boutaba, "A survey of network virtualization," *Computer Networks*, vol. 54, no. 5, pp. 862–876, 2010.

[7] S. Baset and H. Schulzrinne, "An analysis of the skype peer-to-peer internet telephony protocol," in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, April 2006, pp. 1–11.

[8] M. Savari, M. Montazerolzohour, and Y. E. Thiam, "Comparison of ecc and rsa algorithm in multipurpose smart card application," in *Cyber Security, Cyber Warfare and Digital Forensic (CyberSec), 2012 International Conference on*. IEEE, 2012, pp. 49–53.

[9] K. Gupta and S. Silakari, "Ecc over rsa for asymmetric encryption: A review," *IJCSI International Journal of Computer Science Issues*, vol. 8, no. 3, 2012.

[10] M. Bafandehkar, S. Md Yasin, R. Mahmod, and Z. Hanapi, "Comparison of ecc and rsa algorithm in resource constrained devices," in *IT Convergence and Security (ICITCS), 2013 International Conference on*, Dec 2013, pp. 1–3.

IoT Cloud-Sensor Secure Architecture for Smart Home

Syed Abdul Muqtader Razvi, Abdullah Al-Dhelaan, Mznah Al-Rodhaan and Riman A. Bin Sulaiman
Department of Computer Science, King Saud University, Saudi Arabia
Email: muqtader_razvi@student.ksu.edu.sa, {dhelaan, rodhaan}@ksu.edu.sa, dr.riman@gmail.com

Abstract— Automatize the handling of smart objects by providing an extra edge technology (anytime anywhere feature) by bringing those objects under the umbrella of Internet is the fundamental ideology of Internet of Things (IoT). With such perceptive, each device and any associated sensors needs to have an IP address to enable anytime anywhere feature. Moreover, for smart object's automating to be efficient, these devices must be fitted with extra hardware that handles the process of decision making. However, these smart objects are energy deficient devices and attaching an extra hardware to these devices will not be feasible solution. An efficient solution to this problem is the teaming of IoT and Cloud Computing technologies. Cloud Computing will provide virtualization feature to smart objects in IoT environment by enabling the processes of decision making to be on Cloud side thereby conserving the energy at smart object's side. The Cloud Computing models IaaS, PaaS and SaaS can provide virtualization at user side. A Cloud-Sensor secure architecture for smart home in IoT environment is proposed in this paper, automation of processes is provided by cloud infrastructure, while the slave server on client side ensures the security of sensors and devices.

Keywords— Cloud Computing, Internet of Things, Smart Home

1. INTRODUCTION

Internet of Things (IoT) is the word coined to extend the current world of Internet to all physical quantities. It encompasses of key technologies relating to identification, networks and data mining fields. Recent valuable researches and innovations in these related technologies are paving the way for IoT. Before connecting smart objects to the world of Internet, it is essential to ensure that all security properties are met. Cloud Computing can play a crucial role to ensure all security properties are met.

Cloud Computing is a technology that allows the individual to efficiently run applications and utilize the resource placed over the internet. It provides a request service model that makes the user worry-free about the implementation of applications.

One of the important aspects for teaming Cloud Computing and IoT technologies is the automation of services. The feasible solution for fully utilizing the potential cloud computing and ubiquitous sensing is to develop a framework where cloud is at center [1]. With such internet centric architecture associated sensor and actuators is the only need for devices in IoT. All the decision-making capabilities will be facilitated by the cloud utilizing the data from sensors. With cloud computing model the most effective and efficient tools from different providers like machine learning tools for converting information into knowledge, graphics for visualization etc, can be clustered into single environment to increase the productivity at the client side.

In the proposed paper, a cloud-based model is presented for smart home. In an overview, the features that are required for smart home are: (1) hiding of software's that is required for fulfilling of any particular service. (2) Supporting automatic decision making without users intervention. (3) Management of devices and associated sensors. The service provider through cloud provides all the services related to smart home management. Scheduling management of the service is done by the service provider i.e deciding what time it is appropriate to run a service. To do so service provider can use decision-making software to ease and automatize the process of scheduling.

The paper is organized as follows. Section 2 provides related data. Section 3 compares some of the available IoT tools against each other in terms of features that are essential for a tool to be efficient. Section 4 introduces cloud-based architecture, explains features of related terminologies, explains the working and describes the lifetime of service. Section 5 provides a working scenario of smart home that explains automation processes briefly. Finally, Section 6 presents the conclusion.

2. RELATED WORK

Internet of Things is ought to establish its base in the world of internet and cloud computing can be nominated as a pillar to IoT. The clouding models IaaS, PaaS and SaaS can facilitate the end users to efficiently accommodate the services. The cloud can be integrated with wireless sensors that allow easy accessing, sharing, collecting and searching of appropriate data for distinct applications. These sensors and their associated data can be provided to the clients' on-fly terming new word "Sensing as a Service (SnaaS)" [2].

Architecture for IoT based on cloud [3] was proposed. Constrained Application Protocol (CoAP), a request/response web transfer interaction model is used for interacting with the things. The requesting and identification of resources is done through URIs using REST protocol. HTTP protocol was used for sending the data in between cloud application and IoT things.

According to [4] the barriers that are coming in path of sensor-cloud integration, to larger extend is related to establishment of data channels. It later on moved on to provide software tools for connector data channel. These tools eased the designer's task in handling of different network protocols. These provided tools work in self-contained manner for each user depend on the needs. It is also responsible to handle the job migration and mobility of users by recommencing the broken links. For rapid creations of applications and their

deployment [1] provided a framework for mapping with cloud API's.

IoT devices and sensors have limited resources due to which processing of information at the data-site is not efficient. Thus to increase the efficiency of devices and sensors in IoT environment, it is appropriate to send the data to nodes or cloud where processing of information is possible. The on-demand model characteristic of Cloud will allow to increase the processing complexity of IoT environment [5][6]. Cloud will provide models that will allow development of decision making and prediction algorithm for IoT environment and these algorithms are supposed to be at low cost, efficient and also with low risk [7]. IoT involves huge amount of data that is non-structured or semi-structured [8], hence Cloud is assumed to be most effective solution to the data produced by IoT environment. Several advantages can be seen in integration of IoT with Cloud [8], where on one hand IoT can benefit from on-demand model characteristic of Cloud while on other hand Cloud will benefit from IoT by directly dealing with world of real things.

ThingsWorx [9] first software development platform designed for the needs of the connected world with an explosion of connected sensors, devices, and equipment ("Things"). It provides a complete application design, runtime, and intelligence environment and allows organizations to rapidly create M2M applications and innovative solutions that allow the value found at the intersection of people, systems, and intelligent connected Things.

Cosm [10] is a secure, scalable platform that connects devices and products with applications to provide real-time control and data storage. Using Cosm's open API, individuals and companies can create new devices, develop prototypes, and bring products to market in volume. Cosm offers a way to launch internet-enabled products without having to build any backend infrastructure. The platform runs within LogMeIn datacenters, providing world-class security and reliability.

Paraimpu [11] is a social tool with the aim to allow people to connect, use, share and compose Things, services and devices to create personalized applications in the field of the Web of Things. It not only connects with the present social networks but also provide facility to share things between friends. It is easily inter-connect and mash-up the Things and let them automatically communicate through the Web. To interact with internet the users can use sensors, motors and various object provided by Paraimpu.

The MAGIC Broker 2 (MB2) [12] platform (formerly called the OSGiBroker) is designed to offer a simple and consistent programming interface for collections of things. MB2 is based on the RESTBroker, a pub/sub middleware supporting REST protocol for connecting subscribed clients to the publishers. Like the RESTBroker, MAGIC Broker 2 provides a set of common abstractions and a RESTful web services protocol to more rapidly develop Internet of Things (IoT) applications such as interactive public large screen display applications and web-based wide area sensor networks.

The Web of Things Toolkit (WoTKit) [13] is a platform as a service that allows you to connect things to the web. The system serves as a sensor data aggregator, dashboard, remote control and data processing tool. Developers can also create their own applications by using the RESTful API supplied with the platform. A lightweight toolkit and platform (run as a service) that provides a simple way for end users to find, control, visualize and share data from a variety of things.

The SensorCloud [14] is a platform that embeds cloud technologies that features scalability, visualization and analysis of data. It provides the users with OpenData API, which facilitates the user in uploading of sensor data. A sophisticated graphing tool is provided in form of FastGraph. One of the major featuring of SensorCloud is LiveConnect, it allows the users to access every function present on network from anywhere round the earth. The users using MathEngine feature can process huge quantity of data from sensors.

3. IOT TOOLS AND ANALYSIS

We determined some of the essential features that can make an IoT tool feasible and we gauged the importance of each feature by weighing them against each other. For this purpose, we divided 100 points between those features. The column "weight (%)" describes points given to each feature. The weight percentage reflects the importance of the feature in the IoT tools. We then rated the presence of each feature in some of the currently available IoT tools on the scale of 5. Rating '0' indicates feature is not present in that tool. Rating 1 indicates efforts are in progress to incorporate the feature. Rating 2-3 indicates feature is present and efforts are in progress to upgrade. Rating 4-5 indicates that feature is present. Table 1 shows the analysis table that compares some of the currently available tools.

Criteria	Weights (%)	Tool-1	Tool-2	Tool-3	Tool-4	Tool-5
scalability	3	3	2	1	1	2
heterogenity	1	3	2	1	1	1
federation	2	3	3	2	1	3
concurrency	3	3	3	3	3	3
mobility	2	3	3	3	3	3
network management	1	3	3	3	1	3
unique identifier	3	3	3	0	0	3
source distribution	2	1	3	2	3	2
cloud service	5	3	3	1	2	3
energy optimization	3	2	2	2	1	2
decision making	4	3	3	2	2	3
event driven execution	5	3	3	3	3	3
search-based intelligence	3	3	1	2	2	3
security	5	3	3	2	1	3
energy efficient security	1	2	1	1	1	1

(encryption and hashing)						
anonymity and forward security	3	3	3	1	1	2
model based developmen t	4	3	3	2	3	3
less overhead	3	2	2	2	2	2
privacy	2	1	1	1	1	1
remote access	2	3	3	3	1	3
availability	1	3	3	3	2	3
smart object API	3	3	2	1	1	2
IPV6 support	1	2	2	1	1	2
resource discovery management	3	2	1	1	1	1
data filtering and management	3	3	2	3	1	2
language	1	1	1	1	1	1
developmen t effort	3	3	2	2	1	3
M2M support	5	3	2	1	1	3
data storage	3	3	1	1	2	2
policy based access control	3	3	3	3	0	3
API documentati on	2	3	3	1	2	3
different protocols	3	2	2	1	3	2
controling things	3	2	2	1	1	2
monitoring things	3	2	2	1	1	2
virutal objects	2	1	1	3	1	1
email module	2	2	1	1	1	3
map visualization	2	1	1	1	1	2

Table 1: Analysis of features available in current IoT Tools.

**Tool-1= Things Worx; Tool-2= Cosm; Tool-3= Paraimpu; Tool-4= Magic Broker2; Tool-5= WoTKit*

4. PROPOSED METHOD

The proposed method utilizes the design of Master and Slave, where Cloud Server (which is a public Cloud) is the master and Slave Server is the slave. Cloud Server may have large number of Slave Servers or slaves registered with it. Slave Server relies on Cloud Server for making decisions.

- **Features of Slave Server:**

Slave Server is responsible for the collection of data from the sensor displaced over an area. It will create functional groups and assign sensors to each group depending upon their functionality. It will assign unique identifier like RFID to these groups in order to distinguish them and to the devices so that it can control them as needed. For example, refrigerator has its unique ID with its groups of sensors, temperature sensor group have their unique ID, AC has its uniqueID, and so on. Slave Server timely communicates with the Cloud Server sending the data received from sensors. On receiving an event from Cloud Server, i slave server will direct particular group of sensor or device to perform the action (as directed by Cloud Server). Some time the owner will request certain action through Cloud Server (like switching ON AC while he is near to home), to ensure the authenticity the Slave Server will also maintain a RSA secureID as a security token feature. The owner will have RSA key fob with the secure token similar to the one on the Slave Server.

- **Features of Cloud Server:**

Cloud Server is a cloud model of computing; which comprises number of applications and services. Developers do develop applications that provide decisions as their output and service are those sub functions that helps in decision-making (E.g.: Temperature sensor's data as a service). An application may consist of one or more services. Cloud Server will enable the registered users to utilize any of its decision making feature only if the user is registered with all the services needed by that particular feature. On receiving the data from the Slave Server, it makes its decision depending upon the current and past situations. It will also store certain amount of data for future perspective. It will also study the individual decision for current situation and takes same actions if same scenario occurs in the future. Cloud server is also responsible for tracking of individual at the daily routine places like home, office, gym etc; to make decisions depending on current location of individual. It will also allow users to arrange meetings; a suggestion for meeting will be popped up in the scheduler of the individual that tells about the time, place and the meeting invitee. This popping for suggestion is time constraint i.e the cloud server behaves intelligently, checking the current busyness of the individual. A daily report containing energy consumption, individual's time management and some suggestion (if any) is generated for each registered individual and provided to slave server located at a particular home.

- **Registration of Users:**

The Cloud Server providers will prepare a catalogue that tells about the services particular server provides. This registration can be via user interface like browser. The individual will select the needed services. The user can rent out the temperature sensors and light sensors from cloud server's organization so that they can utilize their data effectively when they like. On registration of the services, the individual will be provided with unique identifier (for managing account), tags with some identifier (just to distinguish different places), user interface software at mobile, a reader linked with mobile device and a RSA secureID key fob. It is on to the individual where he wants to place the provided tags suggested in home, office, gym and car. At the same time, a slave server is set at the site that monitors the sensors available on site and those which can be effective for fulfilling the registered services requirement. A user interface is provided at mobile device; this will help the individual in management of things. The user interface will synchronies the cloud scheduler and reminder with mobile scheduler and reminder i.e. the individual does not need to manage the scheduler and reminder section of cloud personally. At the time of registering of service 's'_file is created at cloud server that manages the sensors associated with particular service.

- **Architecture and its Functionality:**

The most important characteristic of cloud based Internet of Things is the aptness of architecture that allows online deployment, management and running of applications and services. Cloud server is the server that manages different applications and service like energy management of home. Thus, it is SaaS model of cloud computing.

The cloud server will follow some specific series, for allocating identifiers to each different sensor with different functionality. The slave server will group the different sensors and controllers depending upon their capability. A unique identifier is assigned to each group by the cloud server following any particular series at registration time depending upon the functionality of the sensors. The slave server will maintain a file for each group of sensors and controllers that contains their data. This file is known as 'se'_data, where 'se' is identifier of particular group. This 'se'_file of each group is also maintained at cloud server with the same name and these files are synchronized from time to time. The cloud server will have a dedicated file that maintains the record of sensors associated with each service. This file is known as 's'_sensors, where 's' is the name of service eg. energymanagement_sensors.

The cloud server will have a special mapping file termed as "map_file" that is responsible for creating a session with slave server. Mapping file "map_file" will contain information related to slave server's address. The cloud server will also have another file termed as "mobile_file"; it will contain the information related to current IP address of the mobile device of the individual. Cloud server will timely create a session using these file. The Cloud server and Slave server will maintain RSA token generator software for authentication purpose. Figure 1 visualizes the proposed architecture of cloud-based model for smart homes and shows the location of different files.

To initiate any particular service, the cloud server will check 's'_sensors of that service to determine the associated sensors. Then the cloud server will request 'se'_data of each associated sensors from slave server. Whenever the cloud server is to receive a file from slave server it will first create SSH tunnel with the slave server. Using the information from "map_file" Cloud server will first send the request to the slave server to send 'se'_data file. Along with the request, it will also send the RSA token at current point of time. If slave server is satisfied with the received token it will acknowledge the cloud server to create a SSH tunnel. The requested 'se'_data files are exported only after the creation of SSH tunnel. The cloud server will then update each received 'se'_data file at its side and will take decision depending upon the computation of data. These decisions are generally updating of 'se'_data files associated with controller's groups. If this decision is to be conveyed to the slave server for taking any action at site, it will transfer the updated files to the slave server. For this transferring, Cloud server will again send a message containing RSA token at current time, then SSH tunnel is created if authenticated and then files are transmitted. On receiving files, the slave server will overwrite the controller group's file and take appropriate action.

If a slave server is to initiate a communication, it will send a message to cloud server to create a SSH tunnel. The message will contain information related to user's ID and RSA token at current time. The cloud server will then check for RSA token for specified registered user; if authentic, the cloud server will then create SSH tunnel and then the slave server will send the required files.

The individual also can handle the devices at home remotely using the user interface present at the mobile device of the individual. The user interface will contain the information such as cloud server address and user identifier. The cloud server and the user interface will maintain special software to generate RSA tokens. The user interface will provide a menu of devices that the individual can control. Later on depending upon the selected device from the menu, related controls are

provided. The user interface then sends the message to cloud server that tells the cloud server about the user identifier, RSA token at current, and what he want to do on particular device. On receiving the message the cloud server, will checks for the registered user with the received identifier in the message. The cloud server will then authenticate the user by checking the RSA token, it then again asks for password since the user is asking for performance of action on controllers. The user will then enter the password. If the user turns to be authentic, the cloud server will then update the 'se'_data file of associated sensors and controllers and later on sends the control information to the slave server. The cloud server will also send any alerts like not working of any sensors at the user interface. The user interface timely checks for IP address of the device, if there is any change it will communicate it to cloud server. The cloud server then updates "mobile_file" file.

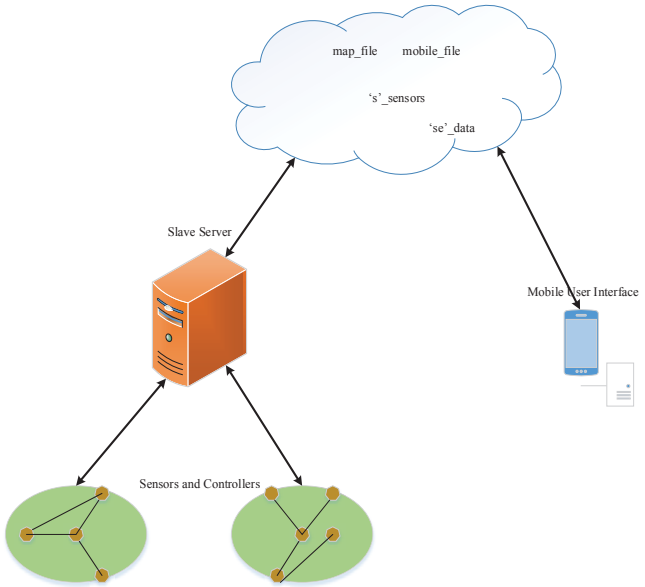


Figure 1: Architecture of Cloud Based IoT model for Smart Home

The cloud server also has to synchronize with the scheduler that is present at the mobile of the individual. This synchronization occurs through user interface. If any of other registered user wants to fix a meeting with the individual it will send the request from its user interface specifying the time and date. The cloud server will then checks the scheduler for the desired individual, if the scheduler already has appoint at specified time it will check the priority of both meetings. If it finds that incoming meeting, suggestion is of low priority then it will reject the suggestion for meeting. If there is no meeting or there is already a meeting which of low priority than the incoming suggestion, the cloud server will ask the individual to acknowledge whether he want to go on with the suggested meeting. These suggestions for the meeting are popped at the user interface of the individual mobile depending upon the time and the busyness of the individual. When these suggestion pops at user interface, it will check the authenticity of the received message by checking the RSA token present in the message. The message is displayed to the user only after checking the authenticity.

- **Lifetime of the service:**

The cloud providers will prepare a catalogue that tells about the offered service. The individual has to register with that service in order to use it. The life cycle of the service will be divided into two phases [15]. One phase is responsible for creating a service and another is responsible for providing it. Figure 2 depicts the lifecycle of services.

Service Creation: The service provider will create software or a set of interlinking software that will fulfill the requirements of the particular service. Each service will require a group of sensors in order to perform the desired task and it is responsibility of the requestor to fulfill it. The requestor can waive away from requirement of some sensors like temperature, light and sound by asking the service provider to arrange it. The service provider will then look for neighborhood of requestor that is registered with it and is ready to provide to rent out the service-to-service provider. If service provider does not find such neighborhood, then there is no option and requestor has to provide that required sensors. Each service will have a dedicated files associated with it and it will be stored in the database for each registered user. These files will contain the information related to the data of associated sensors. Depending upon the priority service invocation is done from time to time.

Service providing phase: The end user will register for the service by selecting it from the catalogue. The registered service will be stored in the account of the individual along with its dedicated files. Timely depending upon the functionality the dedicated files of the service are updated by querying the slave server or user interface. These files can also be updated when there is event trigger from user interface or slave interface. When there is any updating, the data is parameterized to the software linked with the particular service. The outputs of these software's are usually updating of files associated with controllers. These file updating is then communicated to the slave server who then perform or direct for performing of action.

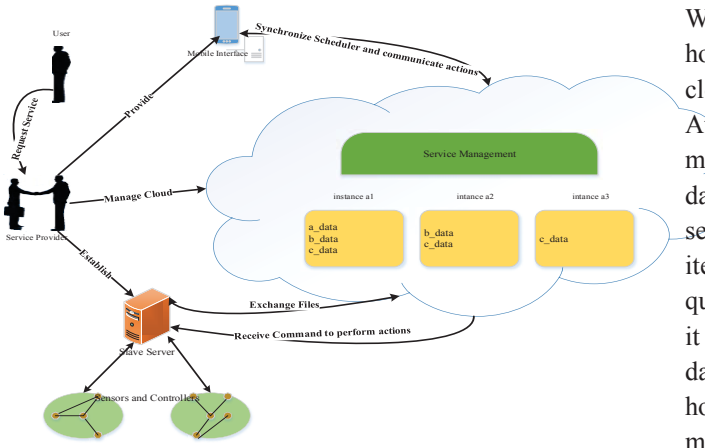


Figure 2: Lifecycle of Services

- **Security Analysis:**

  It is essential that smart home with cloud-based model also ensures the security properties. The creation of SSH tunnel for communication ensures the security properties like integrity and confidentiality (in terms of exchange between cloud server and slave server). SSH tunnel also ensures that there is no eavesdropping case in communication. Authentication is established by the presence of RSA token in the communication. The owner can control the device remotely from the user interface; a cross examination is done by asking password to ensure that it is authorized user who wants to access any particular device at a given time. The trust for service provider is essential in cloud-based environment of IoT as all data rest at server for decision-making.

5. SCENARIO OF SMART HOME

The devices that are present in smart home are sensors (indoor and outdoor), electronic equipment like microwave oven, refrigerator, television, computational devices like PC, mobile phones and controllers (for doors windows).

An early morning individual wakes up on running of background music at appropriate time determined by the cloud server. At the same time curtains of bedroom windows open up slowly letting sun light to enter. When the individual rise and starts using taps for washing face the bedroom music stops automatically and there starts morning news in bathroom. When the individual is preparing breakfast the TV in the kitchen opens up automatically displaying the schedule, reminders and notes for the present day. Timely sound alerts are set if there is any delay in task performing depending upon the present day schedule. When the individual is on the verge to leave the house, a weather forecast for the day is dictated. When the individual leave for work and if there is no one at home, the AC is turned off and the doors and windows are closed locking system is then enabled.

At any certain time the home system goes in surveillance mode depending upon the busyness of the system, looking for daily routine eating products and clothes for washing. The sensors in refrigerator provides with the quantity of available items. If the cloud determines that there is a scarcity of quantity in any particular product, it will then decide to order it or not depending upon the schedule of present day and next day. If the cloud determines that there is some special event at home like party with friends, depending upon the planned menu for party, the cloud check for items available and their quantity. It will then proceed with ordering for items that are unavailable or less in quantity. The individual will place the cloth into washing machine as soon as they get dirty, then the sensors in washing machine provide Cloud server with the data that details about the quantity of clothes present. Cloud server will direct the washing of clothes after considering various factors like energy consumption, time etc.

At evening the temperature of outside environment cools and by using the data from temperature sensors and light sensors the cloud server will direct the slave for the watering of plants and garden. At any pleasant evening if the individual is at his home, cloud will check the data from temperature sensors, light sensors and sound sensor, if all conditions are favorable for the individual it will open the windows letting cool air to enter.

If the individual is travelling (which is known by the identifier received of the tag placed in car), the cloud will check the schedule of the individual if there is any meeting or appointment at any particular place the cloud will direct the individual by updating the status of traffic in his path. If the individual is coming back to home from office or anywhere else, the cloud will calculate the estimated time of arrival. It will also check the outside temperature through temperature sensors and will switch ON the AC depending upon the arrival of the individual.

At evening the individual arrives from the work; as he enters the compound of the house, the tag situated there is read by the reader of mobile and communicated to the cloud. The cloud server then directs the slave server, ascertaining the presence of the individual. The slave then initiates appropriate sensors as directed. As the individual enters, the living room lights open up slowly and he finds the room temperature that is comfortable for him. Slow music starts in background for some time just to relieve stress. At the same time, the cloud server will generate a report that states the energy consumption for last 24 hours. The report will also contain

some suggestion for diet and exercise (by utilizing the data received from wearable sensors).

At night when the individual is preparing dinner music starts automatically in background. When the individual sits to eat dinner the TV in living room starts automatically playing his favorite program from where he ended the last time. When the individual moves to the bedroom to sleep, he will find favorable environment with dim lights. The lights in living room and TV are turned OFF. After specified amount of time and sensing no motion in home, all the devices moves into sleep mode.

6. CONCLUSION

A cloud-based model for smart home automation is proposed. Architectural designing was done with a vision of meeting security properties. The concept of slave server was introduced to ensure that sensors and actuators are not directly connected public network like internet. The main vision was to encourage cloud-based model for IoT. The smart home scenario provides a vision to automation of IoT processes in cloud based environment.

REFERENCES

[1] Gubbi J, Buyya R, Marusic S, Palaniswami M. Internet of Things IoT): A vision, architectural elements, and future directions. Future Generation Computer Systems. 2013; 29(7): 1645-1660. doi: 10.1016/j.future.2013.01.010.

[2] Rao BBP, Saluja P, Sharma N, Mittal N, Sharma SV. Cloud Computing for Internet of Things & Sensing Based Applications. 2012 Sixth International Conference on Sensing Technology. 2012; 374-380. doi: 10.1109/ICSensT.2012.6461705

[3] Zhou J, Lappanen T, Harjula E, Yu C, Jin H, Yang LT. CloudThings: a Common Architecture for Integrating the Internet of Things with Cloud Computing. 2013 IEEE 17th International Conference on Computer Supported Cooperative Work in Design. 2013; 651-657. doi: 10.1109/CSCWD.6581037

[4] Melchor J, Fukuda M. A Design of Flexible Data Channels for Sensor-Cloud Integration. 2011 21st International Conference on Systems Engineering. 2011; 251-256. doi: 10.1109/ICSEng.2011.52

[5] Dash SK, Mohaapatra S, Pattnaik PK. A Survey on Application of Wireless Sensor Network Using Cloud Computing. International Journal of Computer Science and Engineering Technologies. 2010; 1(4):50-55.

[6] Parwekar P. From Internet of Things towards Cloud of Things. 2nd International Conference on Computer and Communication Technology. 2011, 329-333. doi: 10.1109/ICCCT.2011.6075156.

[7] Zaslavsky A, Perera C, Georgakopoulos D. Sensing as a Service and Big Data. arXiv preprint arXiv:1301.0159, 2013.

[8] Botta A, Donato W, Persico V, Pescape A. On the Integration of Cloud Computing and Internet of Things. Internation Conference on Future Internet of Things and Cloud. 2014; 23-30. doi: 10.1109/FiCloud.2014.14

[9] ThingsWorx, ThingWorx platform overview. "http://www.thingworx.com/platform/". (accessed January 2, 2014)

[10] Xiverly, Cosm or Xively platform. "https://xively.com/whats_xively/". (accessed January 2, 2014)

[11] Paraimpu, Paraimpu..what?. "http://paraimpu.crs4.it/about". (accessed January 2, 2014)

[12] Magic Broker 2. "http://www.magic.ubc.ca/pmwiki.php?n=Projects.MAGICBroker2". (accessed January 2, 2014)

[13] Sensetecnic, Sensetecnic about. "http://sensetecnic.com/?page_id=2". (accessed January 2, 2014)

[14] Sensorcloud, "http://www.sensorcloud.com/". (accessed January 2, 2014)

[15] Yuriyama M, Kushida T, Itakura M. A New Model of Accelerating Service Innovation with Sensor –Cloud Infrastructure. 2011 Annual SRII Global Conference. 2011; 308-314. doi: 10.1109/SRII.2011.42

Key Management for Secure Multicast Communication in Sensor Cloud

Shoroq Odah Al Beladi, Firdous Kausar
Department of Computer Science, Al Imam Mohammad Ibn Saud Islamic University, Riyadh, Saudi Arabia
shorogodah@gmail.com, firdous.kausar@ccis.imamu.edu.sa

Abstract - *Recently, data size has been largely increased and there is no enough storage and computation resources to handle these data. Sensor-Cloud scheme represents the best solution to solve this problem. In this paper, secure-multicast group key management protocol has been has been proposed in order to provide secure group communication within a dynamic Sensor-Cloud. The analysis of proposed protocol shows that it provides the properties of forward secrecy, backward secrecy, self-healing and periodic rekeying. We implement the proposed protocol on Tosssim simulator by using programming language nesC and TinyOS operating system. Simulation results are measured in term of throughput and packet loss and it has been found that it has high throughput and low packet loss.*

Keywords: Sensor-Cloud, Secure Multicast, Group Key Management Protocol, Wireless Sensor Networks, TinyOS

1 Introduction

In the few last years, the "Wireless Sensor Network (WSN)" earns high attention from a large number of people, because it provides many effective solutions for many fields, like; monitoring of air pollution, forecasting of weather, traffic monitoring for citywide roads, and e-healthcare. On the other hand, expanding WSNs to large networks can lead to many problems and limitations [1].

Many designs of the vendor show that there is no ability to connect various sensor networks with others and it is impossible to sharing the data of sensor between various user groups. In addition, there are no enough resources of storage and computation to overcome the applications of large-size. The sensor network with large-size is a very important issue, so the Sensor-Cloud is used to handle this issue. This model can be described by combining the cloud with WSNs and it is a very efficient solution for this type of networks. This combination provides the data processing with high swift through the available Cloud structure with larger routing and through the massive processing power to support the user with fast response. The management of secure multicasting is a very important issue, so it is discussed here. The figure 1 below shows the structure of Sensor-Cloud [1].

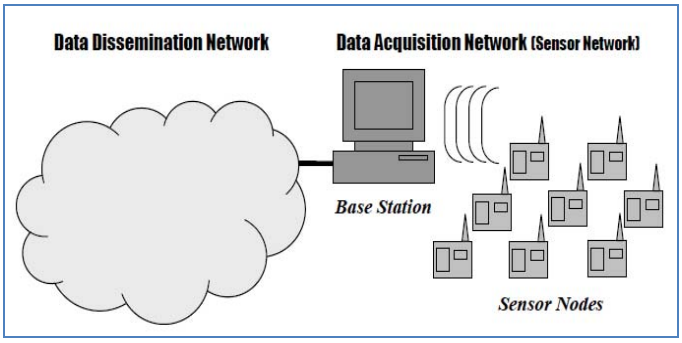


Figure 1: Sensor-Cloud Structure [1]

The security of sensor network can be improved by many services of security like; key management. The protocol of key management for WSNs must be light and simple because the communication bandwidth, processing power, memory space and battery life of the distributed nodes are limited. The scheme of random-key pre-distribution is very suitable for widely WSNs, where this scheme enables the communication between neighbor nodes only. In this scheme, the nodes selected a specified number of used key randomly before being scattered. Secure links are constructed between these nodes and their neighbors that possessing shared keys after the deployment. Furthermore, a path-key is constructed between every two neighbors that have not shared keys [2].

The techniques of encryption are the best solution to achieve secure group communication. All keys should be managed in a secure manner to distribute, update and create these keys in an efficient way in order to provide a secure communication for the group. Furthermore, the protocols of key establishment should be used to allocate group key among the group of entities in an effective and secured way. There are two main kinds of these protocols which are; "Key Agreement Protocol (KAP)" and "Key Transfer Protocol (KTP)". KTPs based on "Key Generation Center (KGC)" in order to provide the contacting information with appropriate group key. However, in KAPs, group key can be identified via interchanges the public keys that refer into two parties of communication and this can be achieved by communication bodies' presence [3].

The schemes of "Self-Healing Key Distribution" aim to propagate useful information into the trusted users. By integrating the propagated information with the pre-distributed secrets, the trusted users became able to rebuild shared keys. However, useful information cannot be reached

by the revoked users. Self-healing enables users to recover any lost key and this only requires the user to be a member in the key group. So, the off-line group member can immediately recover the lost keys of the session after the return to the on-line state [4,8].

In this work, a secure and efficient multicast key management protocol has been suggested to provide secure group communication in dynamic Sensor-Cloud with self-healing and periodic rekeying properties. Furthermore, the "Backward Secrecy (BS)" and the "Forward Secrecy (FS)" have been used in this design to protect the keys of the group. A secure-multicast has been used to securely transmit the data between WSNs and the Cloud. TinyOS has been used to construct and simulate the suggested scheme architecture.

This paper consists of other four sections: many related studies that related to this work have been discussed in section 2. The proposed scheme has been illustrated in section 3, where the results obtained have been discussed in section 4. A summary of the entire work has been provided in section 5.

2 Related Work

Please A. Herrera et.al [5] proposed a "Key distribution Protocol (KDP)" using prevailing primitives of security to confirm interoperability and security for WSN. This suggested protocol was able to support the distributed nodes within WSN with a key of symmetric cryptography. The "Trusted Platform Module (TPM)" has been applied rather than ECC or RSA primitives of encryption. The obtained results showed that this suggested protocol was able to provide a strong level of interoperability and security. Also, it was able to preserve the efficiency of energy and the scaling ability.

D. M. Mani [6] presented the "wireless Sensor Networks (WSNs)" secure multicasting protocol. Many nodes of sensors and base stations were included in "Wireless Sensor Networks (WSNs)", where the external events provide simulated to those nodes. One of the most important services of security in WSN is the transmission of the message, which is susceptible to many attacks kinds. In the proposed research, there were many schemes that proposed to get services to WSNs, like multilevel μ TESLA and μ TESLA. But, the problem of message authentication delay led to suffer these schemes from many attacks of "Denial of Service (DoS)". In wireless devices, the "Elliptic Curve Cryptography (ECC)" is largely deployed because its features over RSA, also ECC is used in the devices that their battery life, memory life, and computing power are bounded. In addition, there is another scheme is largely used in authentication of multicast, which is "Public Key Cryptography (PKC)". But using PKC in a dense way for authentication process is very expensive to supply restricted sensor nodes.

The most important parameter in WSN is the lifetime of sensor nodes. The scheme of "Low Energy Adaptive Clustering Hierarchy (LEACH)" has motivated several researchers that focus on the extension of node lifetime. A short survey has been proposed in [7] to propose various strategies to select the cluster-head and to compare the cost that demanded to select the cluster-head based on transmission method, cluster creation, rounds, information of cluster and cluster-heads distribution.

K. Ramesh et.al [7] compared many schemes such as; deterministic schemes, schemes of cluster-head selecting, using a hybrid type of clustering, probabilistic schemes of constant parameters and probabilistic schemes of adaptive resources. The obtained results show that the distance between middle cluster-head and sending cluster-head in multi-hope sending of data should be equal through various rounds of data collecting. The purpose of this set was to make the amount of consumed energy equal among the transmitted data from or to the base-station. Furthermore, the results confirm that these schemes were not the best, so more stable, energy efficient and scalable schemes of clustering should be found.

A distribution scheme of self-healing distribution has been evolved in [8] to achieve the secure multicast communication of groups in the environment of WSN. Many techniques to leave and combine the groups and a strategy to scatter the rekeying messages securely have been also applied. According to this suggested scheme, the nodes have been separated into many collections, where the "Group Controller (GC)" has been used to manage all these collections. But, the dynamicity of these collections leads to perform the regrouping process after a certain duration of time. The obtained results confirmed that the proposed scheme met the requirements of security for backward and forward secrecy. Furthermore, applying the "Message Digest 5 (MD5)" and "Secure Hash Algorithm-1 (SHA-1)" algorithms; which are one-way segmentation algorithms, confirms that the suggested scheme feasibility is suitable for the present technology of WSN. Also, the results of this scheme are attractive and scalable for the huge and dynamic collections.

A novel scheme of randomized-key pre-distribution has been implemented in [9] through "TinyOS mot Simulator (TOSSIM)" in order to provide WSN with high secure simulation. The TinyViz, has been used to transfer messages among nods and to visualize this proposed scheme. Through this scheme, the nodes have been randomly scattered over a specific area. The pseudo generator of random number was used to construct a pool of key with their IDs. During the discovery shared of shared key, the IDs of key have not broadcasting into all nodes but these IDs only sent to the desired nodes for communication in order to improve the efficiency of communication and to save resources loads. In this study, this proposed model has been analyzed depending on Cryptography and "Erdos-Renyi (ER)" model. The obtained results show that the model of Cryptography was more appropriate than an ER model for safe WSNs.

3 Proposed Scheme

This section describes the scheme architecture and the main protocols that applied within this scheme.

3.1 Sensor-Cloud Architecture

The architecture of Sensor-Cloud consists of two main sides; WSN Side and Sensor-Cloud Side. This architecture links the WSNs with three distinct network elements. These are defined below:

1) **Sensor Nodes (SN)** –these are the leaf nodes (end-nodes) in the sensor-network which do the actual sampling. Each sensor-node has a unique sensor node id.

2) **Group-Coordinators (GC)** –the large amount of leaf nodes are harder to manage, therefore, they are grouped. These groups consists of the Group-Coordinator which manages various administrative task such as renewing session keys/ revoking node permission, etc. They also act as the data aggregation point for the all the nodes assigned to its particular group.

3) **Cloud-Gateway (CG)** –all the Group-Coordinators are connected to the Cloud-Gateway via the Internet. Also the Cloud-Gateway handles all the data-aggregation from each of the sensor-node groups connected to it. This architecture is shown in the Figure 2.

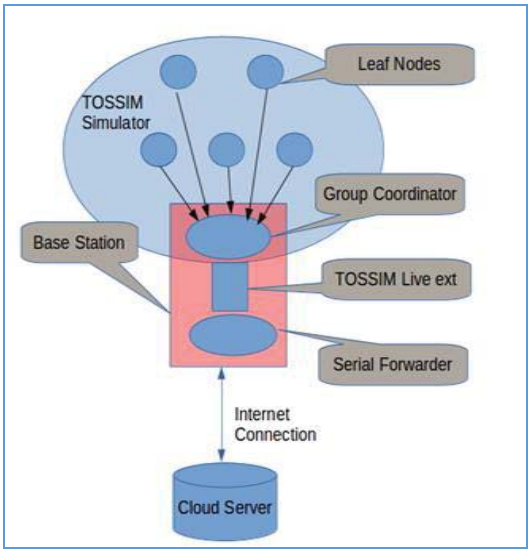


Figure 2: Sensor-Cloud Architecture

3.2 Protocol Features

The protocol is optimized for the following features:

1. Optimized for Data Aggregation: We take advantage of the fact that many of these sensor-networks operates in particular manner, where their main operation is to aggregate the sensor data into the back-end infrastructure. The leaf-nodes rarely have to communicate among themselves. Therefore the protocol-suit can optimized for that particular operation.

2. Low Cost Session re-Keying: The symmetric keys used for communication sessions between the leaf-nodes and the group-coordinator need to be updated periodically to avoid security vulnerabilities. Given the low-cost and simple hardware capabilities of the leaf-nodes, it is not possible to store large number of session keys at the node installation as that would costly in-terms of storage. Therefore we propose a mechanism that allows the session keys to be derived from a simple sequence of random seeds installed at the node set-up stage.

3. Key revoking: When a node is suspected of being compromised the node permission required to be revoked as the data it provides will potentially corrupt the aggregated data from the entire set of nodes. This is often achieved through the revoking the session-key for that particular node. But often in multi-cast group communication protocols this operation requires the participation of all the nodes in the group and is costly. In the protocol proposed below, the revoking permission for a particular leaf-node only require simple operation at the Group-Coordinator.

3.3 Protocol Steps

The protocol execution is divided into two stages

1) The initiation and set-up stage

2) Operational stage

These two stages are described in the following section:

3.3.1 Initiation and Setup Stage

1. Node Ids assignment: Each of the Leaf-Nodes are assigned a unique id, also the group-coordinators are assigned a unique Id.

2. The Topology: we assume for simplicity that the leaf-nodes arranged in a star-topology, where each leaf-node can reach the Group-Coordinator in one hop.

3. Administrative Key Assignment: Each leaf-node is installed with a random and unique administrative-key. The corresponding group-coordinator is also installed with the set of administrative keys for all the leaf-nodes within its group.

4. Assignment of random sequence of seeds: Each leaf-node is installed with a random sequence of seeds (random numbers). The corresponding group-coordinator is installed with all the seed sequences belong to all the node-ids within its group. This is depicted in the table below.

5. The second stage of the data-transfer: the data that is sent from the leaf-nodes to the Group-Coordinators need to be sent to the back-end Cloud Infrastructure for final storage and analysis. This is node using the asymmetric cryptographic

methods. To this end, the all the GC nodes are installed with a public key of the Cloud Gateway.

Leaf Node	Sequence of Seeds
Node id 1	S_1, S_2, S_3 …. S_N
...	...
Node id N	S_1, S_2, S_3 …. S_N

3.3.2 Operational Stage

1. Join Message: Leaf Node wishing to initiate a session with GC will send join message to the GC

SN → GC : {node_id}, {join:message_type, seed_id} node_Admin_key

This message request the GC to initiate the session with the Leaf-Node. The message contain the node_id of the leaf node, the message type and the random seed id. The random seed id is randomly picked from the sequence of seeds contained within the leaf-node. The latter two elements are encrypted with the admin_key installed at the initiation.

2. When GC receives the join message from the Leaf-Node, it look up the admin key for the node id in the admin key table and decrypts the message. Then it retrieves the seed_id and looks up the seed_id in the seed sequence table using: lookup (node id, seed id) => seed. Then it uses the seed to produce the session key as described below.

3. The session key is produced using the admin key material and the random seed as follows: MD5_Hash (admin_key XOR seed) =>Session key material

4. Now since both the Leaf-Node and GC is able to create the session key, a session can now be initiated. This is done by GC sending a session acknowledgement message to notify the leaf-node that a session was successfully initiated.
GC → SN: {node_id}, {session_ack:message_type } session-key

5. Data Transfer Mode: this is the standard mode of operation for the system, where the sensor data are transferred for the cloud infrastructure.

SN → GC: node_id {data: message_type, data_byte1, data_byte_2, …, data_byte_N} session-key

When the Group-Coordinator receive the data, it looks-up the current session-key table where all the keys for active sessions are stored using the node_id. Then uses that key to decrypts the message. See below how the next stage of the protocol where the data is sent to the cloud gateway is handled.

6. Transfer to Cloud: The Group-Coordinator then encrypts all the messages it receives from the Leaf- Nodes using the public key of the Cloud Gateway (CG) as flows:

GC → CG : {node id, Group_Coordinator id, data } Cloud_Gateway_public_key

If there a need for communicating between leaf-node with a group or nodes between groups. This can be achieved by addition another two message types (intra-group, inter-group). These messages need to send via the GC. If the message is of type “Intra-Group” the message will be routed to the node within the group. Else, if the message type is “Inter-Group”, then the message need to contain the Group id in addition to the node id, which can be sent to the appropriate GC via the Cloud-Gateway.

4 Performance Analysis

4.1 Scheme Framework

The suggested system has been constructed by the TOSSIM Simulator, where it aims to provide the applications of TinyOS with a simulation with high validity. Due to this, this simulator concentrates on simulating the execution of the proposed protocol on TinyOS. By this simulator, the users can analyze, test and debug the algorithms within the repeatable and controlled environment. Also, this simulator operates on the PC, so the TinyOS code can be examined by the tools of developments and the debuggers. The architecture of TinyOS allows the rapid implementation and innovation while reducing the size of code as demanded by the memory of server limits the network of inherent sensor. In addition, TinyOS has three main features that impact on the design of nesC, these features are; operations of split-phase, the simple concurrency model that based on the events and component-based construction.

In this work, WSNs that consist of sensor nodes have been constructed. The wireless BSs provide the communication channels between the WSNs and the Cloud. The required code to define the WSNs and the BS has been written by nesC, where the code of applying the protocol of key management, symmetric key, Backward Secrecy and Forward Secrecy have been included within this code. The flowchart of the designed system is shown in Figure 3.

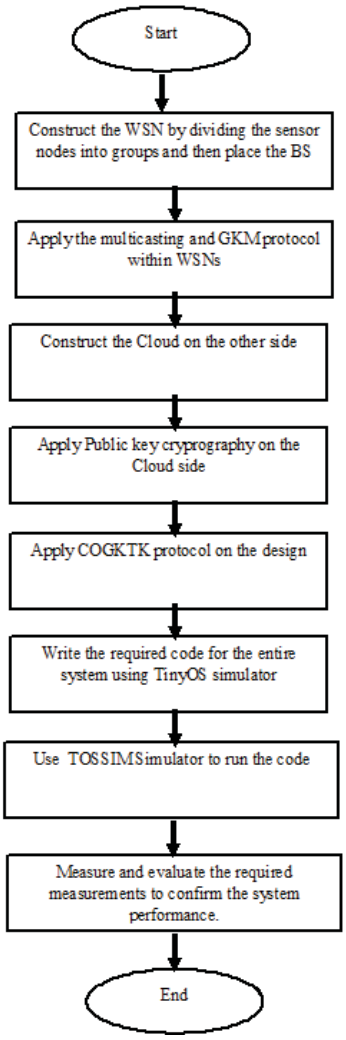


Figure 3: Scheme Flowchart.

4.2 Measurements

By this work, many parameters can be measured to evaluate the performance of the suggested scheme. These measurements are;

4.2.1 The throughput

This concept can be described by the amount of the data transmitted from the source node to the node of the destination. Furthermore, it can be defined as the amount of data processed within a certain time amount. Also, it defines as the average bits' number that successfully transmitted per second. Within this work, the throughput has been measured for the data transmitted from the WSNs to the Cloud. This term ensures the reliability of this proposed system.

4.2.2 Packet Loss

By this term, the number of packet losses can be measured. The heavy traffic, delay, collisions and buffer overflows, are the key reasons for this loss. Within the WSNs, packets are lost due to the packet dropping or malicious non-forwarding that caused by the compromised nodes or the adversary. Within this work, the number of pack lost has been measured for the total data that transmitted from WSNs to the Cloud.

5 Obtained Results

The results of the simulation and simulation scenario have been shown and discussed in this section.

5.1 Simulation Setup

The initial simulation set up consists of two groups each consisting of 10 nodes, node id 1 connected to nodes 2-10, and node id 11 connected to nodes 12-20. The network topology for each group is a star- topology where each of the leaf-nodes are connected to the group-coordinator with a single hop.

In this setup nodes 1 and 11 are the group-coordinators. The leaf nodes will be sending packets of data periodically to the group-coordinators (at 5Hz frequency). Since we do not have actual sensor data values to send in the packets, we use a counter that is incremented at 5Hz (that is every 200 miliseconds). This counter value is used as the data that is sent over the network. This setup simulate a typical data-aggregation sensor network.

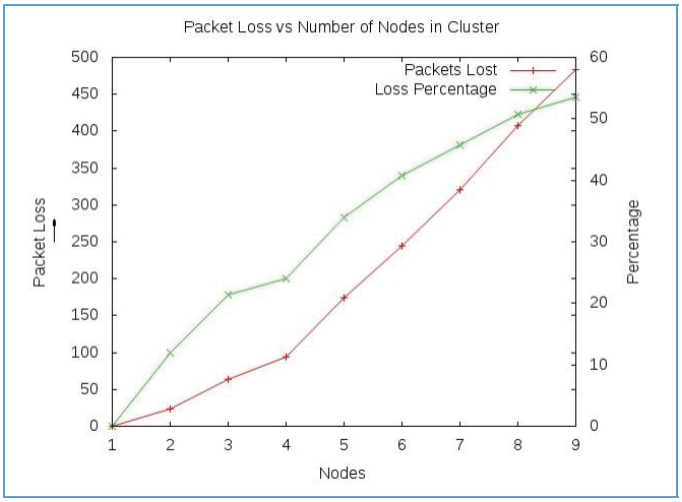


Figure 4: Packet loss vs number of nodes.

As shown in Figure 4 above the value of packet loss is increased with the number of nodes that existed within the cluster. This figure represent the behavior of packet loss in general. However within this design, applying GKMP helps to minimize the number of lost packet through the self-healing property.

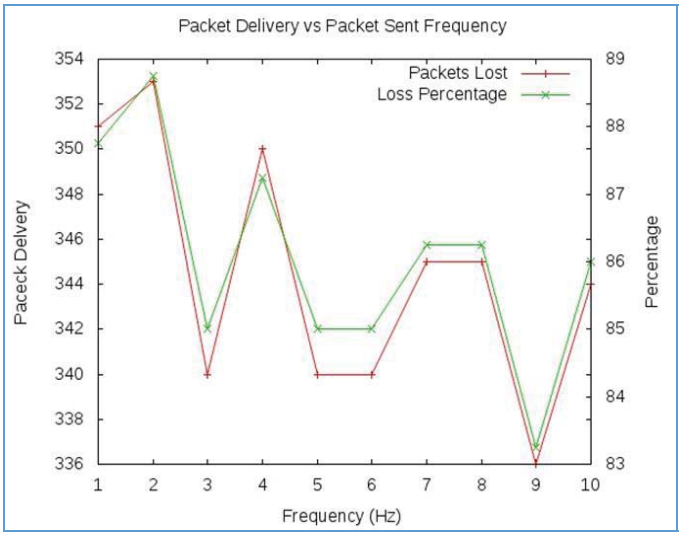


Figure 5: Packet delivery vs packet transmission frequency.

The Figure 5 illustrates the number of packet delivery with respect to the packet sent frequency. The data show the packet delivery reaches the equilibrium at around 85% between the 1-10 Hz packet injection rates from each individual node. The effective data throughput as opposed to the overall data-throughput depends on the efficient and effective design of the aggregation protocol.

6 CONCLUSION

This paper presents a secure multicast group key management protocol for Sensor-Cloud in order to solve the problem of secure transfer of large size of data between WSN and cloud network. The suggested system architecture consists of two main sides; WSNs side and Cloud's side. The secure-multicast method has been used as a way to transmit the data. Furthermore, proposed secure multicast group key management protocol have been analyzed and provide the features of secure transmission of data, prevent data loss and to enhance the number of updated keys. We measure the throughput and packet loss to assess the performance level of the proposed protocol and determine, it achieve the desired results and performance. The obtained results confirm the stability of this protocol, where it has high throughput and low packet loss.

7 References

[1] T. Nguyen and E. Huh, "An efficient key management for secure multicast in Sensor-Cloud", ACIS/JNU International Conference on Computers, Networks, Systems, and Industrial Engineering, pp 1-7, 2011.

[2] P.J. Chuang, T. H. Chao and B.Y. Li, "Scalable Grouping Random Key Predistribution in Large Scale Wireless Sensor Networks", Tamkang Journal of Science and Engineering, 12(2), pp. 151-160, 2009.

[3] R. V. Rao, K. Selvamani and R. Elakkiya, "A Secure Key Transfer Protocol for Group Communication", Advanced Computing: An International Journal (ACIJ), 3 (6), pp. 83-90, 2012.

[4] B. Tian, S. Han, D.S. Tharam, S. Das, " A self-healing key distribution scheme based on vector space secret sharing and one way hash chains", IEEE international symposium on a world of wireless, mobile and multimedia networks, WoWMoM 2008 , pp.1,6, 23-26 June 2008.

[5] A. Herrera and W. Hu, "A Key Distribution Protocol for Wireless Sensor Networks", 37th IEEE Conference on Local Computer Networks (LCN), pp.140,143, 22-25 Oct. 2012.

[6] D. M. Mani, "Secure Multicasting for Wireless Sensor Networks", IJREAT International Journal of Research in Engineering & Advanced Technology, 1(5), pp 1-8, 2013.

[7] K. Ramesh and K. Somasundaram, "A Comparative Study of Cluster-head Selection Algorithms in Wireless Sensor Networks", International Journal of Computer Science & Engineering Survey (IJCSES), 2(4), pp. 153-164, 2011.

[8] F. Kausar, S. Hussain, J. H.Park and A. Masood, "Secure Group Communication with Self-healing and Rekeying in Wireless Sensor Networks", Third International Conference on Mobile Ad-Hoc and Sensor Networks, MSN 2007, Lecture Notes in Computer Science, pp. 737-748, Beijing, China, December 12-14, 2007.

[9] S. Verma and Prachi, "Analysis of a New Random Key Pre-distribution Scheme for WSN Based on Random Graph Theory and Cryptography", Journal of Information and Data Management, 1(1), pp. 14-18, 2012.

[10] M. H. Ullah, J. No, G. H. Kim and S.Park, "A Collaboration Mechanism Between Wireless Sensor Network and a Cloud through a Pub/Sub-based Middleware Service", The Fifth International Conference on Evolving Internet, pp.38-42, 2013.

[11] K. Sun, P.Peng, P. Ning and C. Wang, "Secure Distributed Cluster Formation in Wireless Sensor Networks", Computer Security Applications Conference, ACSAC '06. pp. 131-140, Dec 2006.

[12] O. Gaddour, A. Koubˆaa and M. Abid, "SeGCom: A Secure Group Communication Mechanism in Cluster-Tree Wireless Sensor Networks", Communications and Networking, ComNet 2009, pp. 1-7, Nov 2009.

[13] A. Diop, Y. Qi and Q. Wang, "Efficient Group Key Management using Symmetric Key and Threshold Cryptography for Cluster based Wireless Sensor Networks", I.J. Computer Network and Information Security, pp. 9-18, 2014.

SESSION

NETWORK SECURITY + SECURITY MANAGEMENT

Chair(s)

Dr. Younchan Jung
Dr. Xinli Wang

Prevention of Toll Frauds against IP-PBX

James Yu
DePaul University
Chicago, IL 60604

Abstract – This research is motivated by a toll fraud case against an enterprise IP-PBX, and further investigation of an asterisk server log shows a growing threat of VoIP attacks against enterprise IP-PBX. Although the Session Initiation Protocol (SIP) has a comprehensive security measures, its implementations are optional and VoIP administrators could be confused about which security measures are required for their specific environments. This study identifies several vulnerabilities in the VoIP implementation, and hackers could explore the vulnerabilities to launch various security attacks. Based on the analysis of the log data and the protocol, this study proposes several counter measures to prevent the security attacks for different VoIP implementations.

Key-Words —VoIP, SIP, Registration Hijacking, Toll Fraud, PRS

I. INTRODUCTION

THE ubiquity of Internet Protocol (IP) and economics of IP-PBX make Voice over IP (VoIP) a cost-effective alternative to the legacy PBX or key system. An Infonetics report estimated the service market of VoIP is $68B in 2013, and expected to grow to $88B in 2018 [1]. The growth of IP-PBX seats (end-user ports) is estimated at 35% annually. However, there is also an increase in the phone fraud. A 2011 survey shows that phone fraud was estimated at $4.96B which is more than double the credit card fraud $2.40B [2]. In the same report, toll fraud is a major fraud category where hackers explore the vulnerability of the system for financial gain.

The increased use of IP-PBX in the enterprise environment makes it a new target of security attacks. In the taxonomy of VoIP security, researchers classify security requirements as follows [3][4][5]:

- *Confidentiality* – the communication is between the sender and the intended receiver. The security measure is to protect and prevent eavesdropping of the communication.
- *Integrity* – the content of the communication does not change during the communication. The security measure is to protect both signaling traffic and bearer traffic. In the case of bearer traffic, it needs to protect both the header and payload content.
- *Authentication* –both the caller and the callee are authentic users as they are claimed in the call messages and content. Authentication is assured by the server.
- *Availability* – this is the case of protecting against Denial of Service (DoS) attack.

Toll Fraud is an issue in the category of *authentication* where a hacker falsifies the caller ID and makes a call from the caller system for financial gains. Researches on toll fraud can be classified as fraud detection and fraud prevention. An example of fraud detection is to study real-time Call Detail Record (CDR) and identify anomalies in CDR [6]. This research is from a network perspective on fraud prevention. A major incentive of toll fraud by hackers is for immediate financial gain. Hackers explore the potential IP-PBX vulnerabilities and try to access it to make long distance (toll) calls. We can further categorize toll frauds into two categories:

1. The first category is that hackers gain access to enterprise IP-PBX and use it as a gateway for commercial use. A SANS report published a case where a hacker created several phone companies and route toll calls of his customers to multiple *hacked* IP-PBX. He made over $1M by charging his customers before being caught [7].
2. The 2nd category is the fraudulent use of Premium Rate Sharing Service (PRS). In the U.S., this is the 900- calls where each call is charged a high premium and the callee (receiver) gets a share of the service premium. Due to many frauds of the 900- calls, Federal Communication Commission (FCC) has a strict regulation. As a result, the frauds of 900- calls have been significantly reduced. However, the case of International Premium Rate Sharing (IPRS) is a new threat of phone fraud.

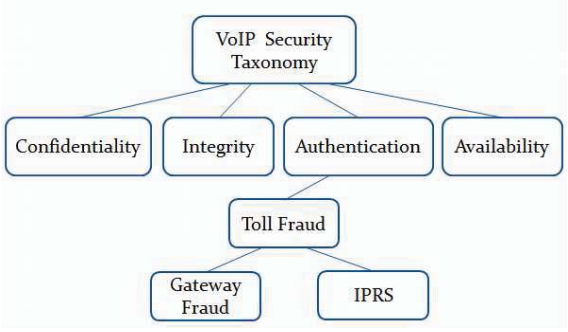


Figure 1. Taxonomy of VoIP Security

An example of the phone bill of IPRS fraud is given below:

Itemized Phone Bill

		Time	Charge $$
2522160333	07/20/11 9:49 p Somalia	19.8	16.6657
2522160333	07/20/11 10:15 p Somalia	16.8	14.1406
2522160333	07/20/11 10:32 p Somalia	17.4	14.6456
2522160333	07/20/11 10:49 p Somalia	20.7	17.4232
2522160333	07/20/11 11:10 p Somalia	20.7	17.4232
2522160333	07/20/11 11:31 p Somalia	11.9	10.0162
2522160333	07/20/11 11:44 p Somalia	16.6	13.9722

This is the case that a hacker intruded into an enterprise IP-PBX, and then made continuous international calls to many African countires (one of them is Somalia.) It is apparent that the hacker had a program to automatically generate calls, and each call lasted for 10-20 minutes. The fraud case started at 06:00pm and continued until 06:00am next morning. Because the calls were made at night, the users were not aware of this fraud case. The hacker then tried it again the next day. It took several days for the phone compnay to identify the fraud and cut the phone service. However, the company already accumulated a phone bill of thousands of dollars. After the first fraud case, the company implemented a strict dialing plan to prevent international calls. Any 011 prefix dialing was blocked. However, the North American Numbering Plan (NANP) includes many Carribbean countires which follow the NXX-XXX-XXXX dialing plan. The dialing plan to any NANP country is the same as a US domestic call. As a result, this company was hacked again with another large phone bill to a Carribbean country.

The purpose of this research is to study the vulnerability of VoIP implementations, and develop protective measures to prevent toll fraud against IP-PBX in an enteprise environment.

II. VoIP Networks

An example of IP-PBX for an enterprise environment is illustrated in Figure 2.

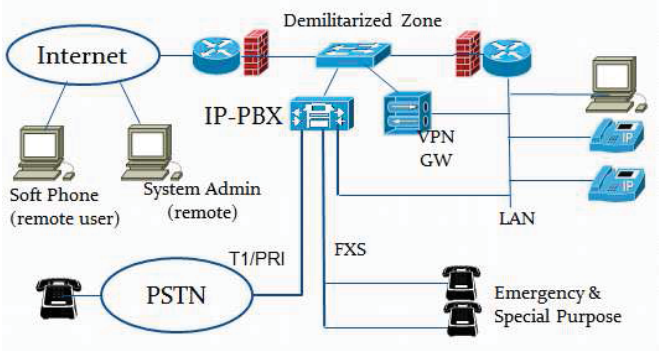


Figure 2. IP-PBX for an Enterprise environment.

An IP-PBX has the functions of both Session Initiation Protocol (SIP) proxy and gateway, and it usually has the four physical interfaces:

1. LAN ports – a LAN port connects to an Ethernet switch which connects to the enterprise Local Area Network (LAN). IP phones and workstation with soft phone are connected to the LAN.
2. WAN (Internet) port – a WAN port connects to the public Internet. The purpose of WAN port is to support remote clients and remote system administration. For security protection, the WAN connection terminates at the demilitarized s zone (DMZ) which is protected by a firewall.
3. Foreign Exchange Subscriber (FXS) lines – An IP-PBX usually has one to four FXS ports. This is to support emergency calls (e.g., E911) or special use (such as Fax).
4. PSTN interface – an IP-PBX may use either Foreign-Exchange-Office (FXO) lines or T1/E1 lines to connect to the Public Switch Telephone System (PSTN). In a typical office environment, we usually follow an engineering rule of 1:8 where one FXO line serves up to 8 users. For example, a small office of 30 staff would subscribe to four FXO lines. If a company needs more than 10 FXO lines, a T1 would be more economical. A T1/PRI (Primary Rate Interface) trunk has 23 B- channels and could support an office up to 184 users.

VoIP is an *application* on IP-based data network, and all security measures of IP network are applicable to VoIP [8]. This is the reason that IP-PBX needs to be positioned at DMZ and relies on firewall to protect typical threats against data network. We also observe a growing trend toward *all* IP networks in an enterprise environment, which is to use SIP trunking to replace FXO/T1 connections to PSTN. The SIP trunking configuration is illustrated in Figure 3.

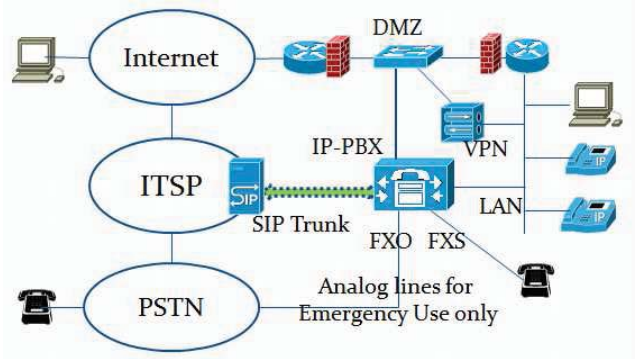


Figure 3. SIP Trunking Configuration

A SIP trunk is not a physical connection, but a secured IP tunnel from the enterprise to an Internet Telephony Service Provider (ITSP). Both signaling and bearer traffic is sent on this IP tunnel to the ITSP which routes the bearer traffic to PSTN. The pricing of each SIP trunk is comparable to an FXO line. Some of the advantages of SIP trunk are (a) simpler device configuration, (b) more scalable for service growth, and (c) lower toll cost. A disadvantage is service availability and reliability of emergency calls (e.g., E911).

The objective of security protection is to allow *normal* use of the network and to prevent *misuse* of the network. The scope of normal use includes the following:

1. Internal users (LAN side) can make and receive calls internally, and also make and receive calls to and from PSTN.
2. Remote users (Internet side) have the same capability as internal users, including making and receiving calls to and from PSTN. One important *built-in* VoIP feature is global *voice* Virtual Private Network (VPN). A user can keep his/her local phone number regardless of his/her physical location in the world.
3. Administrator have the capability to monitor the system either from the LAN side or through the public Internet.

The security measure agaisnt toll fraud should support *normal* use of phone services and protect *fraudlent* use from both internal and external hackers.

III. Hacking Scenarios and Prevention

In the legacy PBX system, toll frauds are usually from stolen pass codes that allow unauthorized access to PBX for free toll calls. If an IP-PBX has an interface to the Internet with a public IP address, it opens the door for a new wave of hacker attacks.

A. Hacking and Intrusion Attempt

In our VoIP lab, we have installed multiple asterisk servers for our VoIP course [9], and one of the servers has a public IP address for students to make proxy-based SIP calls. About half of our students are distance learning students, so the lab environment is accessible to these students via the public Internet. The server does not have PSTN interfaces, so it would not have an issue with toll fraud. To investigate the potential threat of the VoIP attack, we studied the Asterisk log and noticed an alarming attacking rate of the following pattern:

```
[Mar 14 03:00:25]Registration from '"7121" <sip:7121@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:00:36]Registration from '"7957" <sip:7957@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:00:47]Registration from '"7957" <sip:7957@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:00:59]Registration from '"7955" <sip:7955@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:01:11]Registration from '"7955" <sip:7955@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:01:24]Registration from '"7937" <sip:7937@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:01:37]Registration from '"7937" <sip:7937@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:01:49]Registration from '"7734" <sip:7734@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:01:58]Registration from '"7734" <sip:7734@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:02:11]Registration from '"7774" <sip:7774@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:02:22]Registration from '"7774" <sip:7774@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:05:24]Registration from '"7633" <sip:7633@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:05:24]Registration from '"7633" <sip:7633@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:05:24]Registration from '"7633" <sip:7633@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:05:24]Registration from '"7633" <sip:7633@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:05:24]Registration from '"7633" <sip:7633@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:05:24]Registration from '"7633" <sip:7633@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:05:24]Registration from '"7633" <sip:7633@140.192.40.4>' failed for '212.129.38.252'
[Mar 14 03:05:24]Registration from '"7633" <sip:7633@140.192.40.4>' failed for '212.129.38.252'
```

Figure 4. Asterisk Log of VoIP Intrusion Attempt

The log data shows that hackers first check the open SIP port (UDP port 5060). When they find a VoIP server with this port open, they will try every possible extension to explore the VoIP system and they will also try many passwords for each extension. Our Asterisk server is available during the academic quarter only, and its Internet connection is disabled after the class is over. The attacking rates of the fall quarter of 2014 and the winter quarter of 2015 are illustrated in Figures 5a and 5b. Note that we experienced 8.67 million intrusion attempts in one day (03/14/2015) (Figure 5b). Although these are all failed attempts, the amount of hacker traffic affects the normal operation of the server and our network. This intrusion data shows a more serious case (many more intrusion attempts) than other researches which used a honeypot approach to trap and analyze the attacking data [10][11].

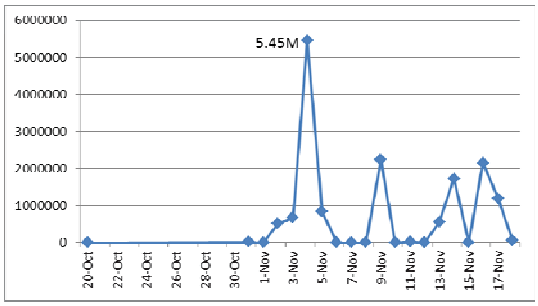


Figure 5a. Daily SIP Intrusion Attempts in Fall 2014

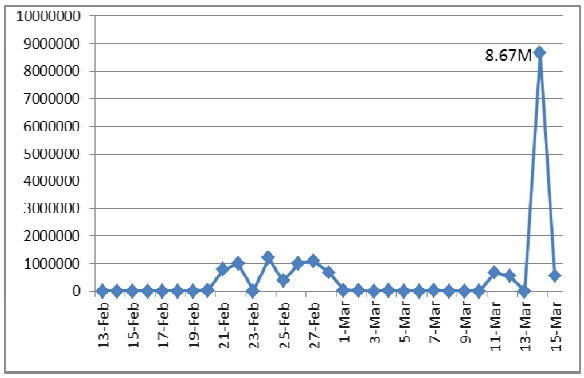


Figure 5b. Daily SIP Intrusion Attempts in Winter 2015

We also checked the source location of IP addresses of those SIP intrusion attempts, and the results are given in Figure 6.

Attempts	IP	Country		
7938129	62.210.245.11	FR		
1795703	195.154.38.225	FR		
1784133	195.154.42.18	FR		
1575740	195.154.41.250	FR		
1303322	212.129.38.252	FR		
1227472	212.83.128.50	FR		
1009644	62.210.245.132	FR		
57656	209.126.100.162	US	MO	Saint Louis
21344	195.154.38.97	FR		
19860	195.154.33.127	FR		
10573	85.25.214.254	DE		
1580	37.75.213.132	PS		

Figure 6. SIP Intrusion Source in Winter 2015

From our analysis, we concluded that this is similar to Distributed Botnet attacks [12]. The hacker builds a botnet and recruits an army of bots from multiple sources. When the hacker finds a potential target, he/she will send SIP

registration requests to find a valid, unsecured phone number. If he/she finds one, his/her Command and Control Center will make hundreds of automated IPRS calls from which the hacker would share the premium.

B. SIP Registration and Session Hijacking

According of the SIP standard (RFC 3261), the password for SIP clients is optional. A SIP client (e.g., IP Phone) is first registered with a SIP proxy server. After the success of registration, the SIP client can make a call request via the INVITE message. The message flow of registration and call is captured in the wireshark packet trace as illustrated in Figure 7. Note that the call setup time (INIVTE to 180 Ringing) is

$$10.009 - 9.795 = 0.214 \text{ (sec)} = 214 \text{ ms}$$

No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.0.106	140.192.40.4	SIP	Request: REGISTER
3	0.096954	140.192.40.4	192.168.0.106	SIP	Status: 200 OK (1
6	9.795298	192.168.0.106	140.192.40.4	SIP/SDP	Request: INVITE si
8	10.009022	140.192.40.4	192.168.0.106	SIP	Status: 180 Ringin
9	27.064595	140.192.40.4	192.168.0.106	SIP/SDP	Status: 183 Sessio
11	27.070765	140.192.40.4	192.168.0.106	SIP/SDP	Status: 200 OK |
13	27.105583	192.168.0.106	140.192.40.4	SIP	Request: ACK sip:2
555	32.536903	192.168.0.106	140.192.40.4	SIP	Request: BYE sip:2
557	32.567092	140.192.40.4	192.168.0.106	SIP	Status: 200 OK |
558	46.526187	192.168.0.106	140.192.40.4	SIP	Request: REGISTER
560	46.570758	140.192.40.4	192.168.0.106	SIP	Status: 200 OK (0

Figure 7. SIP Registration and Call Flow (no password)

If an IP-PBX is open to the public Internet and one of the phones (SIP client) is not password protected, a hacker can easily find this phone through exhaustive search. Also note that an *easy-to-guess* password is the same as no password because hackers will try hundreds of common-used passwords for each account.

C. SIP Authentication

To prevent the above attacking scenario, the SIP standard provides an authentication procedure. A password is provisioned for each SIP account and the password is configured on both the server and the client. The registration process and the call flow diagram are captured in the wireshark packet trace as illustrated in Figure 8. The highlighted messages are authenticated. Note that the call setup time (INVITE to 180 Ringing) is

$$(18.460 - 18.147) = 0.313 = 313 \text{ ms}$$

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.106	140.192.40.4	SIP	588	Request: REGISTER sip:140.192.40.4 (1 bi
3	0.032525	140.192.40.4	192.168.0.106	SIP	584	Status: 401 Unauthorized |
4	0.135334	192.168.0.106	140.192.40.4	SIP	742	Request: REGISTER sip:140.192.40.4 (1 bi
6	0.167877	140.192.40.4	192.168.0.106	SIP	631	Status: 200 OK (1 binding) |
11	18.147348	192.168.0.106	140.192.40.4	SIP/SDP	990	Request: INVITE sip:20370140.192.40.4 |
12	18.182377	140.192.40.4	192.168.0.106	SIP	597	Status: 407 Proxy Authentication Required
13	18.182781	192.168.0.106	140.192.40.4	SIP	364	Request: ACK sip:20370140.192.40.4 |
14	18.186305	192.168.0.106	140.192.40.4	SIP/SDP	1155	Request: INVITE sip:20370140.192.40.4 |
16	18.460265	140.192.40.4	192.168.0.106	SIP	531	Status: 180 Ringing |
18	39.749511	140.192.40.4	192.168.0.106	SIP/SDP	856	Status: 183 Session Progress |
21	39.760445	140.192.40.4	192.168.0.106	SIP/SDP	842	Status: 200 OK |
33	39.865245	192.168.0.106	140.192.40.4	SIP	634	Request: ACK sip:20370140.192.40.4 |
2105	60.470602	192.168.0.106	140.192.40.4	SIP	591	Request: SUBSCRIBE sip:20360140.192.40.4
2109	60.502533	140.192.40.4	192.168.0.106	SIP	585	Status: 401 Unauthorized |
2121	60.605349	192.168.0.106	140.192.40.4	SIP	750	Request: SUBSCRIBE sip:20360140.192.40.4
2516	64.540309	192.168.0.106	140.192.40.4	SIP	674	Request: BYE sip:20370140.192.40.4 |
2519	64.574150	140.192.40.4	192.168.0.106	SIP	523	Status: 200 OK |

Figure 8a. Caller Side (Wireshark Packet Trace)

No.	Time	Source	Destination	Protocol	Info
1	0.00000000	192.168.0.20	140.192.40.4	SIP	Request: REGISTER sip:140.
3	0.03128200	140.192.40.4	192.168.0.20	SIP	Status: 401 Unauthorized |
4	0.03843000	192.168.0.20	140.192.40.4	SIP	Request: REGISTER sip:140.
6	0.07162900	140.192.40.4	192.168.0.20	SIP	Status: 200 OK (1 bindin
11	16.8360960	140.192.40.4	192.168.0.20	SIP/SDP	Request: INVITE sip:2037@1
13	17.0105860	192.168.0.20	140.192.40.4	SIP	Status: 180 Ringing |
15	24.1985060	192.168.0.20	140.192.40.4	SIP/SDP	Status: 200 OK |
17	24.2291050	140.192.40.4	192.168.0.20	SIP	Request: ACK sip:2037@192.
868	32.7948650	140.192.40.4	192.168.0.20	SIP	Request: BYE sip:2037@192.
870	32.8391460	192.168.0.20	140.192.40.4	SIP	Status: 200 OK |

Figure 8b. Callee Side (Wireshark Packet Trace)

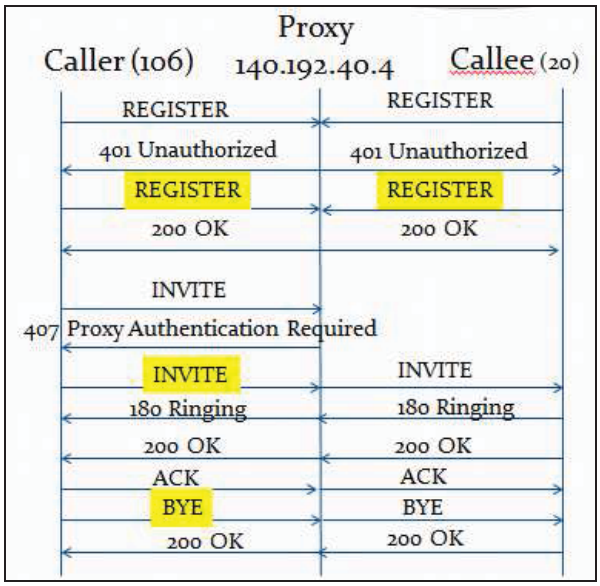


Figure 8c. SIP Registration and Call Flow

As llustrated in Figure 8, the authentication follows a *three-way* hand-shaking procedure for the registration process. The 1st REGISTER message from the client is a request, and the server responds with a challenge (401 Unauthorized). The client then resends the REGISTER message with a response to the challenge. The password is not sent in the authentication process, and the password is used to decrypt the challenge and to encrypt the response on the client side. There is a unique *nonce* value generated on the server for each challenge-response. It should also be noted that this registration process is repeated every 60 seconds as shown in Figure 8. The authentication process is also applied to each call. When a client makes a call request, it also uses the three-way hand-shaking process to authenticate each call request. The nonce value, generated on the server, is used as a challenge in the 407 Proxy Authentication Required message. This nonce value is used for the next INVITE message to authenticate the call.

It should be noted that this SIP authentication process is a one-way procedure. It is for the proxy server to authenticate the clients, but it does not support clients to authenticate the proxy server. In Figure 8, there is no authentication for any message on the callee side (except for registration). This one-way authentication has a potential issue of Denial of Service (DoS) attacks. If a call is terminated by the caller, the SIP BYE message is authenticated by the same nonce value of the original call. However, if a call is terminated by the callee, the SIP BYE message is not authenticated as illustrated in Figure 9.

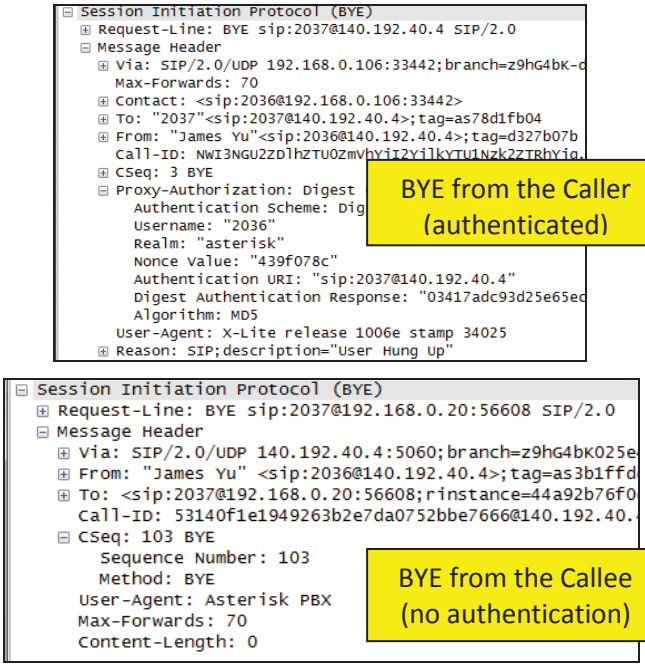


Figure 9. SIP BYE Message

Because this SIP messages on the callee side are not encrypted, a hacker could sniff the traffic from the network. The hacker could *impersonate* as a callee and send a BYE message to the proxy server. Without authentication, the proxy simply relays the BYE message to the caller and terminates the call. Abdelnur identified another vulnerability in SIP where an authenticated user could obtain credentials of other legitimate users and make fradulent calls from their accounts [13]. This is an example of internal hacking by legitimate users.

D. SIP over TLS

In the previous section, we identified three vulnerabilities of the SIP authentication process: (a) not a mutual authentication, (b) unauthorized call termination, and (c) weakness in protecting user credentials. These three issues could be addressed by SIP over Transport layer Security (TLS) which is also specified in RFC 3621 [15]. The procedure of SIP over TLS, aka as SIPS, is the same as HTTP over TLS, aka HTTPS. Shen and his colleagues did a thorough performance analysis of SIPS overhead [16]. According to their study, the most secured case of TLS mutual authentication would reduce the call capacity (calls per seconds) from 460 cps down to 60 cps. Although this reduction seems significant, a capacity of 60 cps is equivalent to 216,000 Busy Hour Calls (BHC). Assuming a user makes an average of four calls during *busy* hour, this capacity could support an enterprise of 54,000 users. Therefore, the performance issue is not a concern of using SIP over TLS.

IV. VoIP Security Protection

Given the severity of hacker attacks on the VoIP, our first recommendation is that if an IP-PBX has a direct connection to PSTN, the IP-PBX should not have a public IP address.[1] We recommend to move IP-PBX out of the DMZ and to put it behind the 2nd firewall as illustrated in Figure 10.

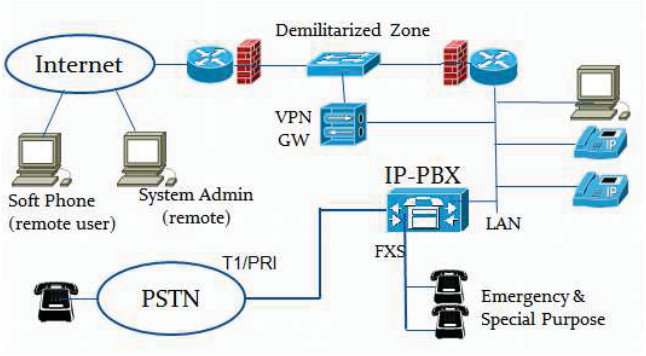


Figure 10. Secured IP-PBX for an Enterprise

Without a public IP address, external hackers cannot access the IP-PBX, and we prevent any threat of external intrusion attempts or attacks. However, we still need to support remote users and remote administration over the public Internet. Our recommendation is to require remote users to access the internal IP-PBX via IP-VPN. The protocol stacks of VoIP over IP-VPN are given in Figure 11.

Bearer	Signaling
RTP	SIP
UDP	UDP
IP	
IPSec (ESP)	
UDP	
IP	

Figure 11. Protocol Stacks of VoIP over IP-VPN

Our lab test, which is based IPSec/ESP (Encapsulating Security Payload), shows that the performance overhead of ESP on the client is low, and the set up time (270 ms) is comparable to non-VPN cases. There is no performance overhead on IP-PBX as it does not *see* IP-VPN. A remote user over IP-VPN is the same as a local user from the IP-PBX perspective.

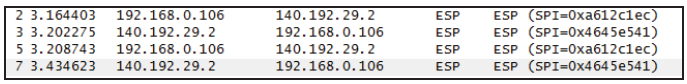


Figure 12. Encrypted SIP messages
(Invite, Status 407, Invite and180 Ringing)

We also recommend that all SIP clients must be password protected. This could be an administration issue to manually configure passwords on individual clients. Fortunately, most VoIP servers (including Asterisk) support the generation of client configuration files and

[1] In some environment, its *public-like* IP address is not routable on the Internet. It is considered the same as a private IP address.

then automatically distribute these files to the clients. This step is essential for the provisioning of the VoIP service. It is possible and also acceptable to set up test accounts without passwords, but it should be limited during testing only. When an IP-PBX is in the production environment, administrator should conduct regular audits to assure that all SIP accounts are password protected and have not-easy-to-guess passwords.

In the case SIP trunking service, the IP-PBX requires a public IP address to connect to the SIP proxy server managed by the ITSP. In this case, SIPS (SIP over TLS) should be required for *signaling* traffic, and Secured RTP (SRTP, RFC 3711) should be required for *bearer* traffic. If the ITSP does not support SIPS or SRTP, the enterprise should use a security measure comparable to SIPS and SRTP. For example, an IP-VPN tunnel based on IPSec and L2TP provides a strong security protection comparable to SIPS and SRTP. It should be noted that this public IP address on IP-PBX is for the ITSP only. The firewall policy should prevent any other service or any other external connection using this IP address.

The enterprise also needs to consider hacking/abuse within the network (LAN side). We recommend to use Virtual LAN (IEEE 802.1Q) to segregate voice and data traffic [17], and also to implement Quality of Service (802.1p) to give priority to voice traffic over data traffic. This design prevents internal hackers from sniffing voice traffic. The network administrator could also monitor traffic on individual voice ports on the Ethernet switch. If a voice port has unusual traffic spike, it would trigger a security alert for further investigation. In our lab environment, we use mrtg (www.mrtg.org) to monitor the lab traffic, and an example of our monitoring chart is illustrated in Figure 13.

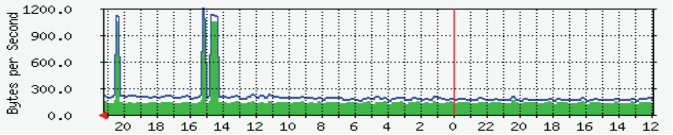


Figure 13. Monitoring VoIP Traffic on Individual Ports

Other recommendations of security counter-measures are given as follows:

1. Disable non-service related ports [18]. This is a standard practice of hardening a server.
2. Restrict international calls to designated phone numbers. As discussed earlier, international calls are not limited to 011 calls. They are many NANP countries, and administrator needs to identify their area codes and to restrict calls to these countries [19].
3. Constantly monitor Call Detail Record (CDR) to identify unusual usage patterns. A CDR is created after each call, and it contains the billing information of the call. Administrator should not wait until the phone bill; instead, administrator should monitor CDR and identify abnormal events.

V. Conclusion

This research is motivated by a real case of toll fraud, and further study of the lab log shows an alarming and growing threat of VoIP attacks. This paper presents a detailed study of the authentication process in the VoIP protocol (SIP) and identifies several vulnerabilities of its use. Note that we did not identify issues with the protocol (SIP) itself, but its security features are optional in implementation. Our study shows that the use of security measures depends on the enterprise network configuration. For example, SIPS should be mandatory for IP-PBX with a public IP address. Certain security measure (e.g., password protection for clients) should be mandatory regardless of any environment.

It should be noted that the scope of our study is on the IP side, and it does not cover the protection on the PSTN side. Our conclusion is that an IP-PBX, if properly protected, could be as safe as any legacy PBX, but not better. The reason is that all attacks from the PSTN side are equally applicable to both IP-PBX and legacy PBX.

References

[1] Diane Myers, "2014 VoIP and UC Services and Subscribers," http://www.infonetics.com/pr/2014/2H13-VoIP-UC-Services-Market-Highlights.asp

[2] Wikepedia Phone Fraud http://en.wikipedia.org/wiki/Phone_fraud

[3] D. Butcher, X. Li, and J. Guo, "Security Challenge and Defense in VoIP Infrastructures," IEEE Transactions on Systems, Man, and Cybernetics, , Vol. 37-6, November 2007, pp. 1152-1162.

[4] Angelos D. Keromytis, "A Comprehensive Survey of Voice over IP Security Research," IEEE Communications Survey and Tutorial, Vol. 14-2, 2012 pp. 514-437.

[5] D. Hoffstadt et. al. "A comprehensive framework for detecting and preventingVoIP fraud and misuse," International Conference on Computing, Networking and Communications (ICNC), February 2014, pp. 807-813.

[6] Hofbauer, S. et. al., "A Lightweight Privacy Preserving Approach for Analyzing Communication Records to Prevent VoIP Attacks using Toll Fraud as an Example," IEEE 11th Intl. Conf. on Trust, Security and Privacy in Computing and Communications (TrustCom), June 2012, pp. 992 – 997.

[7] David Persky, "VoIP Security Vulnerabilities," SANS Institute, Fall 2007. http://www.sans.org/reading-room/whitepapers/ voip/voip-security-vulnerabilities-2036

[8] D. R. Kuhn, T. J. Walsh, and S. Fries, "Security Considerations for Voice over IP Systems," National Institute of Standard and Technology, 800-58, January 2005.

[9] Asterisk VoIP server. http://www.asterisk.org/

[10] Gruber, M. et. al., "Voice calls for free: How the black market establishes free phone calls - Trapped and uncovered by a VoIP honeynet," 11th Intl. Conf. on Privacy, Security and Trust (PST), July 2013, pp. 205 – 212

[11] Aziz, A. et. al., "A distributed infrastructure to analyse SIP attacks in the Internet," IFIP Networking Conference, June 2014, pp. 1-9.

[12] L. Zhang, S. Yu, Di Wu, P. Watters, "A Survey of Recent Botnet Attacks and Defenses," IEEE 10th International Conference on Trust, Security, and Privacy in Computing and Communications, November 2011, pp. 53-60.

[13] Abdelnur, H. et. al. "Abusing SIP Authentication," 4th Intl. Conf. on Information Assurance and Security, September 2008, pp. 237-242.

[14] A. N. Jaber, S.Manickam, S. Ramdas " A study of SIP trunk security and challenges," IEEE International Conference on Electronics Design, Systems and Applications (ICEDSA), November, 2012 pp. 239-243.

[15] G. Sonwane, B. Chandavarkar, "Security Analysis of Session Initiation Protocol in IPv4 andIPv6 Based VoIP Network," 2nd International Conference on Advanced Computing, Networking and Security (ADCONS), December 2013, pp. 187-192.

[16] Shen, C. et. al. "The Impact of TLS on SIP Server Performance: Measurement and Modeling," IEEE/ACM Transactions on Networking, Volume: 20 , Issue: 4, August 2012 , pp.1217 - 1230

[17] G. Sonwane, B. Chandavarkar, "Security Analysis of Session Initiation Protocol in IPv4 andIPv6 Based VoIP Network," 2nd International Conference on Advanced Computing, Networking and Security (ADCONS), December 2013, pp. 187-192.

[18] X. Wei, K. Sellal, Y. Bouslimani, "Security Implementation for a VoIP Server," International Conference on Computer Science & Service System (CSSS), August 2012 pp. 983-985.

[19] Countries of North American Numbering Plan (NANP) http://www.nanpa.com/pdf/NANP_Member_Country_Maps.pdf

An Examination of Recent Network Security Failures

Sean Maples
Southern New Hampshire University
2500 North River Road
Manchester, NH, 03106, USA
Email: sean.maples@snhu.edu

Weifeng Chen
Dept. of Math, Computer Science & Computer Information Systems
California University of Pennsylvania
California, PA
Email: chen@calu.edu

***Abstract*—The purpose of this paper is to determine whether recent high profile network security breaches of large corporations is the result of attackers growing sophistication and techniques defeating available network security tools and techniques, or if the attackers are being successful due to poor security practices. This paper outlines the types of attacks in which malicious entities attempt to breach a network. The paper then details the tools available to prevent and defend against attacks. The high profile security breaches that are examined are the data theft of millions of credit and debit card information from Home Depot in 2014 and Target in 2013. In both cases this paper details how the attacks occurred, and why the security in place failed to prevent them. The results of the examination show that the techniques used by the attackers were not sophisticated enough to defeat modern security measures. The techniques succeeded due to poor security practices in place at Home Depot and Target. In conclusion, the tools available for network security today will prevent the attacks being used in these high profile cases; that is if they are utilized.**

***Keywords*—Network security; Home Depot security breach; Target security breach; Security policy;**

I. INTRODUCTION

Modern network security is tasked with stopping an ever expanding list of threats. The sophistication of these threats are increasing at an incredible pace. In the last few years a number of high profile attacks have raised the question as to whether a secure network is obtainable. As large corporations with billions of dollars in yearly revenue fall victim to crippling attacks that take down their networks, steal their confidential data, and the data of their customers. This begs the question, is the state of network security such that the attacker has the upper hand?

Network security exists in a general sense to protect the network. This protection has several goals that need to be achieved. The most important of these goals is to prevent unauthorized access to the network, protect the network's operational capability from external threats, protect the integrity of network data, and to ensure secure communication. To prevent unauthorized access to the network each network administrator implements access control policies to authorize or de-authorize what devices can communicate with the network. To protect the network from external threats a network administrator enables a firewall, an intrusion detection system, an intrusion prevention system, and other options. To ensure secure communication encryption and end-point authentication can be utilized.

In this paper, we examine in details two recent security breach cases, to understand whether network security is effective or not. Section II describes security threats. Security tools that could be used to protect against threats are presented in Section III. Section IV analyzes the security breaches on Home Depot in 2014 and Target in 2013. Lessons learned from the breaches and suggestions to improve network security are discussed in Sectionřefsec:lessons. Finally we conclude the paper in VI.

II. SECURITY THREATS

The threats to network security can be broken down into two main types, active and passive. Active threats are when an attacker attempts to bypass or break into secured systems. There are many different kinds of active attacks such as viruses, worms, trojan horses, denial of service, spoofing, and more. Active attacks exist to attempt to "circumvent or break protection features, to introduce malicious code, and to steal or modify information." [12]. As such active attacks can be considered the most dangerous.

Passive attacks are when an attacker takes a stealthier option. Rather than trying to break through multiple security layers to imbed malicious code or cripple a device, passive attacks go after the data that is being sent back and forth. This data is often unencrypted allowing the attack to gain access to "clear-text passwords and sensitive information that can be used in other attacks." [12] Passive attacks can often be utilized to case the security of a network, in much the same way a robber might case a bank. Finding a potential piece of information that will allow for a devastating active attack. Types of passive attacks include eavesdropping (password sniffing, traffic analysis), scavenging, system mapping (scanning for vulnerabilities), and side channel attacks [5].

On top of these active and passive attacks the network must be secured against attacks aimed at the users of the network. The most notorious of these attacks is phishing, using e-mail or other messaging to trick the user into revealing critical information or running malicious code. For a network to be secure it must be able to withstand the impact of active threats,

prevent passive threats, and prevent users from unwittingly attacking it or disclosing security information.

III. SECURITY TOOLS

While the attacks being used against networks have advanced over the years, network security has not been standing still. Organizations and network administrators have been building on the successes and failures of the past. Security updates are releasing faster, firewalls are looking past the perimeter, and encryption has become easier to implement. Zero day threats are discovered, yet software companies are patching them faster than ever. Anti-virus and malware protection tools are constantly updating both their detection files to find new threats, but also their techniques. Real-time monitoring of systems is now accomplished in even basic commercial malware protection protects. Multi-factor authentication, requiring two or more methods of unique identification from a user, such as a password and a code texted to the user's phone, is seeing increasing use, forcing attackers to steal not just a login ID and password, but also to steal or clone the user's phone, or other code generating hardware.

In the following we will summarize three common security tools: Firewalls, Intrusion Detection & Prevention, and Encryption.

A. Firewalls

Firewalls are a good example of security tools that has been long in service and has seen changes over the years to meet the challenges of evolving threats. A firewall, in this case a network-based firewall, uses both hardware and software to isolate an internal network from the Internet. Firewalls serve three main goals: make all traffic from inside and outside the firewall pass through it, make sure only authorized traffic (as defined by the local security policy) can pass through it, and make sure it itself is immune to penetration [8]. Firewalls can be a key tool for network administrators, as they allow the network administrator a way to control access to their network. Giving the administrator control over the traffic flowing to and from their network. Firewalls often implement this control through packet filtering, which looks at the packet's Internet Protocol (IP) addresses, protocol used, and ports to verify that the package is coming from the correct source, and allowed into the network.

Though the usage of firewalls has come under scrutiny in today's changing landscape of advanced threats, remote connections, and remote devices; the firewall "continues to be critical in enabling network segmentation and in ensuring critical business and corporate systems are separated." [18] Earlier firewalls focused on the perimeter of the network. Stopping traffic attempting to get in or out when it didn't meet the access policy setup by the network administrator. Firewalls though have seen advancement, and next generation firewalls offer added layers of security including intrusion prevention systems (IPS), application visibility and control, high-performance protection, and controls integrated through a central management system [9].

B. Intrusion Detection and Prevention

Another tool that can be of great help in combatting network attacks is the usage of intrusion detection systems (IDS) and intrusion prevention systems (IPS). IDS and IPS perform packet inspection, except they take that inspection of incoming traffic a step further than standard firewalls. They perform a deep packet inspection, which is looking "beyond the header fields and into the actual application data that the packets carry." [8] For instance say a packet was a package a company received in the mail. The packet inspection performed by the firewall would be like having the mail room look at the sender, receiver, and packaging to make sure it looks legit. While with deep package inspection it would be like having the mail room open up the package and examine the contents. The difference between an IDS and IPS system is what happens after the deep packet inspection. An IDS system will generate an alert if malicious traffic is found, while an IPS system will filter that traffic out. A network administrator then can implement an IPS or IDS setup to filter out known threats, or alert to a possible threat, and in doing so prevent an attack.

C. Encryption

Encryption has also come along ways over the years, becoming more robust, easier to implement, and rising to the challenge of advancing computational performance and encryption breaking techniques. There are two types of modern encryption, symmetric key algorithms and asymmetric key algorithms. Symmetric key algorithms use the same or related encryption keys for both encryption and decryption, while asymmetric key algorithms use different keys. Asymmetric key algorithm's usage of different keys can also be referred to as public-key cryptography, since one key is made public, while the other is kept private. Symmetric key algorithms work by encrypting the data and then sending that data, and sending the encryption key, to the recipient so they can decrypt it. Asymmetric though does not need to send the encryption key, as the public key of the recipient can be used to encrypt the message going to them, and then their private key can decrypt it. Encryption can be used to both secure the data on the networks and ensure secure communication. For instance for secure communication the recipient of the message can rest assured it is from the sender, as either the key they received in the symmetric algorithm encryption worked, or whether their private key in the asymmetric system worked.

1) AES, DES, and key size: The strength of encryption often falls on the bit size of the key. With the implementation of the advanced encryption standard (AES) the key bit size was increased the 56 bit size of the previous data encryption standard (DES) standard to 128, 192 or 256 bit with AES. This increase in key size has effectively made brute force decryption, which is trying each possible combination until the key is found, almost impossible with the current computational power available. As now a brute force attempt would have

to try 2^{128} or 2^{192} or 2^{256} combinations instead of only 2^{56}. AES though is only one of many modern encryption standards. Other include RC4, which is "the most widely-used stream cipher and is used in Secure Socket Layer (SSL) and Wired Equivalent Privacy (WEP)" [10]. SSL is used to secure web traffic, WEP is often used to secure Wi-Fi connections.

IV. CORPORATE NETWORK SECURITY BREACH EXAMINATIONS

Now that a basic overview of various network security tools and techniques has been gone over, it is time to look at some real world examples of network security failures.

A. Case Study One: Home Depot

In April 2014, Home Depot's network security failed. Home Depot is a large corporation operating 2,266 stores hardware stores with $79 billion in annual revenue. In September 2014, Home Depot announced that 56 million credit and debit cards were stolen in a massive data breach. Home Depot later added 53 million e-mail addresses to the list of what was stolen. Home Depot expects to pay around $62 million to recover from the intrusion [3].

1) How it happened: How does a corporation the size of Home Depot end up having the credit and debit card numbers of 56 million customers stolen from their system? The first step was to obtain access to Home Depot's network. Access was gained by using a vendor's stolen log-on credentials [19] By doing so the security in place on the network to protect it from external threats was completely thwarted, as with the stolen credentials from the vendor's system, the attackers appeared to the system as legitimate users. Next the attackers needed to move from the vendor's system to the broader system. This was achieved by exploited a vulnerability found in Microsoft's Windows XP operating system. This exploit was later patched, but not before the damage was done [2]. Once into the broader system the attackers then were able to find a target in Home Depot's 7,500 self-checkout lanes, "because the register's reference names in the computer system clearly identified them as payment terminals." [2] The final step of the Home Depot attack was to then infect the 7,500 self-checkout terminals to get the customer data. The attackers did this "with a new variant of 'BlackPOS' (a.k.a. 'Kaptoxa'), a malware strain designed to siphon data from cards when they are swiped at infected point-of-sale systems running Microsoft Windows." [6]

To quickly recap, the attackers avoided the external network security by stealing the credentials of a vendor, then after getting into the vendor system utilized an exploit in Windows XP to make it into the main system, then once in the main system they were able to find the checkout terminals as they were named as such, then finally they installed malware that siphoned the credit and debit card information as they were used.

2) Where did Home Depot go wrong: Going back to the primer on network security we can see that the firewall and any IDS or IPS proved useless when the attackers could gain access with stolen credentials. Yet what wouldn't have proved useless, and may have saved millions of credit cards numbers from being known, is encryption. According to former managers at Home Depot the credit card numbers were not being encrypted, and that data was "sent from the stores to central servers in clear text." [4] With proper encryption the stolen data would have proven almost impossible to decrypt; that is without the attackers also gaining access to the encryption key.

As mentioned earlier, the naming scheme for devices on Home Depot's network contributed to the success of the attack. As since the self-checkout PoS machines were referenced by name, allowing the attackers to easily find their target. The 70,000 standard cash registers were not referenced on the network by name, instead only using a number, and they were not infected with the malware and did not siphon customer credit data. By simply sticking with the numerical naming scheme used on the standard cash registers for the self-checkout registers, the malware infections of the PoS machines and the 55 million credit and debit card machines likely would have been prevented.

3) Software Updates: The larger error in Home Depot's network security is one that many corporations and individuals share. They did not update their software. While the decision to keep using Windows XPe over a decade after its release might be understandable for a retailer with thousands of machines to upgrade in a poor retail economy, the reality remains that if the attackers were faced with the current version of Windows, Windows 8.1 embedded, or even the earlier Windows 7 embedded, the malware attack on the PoS systems would have failed [11]. The attackers may have not even made it into the main system, and been stuck in the vendor system. Windows XPe was also well known as an exploitable system, as its weak memory access protection enabled an attack technique called RAM scraping. This vulnerability was previously exploited by hackers to steal credit card information from "TJX Companies, TJ Maxx stores, Office Max, Dave & Busters, DSW, Heartland Payment, BJ's Wholesale Club, Barnes & Noble, and Sports Authority." [13] All the improvements made to network security software, operating system security, and application security over the years are pointless if the updated versions are not installed.

B. Case Study Two: Target

The biggest failure of the Home Depot breach may have been that they did not learn from the network security mistakes of their peers. As 9 months prior to Home Deport being breached an almost identical attack happened on the large retail chain of Target. On December 19th, 2013 Target confirmed reports that unauthorized access of customer credit and debit cards had occurred. Announcing "approximately 40 million credit and debit card accounts may have been impacted between Nov. 27 and Dec. 15, 2013." [17] Target later announced that personal information on 70 million customers was also stolen. With final estimates for stolen personal information at about 110 million customers. The New York Times estimated

the cost of replacing stolen cards from Target's breach at "roughly $400 million." [14]

1) How it happened: The tale of how the Target breach happened will undoubtedly sound familiar. According to sources close to the investigation into the breach, the breach appears to have begun with a malware-laced email phishing attack sent to employees at Fazio Mechanical, an HVAC firm, which did business with the nationwide retailer [7]. The attackers used that e-mail campaign to get employees of Fazio Mechanical to unwittingly install a piece of malware that is referred to as Citadel, which is a "password-stealing bot program that is a derivative of the ZeuS banking trojan." [7] It is that password-stealing capability that would allow the attackers to gain the credentials that Fazio Mechanical employees had received from Target to access Target's systems. Then once they had the credentials the attackers were able to access Target's external billing system called Ariba. How the attackers managed to move from the Ariba system and into Target's main network is not confirmed. Without any need to defeat Target's authentication system, Firewall, or any other external security measures they were able to move around in the external billing system. It is speculated that the Ariba system used the same Active Directory (AD) credential system as many other internal applications at Target [7]. Meaning that although the credentials for the HVAC contractors likely would not have had access outside the server running Ariba, the administrators on that server would have, and once inside Ariba the attackers could have found a way to gain working AD credentials for the rest of the network.

Once onto the larger network the attackers managed to find the point-of-sale machines, install the credit information siphoning malware. This malware was found to have the attributes of an advanced threat, namely that it carried out its attack in two stages. "First, the malware that infected Target's checkout counters (PoS) extracted credit numbers and sensitive personal details. Then, after staying undetected for 6 days, the malware started transmitting the stolen data to an external FTP server, using another infected machine within the Target network." [15]

2) Where did Target go wrong: This style of attack would seem to have two obvious weaknesses. One it depends on gaining authenticated access to the network. The attackers bypassed authentication by stealing credentials from a 3rd party using malware installed from an e-mail scam. Yet it would appear that this tactic would have been defeated easily if Target had implemented proper multi-factor authentication on their external billing server. Though if Target's external billing system was properly segmented from their main network; the attackers would only have been able to move around in the external-billing system, and the credit data would have been protected.

The second obvious weakness with this attack is that it is dependent on having the malware active on the network for several days. This would mean that the target of the attack would have to have ineffective or no malware detection. Meaning no real-time monitoring applications on the servers and clients, or an IPS or IDS monitoring the network traffic, that could detect the malware. In the case of Fazio Mechancial, the HVAC company Target gave access to, they were not running a real-time protection anti-malware tool. Investigators found that the "company's primary method of detecting malicious software on its internal systems was the free version of Malwarebytes Anti-Malware." [7] Malwarebytes does provide real-time monitoring, yet only in the Pro version. As such the malware was able to remain undetected on their system, find the login credentials for the Target external-credit system, and transmit that data back to the attackers.

3) Human Error: As for Target, their malware protection appears to have been excellent. They had a "$1.6 million malware detection tool made by the computer security firm FireEye, whose customers also include the CIA and the Pentagon." [16] Which was backed up with a team of security specialists monitoring the network in Bangalore. With the idea that if FireEye detected a threat, the Bangalore office would contact Target's main Minneapolis security operations center, and they would go in and cut out the infecting malware from their system. In this instance it is reported that FireEye did catch the malware being uploaded. The Bangalore monitors then were alerted to the threat, and then notified the security team in Minneapolis. Yet according to 10 former Target employees who spoke with Bloomberg Businessweek, the Minneapolis team did not follow through on the warning from Bangalore. The 1.6 million dollar anti-malware system worked, the alert went off, and the monitoring team notified the security office, and the security office did nothing [16]. The malware used against target was not an advanced next generation technique that defeated the system. As Jim Walter, director of threat intelligence operations at technology security company McAfee said, "The malware utilized is absolutely unsophisticated and uninteresting." [16]

V. MODERN SECURITY TOOLS AND PRACTICES COULD HAVE PREVENTED THESE ATTACKS

None of the techniques employed by the attackers in either case would have prevailed against a network employing good security practices and modern tools. Good security practices means that the company keeps its software up to date, that a company wall off external systems from internal ones on the network, that a company utilizes encryption for all sensitive information being stored on the network and being sent from or to the network. That for authentication the network requires multiple identification factors, especially for external connections. That the company employs a modern firewall with proper filtering of packets, an intrusion detection or prevention system to monitor traffic in the network, and malware monitoring software on the servers and client machines. Finally, the best security practice is having staff that are proactive and serious about security. These attacks on Target and Home Depot did not have to occur.

A. Malware Should Have Been Ineffective and Caught

In the case of the Home Depot the malware that was able to infect the point-of-sale machines to siphon off the credit and debit card data was only effective because the software being run on the point-of-sale machines was antiquated. In the case of Target the malware that was utilized was actually caught by their active monitoring system. Nor could the attackers have managed to get access to Target's system if not for their supplier, Fazio Mechanical, failing to utilize an active malware protection system. The malware was able to succeed do to lax security policies implemented at all three companies. In the case of Home Depot and Fazio Mechanical it succeeded because of poor software choices, which appear to have been made because of the cost involved. In the case of Target the malware succeeded because the security personal did not care to stop it. The real-time malware monitoring software worked, the around the clock monitoring office did its job, but then the people whose job it was to go in and take care of the malware did not do so. In all three of these examples the malware would have failed entirely if the companies simply utilized modern operating systems, modern active anti-malware software, and had employees who cared to follow through.

B. Network Setup Choices Contributed to Attacks Success

The startling realization of the Home Depot analysis is that the point-of-sale machines infected were only the self-checkout models. Yet that was not the result of the attackers only wanting to go after self-checkout machines, it was because the regular point-of-sale machines were hidden on the network do to a numerical naming scheme, while the self-checkout machines were easily seen. Over at Target the network was setup so that the external-client billing server was not properly segmented from the network. If the server was properly segmented, the attackers would not have been able to turn their unauthorized access for one external billing server into access for greater network. In both cases the choice to not implement multi-factor authentication is what allowed the stolen credentials to work. If the attacker masquerading as an external contractor had to not have that contractor's login information, but also the contractor's cell phone or authentication device, they would not have been able to even access the networks.

VI. Conclusion

In the case of Home Depot, Target, and their external contractors, there is a clear pattern that emerges. The attack sophistication is high, yet not very capable. Attackers are going after external users to the system, in the case of Home Depot and Target the vendor, to get credentials to get access to the system, rather than attempting to break into the system through the firewall, IPS, and IDS. Attackers are then finding their targets often due to luck, as the machines they need to attack are recognizably referenced, and in the Home Depot's case by actual name. The attackers are not having to thwart encryption schemes, as the encryption was not being done, with credit information being sent in clear text from local stores to the main network.

A. Malware Used Capable Against Weak Targets

The malware the attackers used was capable of doing what they needed, but only because of the weakness of their targets. In the case of the Home Depot it proved capable against a decade old operating system. In the case of Fazio Mechanical it proved capable against a system with no active malware protection. Then in the case of Target, the most egregious one, it proved capable against a system where the security staff did not act on the warning given by the FireEye anti malware tool and their network monitoring operation. The malware utilized in these three attacks would have failed if the three companies were utilizing a recent operating system, with an active anti malware program, and had employees who were proactive about security.

B. Attacks Succeeded Through Weakness of the Target

These cases paint a clear picture that while the attackers were clearly technically adept, it was the failing of the companies to follow good security practices that led to the massive data breaches. Software was not being updated, even when it was known to be a risk. Network security policies were not being properly implemented and carried out. Encryption was not being utilized, even for highly sensitive data. Security personal were not being proactive and diligent.

C. Who Has the Upper Hand?

As such the answer to the question, does the attacker have the upper hand, is both yes and no. The issues in these cases were not that the network security techniques available were being defeated. The firewalls worked, the authentication system worked, the malware software did the job it was designed to do. These were not grand attacks that broke through multiple level of security with complex new software tools. There were attacks that could have been easily prevented by simply staying current on updates, or using an operation system that is not more than one generation old, or even having employees who care enough to act. The attackers in these cases had the upper hand because their victims failed to properly protect their network. Yet those companies did not have be victims. Companies can easily gain the upper hand as the techniques and the actual malware being utilized by these attackers will not succeed against a well-protected network. With a firewall a company can block most unwanted access to their network. With multifactor authentication the company can prevent unauthorized access from an attacker who manages to guess a poorly chosen password, or steal it. With an active and alert staff using these tools, when an attacker does breach the network, they can find it and stop it. There is no reason the massive attacks on Home Depot and Target happening again has to be a foregone conclusion. No reason hundreds of millions of people need to have their private data exposed. We have the tools to prevent it. All that is necessary is the will to take network security seriously.

REFERENCES

[1] G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529-551, April 1955. (references)

[2] Banjo, S. (2014, November 6). "Home Depot Hackers Exposed 53 Million Email Addresses". Retrieved January 11, 2015, from http://www.wsj.com/articles/home-depot-hackers-used-password-stolen-from-vendor-1415309282

[3] Elgin, B., Riley, M., and Lawrence, D. (2014, September 12). "Former Home Depot Managers Depict 'C-Level' Security Before the Hack". Retrieved January 11, 2015, from http://www.businessweek.com/articles/2014-09-12/home-depot-didnt-encrypt-credit-card-data-former-workers-say

[4] Elgin, B., Riley, M., and Lawrence, D. (2014, September 18). "Home Depot Hacked After Months of Security Warnings". Retrieved January 11, 2015, from http://www.businessweek.com/articles/2014-09-18/home-depot-hacked-wide-open#p1

[5] Herzog, A., Shahmehri, N., and Duma, C. (2007). "An Ontology of Information Security". International Journal of Information Security and Privacy, 1-23.

[6] Krebs, B. (2014, February 12). "Email Attack on Vendor Set Up Breach at Target". Retrieved February 2, 2015, from http://krebsonsecurity.com/2014/02/email-attack-on-vendor-set-up-breach-at-target/

[7] Krebs, B. (2014, September 7). "Krebs on Security". Retrieved January 11, 2015, from https://krebsonsecurity.com/2014/09/home-depot-hit-by-same-malware-as-target/

[8] Kurose, J., and Ross, K. (2013). *Computer networking: A top-down approach featuring the Internet* (6th ed., p.516). Saddle River, NJ: Pearson.

[9] Lawson, R. (2014, July 10). From Brick to Brilliance: It's Time for the Next Generation Firewall. Retrieved January 11, 2015, from http://www.securityweek.com/brick-brilliance-its-time-next-generation-firewall

[10] McDonald, N. (n.d.). "PAST, PRESENT, AND FUTURE METHODS OF CRYPTOGRAPHY AND DATA ENCRYPTION". Retrieved January 11, 2015, from http://www.eng.utah.edu/ nmcdonal/Tutorials/EncryptionResearchReview.pdf

[11] Mick, J. (2014, September 8). "Appalling Negligence: Decade-Old Windows XPe Holes Led to Home Depot Hack". Retrieved January 11, 2015, from http://www.dailytech.com/Appalling Negligence DecadeOld Windows XPe Holes Led to Home Depot Hack/article36517.htm

[12] Network Security Types of attacks. (n.d.). Retrieved January 11, 2015, from http://computernetworkingnotes.com/network-security-access-lists-standards-and-extended/types-of-attack.html

[13] Patrizio, A. (2014, September 18). "Home Depot, Target breaches exploited Windows XP flaw, report says". Retrieved January 11, 2015, from http://www.networkworld.com/article/2685295/microsoft-subnet/home-depot-target-breaches-exploited-windows-xp-flaw-report-says.html

[14] Perlroth, N. (2014, December 4). "Banks' Lawsuits Against Target for Losses Related to Hacking Can Continue". Retrieved February 2, 2015, from http://bits.blogs.nytimes.com/2014/12/04/banks-lawsuits-against-target-for-losses-related-to-hacking-can-continue/

[15] Rath, A. (2014, January 16). "PoS Malware Targeted Target - Seculert Blog on Breach Detection". Retrieved February 2, 2015, from http://www.seculert.com/blog/2014/01/pos-malware-targeted-target.html

[16] Riley, M., Elgin, B., Lawrence, D., and Matlack, C. (2014, March 13). "Target Missed Warnings in Epic Hack of Credit Card Data". Retrieved February 2, 2015, from http://www.bloomberg.com/bw/articles/2014-03-13/target-missed-alarms-in-epic-hack-of-credit-card-data#p2

[17] Target Confirms Unauthorized Access to Payment Card Data in U.S. Stores — Target Corporate. (2013, December 19). Retrieved February 2, 2015, from http://pressroom.target.com/news/target-confirms-unauthorized-access-to-payment-card-data-in-u-s-stores

[18] Vijayan, J. (2014, March 5). "Network firewalls aren't dead yet". Retrieved January 11, 2015, from http://www.computerworld.com/article/2488179/endpoint-security/network-firewalls-aren-t-dead-yet.html

[19] Winter, M. (2014, November 7). "Home Depot hackers used vendor log-on to steal data, e-mails". Retrieved January 11, 2015, from http://www.usatoday.com/story/money/business/2014/11/06/home-depot-hackers-stolen-data/18613167/

Domain Based Certification and Revocation

Xinli Wang
School of Technology
Michigan Technological University
Houghton, Michigan 49931, USA
xinlwang@mtu.edu

Yan Bai
Institute of Technology
University of Washington Tacoma
Tacoma, WA 98402, USA
yanb@uw.edu

Lihui Hu
Department of Computer Science
Michigan Technological University
Houghton, Michigan 49931, USA
lhu@mtu.edu

***Abstract*—Certificate Authorities (CAs) are considered as a single point of failure in the design of Public Key Infrastructure (PKI). Adversaries can take the advantage of a compromised CA to issue certificates for any domains without being noticed by the domain owners. Another argument regarding PKI is the adoption of Certificate Revocation List (CRL) and Online Certificate Status Protocol (OCSP) for publishing revoked certificates. It is difficult to synchronize related databases and provide a positive answer for certificate verification. Finally, the requirement of a globally unique name to identify a subject can be a fundamental flaw in applications. In response, many approaches have been proposed to use logs of certificates and CA history for certificate verification.**

In this study, we propose an alternative approach to limit the damage of a breached CA and improve the performance of certificate revocation with domain-based certification and revocation. With our approach, a unique CA is set up for each individual domain and is responsible for issuing and managing certificates for its domain. Information of certificate issuance and revocation is maintained locally at the corresponding CA to facilitate certificate verification and management. A subject is named with a local identifier and its domain. With this naming scheme, the requirement for a globally unique name can be resolved.

***Keywords*-Public Key Infrastructure; PKI; PKIX; Public Key Certificate; Certification; Revocation**

I. Introduction

PKI has been widely employed by many applications for public key distribution [1]–[4]. However, there have been concerns about its security [5]–[8]. Previous work has identified its weaknesses.

First, a compromised CA can be a single point of failure. When a breached CA is used by adversaries to issue certificates for a domain without being noticed by the domain owner, these bogus certificates can result in Man-in-the-Middle (MitM) attacks [9]–[11]. Such attacks occur because 1) there are more than 600 CAs located in more than 50 countries [12]; 2) a number of techniques have been identified to attack a CA [13]; and 3) a government agency may compel CAs to issue forged certificates for intended purposes [14]. Apparently, this inherent weakness exposes a huge vulnerability in current implementations of PKIs.

Using CRLs and OCSP to publish revoked certificates brings up another issue [15]–[18]. The delay between the time when a certificate is revoked and when this revocation is published cannot be avoided. Management of related databases is not trivial.

The third issue is the requirement of a globally unique name to identify a subject. It does not make sense for users in general [19]. Due to the fashion PKIs are currently deployed, it is impossible to have a globally unique name for an entity across multiple independent PKIs [20].

To address these issues, we propose an alternative approach with domain-based certification and revocation to limit the damage of a compromised CA and facilitate certificate verification. With our approach, a unique CA is established in each individual domain. This and only this CA can create, issue and manage certificates for its domain. A subject is identified with a local name followed by its domain name. In this paper, we describe our approach and compare it with the current design.

This paper is organized as follows. In section II, we provide a brief overview of related work on PKI design and approaches to minimizing damages of a compromised CA. A system model is described in section III to draw a boundary between a PKI and its environment. We outline the approach of domain-based certification and revocation in section IV. Integration of our proposal into current PKIs is described in section V. Finally, we conclude our work, discuss the advantages of our approach and describe our future work in section VI.

II. Related Work

In PKI practice, a *subject* is generally defined as an entity that is either connected or has access to a network. A subject is also considered as a certificate *user*, that can be either a *principal*, who is a certificate owner, or a *verifier*, who receives a certificate and verifies it.

There have been a number of different proposals to deploy a PKI, such as Pretty Good Privacy (PGP) [21], plug-and-play PKI [22], Public Key System (PKS) [23], biographic key infrastructure [24] and notary based PKI [25]. However, they are not widely implemented due to various reasons such as scalability and security concerns.

Current PKI deployments mainly implement the trustworthy-based X.509 standards (PKIX) [26], where

a globally unique name is used to identify a subject on a certificate. CAs are equivalently trusted by users and allowed to issue certificates to users in any domains without the consent of the user and the domain owner.

Two fundamental problems have been recently investigated in current PKI implementations.

First, many approaches have been proposed to mitigate the risk of a compromised CA by utilizing history or audit logs for certificate verification. Examples include notary-based approaches [27], public key and certificate pinning [28] and Sovereign Keys [29], [30]. More recently, Google leads an effort on Certificate Transparency [31], [32] in order to stop a CA from issuing a certificate for a domain without it being visible by the domain owner. With the application of publicly accessible logs, Cheval *et al.* [33] have proposed a PKI design without relying on trusted parties and approved this design using a formal method. Ryan [34] has extended certificate transparency to allow secure end-to-end email or messaging system using a PKI independent of the trustworthy of CAs. The Accountable Key Infrastructure [35] integrates a system for key revocation with an architecture for accountability of all parties through checks-and-balances among independent entities.

In addition to publicly available logs, Attack Resilient Public-Key Infrastructure (ARPKI) allows certificates with multiple signatures to make them attack resilient [36]. PolicCert [37] grants domain owners more control over how their certificates are used and verified by specifying detailed policies on it.

The DNS-Based Authentication of Named Entities (DANE) embeds certificate information into a record of Domain Name System (DNS) [38]–[40]. A client can receive authentication data from a DNS record. The operation of DANE relies on the deployment of Secure DNS (DNSSec). The authors of CAge [41] propose to restrict the scope of top-level domains a CA can issue certificates for.

However, Grant [42] has analyzed most of the approaches with a set of well designed criteria and concluded that few of them have a brighter promise than the existing design of PKIX.

Secondly, researchers have argued recently that the requirement of a globally unique name to identify a subject on an X.509 certificate is a fundamental flaw [20] due to the following observations:

- In practice, multiple PKIs are deployed. Globally unique names do not exist in multiple deployments of independent PKIs.
- Global names contract the human use of names, since the exact meaning of a name depends on the local domain.
- The scheme to construct a global name may not be agreed on by the issuer of certificates and their verifiers. This disagreement may create security flaws in applications.

In contrast, we have the problem of "Which John Robinson is he?" [19] if the requirement of a globally unique name is removed. A mechanism is needed to resolve this dilemma in a PKI design.

III. System Model

We describe our system model in this section. The functionality of a PKI and trust in the context of PKI are also defined.

A. PKI and Its Environment

We define a PKI as a collection of organized and networked hardware and software components, along with protocols, processes, policies and procedures that are designed to effectively and efficiently create, issue and manage public key certificates. Figure 1 depicts our system model. A principal registers with and receives a certificate from a CA. A verifier contacts the CA to verify the certificate. The processes of user registration and certificate request, delivery and verification outline the interactions between a PKI and its users.

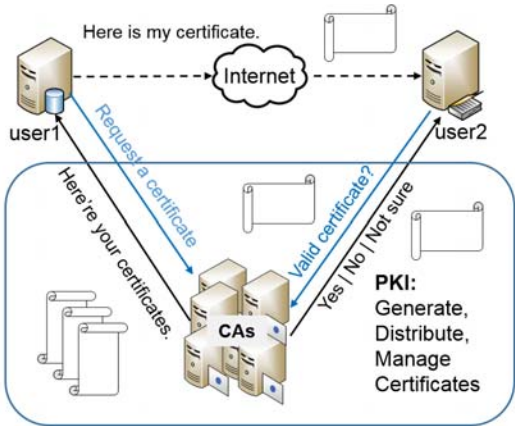


Figure 1. A diagram of the proposed system model

In practice, the communications between a PKI and its users can be done through various media beyond the Internet. Examples include physical delivery of certificates, off-line registration and certificate revocation.

B. Functionality

The functionality of a PKI can be classified into the following three categories:

- **Certificate Creation:** Upon requests from principals, certificates are generated by CAs with a chosen cryptosystem. Before a certificate can be created, a data structure must be defined to represent a certificate, including the data items that should be included in a certificate. A naming scheme must be developed to name a principal and map a public key to this principal. Furthermore, a digital signature on a certificate is usually employed to ensure the authenticity of certificate

contents. In addition to the cryptographic algorithm of the digital signature, a procedure to sign the data structure needs to be defined as well.

- **Certificate Issuance:** A CA certifies that the mapping from a public key to a principal on a certificate is correct and signs it. Then, the signed certificate is delivered to its owner.
- **Certificate Management:** The goal of certificate management is to ensure its validity. Main tasks include certificate revocation and status verification.

Other factors such as ease of use and cost efficiency must be considered to implement the functionality of a PKI.

C. Trust

In the domain of PKI research and practice, a CA is commonly recognized as a trusted entity. However, the interpretation of trust is rather ambiguous [1], [2], [43]–[46]. For example, it is unclear in what we trust on CAs [19]. It has been argued in some proposals that trusted entities are not needed to build a PKI [33], [47], [48].

To clarify the concept and its connotation, we define *trust* as a firm belief in the competence and honesty of an entity to function correctly as specified. More specifically, trusting a CA declares our firm belief that 1) this CA is capable of performing the functionality defined in subsection III-B correctly, and 2) it conducts its tasks honestly.

Our trust definition clearly specifies the technical capacity and moral behaviors of a CA. Violation of any of these breaks a trust relationship.

By this definition, we argue that some sort of trust is needed to distribute certificates on large scales. Establishing a trust relationship between two parties is not trivial. It involves technical considerations and non-technical factors. Laws, regulations and service agreements may be demanded to establish and maintain the trust.

IV. Domain Based Certification and Revocation

Based on the system model and functionality described in section III, we propose an improvement to the current PKI design with domain-based certification and revocation. With our approach, a unique CA is built in each domain. This CA is referenced as a *domain CA*. A domain CA and only a domain CA can create, issue and manage certificates for its domain. A framework of our approach is described in this section.

A. The Certificate

We classify the data items contained in a certificate into two categories as described below:

- **Operational Fields:** Values in operational fields are used in PKI functional operations. For example, the name of a principal is used to identify the owner of a certificate. The name of issuer is used to locate the server that issued this certificate. Operational fields are required to create a certificate in order to satisfy the need for interoperability of certificates from different PKIs.
- **Optional Fields:** Values in optional fields are used to include additional information. They also make a certificate design flexible and extensible. Values in optional fields should not be used for the purpose of PKI functional operation since they are not required. Extensions in an X.509 v3 certificate are good examples of optional fields. In practice, some optional fields can be required to meet the need of an application. However, this requirement should be specified in the application design.

We adopt a data structure similar to X.509 v3 certificate [26] in order to accommodate current deployments with minimum modifications. However, we propose to identify a principal with a name of two parts: a local identifier (or local name) and a domain name. The domain name is determined by the domain CA that has issued this certificate. It also indicates the relationship between the principal and the CA. The existing registered domain names in the operation of DNS can be used for this purpose. Local identifiers are given by local administrators and meaningful locally. They can be constructed in different ways according to the naming scheme adopted in the associated domain. No matter how local identifiers are determined, they must be unique in their local domain. The combination of a locally unique identifier and a globally unique domain name gives a subject a name that is globally unique. Apparently, local identifiers are not necessarily unique across different domains. In order to make a certificate more readable to human beings, a common name and organization can be included as optional fields in the data structure of a certificate.

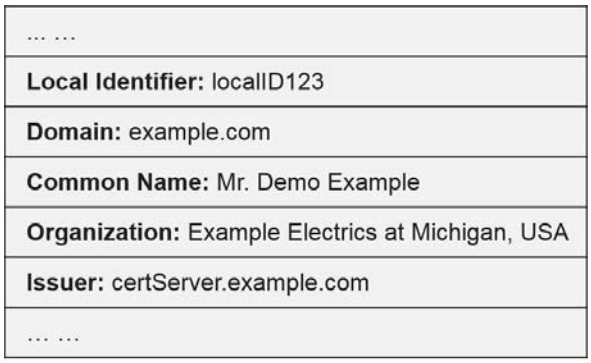


Figure 2. Sample fields on a proposed certificate

Figure 2 shows sample fields on a certificate with our approach. The certificate shows that it is owned by *localID123* in the domain of *example.com*. The combination of the local identifier and its associated domain forms a *fully qualified name* that can be easily located on the Internet with the assistance of DNS. The fields of *Common Name*

and *Organization* are used to expose more principal-related information to verifiers. The value of *Issuer* indicates the server that issues this certificate. It also gives the location of the domain CA that a verifier should contact for verification purpose.

There are other operational and optional fields that are described in the document of X.509 certificate [26]. They are not repeated here.

B. Certificate Creation, Issuance and Revocation

A domain CA must be built in order for the domain to create and issue certificates that are publicly verifiable. In an X.509 v3 certificate [26], a certificate can be used to issue other certificates if the *cA* bit in the extension field of *Basic Constraints* and the *keyCertSign* bit in the extension field of *Key Usage* are asserted. We reference such a certificate as a *CA certificate*. Apparently, we can define the aforementioned extension fields as operational fields to facilitate the PKI operation of our approach. A domain CA can be built by obtaining a CA certificate for a server from an existing root CA or one of its subordinate CAs.

Supposedly, obtaining a CA certificate must follow a set of rules for security considerations. The server for a domain CA must be carefully hardened.

The process of certificate creation and issuance is straightforward when a domain CA is set up. Certificates for a domain are created, signed and issued by the domain CA with chosen software tools.

We argue that our approach has the following benefits:

- The effort for users to register with a domain CA will be minimized. Usually, users (human users and computing devices) are physically located on the same local area network (LAN) as the domain CA. They may be with the same Internet Service Provider (ISP). Before a certificate is issued, domain administrators are able to make physical checks on user's identification to ensure that the name on a certificate is not forged in the first place. In addition, existing databases such as the database of an existing directory service can be used to enroll principals for certificate issuance [49], [50]. Additional schema and attribute variables can be defined to meet the need for certificate creation and issuance.
- Certificate delivery is simplified since all users are from the same domain. Physical delivery is one option. Centralized access and management is another option because the users are logically local in the same domain.
- Certificates are issued with a minimum cost or even free of charge. Short-lived certificates can be used for the purpose of authentication and encryption, which will decrease the need for a check on certificate revocation [15], [51]. This can be beneficial for certificate verification.
- In the perspective of IT (Information Technology) management, our approach has the benefits of ease of use and cost-efficiency. Policies for certificate creation, revocation and utilization can be defined within a domain. IT staff for a domain can determine whether to manage their certificates in a centralized approach or a distributed fashion.

To facilitate certificate verification and management, each domain CA maintains a database that contains relevant information of the issued certificates. This database is referenced as a *certificate database*. An entry is added to the database when a new certificate is created and issued.

In addition, a *revocation database* is set up and maintained by a domain CA. When a certificate, which is issued by this domain CA, needs to be revoked due to some reason, an entry is added to the database specifying the revoked certificate.

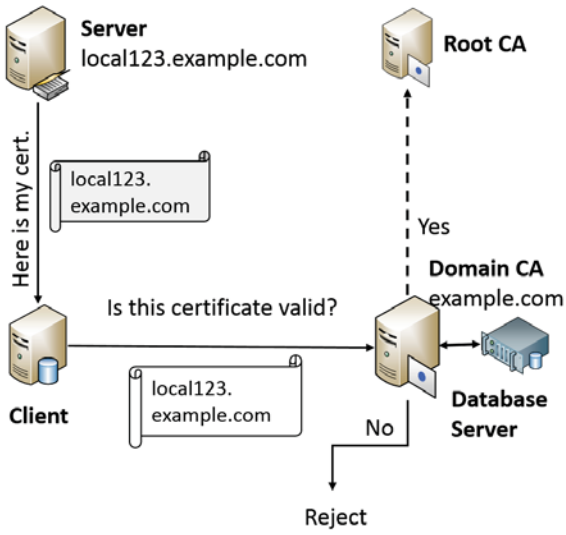


Figure 3. Process of certificate verification

C. Certificate Verification and Management

The process of certificate verification starts with the domain CA that issued the certificate. As shown in Figure 3, when an application server such as a web server (top-left corner) hands out its certificate to a client (bottom-left corner) in protocol negotiation, the client contacts the domain CA (bottom-right corner) for certificate verification. Since the domain CA has the information regarding all of the certificates it has issued and those it has revoked, it is able to verify whether this certificate is valid or not. For example, if a certificate has an entry in its certificate database, but no entry in its revocation database, this certificate is valid. Otherwise (either revoked or not issued by this domain CA), the certificate is invalid. The client should reject an invalid certificate. When the validity of the certificate is asserted by

the domain CA, the verification process goes further through a trust-chain to a root CA (top-right corner).

If we require every CA, including domain CAs, root CAs and the CAs in between (delegated CAs), maintain a certificate database and a revocation database, certificate verification at each CA goes through the same procedure as that at a domain CA. This will remove the need to establish CRL databases or OCSP servers.

V. Integration with Current PKIs

The following modifications need to be made in order for the design of current PKIs to accommodate our approach:

- A minor change to the certificate definition is needed to use a local identifier combined with a domain name to uniquely identify a subject.
- The procedure for certificate verification needs to be modified to incorporate the changes described in section IV-C.
- The functionality of a domain CA is different from current CAs. In addition to creating, issuing and managing certificates for its domain, a domain CA needs to maintain databases for certificates it has issued and revoked. It answers queries from clients for certificate verification.

With those modifications, a domain CA can issue and manage certificates for its domain.

VI. Conclusion, Discussion and Future Work

In this paper, we have presented a system model to define the environment a PKI works in and the interactions between a PKI and its users. The functionality of a PKI has been outlined under this model. With the definition of trust and its connotation, we argue that certain sort of trust is needed to issue and manage public key certificates on large scales. We have then proposed an improved PKI design with domain-based certification and revocation. In this design, a subject is identified with a local name followed by a domain name. A domain CA is built to create, issue and manage certificates for its domain users. The domain CA also establishes and maintains a certificate database for all of the certificates it has issued and a revocation database for the revoked certificates in this domain. With these two collocated databases, certificate verification is straightforward and effective.

If we command that a CA cannot issue certificates for domains other than its own domain, domain-based certification effectively limits the damage of a breached CA since this CA can issue certificates for its domain only. This is particularly important when governments compel CAs to issue forged certificates for state level MitM attacks [14]. Our design adds another layer of protection for compromised CAs. Since a certificate database is maintained by a domain CA, a forged certificate will be rejected immediately at the domain CA in the process of certificate verification.

Certificate verification is more effective and efficient compared with that in current PKIs. Data regarding the status of certificates is naturally distributed over the Internet. It is not necessary to maintain separate databases for CRLs and OCSP operations. Clearly, our approach resolves the issue of synchronization of revocation-related databases and provision of an assured certificate verification in a timely manner.

Apparently, a name of two parts in our approach solves the issue of a globally unique name in current PKI practice, while the local identifier provides a meaningful interpretation for a local subject.

Finally, our design facilitates the management of certificates in a domain. Since all of the certificates are created and issued locally, various databases can be established to store and manage certificates [52]. Existing authentication mechanisms, such as directory services and domain authentication, can be used by principals to access their certificates. Face-to-face check for registration and physical delivery of critical certificates by IT staff are a possible option.

We plan to look into the scalability of domain-based certification and coexistence of domain-based CAs with conventional CAs. An implementation and performance evaluation of domain-based certification and revocation is underway.

Acknowledgment

We would like to thank the anonymous reviewers. Their comments are very helpful for our future work. The National Science Foundations (NSF) support via TUES grants (Award#: 1140310 and Award#: 1140308) is also gratefully acknowledged.

References

[1] M. Pala and S. A. Rea, "Usable trust anchor management," in *Proceedings of the 8th Symposium on Identity and Trust on the Internet*, ser. IDtrust '09. New York, NY, USA: ACM, 2009, pp. 61–72. [Online]. Available: http://doi.acm.org/10.1145/1527017.1527025

[2] C. Wallace and G. Beier, "Practical and secure trust anchor management and usage," in *Proceedings of the 9th Symposium on Identity and Trust on the Internet*, ser. IDTRUST '10. New York, NY, USA: ACM, 2010, pp. 97–107. [Online]. Available: http://doi.acm.org/10.1145/1750389.1750403

[3] J. Buchmann, E. Karatsiolis, and A. Wiesmaier, "PKI in practice," in *Introduction to Public Key Infrastructures*. Springer Berlin Heidelberg, 2013, pp. 143–164. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-40657-7_10

[4] D. Hunt and W. A. Al-Hamdani, "Theoretical analysis of using identity based PKI as the authentication method in SQL," in *Proceedings of the 2013 on InfoSecCD '13: Information Security Curriculum Development Conference*, ser. InfoSecCD '13. New York, NY, USA: ACM, 2013, pp. 33:33–33:41. [Online]. Available: http://doi.acm.org/10.1145/2528908.2528915

[5] E. Gerck, "Overview of certificate systems: X.509, PKIX, CA, PGP & SKIP," *The Bell*, vol. 1, no. 3, p. 8, Jul. July 2000. [Online]. Available: http://www.thebell.net/papers/

[6] P. Gutman, "PKI: It's not dead, just resting," *IEEE Computer*, vol. 35, no. 8, pp. 41–49, 2002.

[7] A. Lioy, M. Marian, N. Moltchanova, and M. Pala, "PKI past, present and future," *International Journal of Information Security*, vol. 5, no. 1, pp. 18–29, 2006.

[8] J. Clark and P. van Oorschot, "SoK: SSL and HTTPS: Revisiting past challenges and evaluating certificate trust model enhancements," in *Security and Privacy (SP), 2013 IEEE Symposium on*, May 2013, pp. 511–525.

[9] S. Bhat, "Gmail users in Iran hit by MITM attacks," Online, August 2011, http://techie-buzz.com/tech-news/gmail-iran-hit-mitm.html, Retrieved on December 3, 2014.

[10] P. Roberts, "Phony SSL certificates issued for Google, Yahoo, Skype, others," threat post, online, March 2011.

[11] C. Wisniewski, "Turkish certificate authority screwup leads to attempted Google impersonation," naked security, online, January 2013.

[12] Electronic Frontier Foundation, "The EFF SSL observatory," online, https://www.eff.org/observatory, last retrieved on December 21, 2014.

[13] P. Eckersley, "How secure is HTTPS today? How often is it attacked?" online, October 2011, https://www.eff.org/deeplinks/2011/10/how-secure-https-today, last retrieved on December 22, 2014.

[14] C. Soghoian and S. Stamm, "Certified lies: Detecting and defeating government interception attacks against SSL (short paper)," in *Financial Cryptography and Data Security*, ser. Lecture Notes in Computer Science, G. Danezis, Ed. Springer Berlin Heidelberg, 2012, vol. 7035, pp. 250–259. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-27576-0_20

[15] R. L. Rivest, "Can we eliminate certificate revocations lists?" in *Proceedings of the Second International Conference on Financial Cryptography*, ser. FC '98. London, UK, UK: Springer-Verlag, 1998, pp. 178–183. [Online]. Available: http://dl.acm.org/citation.cfm?id=647502.728327

[16] C. A. Gunter and T. Jim, "Generalized certificate revocation," in *Proceedings of the 27th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, ser. POPL '00. New York, NY, USA: ACM, 2000, pp. 316–329. [Online]. Available: http://doi.acm.org/10.1145/325694.325736

[17] S. Misra, S. Goswami, G. Pathak, N. Shah, and I. Woungang, "Geographic server distribution model for key revocation," *Telecommunication Systems*, vol. 44, no. 3-4, pp. 281–295, 2010. [Online]. Available: http://dx.doi.org/10.1007/s11235-009-9254-x

[18] M. Ingle and M. Kumar, "Comparative analysis of methods for distribution of certificate revocation information in mobile environment," in *Communication Systems and Network Technologies (CSNT), 2011 International Conference on*, June 2011, pp. 166–169.

[19] C. Ellison and B. Schneier, "Ten risks of PKI: What you're not being told about public key infrastructure," *Computer Security Journal*, vol. 16, no. 1, pp. 1–7, 2000.

[20] S. Wiesner, "Simple PKI," *Innovative Internet Technologies and Mobile Communications (IITM)*, vol. 83, 2013.

[21] J. Callas, L. Donnerhacke, H. Finney, D. Shaw, and R. Thayer, "OpenPGP message format," Internet Requests for Comments, RFC Editor, RFC 4880, November 2007. [Online]. Available: http://tools.ietf.org/html/rfc4880

[22] P. Gutmann, "Plug-and-play PKI: a PKI your mother can use," in *Proceedings of the 12th Conference on USENIX Security Symposium - Volume 12*, ser. SSYM'03. Berkeley, CA, USA: USENIX Association, 2003, pp. 45–58. [Online]. Available: http://dl.acm.org/citation.cfm?id=1251353.1251357

[23] M. Pala, "A proposal for collaborative internet-scale trust infrastructures deployment: the public key system (PKS)," in *IDtrust*, 2010, pp. 108–116.

[24] W. Scheirer, W. Bishop, and T. Boult, "Beyond PKI: The biocryptographic key infrastructure," in *Security and Privacy in Biometrics*, P. Campisi, Ed. Springer London, 2013, pp. 45–68. [Online]. Available: http://dx.doi.org/10.1007/978-1-4471-5230-9_3

[25] M. A. Vigil, C. Moecke, R. CustÃşdio, and M. Volkamer, "The notary based PKI," in *Public Key Infrastructures, Services and Applications*, ser. Lecture Notes in Computer Science, S. Capitani di Vimercati and C. Mitchell, Eds. Springer Berlin Heidelberg, 2013, vol. 7868, pp. 85–97. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-40012-4_6

[26] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk, "Internet X.509 public key infrastructure certificate and certificate revocation list (CRL) profile," Internet Requests for Comments, RFC Editor, RFC 5280, May 2008. [Online]. Available: http://tools.ietf.org/html/rfc5280

[27] Perspectives Project, "What is perspectives?" online, http://perspectives-project.org/, last retrieved on December 20, 2014.

[28] C. Evans, C. Palmer, and R. Sleevi, "Public key pinning extension for HTTP," Internet-Draft, October 2014. [Online]. Available: https://tools.ietf.org/html/draft-ietf-websec-key-pinning-21

[29] P. Eckersley, "Sovereign key cryptography for internet domains," online, 2012, https://git.eff.org/?p=sovereign-keys.git;a=blob_plain;f=sovereign-key-design.txt;hb=master, last retrieved on December 21, 2014.

[30] Electronic Frontier Foundation, "The sovereign keys project," online, 2011, https://www.eff.org/sovereign-keys, last retrieved on December 21, 2014.

[31] B. Laurie, A. Langley, and E. Kasper, "Certificate transparency," Internet Requests for Comments, RFC Editor, RFC 6962, June 2013. [Online]. Available: https://tools.ietf.org/html/rfc6962

[32] B. Laurie, "Certificate transparency," *Communications of The ACM*, vol. 57, no. 10, pp. 40–46, Sep. 2014. [Online]. Available: http://doi.acm.org/10.1145/2659897

[33] V. Cheval, M. Ryan, and J. Yu, "DTKI: a new formalized PKI with no trusted parties," *CoRR*, vol. abs/1408.1023, 2014. [Online]. Available: http://arxiv.org/abs/1408.1023

[34] M. D. Ryan, "Enhanced certificate transparency and end-to-end encrypted mail," in *Proceedings of Network and Distributed System Security (NDSS)*. NDSS, February 2014.

[35] T. H.-J. Kim, L.-S. Huang, A. Perring, C. Jackson, and V. Gligor, "Accountable key infrastructure (AKI): A proposal for a public-key validation infrastructure," in *Proceedings of the 22Nd International Conference on World Wide Web*, ser. WWW '13. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2013, pp. 679–690. [Online]. Available: http://dl.acm.org/citation.cfm?id=2488388.2488448

[36] D. Basin, C. Cremers, T. H.-J. Kim, A. Perrig, R. Sasse, and P. Szalachowski, "ARPKI: Attack resilient public-key infrastructure," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '14. New York, NY, USA: ACM, 2014, pp. 382–393. [Online]. Available: http://doi.acm.org/10.1145/2660267.2660298

[37] P. Szalachowski, S. Matsumoto, and A. Perrig, "PoliCert: Secure and flexible TLS certificate management," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '14. New York, NY, USA: ACM, 2014, pp. 406–417. [Online]. Available: http://doi.acm.org/10.1145/2660267.2660355

[38] P. Hoffman and J. Schlyter, "The DNS-based authentication of named entities (DANE) – transport layer security (TLS) protocol: TLSA," Internet Requests for Comments, RFC Editor, RFC 6698, August 2012, https://tools.ietf.org/html/rfc6698, last retrieved on 3/12/2015.

[39] O. Gudmundsson, "Adding acronyms to simplify conversations about DNS-based authentication of named entities (DANE)," Internet Requests for Comments, RFC Editor, RFC 7218, April 2014, https://tools.ietf.org/html/rfc7218, last retrieved on 3/12/2015.

[40] P. Hoffman and J. Schlyter, "Using secure DNS to associate certificates with domain names for S/MIME," Internet Requests for Comments, RFC Editor, RFC, February 2015, https://datatracker.ietf.org/doc/draft-ietf-dane-smime/?include_text=1, last retrieved on 3/12/2015.

[41] J. Kasten, E. Wustrow, and J. Halderman, "CAge: Taming certificate authorities by inferring restricted scopes," in *Financial Cryptography and Data Security*, ser. Lecture Notes in Computer Science, A.-R. Sadeghi, Ed. Springer Berlin Heidelberg, 2013, vol. 7859, pp. 329–337. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-39884-1_28

[42] A. C. Grant, "Search for trust: An analysis and comparison of CA system alternatives and enhancements," online, Dartmouth Computer Science, Technical Report TR2012-716, 2012. [Online]. Available: http://www.cs.dartmouth.edu/reports/TR2012-716.pdf

[43] A. Abdul-Rahman, "The PGP trust model," in *EDI-Forum: the Journal of Electronic Commerce*, vol. 10, no. 3, 1997, pp. 27–31.

[44] R. Perlman, "An overview of PKI trust models," *Network, IEEE*, vol. 13, no. 6, pp. 38–43, Nov 1999.

[45] T. Grandison and M. Sloman, "A survey of trust in internet applications," *Communications Surveys Tutorials, IEEE*, vol. 3, no. 4, pp. 2–16, Fourth 2000.

[46] P. Liss, "Trust revisited: branding and PKI," *Card Technology Today*, vol. 14, no. 4, pp. 11 – 11, 2002. [Online]. Available: "http://www.sciencedirect.com/science/article/pii/S0965259002004231

[47] C. M. Ellison, "Establishing identity without certification authorities," in *Proceedings of the 6th Conference on USENIX Security Symposium, Focusing on Applications of Cryptography - Volume 6*, ser. SSYM'96. Berkeley, CA, USA: USENIX Association, 1996, pp. 67–76. [Online]. Available: http://dl.acm.org/citation.cfm?id=1267569.1267576

[48] R. L. Rivest and B. Lampson, "SDSI-a simple distributed security infrastructure." Crypto, 1996.

[49] M. Lippert, E. Karatsiolis, A. Wiesmaier, and J. Buchmann, "Directory based registration in public key infrastructures," in *Proceedings of the 2005 conference on Applied Public Key Infrastructure: 4th International Workshop: IWAP 2005*. Amsterdam, The Netherlands, The Netherlands: IOS Press, 2005, pp. 17–32. [Online]. Available: http://dl.acm.org/citation.cfm?id=1564104.1564108

[50] M. Lippert, V. Karatsiolis, A. Wiesmaier, and J. Buchmann, "Life-cycle management of X.509 certificates based on LDAP directories," *Journal of Computer Security*, vol. 14, no. 5, pp. 419–439, Sep. 2006. [Online]. Available: http://dl.acm.org/citation.cfm?id=1239313.1239316

[51] P. McDaniel and A. D. Rubin, "A response to "can we eliminate certificate revocation lists?"," in *Proceedings of the 4th International Conference on Financial Cryptography*, ser. FC '00. London, UK, UK: Springer-Verlag, 2001, pp. 245–258. [Online]. Available: http://dl.acm.org/citation.cfm?id=647504.728491

[52] P. Gutmann, "PKI design for the real world," in *Proceedings of the 2006 workshop on New security paradigms*, ser. NSPW '06. New York, NY, USA: ACM, 2007, pp. 109–116. [Online]. Available: http://doi.acm.org/10.1145/1278940.1278958

Calculation Model of the Status and Staffing for Security Management – A Case Study

Lilian Noronha Nassif, Daniel Silva Carnevalli
Information Technology Department, Public Ministry of Minas Gerais, Belo Horizonte, Minas Gerais, Brazil
liliannassif, dcarnevalli{@mpmg.mp.br}

Abstract - *Security management involves a great variety of themes. The easiest way to make an organization more secure is by installing and appropriately configuring several security tools. Nevertheless, this is insufficient. Usually processes and methodologies are put in a second plane, allowing gaps that can be explored. However, analyzing whether an enterprise security status is adequate and if the number of security staff is sufficient remain difficult. This work presents a method to measure the security status in an organization. It also presents an analytical model with metrics to calculate the security staff size. Both models are simulated using real data collected in surveys from 28 organizations. The results are feasible and can be used as benchmark.*

Keywords: *security metrics; information security management; security auditing*

1 Introduction

Information security management is a dynamic area. Technological factors alone cannot prevent security problems. Other factors such as institutional organization, supplier interactions, and information security training of users and the Information Technology (IT) team are also key instruments to provide confidentiality, integrity, and availability of information resources.

A challenge that Chief Information Officers (CIOs) face is determining how many people are necessary to manage security issues. This decision must consider several factors such as environment complexity and security attributions.

This paper presents comprehensive analytical models to calculate the information security status in an enterprise and the security staff required. A survey with 51 questions was conducted within 28 organizations. The results can help IT leaders structure their security departments according to their main faults and compose a security team appropriately.

The paper is structured as follows: section 2 presents studies about security demands and staff sizing. Section 3 presents a case study conducted in 28 organizations. Section 4 shows our models to calculate the security status and staffing, presenting real numbers according to metrics obtained from interviews. Finally, section 5 concludes the paper.

2 Information security management and staffing metrics

Defining the information security department procedures and the number of staff to carry on such procedures are elementary aspects that concern IT leaders. The following sections discuss information security management and IT staffing metrics based on standards and surveys.

2.1 Information Security Management

The activities associated with information security management are widely discussed in IT. Standard organizations such as the International Organization for Standardization (ISO) [1], the Control Objectives for Information and related Technology (COBIT) [2], and the Information Technology Infrastructure Library (ITIL) [3] propose guidelines that can be widely applied in organizations. The focus of this paper is on the ISO 27001:2013 [4], as it is a detailed description about the suite of activities concerning information security management.

Adopting an Information Security Management System (ISMS) is a strategic decision for an organization. The ISO 27001:2013 standard was prepared to provide a process model to implement, maintain, and improve the ISMS. This standard defines 114 controls grouped in 14 domains as related below:

1. Information security policy
2. Organization of information security
3. Human resources security
4. Asset management
5. Access control
6. Cryptography
7. Physical and environmental security
8. Operations security
9. Communications security
10. System acquisition, development, and maintenance
11. Supplier relationships
12. Information security incident management
13. Information security aspects of business continuity management
14. Compliance

Although the ISO 27001:2013 standard is one of the most complete references about IT security activities, it must be

adapted to the individual institution objectives, processes, employees, size, and structure. Section 3 presents real data of these domains in 28 organizations.

2.2 IT staffing size metrics

An efficient management must adjust the staff size according to work necessities and environment reality. The following studies relate which metrics can help estimate the number of people on the information security staff.

The Computer Security Institute (CSI) estimated that an information security team is composed of 3% to 5% of an IT team [5].

The work conducted by Computer Economics in 2008 relates that an information security team corresponds to 2% of an IT team. This study refers to security teams limited to security auditing, management, developing, and policy and process implementations. This low percentage is because other groups contribute to ensure the information security at the organization, including network and system administrators, helpdesk, and other operational areas[6].

Another study made in 2003 by Deloitte Touche Tohmatsu (DTT) recommends one information security professional for each 1,000 users [6].

A study realized in a university environment by Educause, concluded that one information security professional is necessary for every 5,000 interconnected network devices [7].

Vostrom [8] presented another way of calculating the adequate number of professionals on an information security team. The analysis was made considering the time spent on each security topic. The calculation model used the concept of Full Time Equivalent (FTE) and is presented in Table 1. The FTE is a method that measures employee workload in a year contract. An FTE of 100% means the employee is a full-time worker, while an FTE of 50% means the employee is a part-time worker. Table 1 shows that 3.8 people/year in a minimal situation and 6.15 people/year in an ideal situation would be necessary to execute information security functions.

Although these studies confirm that some best practices are related to IT security team size, it is important to consider other factors such as environment complexity and the quality of the IT solutions. The next section presents a case study that identifies such questions.

Table 1: Amount of time spent per key security area. Source [8]

Security Staff Function	**Ideal % of time**	**Minimum % of time**
Audit	50%	35%
Physical Security Technologies	10%	5%
Disaster Recovery / Contingency Planning	25%	15%
Solution Investigation / Procurement	15%	5%
Security Education, Training, and Awareness	100%	75%
Personnel / Credential Issues	100%	75%
Risk Management / Planning	50%	15%
System and Network Management	100%	50%
Telecommunications Security	50%	25%
Helpdesk	15%	5%
Maintenance of Security Program	100%	75%
TOTAL	**6.15 staff years**	**3.80 staff years**

3 Case Study

The demand for information security services was identified in a case study addressed to 28 CIOs from different state government organizations. The study contained 51 questions concerning aspects of the methodology described in [9], the quantitative metrics described in section 2, the 27001:2013 standard [4], and the IT security benchmark conducted by Wisegate [10].

The study involved a total of 76,651 employees, 2,101 IT employees, and 117,036 units of interconnected equipment.

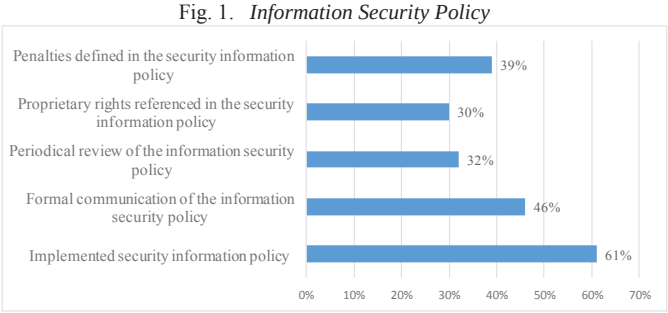
These organizations can be defined as: 57% having fewer than 2,000 employees, 56% having an IT team up to 50 employees, and 48% having fewer than 2,500 units of interconnected equipment.

The following sections 3.1 to 3.14 analyze all obtained answers according to each domain in the 27001:2013 standard.

3.1 Information Security Policy

The domain "information security policy" is presented in Figure 1. It shows that 61% of participants had an information security policy. Nevertheless, this policy was reviewed in only 32% of organizations and formally communicated to employees in only 46%. It is also possible to verify that the security policy was frequently incomplete, since 30% neglect to mention intellectual properties, and 39% include no penalties for policy violations.

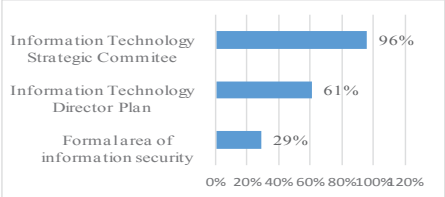
Fig. 1. *Information Security Policy*



3.2 Organization of Information security

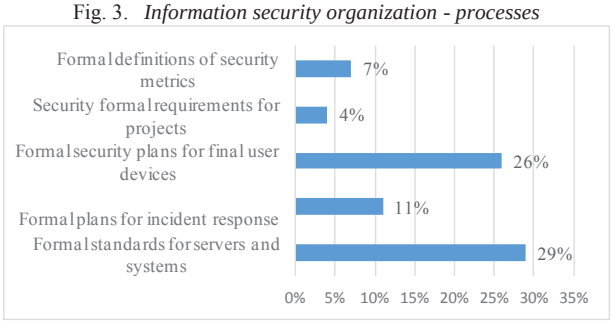
The domain "organization of information security" is presented in Figure 2. It shows that only 29% of participants had a formal information security area installed. However, 61% had an Information Technology Director Plan, and 96% had an Information Technology Strategic Committee.

Fig. 2. *Information security organization – structures*



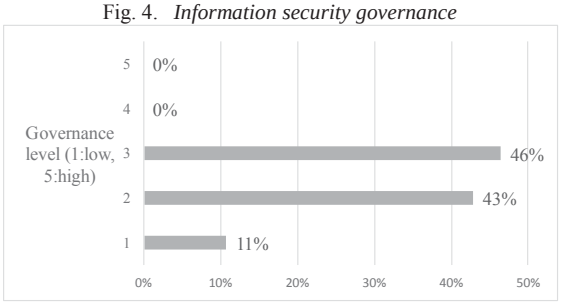
Nevertheless, Figure 3 demonstrates that only 29% of participants had formal standards for servers and systems, 11% had formal plans for incident responses, 26% had security plans for final user devices, only 4% had formal security requirements for projects, and 7% had formal definitions for security metrics.

Fig. 3. *Information security organization - processes*



According to Figure 4, 46% of participants related that the information governance level was medium (3, on a scale from 1 to 5). All participants believed that the governance was below or equal to medium level.

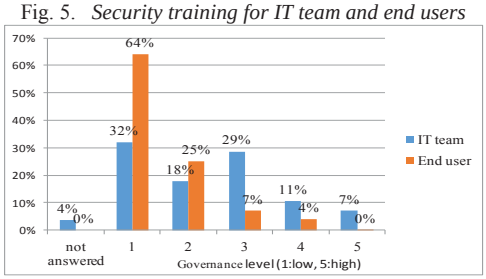
Fig. 4. *Information security governance*



3.3 Human resource security

Figure 5 presents the training level of the IT team and final users. The IT team training level in information security was below 3 for 79% of the participants. The user training level in information security was 1, for 64% of the participants.

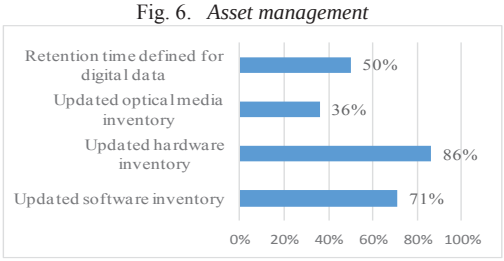
Fig. 5. *Security training for IT team and end users*



3.4 Asset management

Assets were relatively well managed. Figure 6 shows that 86% of the participants had an updated hardware inventory, 71% an updated software inventory, 36% an updated media inventory, and 50% a predetermined time to retain digital information.

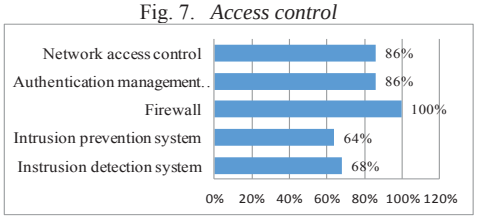
Fig. 6. *Asset management*



3.5 Access control

Access was well controlled. According to Figure 7, all participants had a firewall, 86% had an authentication system, and a network access control system, 68% had an intrusion detection system, and 64% had an intrusion prevention system.

Fig. 7. *Access control*



3.6 Cryptography

Cryptography here is associated with data transmission and storage. Figure 8 verifies the risk level of confidentiality, integrity, and availability in the environment. The risk level was 3 for 39% of participants, and 89% believed this risk level was low to medium.

Fig. 8. *Risk of confidentiality, integrity, and availability*

3.7 Physical and environmental security

Figure 9 presents aspects of physical and environment security in IT. In datacenters or server rooms, an Uninterruptible Power Supply (UPS) was installed in 96%, a fire protection system in 89%, restricted access in 79%, suspended floors in 68%, and temperature, dust, and humidity control in 57%.

Fig. 9. *Physical and environment security in datacenters*

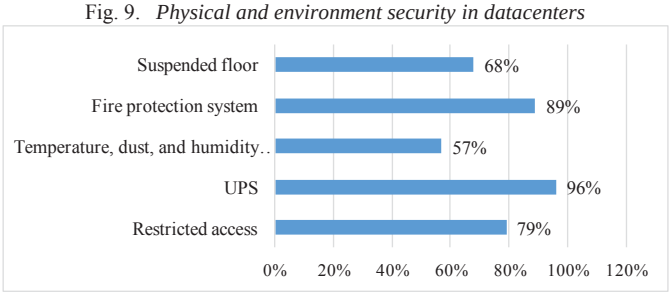
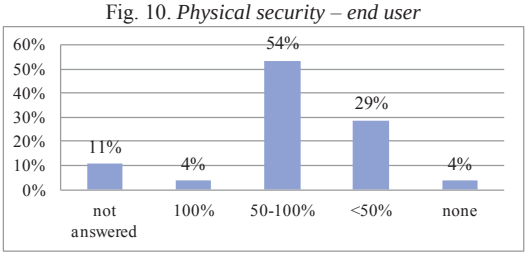


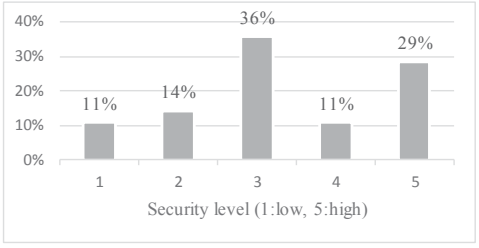
Figure 10 shows that 54% of IT leaders evaluated the physical security level for end user access between 50 to 100%, and 29% evaluated it under 50%.

Fig. 10. *Physical security – end user*



The security level of hosting environments varies from medium to high for 75% of the participants as Figure 11 demonstrates.

Fig. 11. *Server hosting*



3.8 Operations security

Figure 12 presents a list of implemented systems and processes. Backup procedures were documented by 86% of the participants; the backup was retained accordingly to the institution definition by 75% of the participants; and the backup was restored periodically by 71% of the participants. Firewall and application logs were maintained by 89% of the participants.

Fig. 12. *Operation security – installed solutions*

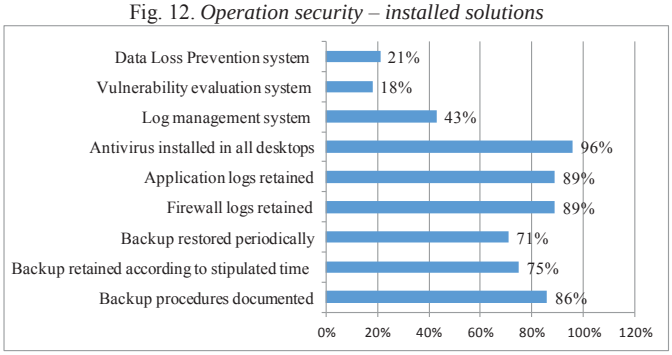
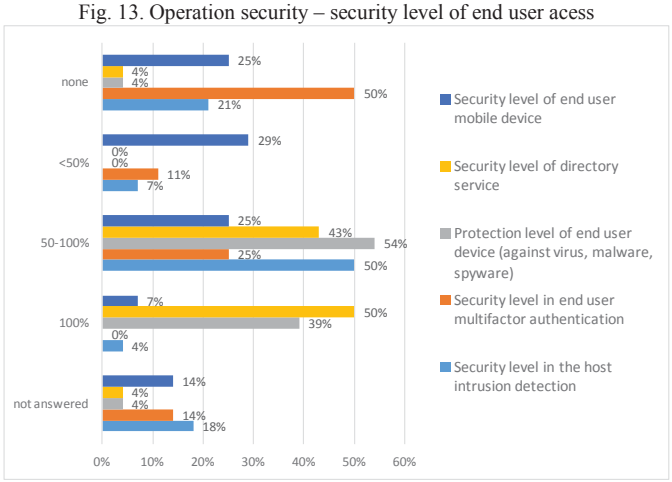


Figure 12 also shows that 96% of the participants had antivirus installed in all desktops, 43% had a log management system, 18% a vulnerability evaluation system, and 18% data loss prevention software.

Figure 13 demonstrates that from all items analyzed, the directory service had the highest trust index (100%) for half of the participants. The graph has the highest concentration in the group of 50-100% of security level.

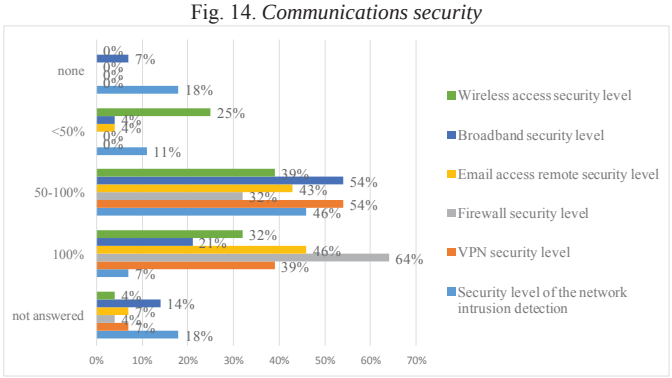
Fig. 13. Operation security – security level of end user acess



3.9 Communications security

Communications security is presented in Figure 14 under several factors. The confidence level in this service concentrates at the 50-100% range and 100%. The security level is 100% for the firewall functions for 64% of the interviewed. For 46% of the interviewed, the confidence level is 100% for email remote access.

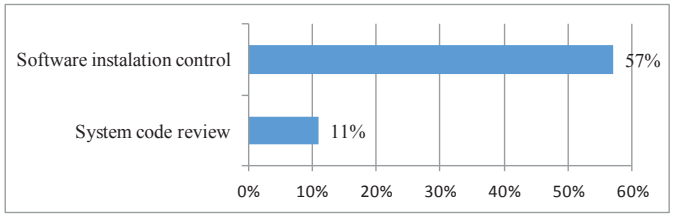
Fig. 14. *Communications security*



3.10 System acquisition, development, and maintenance

Figure 15 shows that 57% of the participants had a mechanism that avoided or controlled the installation of non-authorized software. Only 11% had code revision software.

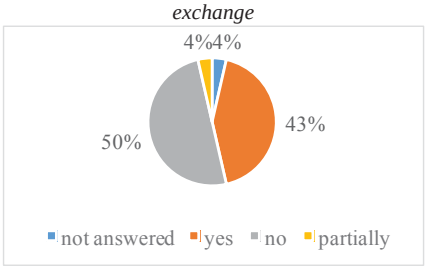
Fig. 15. *Installation control and code review*



3.11 Supplier relationships

Figure 16 demonstrates that 43% of the participants had formal third party contracts that established requirements for electronic data exchange using the internet.

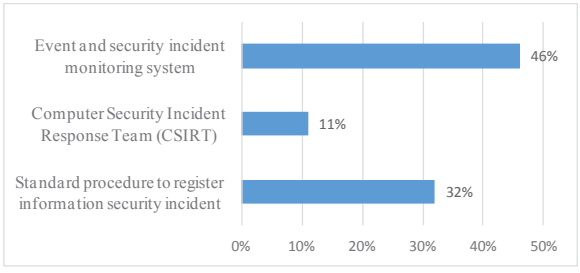
Fig. 16. *Formal third party contracts that established eletronic data exchange*



3.12 Information security incident management

Information security incident management was poorly evaluated by the interviewed. According to Figure 17, only 11% of the participants had a Computer Security Incident Response Team (CSIRT). Institutions had a standard procedure to register security incidents for 32% of the participants, and 46% had an incident and event monitoring system.

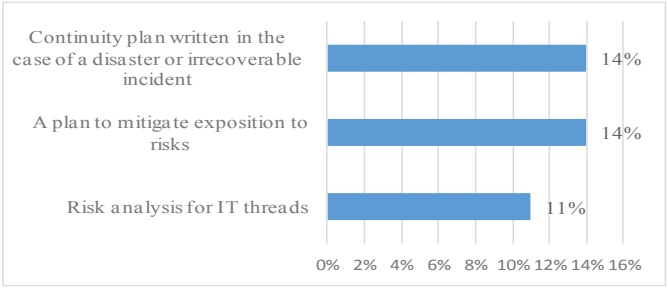
Fig. 17. *Information security incident management*



3.13 Information security aspects of business continuity management

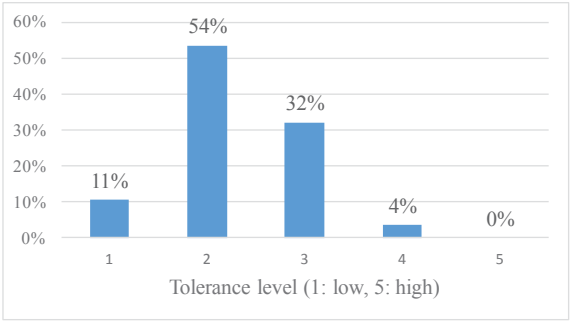
The information security aspects of business continuity management were poorly evaluated as well. Figure 18 shows that only 11% of the participants developed IT risk analysis. A small percentage (14%) of participants had a continuity plan written in the case of a disaster or irrecoverable incident that results in an inoperable IT environment. Only 14% had a plan to mitigate risk exposition.

Fig. 18. *Business continuity management*



The level of risk tolerance was poorly evaluated. In Figure 19, most of the interviewed, 54%, stated that the risk tolerance level was 2 (1:low, 5:high).

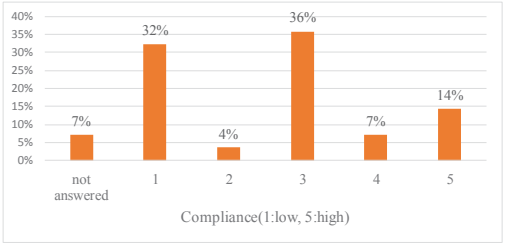
Fig. 19. *Tolerance to security risks*



3.14 Compliance

The security requirements were in high compliance to legislation, contracts, and security policy for only 14% of the participants. Figure 20 shows that 36% believed that the adherence was reasonable and indicated level 3. Nevertheless, 72% of the participants alleged that adherence was low to medium.

Fig. 20. *Compliance to legislation, contracts, and information security policy*



Vulnerability analysis verifies if the security requirements implemented are compliant to the information security policy. 50% of participants had already conducted a vulnerability analysis using a penetration test.

4 Status and staff calculation

Information security can be improved constantly if metrics are used to identify the real management state. We developed here two analytical models to calculate the security status and the staff size. Both used real data collected in the case study described in section 3.

4.1 Status calculation

The calculation of security status was based on the positive responses obtained for each domain of ISO 27001:2013. A positive response depended on the type of question used in the survey and is denoted as *positive_domain*, where:

- For questions whose answers were "yes or no", *positive_domain* is the percentage of equal answers to "yes"
- For questions whose answers were to inform "level 1-5", *positive_domain* is the percentage of responses between "3-5"
- For questions whose answers were to inform percentage, *positive_domain* is the percentage of responses for "50-100% and 100%"

Figure 21 presents the *positive_domain* calculations for all domains considered in section 3. The security status, denoted as *SecStatus* calculated the average of all *positive_domains* and is denoted by Equation (1). According to values presented in Figure 21 and Equation (1), the security status of the survey participants was 51%.

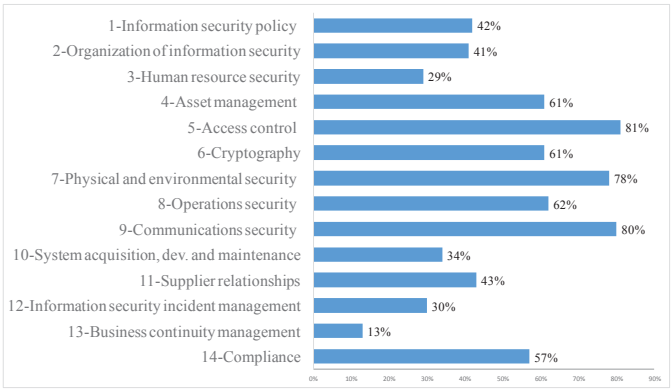
$$SecStatus = \Sigma(positive\_domain)/14 \quad (1)$$

It is important to note that the participants had good information security management for topics related to operational functions such as asset management, communication security, security operations, and physical security.

However, participants had insufficient information security management in the fields related to policy and information security procedures, human resources management, incident management, business continuity management, and compliance with legal and contractual requirements.

Considering that only 29% of the CIOs answered that they had a dedicated team to information security (Figure 2), these findings confirmed, only in this survey, that even without a dedicated information security team, operational activities were performed reasonably well with the existing team. However, the procedural and regulatory issues became impaired and pointed to the need for improvements in these themes.

Fig. 21. *Security status according to ISO 27001:2013 domains*



4.2 Staffing calculation

The staff number of the IT security team was based on several literature studies presented in section 2.2. The study results can be summarized in the following recommendations: 1) Allocate 1 IT security professional per 1,000 users; 2) Allocate 1 person in the IT security team per 5,000 interconnected devices; 3) Allocate 3% of IT staff to the security team; 4) Allocate 3.8 to 6.15 people to the execution of security activities considering FTE.

We developed below a model that considers all of these recommendations to define a multi-factor numerical value to calculate the IT security team size. The recommendations are represented as metrics, defined herein as M1, M2, M3, and M4, and the security team size is defined as *SecS*. The *metrics and SecS* are denoted in Equations (2) to (6) as:

$$M1 = nusu/1000 \quad (2)$$

$$M2 = ndis/5000 \quad (3)$$

$$M3 = nequi * 0.03 \quad (4)$$

M4 = 3.8 people (minimal situation), 6.15 people (ideal situation) (5)

$$SecS = round(\Sigma(M1,M2,M3,M4)/4)|Max=7 \quad (6)$$

Where:

nusu = number of employees in the institution
ndis = number of interconnected devices on the network
nequi = number of IT employees

Table 2 presents the calculation of IT staff size for each survey participant considering real data presented by them. The last column, *SecS*, corresponds to IT security team size calculated according to Equation 6. *SecS* calculated the averages of M1, M2, M3, and M4, and rounded the result to the next integer, maximum 7, considering the study of FTE.

Table 2: IT security team size calculation

ID	nusu	ndis	nequi	M1	M2	M3	M4	SecS
1	304	450	13	0.3	0.1	0.39	3.8	1
2	718	889		0.7	0.2	0	3.8	1
3	500	1000	15	0.5	0.2	0.45	3.8	1
4	600	1167	17	0.6	0.2	0.51	3.8	1
5	716	1200	15	0.7	0.2	0.45	3.8	1
6	1086		14	1.1	0.0	0.42	3.8	1
7	815	1427	21	0.8	0.3	0.63	3.8	1
8	842	2000	33	0.8	0.4	0.99	3.8	2
9	850	1956	40	0.9	0.4	1.2	3.8	2
10	719	1400	35	0.7	0.3	1.05	3.8	2
11	2000	1038	32	2.0	0.2	0.96	3.8	2
12	1082	2800	53	1.1	0.6	1.59	3.8	2
13	1700	2000	42	1.7	0.4	1.26	3.8	2
14	1400	2300	50	1.4	0.5	1.5	3.8	2
15	1602	2580	43	1.6	0.5	1.29	3.8	2
16	2009	4467	22	2.0	0.9	0.66	3.8	2
17	1800	1730	49	1.8	0.3	1.47	3.8	2
18	2967	4000	54	3.0	0.8	1.62	3.8	2
19	3000	4280	55	3.0	0.9	1.65	3.8	2
20	2200	8776	59	2.2	1.8	1.77	3.8	2
21	3000	4000	90	3.0	0.8	2.7	3.8	3
22	3553	5838	67	3.6	1.2	2.01	3.8	3
23	4005	8000	88	4.0	1.6	2.64	3.8	3
24	6647	10143	79	6.6	2.0	2.37	3.8	4
25	4950	10000	153	5.0	2.0	4.59	3.8	4
26	8075	12350	116	8.1	2.5	3.48	3.8	4
27	6000	6245	330	6.0	1.2	9.9	3.8	5
28	13511	15000	516	13.5	3.0	15.48	3.8	7

Two survey participants lacked some required metrics and are detached in Table 2. *SecS* was limited to security activities related to auditing, management, development, and implementation of security policies and processes as reported in the study [6] and as a result of Figure 21. The simulation of data provided in the survey using our model, synthesized in Equation 6, demonstrated that the calculation obtained for the security team size was feasible.

5 Conclusions

This paper carefully addresses information security demands and the amount of staff needed for accomplishing these tasks. From the survey with 28 participants, it was possible to characterize the strengths and weaknesses in information security governance.

IT managers can repeat this experience as a reference for analyzing their security situation to others and use the benchmark technique this work provides.

In the case study, the main difficulties encountered in information security management were related to security governance in aspects of policy, organization, human resource management, system maintenance, supplier relations, business continuity, incident management, and compliance.

Daily operational management, communications management, physical security, access control, and assets management were well rated in the case study. The security staff size proposed disconsidered operational functions, since other areas also took care of information security in the institution.

The study presented quantitative models to calculate security status and team size. The models were simulated with real data obtained in the survey.

The work produced feasible results facilitating its implementation. The calculated metrics can improve security information and help achieve more efficient management.

6 References

[1] ISO. International Standardization for Organization. http://www.iso.org.

[2] COBIT. Control Objectives for Information and related Technology. “Cobit 5 – A management guide“. Van Haren publishing. 2012.

[3] ITIL. “Information Technology Infrastructure Library“. Available at http://www.itil.org.uk/all.htm. Visited in 15/06/2014.

[4] ABNT NBR ISO/IEC 27001. “Tecnologia da informação – Técnicas de segurança – Sistemas de gestão da segurança da informação – Requisitos“. ABNT. 2013.

[5] Tipton, Harold & Krause, Micki. Information Security Management, pp 598-599. 2003.

[6] Computer economics. “IT Security staff levels are declining“. Available at http://www.computereconomics.com/article.cfm?id=1384. Visited in 27/11/2014. 2008.

[7] Pirani, Judith A. “High Stakes: Strategies for Optimal IT Security Staffing“. Educause – Center for Applied Research. 2004.

[8] Vostrom. “Rationalizing Information Security Staffing“. Available at http://vostrom.com. Visited in 27/11/2014. 2004.

[9] OISSG – Open Information System Security Group. « “Information Systems Security Assessment FrameWork (ISSAF) 2.1A “. 2006.

[10] Wisegate. “2013 IT Security Benchmark Report. Crowdsourced Survey Uncovers Key Security Program, Budget and Job Trand Data for CISOs and Security Leaders ». Wisegate Research Report. 2013.

Organizational Information Security Culture Assessment

Areej AlHogail[1] and Abdulrahman Mirza[2]
Department of Information Systems
College of Computing and Information Sciences
King Saud University Riyadh, Saudi Arabia
[1] alhogail@ccis.imamu.edu.sa , [2] amirza@ksu.edu.sa

Abstract—*Information security culture could be used in organizations to construct the appropriate security beliefs and values and to guide the employees' security behavior in order to achieve a secure environment for organizational information assets. In this paper, we adapted Information Security Culture Framework (ISCF) to create a questionnaire to assess the level of an information security culture in organizations. The ISCF is a comprehensive framework that consists of five dimensions and incorporate change management and human factor in information security. Three case studies have been selected to demonstrate the effectiveness of ISCF in describing and explaining the organizational information security culture.*

Keywords— information security culture; information security management; assessment instrumnet; human factor.

1. Introduction

Until now the use of 'information security culture' has not been clearly utilized in practice due to lack of practical frameworks and models. Information security culture provides a guide and structure to human behavior when interacting with Information and Communication Technology (ICT) to avoid actions that may cause risks to the security of information assets. Without a proper information security culture, the enforcement of security policies through the traditional cycle is likely to be less effective [1]. It is apparent that security can only be effective if employees know, understand, and accept the necessary precautions.

Many studies suggest that implementing information security culture inside organizations would help managing and reducing security risks to information assets [2]–[7]. They suggest that organizations need to take formative steps in order to create an environment where security is "everyone's responsibility" and where doing the right thing is the norm [8][8][8](Alfawaz, Nelson, and Mohannak 2010)

Dojkovski et al. [9] suggested that a strong information security culture in organizations might deal with many of the behavioral issues that cause information security breaches in such organizations. Information security culture can be defined as follows: "The collection of perceptions, attitudes, values, assumptions and knowledge that guide the human interaction with information assets in organization with the aim of influencing employees' behavior to preserve information security" [10]. Consequently, organizations need a comprehensive framework and guidelines to build a security-aware culture.

In this paper the Information Security Culture Framework (ISCF) proposed by one of the authors in a previous study [11] is used to create an assessment instrument and applied in an empirical study to validate the instrument. The empirical study includes three case studies. The objective is to validate the ISCF assessment instrument, hence providing a valid and reliable instrument that can be used by organizations to assess information security culture within organizations.

2. Literature review

Organizations need a comprehensive framework and guidelines to build a security-aware culture. AlHogail & Mirza [12] indicated a lack of comprehensive frameworks to guide the cultivation and assessment of effective information security culture. Moreover, most available approaches that address the threats posed by employees' behavior do not focus on the interaction between the behavior of the employee and the organizational culture [5], and on directing the employees' behavior.

Most available framework were lacking a comprehensive view that integrated the human, organization and technology to provide organizations with an all-inclusive framework to aid organizations' information security practitioner in the implementation and adoption of the information security culture [10]. Therefore, there is a strong need for a comprehensive framework to cultivate and assess a security-aware culture.

Moreover, Okere et al. [13] stated that there is no method or toolset to assess information security culture as there is no published or widely accepted and consolidated approach that assigned how to assess the culture and more research in this area is needed. However, one way to measure the status of an organization's information security culture is to use a questionnaire such as those proposed in [14] and [15] to achieve an understanding of factors influence the employees security behavior. Also Da Veiga et al.[16] have validated an instrument for assessing the information security culture created by Martins & Eloff [14]. There is still a need for valid

comprehensive assessment instruments [17].

In information security culture context, assessment plays an important role such as exploring risk sources which help to propose efficient solutions [13]. Da Veiga [18] and Martins [19] have mentioned a number of advantages of using questionnaire to assess the information security culture. Some of these are:

- Identifying areas of concern and areas requiring improvements with regard to the information security culture.
- Help organization to specify the current and the desired information security culture, and recognize change actions required to accomplish the desired information security culture.
- The information obtained can influence future management decisions, such as more awareness, training or resources allocations.
- The questionnaire could be a way of raising awareness regarding information security. It also helps to increase the commitment of organization's employees as they feel that they are part of the process.
- Significant results are yielded due to the statistical robustness.
- Relatively low costs.
- Monitoring the impact of change and performance improvement, as management may use questionnaire results to determine whether its information security culture has had the desired effect.

3. Information Security Culture Framework (ISCF)

Alhogail [11] suggested a comprehensive Information Security Culture Framework (ISCF) for organization consisting of five dimensions: Strategy, Technology, Organization, People, Environment (STOPE). It incorporates four main domains of the human factor diamond: preparedness, responsibility, management, and society and regulations. Furthermore it incorporates change management principles adapted from [20] that guide the cultivation of the information security culture. The framework is portrayed in Fig 1.

Strategy dimension is concerned with the appropriate implementation of different information security strategies such as plans of actions, policies, objectives, best practices, standards, guidelines, and priorities that are designed to guide organization members to reach the goal of protecting information assets. The technology dimension is concerned with security technologies such as hardware, software, services, appliances, and applications that are used within the organization to protect information assets.

The organization dimension is concerned with the collection of information security-related beliefs, values, assumptions, symbols, norms, and knowledge that uniquely identify the organization. The people dimension is concerned with the behavior of any person within the organization who is in direct contact with information assets. The information security culture aims to ensure that information security is everyone's responsibility.

Environment dimension is concerned with the identifiable external elements surrounding the organization that affect its structure and operations and in turn the security of the information assets and the information security culture. It includes national culture, ethical conduct, government initiatives, and legal and regulations systems.

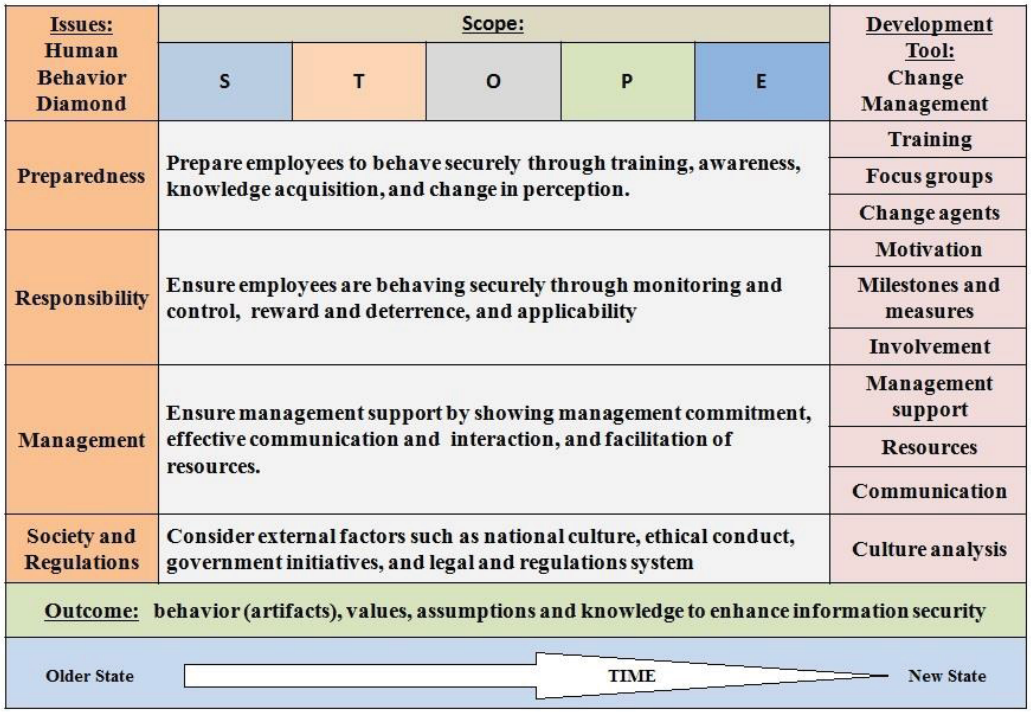


Fig. 1 The Information Security Culture Framework ISCF (AlHogail, 2015)

The human factor diamond presents four domains of human factors that influence information security behavior. The information security culture shall consider each human factor carefully to improve user security behavior. The "preparedness" domain is mainly concerned with training and awareness, knowledge acquisition, and change of old practices. The "responsibility" domain is mainly related to employees' practices and performance such as monitoring and control, reward and deterrence, and acceptance of responsibility. The "management" domain is concerned with security policy, practice, direction, and interaction issues. Finally, the "society and regulations" domain is mainly related to social and cultural aspects and regulation issues

The changeover to adapt information security culture needs to be managed carefully and appropriately to achieve the required strategic goals of creating an information asset-secure environment. Unmanaged change can lead to chaos that exposes critical information assets to security risks as it requires interrupting what employees are used to do [13], [21], [22]. The ISCF has incorporated change management principles in order to guide the changes that are associated with the development of an information security-aware culture inside the organization and to enhance acceptance of and compliance with information security policies and procedures.

In the remaining sections of this paper, the framework will guide the empirical study in order to assess the level of information security culture within each case study.

4. Methodology

Case study research strategy is used to assess the information security culture within the organization based on the framework in real life situation. The researcher can study security culture in a usual setting and learn about the state of the current situation allowing to realize and examine the nature and complexity of the development of information security culture. Also, it is an appropriate way to investigate an area that is exploratory and not confirmatory in nature, and few studies have been carried out. Information security culture area has not been fully investigated.

Three organizations have been selected as the research case studies. All the three organizations are located in Saudi Arabia. The three cases' profiles are summarized as the following:

Case A: A government organization that is responsible for issuing the required rules and regulations and monitoring some financial investments.

Case B: A Small-medium enterprise (SME). Employees may have access to financial and life style records of their clients that should be always kept private and not disclosed. Moreover, the organization holds many projects- related information that may be desirable for many competitors.

Case C: A multinational trading company that manages subsidiaries business all over the Middle East including hotels, restaurants, electronics, food industry and petrochemicals.

4.1 Data gathering

The ISCF assessment instrument is a questionnaire that is used to collect data from employees for the three case studies' regarding their values, beliefs, perceptions, knowledge and practice towards information security in order to assess the information security culture based on the ISCF dimensions and elements. Details of the questionnaire design and distribution will be discussed in the following sections.

4.1.1 Questionnaire development

The questionnaire consists of two parts, firstly, the demographical information that is used to segment the data and to make comparisons possible among the respondents. It collects information regarding age group, education background, job level and years of experience, and information technology use and experience.

The second section aims to assess the employees' information security behavior, perceptions, beliefs and knowledge based on the framework dimensions. Framework dimensions have been mapped into several representative tasks and statements in order to measure them. Then, statements were grouped as clusters to represent each element of the dimension where each cluster is consisted of a number of statements that are grouped together and in relation to one another to represent one issue [16].

The questionnaire questions are divided to cover the following issues

- Strategy: checking whether the organization has any strategy elements, measuring the accessibility and clarity of different strategies, and studying the effects of different strategy elements on the employees' security behaviour and general information security culture.
- Technology: assessing the application of different technical measures, the availability of guidance and support including their effectiveness in information security culture.
- Organization: evaluating the values, beliefs, behaviour and knowledge towards information security and their relation to the information security culture.
- People: considering different human factor of preparedness, responsibility and management role that affect information security behaviour and their relation to information security culture.
- Environment: measure the effect of different external factors of national culture, laws and regulations, ethical conduct and government initiatives on the information security culture.

Statements have been selected to ensure that they represent each human factor for every dimension. For instance, technology preparedness is represented in the statement: "I have received training on using information security hardware and software". Technology responsibility is represented in the statement: "I know that the appropriate use of technical controls is vital to achieve information security" and so on. Likert scale is used to measure the respondent's degree of agreement or disagreement with each statement. It varies from 2 point Likert

scale of Agree and Disagree answers to 5 points Likert scale, depending on the question and the possible responses to avoid bias. The used scale is represented in Fig 2 with an example.

Statement	Agree	Disagree			
I have read information security policy or strategy					
Statement	Agree	I don't Know	Disagree		
The organization has a written information security policy or strategy					
Statement	Strongly Agree	Agree	I don't Know	Disagree	Strongly Disagree
Information security strategy element clearly state what is expected from me.					

Fig.2 Survey scale example

4.1.2 Questionnaire administration

The questionnaire has been sent to all employees for the selected case studies organizations, in order to achieve a convenient sample. It was accompanied by a cover letter that describes the questionnaire and emphasizes the confidentiality of the obtained results, and ensures anonymity of participants.

The questionnaire was distributed via e-mail through sending an invitation to participate in the survey and to fill the survey online. The respondents can answer the questionnaire in their own time and at their own pace. Table I indicates the number of responses obtained for each case study.

Table I questionnaire response rate

Criteria	**Case A**	**Case B**	**Case C**
Number of Employees who in direct contact with information (population)	400	50	100
Number of response	52	20	22
Response rate	13 %	40 %	22 %

After that, obtained data from the information security culture survey is prepared for analysis.

5 Results and analysis

The data collected through the case study assessment instrument was quantitatively analyzed using the Statistical Package for the Social Sciences (SPSS) software. The data preparation process ensured that the data set have no missing values and not distorted significantly by the different opinions of specific groups. The data is ordinal and small in size, therefore, non-parametric test will be used when appropriate. The following statistical analyses describe the results:

5.1 Reliability

Cronbach's alpha is the most common used technique to measure reliability through providing an indication of internal consistency [16]. The Cronbach alpha value of each dimension and sub dimension of the framework has been analyzed in order to establish the reliability of the assessment instrument. Cronbach alpha values must meet the minimum accepted criteria that is to be above 0.6 to confirms the consistency and reliability of the framework [16]. Table II provides the results of the analysis of the data to construct reliability.

Table II Cronbach's alpha attribute value and the analysis results

Factor	No of items	Case A		Case B		Case C	
		α value	**Analysis**	**α value**	**Analysis**	**α value**	**Analysis**
S	8	.705	good	.707	good	.816	good
T	6	.720	good	.622	acceptable	.619	acceptable
O	21	.865	good	.832	good	.928	excellent
P	27	.903	excellent	.824	good	.900	excellent
E	10	.775	good	.631	acceptable	.819	good

The values of the Cronbach's alpha ranged from (Case A: 0.705 to 0.903; Case B: 0.622 to 0.832; Case C: 0.619 to 0.928) which is larger than the threshold. This indicates a good internal consistency and reliability. Therefore, the instrument appears to be composed of a set of consistent variables for capturing the meaning of the framework.

5.2 Validity

In order to establish the validity of the tested framework, SEM analysis using Goodness Fit Index(GIF) [23] will be used to measure fitness between the proposed framework and the empirical data [23]. The Goodness of Fit Index (GFI) has the criteria that if $0.9 \leq GFI \leq 0.94$ then it is acceptable fit and if $0.95 \leq GFI \leq 1.0$ then it is a good fit. If it is less than 0.9, then it is a poor fit. The following table provides the attribute value of GFI for each case study.

Table III validity analysis

Goodness Fit Index	**GFI value**	**Acceptance analysis**
Case A	.927	Acceptable fit
Case B	.969	Good fit
Case C	.902	Acceptable fit

It is clear from the results that the GFI is higher than 0.9 indicating that the theoretical ISCF can be accepted and that there is a good fit between the empirical data and the instrument.

5.3 Descriptive statistics

5.3.1 Demographical information

The results showed that the sample adequately represented all segments of classification in the three case studies and adequately representative.

The influence of demographical information and categories were analyzed in order to figure out any external influences on the level of information security culture among the organizations' members using Independent Sample Kruskal-Wallis non-parametric test at significance level of .05 was used. The results are summarized in Table IV.

From the data in Table IV, it is clear that there is difference between background education in IT and working in IT department and the information security culture because $sig<0.05$. It can be concluded that information security culture,

through knowledge and behavior level, among each case study members was only affected by one of two factors: their background education was IT related, or they are working in IT department.

Table IV demographical data analysis

Null Hypothesis: The distribution is the same across all categories of :	Test	Sig Case A	Sig Case B	Sig Case C	Decision
Age	Independent Sample Kruskal-Wallis test at significance level of .05	0.396	0.905	0.485	retain the null hypothesis
Education		0.051	0.242	0.767	retain the null hypothesis
Years of experience in organization		0.421	0.487	0.138	retain the null hypothesis
Job level		0.043	0.947	0.138	retain the null hypothesis
Gender	Independent Sample Mann-Whitney U test at significance level of .05	0.796	0.842	-	retain the null hypothesis
Education background in IT		0.0297	0.033	0.857	reject the null hypothesis
Working in IT department		0.054	0.089	0.0408	reject the null hypothesis

5.3.2 The STOPE dimensions statistical analysis

To obtain an overall mean for each dimension, the scores from each dimension's items were averaged. Table V presents the summary of the results of each case study.

Table V the summary of statistical analysis of the STOPE

Case A					
	S	**T**	**O**	**P**	**E**
Mean	3.70	4.17	4.10	4.04	3.94
Standard Error	0.10	0.08	0.12	0.11	0.10
Standard Deviation	0.45	0.38	0.49	0.41	0.38
Confidence Interval	{3.6-3.8}	{4.1-4.3}	{4.0-4.2}	{3.9-4.1}	{3.8-4.0}
Information Security Culture Mean				3.99	
Case B					
	S	**T**	**O**	**P**	**E**
Mean	3.26	3.83	3.39	3.51	3.54
Standard Error	0.16	0.15	0.20	0.18	0.16
Standard Deviation	0.42	0.38	0.44	0.34	0.25
Confidence Interval	{3.0-3.4}	{3.7-4.0}	{3.2-3.6}	{3.4-3.7}	{3.4-3.7}
Information Security Culture Mean				3.5	
Case C					
	S	**T**	**O**	**P**	**E**
Mean	3.71	3.79	3.87	3.64	3.97
Standard Error	0.10	0.08	0.13	0.10	0.10
Standard Deviation	0.57	0.37	0.63	0.44	0.39
Confidence Interval	{3.4-3.9}	{3.6-4.0}	{3.8-4.2}	{3.4-3.8}	{3.8-4.1}
Information Security Culture Mean				3.8	

The average mean of weighted means indicate the level of information security culture at each case study. The average mean at the three case studies was between 3.5 and 3.9 which indicates a weak level that requires attention and needs improvements. For Case Study A, as the organization has invested heavily in information security, the mean of T, O, and P dimensions was above 4 indicating a good reflection on these variables. Yet, more investment is needed in strategy and in environmental factors. For Case Studies B and C, all mean values were below 4. In the three cases, the lowest average of weighted mean was in strategy dimension; revealing a critical weakness in strategy planning and implementation at the three organizations which was reflected on information culture level.

The small range of 95% confidence interval, demonstrates an indication of the agreement between organizations members on the level of information security culture and suggest that the mean is adequately representative. The small standard deviation values indicate small deviation from the mean, showing precise results among the organization members and assuring that the mean is a good representative for each data set. Additionally, a small standard error value points out that sample means are similar to the population mean, and therefore, the sample is an accurate representation of the population. Consequently, it can be concluded that the mean value can be used as a representative for the data set. In addition, the small values of the standard error suggest that the sample used was sufficiently representative of the population.

The perceptions of the five STOPE domains have been measured among employees through some statements. The agreement responses on each statement have been accumulated and analyzed based on percentage and presented in Table VI.

Table V The perceptions of the five STOPE

Dimension	**Case A**	**Case B**	**Case C**
Strategy	92%	95%	95%
Technology	97%	93%	100%
Organization	95%	94%	98%
People	95%	94%	98%
Environment	95%	94%	98%

From the data, all STOPE dimension have been positively received among the three case studies' members. More than 90% of respondents are favorable of every dimension statements. Consequently, the five dimensions considered important in creating effective information security culture.

5.3.3The relationship between knowledge and behavior in information security culture

Spearman's rho test is used to determine the degree of relationship and influence between the level of security knowledge and the employee's information security behavior and between knowledge and behavior on the level of information security culture [24]. Spearman's rho test is a non-parametric measure to assesses the degree of relationship between two variables using a monotonic function [24]. In order to conclude a relationship between two variables using Spearman's test, the significance must be ($\leq$0.05). Results as presented in Table VII:

Table VI The analysis of the relationship between knowledge and behavior

Variables	Correlation Coefficient	Sig.	Conclusion
knowledge, employee behavior	0.705	0	Strong positive relationship
Knowledge, How employees view management behavior	0.808	0	Strong positive relationship
employee behavior, How employees view management behavior	0.551	0	Moderate positive relationship
knowledge, ISC	.957	0	Strong positive relationship
employee behavior, ISC	.667	0	Moderate positive relationship
How employees view management behavior, ISC	0.863	0	Strong positive relationship

From the results above, it can be concluded that there is a positive relationship between the levels of knowledge and how employee behave. A strong relationship is also shown between knowledge and information security culture level. Therefore, for effective information security culture the level of knowledge will highly affect the information security behavior and culture and should be considered as a critical factor. Different methods of awareness rising could be used to equip employees with information security knowledge. This supports the findings of [25].

6 Discussion

In the three case studies, the use of IT technology in the organization is high to very high. In addition, all the three cases possess valuable information assets. Organizations that have a high usage of IT technology or hold valuable information assets will have a higher likelihood of being vulnerable to information-related misuse [24]. Consequently, in such organization, it is crucial to establish an information security culture.

The empirical study indicates that the use of the assessment instrument was valid in assessing the level of the information security culture based on ISCF and can be used to produce a list of recommendations to secure information assets.

The results of the empirical study indicate that that level of information security culture at three organizations is below the accepted level. Employees believe in the importance of information security for their organization, which establishes a very good foundation to effective implementation of information security culture. In general, the majority of employees for the three case studies are unaware of the organization information security policy and strategies and they have no idea how to get access to it. Moreover, it has been found that most of management efforts are known only by people who work in IT department. This indicates a gap between the management efforts in planning the security strategies and what employees know and do. Therefore, more awareness efforts to spread the word are needed.

All the three organizations have invested in advance security technologies and ensured that they are well implemented. The three case studies have provided a good example of technical and personal support. On the downside, employees are not well trained on using some technologies efficiently rising the risk of human errors or reducing their effectiveness. For instance, they should be educated on the safe storage and disposal of information and how to securely use storage devices.

The organization artifacts, values, assumptions and knowledge are very critical in determining the effective information security culture[23]. In general, most of the organizations' employees are aware of major information security concepts and believe in the importance of it. This indicates according to Martin et al. [14] a good foundation for the information security culture at the individual level. Nevertheless, the three case studies revealed a lack in effective awareness initiatives to raise awareness. Employees were unaware of their information security responsibilities. Employees in the survey suggested different methods such as workshops, training, e-mail, newsletters, posters and leaflets. Different communication methods are also advised to open more chances for questions and answers to reduce gaps between what organizations do and what employees know. Knowledge sharing between employees should be also encouraged to share information about threats and vulnerabilities in information assets.

In the environment dimension, there is no direct relation between reading the code of ethics by employees and their actions. In addition, it was found that employees in general are not informed about related laws, regulations, standards and legislations. Therefore, investment should be more on understanding culture differences and raising awareness in related laws and regulations to avoid any employee's action that may cause the organization to be subject to any legal troubles. Employees need to be aware of the privacy of organization, customer and third parties information to minimize any information leakage that may threaten the trust in the organization. Moreover, culture analysis is very important to set the correct assumptions that need to be changed.

Employees' knowledge and behavior were affected by two factors: background education was in IT related field, and they are working in IT department. The later could be linked to the fact that they are the closest to what is implemented to protect information assets and they are usually responsible for information security. This emphasizes the weak awareness efforts that cause the gap between what management do and what employees know. Moreover, employee's knowledge level could be linked to the level of awareness initiatives at the organization. Furthermore, it is been concluded from the results that information security knowledge highly affects the information security behavior. This is similar to the finding of [23] that employees with no security knowledge or skills will not be able to act securely in the desired way.

Proposed change management principles will provide support and guidance to the implementation of change within a structured process to achieve a smooth change and help employees perceive change positively. Moreover, there is a positive relationship between the information security culture level and the application of change management principles.

Thus, it can be concluded that change management principles are valuable in creating effective information security culture and the framework proposal of incorporating change management is practical and valid.

7 Conclusion

The information security culture framework provides a comprehensive base for organizations to develop an effective information security culture in order to protect information assets. The application of the framework to any organization would improve its employees' behavior and how they interact with the information assets, leading to a positive impact and guarding against many information security threats posed by insiders.

The empirical study aims to evaluate an assessment instrument based on the ISCF. Assessing a broad range of elements provided a holistic view of the information security culture level at each organization. The results derived from the assessment can guide management with the issues that require improvement. It can identify strengths and weaknesses in the organization with regard to information security.

From the empirical study, it can be noticed that the private sector is not placing enough efforts in information security due to the high cost of information security. This is in line with the findings of [9] in Australian SME organizations. Also it is been noticed that the level of sensitivity and criticality of the organization business activities usually determines the level of efforts management paid to protect the information assets. However, information security should be integrated in the business functions no matter what the business activities are. Information security culture takes care of raising employees' awareness of the correct methods of dealing with information assets. The statistical analysis confirmed that the ISCF assessment instrument was valid and reliable in assessing the information security culture. Other organizations can therefore use it to assess and cultivate an effective information security culture.

This research can be expanded in the future to develop best practice guidelines to assist organization with the cultivation of information security culture.

Acknowledgment

This research was supported by the ‘‘Research Center, College of Computer and Information Sciences, King Saud University.

References

[1] S. Maynard, A. Ruighaver, and P. Chia, “Exploring Organisational Security Culture: Developing a comprehensive research model.,” *IS ONE World Conf. USA*, pp. 1–13, 2002.

[2] A. Ruighaver, S. Maynard, and S. Chang, “Organisational security culture: Extending the end-user perspective,” *Comput. Secur.*, vol. 26, no. 1, pp. 56–62, Feb. 2007.

[3] O. Zakaria, “Internalisation of Information Security Culture amongst Employees through Basic Security Knowledge,” in *IFIP International Federation for Information Processing, Volume 201, Security and Privacy in Dynamic Environments*, S. Fischer-Hubner, K. Rannenberg, L. Yngstrom, and S. Lindskog, Eds. Boston: Springer, 2006, pp. 437–441.

[4] D. Bess, “Understanding Information Security Culture for Strategic Use: A Case Study,” in *AMCIS 2009 Proceedings*, 2009, p. paper 219.

[5] A. Da Veiga and J. Eloff, “A framework and assessment instrument for information security culture,” *Comput. Secur.*, vol. 29, no. 2, pp. 196–207, Mar. 2010.

[6] S. Furnell and K. Thomson, “From culture to disobedience: Recognising the varying user acceptance of IT security,” *Comput. Fraud Secur.*, vol. 2009, no. 2, pp. 5–10, Feb. 2009.

[7] K. Knapp, T. Marshall, R. Rainer, and F. Ford, “Information security: management’s effect on culture and policy,” *Inf. Manag. Comput. Secur.*, vol. 14, no. 1, pp. 24–36, 2006.

[8] S. Alfawaz, K. Nelson, and K. Mohannak, “Information security culture: a behaviour compliance conceptual framework,” in *8th Australasian Information Security Conference (AISC 2010)*, 2010, pp. 47–55.

[9] S. Dojkovski, S. Lichtenstein, and M. Warren, “Enabling information security culture : influences and challenges for Australian SMEs,” in *ACIS 2010 : Proceedings of the 21st Australasian Conference on Information Systems, ACIS*, 2010, p. 61.

[10] A. Alhogail and A. Mirza, “Information security culture: a definition and a literature review,” in *proceedings of IEEE World Congress On Computer Applications and Information Systems*, 2014.

[11] A. Alhogail, “Design and Validation of Information Security Culture Framework,” *Comput. Human Behav.*, vol. 49, no. August, pp. 567–575, 2015.

[12] A. AlHogail and A. Mirza, “A Proposal of an Organizational Information Security Culture Framework,” in *Proceeding of the 8th IEEE International Conference on Information, Communication Technology and Systems ICTS 2014*, 2014.

[13] I. Okere, J. van Niekerk, and M. Carroll, “Assessing information security culture: A critical analysis of current approaches,” in *in the proceedings of IEEE conference on Information Security for South Africa (ISSA)*, 2012, pp. 1 – 8.

[14] A. Martins and J. Eloff, “Information security culture,” in *Security in the information society*, Boston: Kluwer Academic Publishers, 2002, pp. 203–214.

[15] T. Schlienger and S. Teufel, “Tool supported management of information security culture,” in *IFIP Advances in Information and Communication Technology*, 2005, pp. 65–77.

[16] A. Da Veiga, N. Martins, and J. Eloff, “Information security culture-validation of an assessment instrument,” *South. African Bus. Rev.*, vol. 11, no. 1, pp. 146–166, 2007.

[17] N. Martins and A. Da Veiga, “The Value of Using a Validated Information Security Culture Assessment Instrument,” in *8th European Conference on IS Management and Evaluation*, 2014, pp. 146–154.

[18] A. Da Veiga, “Cultivating and Assessing Information Security Culture, Unpublished PhD Thesis,” University ot Pretoria, 2008.

[19] A. Martins, “INFORMATION SECURITY CULTURE,” RAND AFRIKAANS UNIVERSITY, 2002.

[20] A. Alhogail and A. Mirza, “A framework of Information Security Culture Change,” *J. Theor. Appl. Inf. Technol.*, vol. 64, no. 2, pp. 540–549, 2014.

[21] L. Ngo, W. Zhou, and M. Warren, “Understanding Transition towards Information Security Culture Change.,” in *Proceeding of the 3rd Australian Computer, Network & Information Forensics Conference, Edith Cowan University, School of Computer and Information Science*, 2005, pp. 67–73.

[22] J. Bennett, “Effectiveness Of Using A Change Management Approach To Convey The Benefits Of An Information Security Implementation To Technology Users, Unpublished PhD thesis,” Capella University, 2012.

[23] D. Hooper, J. Coughlan, and M. Mullen, “Structural Equation Modelling : Guidelines for Determining Model Fit,” *Electron. J. Bus. Res. Methods*, vol. 6, no. 1, pp. 53–60, 2008.

[24] J. Pallant, *SPSS Survival Manual: A Step by Step Guide to Data Analysis using SPSS for Windows*. Berkshire, England: Open University Press., 2010.

[25] J. Van Niekerk and R. Von Solms, “Information security culture: A management perspective,” *Comput. Secur.*, vol. 29, no. 4, pp. 476–486, Jun. 2010.

SESSION

SECURITY APPLICATIONS + CRYPTOLOGY + BIOMETRICS

Chair(s)

Dr. Luis Hernandez Encinas
Dr. Agustin Martin Munoz

A Unique-ID based Usable Multi-Factor Authentication Scheme for e-Services

Mohammed Misbahuddin, Roshni VS, Anna Thomas, Uttam Kumar

Centre for Development of Advanced Computing (C-DAC)
Electronics City, Bangalore - India
Email: mdmisbahuddin@gmail.com (Corresponding Author)

Abstract: *The current day web offers a wide range of e-Governance, e-commerce and other online services that require strong authentication mechanisms to safeguard user's account. In addition, these services require that a user be verified during registration to prevent duplication of accounts in cases where a fraudulent user creates multiple accounts with different credentials to avail the welfare services. Therefore, the challenge is to protect the e-services using secure multi-factor authentication methods with one account per user without compromising the usability. This paper discusses a multi-factor authentication (MFA) scheme which uses password, mobile token and image as multiple factors for authentication. The scheme uses a unique identity for verification of user accounts. The scheme leverages the identity verification system of Unique Identification Authority of India (UIDAI) System for ensuring the issuance of a unique and verified user identity to prevent duplicate and fraudulent accounts. The scheme does not maintain a verifier table at server to prevent stolen verifier attack. In addition, to achieve high level of usability, the scheme proposes to use Image Passwords and Mobile Tokens. The paper also discusses the security analysis of the proposed scheme against common authentication related attacks and the formal verification of the scheme using Scyther.*

Keywords: Two-factor Authentication, Mobile Tokens, e-Services, e-Governance, Graphical Passwords, Image Passwords, Stolen-verifier Attack, Formal Verification, Scyther.

1. Introduction

The advent of web 2.0 and Mobile Technology has led to enormous growth in web based services. The current statistics of Internet users shows that over 3 billion users are connected. In other words, today, over 40% of the World population has an Internet connection. [13]. The Web-based applications and services have changed the landscape of information delivery and exchange in today's corporate, government, and educational arenas. The common services that are offered to users include e-Governance, e-Banking, e-Shopping, financial services, e-scholarships, e-welfare schemes etc. The access to these services is protected mostly by Single Factor Authentication (SFA) methods. The most common example of SFA is Password based authentication. However, offering services using only SFA has various security challenges such as security against Password Guessing attack, Dictionary attack, Brute Force attack, Stolen-Verifier attack etc.

The security strength of authentication can be increased by deploying Two Factor Authentication (TFA) which requires the user to provide an additional factor for identity verification from separate categories of user credentials such as a password (first factor) and a physical token (as additional factor). There are various two factor authentication schemes that use Biometrics, Cryptographic Tokens and Smart Cards as additional factors. But due to the simplicity, cost efficiency and high security that the Smart Card based schemes offer, many researchers focused their attention on designing Smart Card based schemes without verification table at server [1-9]. However, the wide acceptance of Smart Card based schemes requires every user to have a card reader attached to their PCs.

Most recently, using mobile as a second factor for authentication gained significance as the penetration rate of mobile in many of the countries across the globe has crossed 65% of population [14]. Among the given percentage of mobile phone users, atleast 50% owns smart phones and over 83% of them uses mobile Internet. [15].

The mobile phones have come a long way in a very short time with the rapid advancements in mobile technology. With the increase in Smartphone use, it is evident that the usage of mobile as a second factor will surely reap security and usability benefits.

To avail an e-service, a user need to register with each service provider separately leading to creation of multiple user accounts for a single user with different credentials on different services. There are also cases wherein the fraudulent users register for a single welfare or scholarship service multiple times with different credentials to avail the benefits multiple times illegally. Hence, there is a need to verify the identity of a user that is unique and issued by a trusted third party. This identity can be used as a common verified identity of a particular user for all the web accounts the user has registered with.

It is therefore evident that, offering an e-Service securely requires an authentication scheme which shall address the following challenges: 1) Issuance of a verified identity to a user to prevent misuse. 2) Prevention of duplicate and fraudulent accounts 3) Strong authentication using TFA 4) Usable password methods for ease of password remembrance 5) No verifier table at the server to prevent Stolen-verifier attack.

This paper discusses a TFA scheme which uses an image as the first factor and a Mobile Token (MT) as the second factor. The Mobile Token is a mobile application which

contains user's personalized data and runs on the user's mobile during authentication phase. During registration, the scheme leverages on the services of UIDAI's CIDR [16] to verify the validity of a user before issuing a registration ID to access e-Services. The proposed scheme can be integrated with the SSO component using SAML [17] protocol for seamless access of multiple services by the registered users.

The proposed scheme has the following features:

- Issues a verified user identity
- Employs a mobile based authentication scheme with multiple factors
- Prevents duplicate account creation
- Presents a unique graphical password method
- Offers Multi-factor Authentication (MFA) Security for SSO services
- Resistant to various attacks such as Guessing, Stolen-verifier, Replay etc.
- Secure mutual authentication between mobile token and server

The paper also discusses the security analysis of the scheme besides presenting the results of automated formal verification using Scyther tool [11-12].

The rest of the paper is organized as follows; Section II presents the proposed scheme, Section III presents the security analysis of the proposed scheme, section IV presents the formal verification of the proposed scheme using Scyther and finally section V presents the conclusion

2. The Proposed Scheme

This section presents the Architecture and Protocol of the proposed scheme. The scheme consists of four phases namely, Registration, Authentication, Password Change and Forget Password phase.

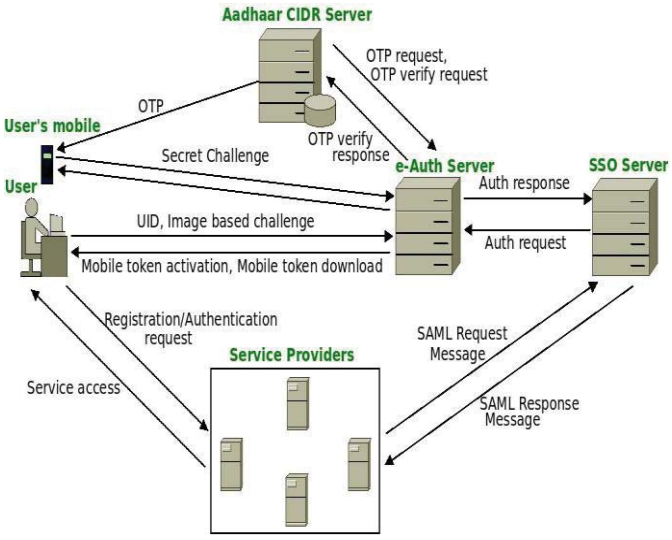


Fig 1: Architecture Diagram

The architecture of the proposed scheme is shown in fig-1 and comprises of the following components in the proposed architecture. These components are described in section 2.1

1) Central Identity Repository (CIDR),
2) User's Mobile Token (MT),
3) Service Provider (SP),
4) e-Authentication Server (AS),
5) SSO Server (SSS).

The proposed scheme uses UIDAI's Aadhaar Authentication Ecosystem (Fig 2) for registration of users of e-services. Aadhaar Authentication is the process wherein an Aadhaar (a unique residents ID), along with other attributes (demographic and/or biometrics and/or OTP) are sent to UIDAI's Central Identities Data Repository (CIDR) for verification; the CIDR verifies whether the data submitted matches the data available in CIDR and responds with a "Yes/No". The personal identity information is never returned as part of the response [16]. The UIDAI provides open APIs to be integrated with e-Services for verification of user's identity. The Aadhaar Authentication eco system is shown in fig 2. This facility helps in ensuring a single verified login identity for each user and also helps in preventing the creation of fraudulent and duplicate accounts on the e-Auth server.

The scheme proposes a unique feature wherein the users' secret credentials are not stored on any of the components given in the architecture except user's mobile token.

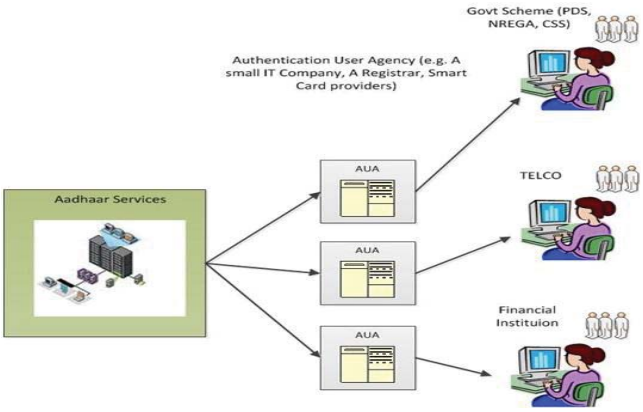


Fig 2: Aadhaar Authentication Eco System

2.1 Architectural Components

Central Identity Repository (CIDR): The CIDR is the central data repository, and functions as a managed service provider. It implements the core services around the UIDAI – it stores resident records, issue unique identification numbers, and verifies, authenticates and amends resident data. The CIDR holds the minimum information required to identify the resident and ensure no duplicates. This includes:

- Unique Identity (UID) Number
- Unique ID agencies
- Demographic data and biometrics

UIDAI collects the user's data such as mobile number, Demographic data and Biometric and issues a Unique Identity (UID) also called as 'Aadhaar' after verification. While certain types of information such as birth date and gender will remain unchanged, other demographic details may undergo changes with time. The agency monitors these changes periodically with the assistance of a network of registrars who oversee the initial enrollment process for the issuance of UIDs and subsequent change requests. The CIDR adheres to the national and international security standards to ensure the security and privacy of users' data.

While CIDR can be used to verify the validity of Indian residents only; most of the developed and developing nations may make use of the Citizen ID or National ID databases for online identity verification of their users during registration time to issue verified online identity for web accounts. [18]

Service Provider (SP): The SPs offers the services in electronic mode through web based services such as e-Governance, e-Shopping, e-Banking etc. The service providers do not register the users directly and instead delegate the registration to e-Auth server (AS). The SPs do not maintain any authentication related data of their users but keeps only the profile data of the users. They may agree for a SSO server environment.

e-Authentication Server (AS): The AS is the most important component of this architecture. It runs the Registration, Authentication engine and provides these services to the users on behalf of SPs. The registration of a valid user account is accomplished with the help of CIDR. The authentication is carried out using MFA scheme which does not maintain password table at the server. The AS communicates with all the entities in the authentication architecture with or without SSO. The AS does not maintain any secret credentials of the users except the profile data.

Single Sign-on Server (SSS): The SSS runs the SAML protocol for providing the SSO services to the user who wishes to access multiple services in a single login session. It is responsible for creating and managing the SSO session, SSO token and SSO communication with all SPs. SSS does not maintain any secret credential of the user.

Mobile Token (MT): MT is a mobile application which is issued to the user by AS after successful registration. The MT contains the personalized registration and authentication related secrets of the user. The user may need to protect it with a PIN for better security. When the user wishes to login she has to activate the MT by keying the PIN.

2.2 Multi-factor Authentication (MFA) Scheme

2.2.1 Registration Phase

If a user 'Ui' wishes to register with AS, she submits the registration request to AS. In case, she submits the registration request to SP, the request is forwarded to AS. The user then proceeds for registration as follows:

Step R1: Ui submits her UID to AS. The AS forwards it to CIDR.
Step R2: CIDR sends OTP to users' registered mobile.
Step R3: Ui submits the OTP which is verified by CIDR
Step R4: Upon successful verification of OTP, CIDR sends the Users' profile data to AS.
Step R5: AS presents the registration form with the profile data populated on it and an image grid to User for choosing an Image (I) as his secret.
Step R6: Ui chooses an 'I' and sets a text password 'p' and submits it to AS.
Step R7: AS computes $h(Pwi) = h(h(I) \oplus p)$; $a = h(UID)$
$Ni = h(Pwi) \oplus h(x \oplus TIDi)$; where x is server's master secret key.
Step R8: AS personalizes an MT with the parameters Ni, h(I), h(Pwi), a, y_i. where y_i is the server's secret key shared with each MT whereas TIDi is the Token ID (unique for each MT) and is stored at the server with a mapping of UID.
Step R4: AS sends email to Ui with downloadable link of MT.

2.2.2 Authentication Phase

A registered user 'Ui' wishes to login to access resources at the server 'SP' will proceed as follows:
Step L1: Ui requests for login to SP for availing services. SP redirects the request to AS.
Step L2: AS asks Ui to enter UID.
Step L3: Ui keys in her UID to AS.
Step L4: AS retrieves TIDi and generates the image grid consisting of random images and the Ui's chosen Image.
Step L5: AS generates random codes 'r' displayed on each image of the grid.
Step L6: AS presents the image grid with 'r' embossed on images as challenge to Ui.
Step L7: Ui activates her MT and enters her text password 'p'. She also identifies her chosen Image from the challenge image grid and keys in 'r' in MT.
Step L8: Ui submits 'p' and 'r' to MT.
Step L9: MT computes h'(Pwi) and checks if
h'(Pwi) = = h(Pwi). If it holds, MT proceeds to compute
$Bi = h'(Pwi) \oplus Ni$; $Ci = h(Bi \oplus r)$; and $E = E_{yi}(Ci, a)$
Step L10: MT sends E to AS where E is encryption of message using y_i. Upon receipt of E, AS proceeds as follows:
Step L11: AS computes $D_{yi}(Ci,a)$ using y_i. where D is decryption. It then retrieves TIDi using 'a',
Step L12: AS computes $Bi' = h(x \oplus TIDi)$; $Ci' = h(Bi' \oplus r)$
Step L13: AS checks if Ci == Ci', if it holds then AS transfers the control to SSO module with authentication success message. SSO provides access to SPi to the user.
The Registration Phase and Authentication Phase are depicted in figures 4 and 5 on last page.

2.2.3 Password Change Phase

A Registered user can change her password by sending password change request to the AS. The change password request is allowed only after successful login.
Step C1: Ui requests for Change Password

Step C2: AS presents image grid to Ui for choosing secret image (I)
Step C3: Ui chooses a 'I' and submits to AS
Step C4: AS sends $E_{yi}(h(I))$ and a request to change text password at MT to Ui
Step C5: Ui activates MT and enters new password 'p'
Step C6: MT computes $Ni_n = h_n(h(I) \oplus p)$ and replaces Ni with the new Ni_n, h(I), h(Pwi).
Step C7: MT sends a confirmation message to AS

2.2.4 Forgot Password

If a user Ui forgets her password (the MT is not lost), she submits the reset password request to AS and proceed as follows:
Step F1: Ui submits request for reset password
Step F2: AS asks Ui to submit her UID
Step F3: Ui submits her UID.
Step F4: AS forwards the UID to CIDR
Step F5: CIDR sends OTP to Ui mobile
Step F6: Ui submits OTP to AS
Step F7: AS forwards the OTP to CIDR for verification
Step F8: Upon successful verification, CIDR sends a positive acknowledgement to AS
Step F9: AS presents a secret question to Ui as challenge or sends email to user with a password reset link
Step F10: When Ui clicks on the link, AS presents a new image grid for choosing a secret image (I).
Step F11: Ui chooses 'I' and submits it to AS
Step F12: AS sends $E_{yi}(h(I))$ and a request to change text password at MT to Ui
Step F13: Ui activates MT and enters new password 'p'
Step F14: MT computes $Ni_n = h_n(h(I) \oplus p)$ and replaces Ni with the new Ni_n, h(I), h(Pwi).
Step F15: MT sends a confirmation message to AS.

2.2.5 Loss of Mobile Token

If a registered user Ui loses her mobile token or accidently deletes it, she has to get a new MT from server by following the steps described in the registration process.

3. Security Analysis

This section presents the security analysis of proposed scheme against various attacks given below.

3.1 Replay Attack

In replay Attack, the adversary intercepts the message transmitted between two parties and then sends it at a later time to gain access to the resources. In the proposed scheme, if the message $E_{yi}(Ci, a)$, transmitted in step L10 is intercepted and replayed by adversary, he cannot gain access to the resources because both the client and server checks for the freshness of nonce every time they receive a message. Hence the attack will fail. Here 'r' is a random nonce which is displayed fresh at every login request.

3.2 Insider Attack

The insider attack is usually performed by an insider of the organization who has access to the sensitive resources by revealing the user secret information to others. In the proposed scheme, the user's password is never stored on any component in architecture. Moreover, none of the parameters required in protocol computation are stored in server in plain text form, instead, to avoid such attack, all the user credentials are stored in the database as message digest which is irreversible.

3.3 Stolen Verifier Attack

To perform this attack, an adversary who has access to the database server steals the password verifier and later uses it for offline guessing attack. But since, the proposed scheme does not maintain a verifier table the attack cannot be performed.

3.4 Server Spoofing Attack

It refers to the situation where in the attacker pretends to be a legitimate server and communicates with client to gain knowledge of the user's secret credentials.
To perform this attack on proposed scheme, the adversary has to fool the user by creating a page which looks similar to a valid server page and then redirect the user's login request in Step L1 to malicious server and subsequently generate the image grid as per step L4. Even if the malicious server generates the image grid (which is highly difficult), it will not be able to generate the Image selected by the user during registration.

3.5 Fraudulently copying the Mobile Token

If an adversary gets access to the user's MT and wants to fraudulently copy the MT (.apk file) which stores the parameters Ni, h(I), h(Pwi), a, y_i, he can neither retrieve the user's secret Pwi nor the server's master secret 'x' from the available parameters as the Pwi, x are stored as a message digest.

3.6 Denial of Service Attack

Suppose, if the adversary who has control over the server, modifies any secret information of the user stored in the database server by replacing it with a newly created digest, then the user will not be able to login even with his valid credentials. This is called as denial of service attack. In the proposed scheme, since there is no secret information stored on the server, the denial of service attack will not work.

3.7 Storage of Registration Data

In the proposed scheme a user profile is maintained, which stores secret questions; answers to secret questions in message digest form so that even if the attacker gets access to the database he cannot figure out the answers of the secret questions the user has set.

4. Formal Verification using Scyther

This section will present a brief description of the automated formal verification tool called 'Scyther' and the specifications of Security Protocol Description Language (SPDL) with .spdl extension for the schemes / protocols to verify the security claims.

Scyther is an automated formal verification tool implemented based on formal and mathematical logics.

Scyther is considered to have many novel features compared to other open source counterparts.

The major objective of Scyther is to guarantee the security of protocol. For this a mathematical model of the protocol and the network is to be created with the assumption that the network is under full control of adversary, meaning that the adversary can intercept, modify, fabricate etc. Since, modeling all the protocol primitives and the cryptographic mechanisms used in the protocol makes a model complex; the cryptographic mechanisms are abstracted with few assumptions, firstly that these mechanisms provides perfect security when the key used to encrypt is known only to the communicating parties. Secondly, it is assumed that the adversary can either decrypt every message or he cannot decrypt any. And finally, that the adversary has full control of the network.

The main objective of formal semantics of Scyther is to clearly distinguish the protocol descriptions from their behavior and the attacker model. Every security protocol has a number of distinct behaviors which are called as roles. For example, in the proposed scheme, the client and server are two 'roles'. A system consists of number of communicating 'agents', where each agent performs one or more role. Therefore, it can be said that, the system does not execute the protocol; instead, it executes the roles performed by agents. Each role performed by an agent is called as 'run'. While agents try to perform roles to achieve security, the attacker tries to oppose them by breaching their security. Each security protocol model can have the following components:

Scyther's specification language is called as Security Protocol Description Language and takes the file extension of .spdl. There are number of basic terms used in .spdl. These include:

Var: Variables that are used to store received messages.

Const: the fresh constants which are generated at each instantiation of role such as nonce, keys etc.

Role: The roles that an agent performs

Func: Represents the function names.

Scyther provides GUI to write the specification in addition to CUI interface. It also assists the protocol analysis by providing classes of attacks as compared to single attack provided by other tools. The tool can be used in three modes i,e. to verify the user defined claims, to verify the automatic claims generated by scyther and to analyze the performance of the protocol in terms of traces by characterization.

Description of Proposed Scheme: As per the specification the Agent model in the proposed scheme has two agents i,e. 'C' – Client and 'S' – Server. Each agent performs the roles, therefore this scheme has two roles that are named after the agents i,e. 'role I' and 'role S'. The adversary model is also designed considering that the adversary has complete control of the network. Therefore, Eve is considered as adversary. Since Scyther checks for the freshness and synchronization by default, those attributes have also been claimed.

Once the tool is run for automated verification, all the claims are analyzed against the automated attacks that the tool has. If any one of the claim is failed to resist the attack, the result of verification will be "Fail"; conversely, if the scheme resists all the attacks then the result will be 'OK' for all claims. For the proposed scheme, the parameters that are being transmitted over the channel were provided as claims. These include, h(UIDi), E_{yi}(Ci, a). The result of the automated verification of proposed scheme is found to be successful as depicted in figure 3. The .spdl is given below.

```
/*
 * mobile-based-TFA protocol
 */
// The protocol description
secret x : Function;
const equal : Function;
const hash : Function;
const XOR : Function;
//const r : Nonce;
const TID : Nonce;
hashfunction H1;
protocol mobile-based-TFA (I,R)
{
        role I
        {
                const i,P;
                fresh HA : Ticket;
                var y : Nonce;
                fresh r : Nonce;

                send_1 (I,R,{I,r}pk(R));
                recv_2 (R,I,{r,y,R}pk(I));
                send_3
(I,R,{H1(XOR(XOR(H1(XOR(H1(i),P)),XOR(H1(XOR(H
1(i),P)),H1(XOR(x,TID)))),r)),y}pk(R),{H1(HA,y)}pk(R));
        claim_i1
(I,Secret,XOR(H1(H1(XOR(H1(i),P))),XOR(H1(XOR(H1(
i),P)),H1(XOR(x,TID)))));
                claim_i2
(I,Secret,H1(XOR(XOR(H1(XOR(H1(i),P)),XOR(H1(XO
R(H1(i),P)),H1(XOR(x,TID)))),r)));
                claim_i3 (I,Secret,HA);
                claim_i4 (I,Niagree);
                claim_i5 (I,Nisynch);
        }
        role R
        {
                var HA : Ticket;
                var r : Nonce;
                fresh y : Nonce;
                var Ci : Ticket;
                recv_1 (I,R,{I,r}pk(R));
                send_2 (R,I,{r,y,R}pk(I));
                recv_3
(I,R,{Ci,y}pk(R),{H1(HA,y)}pk(R));
                claim_r1 (R,Secret,H1(XOR(x,TID)));
                claim_r2 (R,Secret,Ci);
                claim_r3
(R,Secret,H1(XOR(H1(XOR(x,TID)),r)));
                claim_r4 (R,Secret,HA);
                claim_r5 (R,Niagree);
```

claim_r6 (R,Nisynch); } }

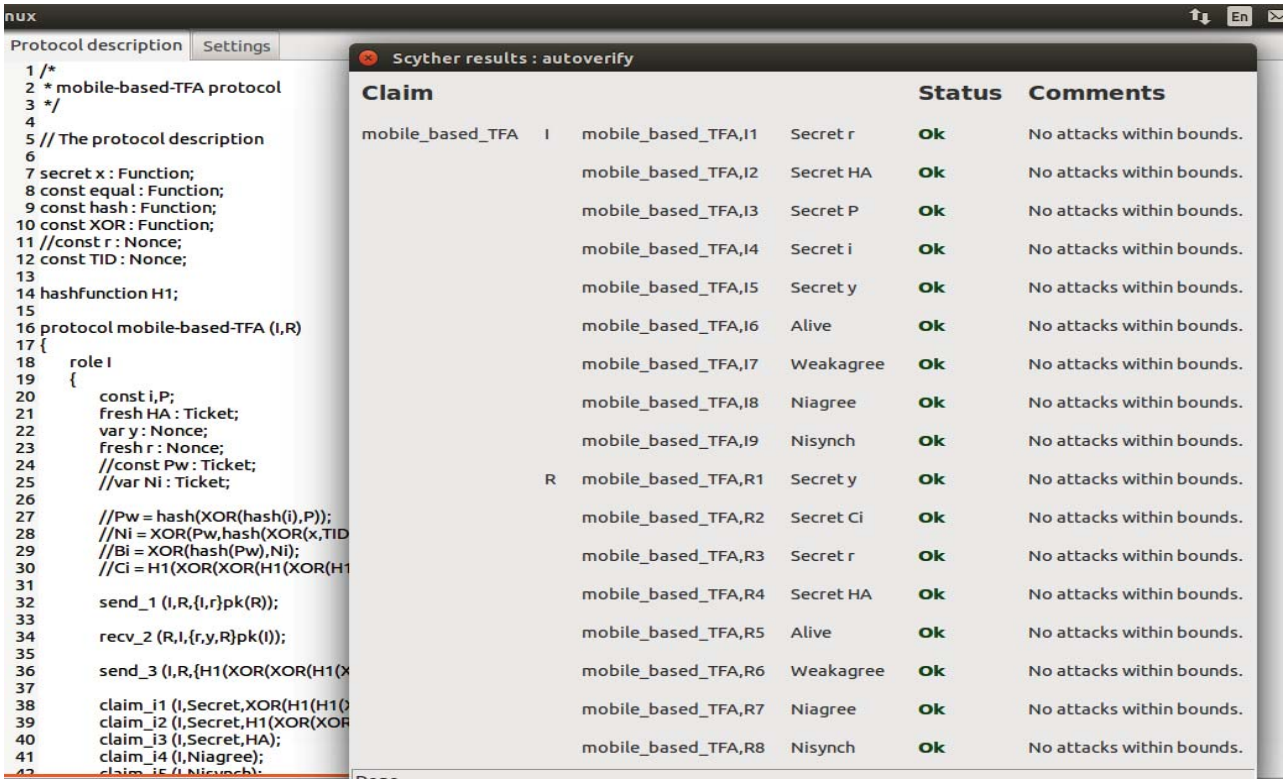


Fig 3: Result of Automated verification of proposed Scheme

5. Conclusion

This paper presented a Multi-Factor Authentication (MFA) Scheme for e-Services which does not require verifier table at the server. The paper discussed the security analysis of the proposed scheme against common attacks and the automated attacks using theoretical and formal analysis respectively. The authentication scheme is usable and secure as it considers image as the first factor, mobile token as second and password as the third factor. The Multi-factor security is required as the scheme is used with SSO for accessing multiple services in a single login session. The registration phase creates a verified identity of the user and prevents the fraudulent creation of duplicate accounts. This is accomplished using CIDR verification of Indian residents. However, for large scale deployment of this scheme, if the Citizen ID and National ID databases of developed and developing nations provide APIs to verify the validity of their residents then the issuance of verified identity would be a reality.

6. References

[1] Shunmuganathan, S. ; Saravanan, R.D. ; Palanichamy, Y., "Secure and Efficient Smart-Card-Based Remote UserAuthentication Scheme for Multiserver Environment" Electrical and Computer Engineering, Canadian Journal of Volume: 38 , Issue: 1, 2015 , Page(s): 20 - 30

[2] Das M. L., Saxena A. and Gulati V. P., "A dynamic ID based remote user authentication scheme", IEEE Trans. Consumer Electronics, May, vol.50, No. 2, 2004, Pg. 629 - 631

[3] Jenq-Shiou Leu ; Wen-Bin Hsieh, "Efficient and secure dynamic ID-based remote user authentication scheme for distributedsystems using smartcards, Information Security, IET Volume: 8 , Issue: 2, 2014 , Page(s): 104 - 113H.

[4] Haw Lee ; Wei-Chih Hong ; Chia-Hung Kao ; Chen-Mou Cheng, "A User-Friendly Authentication Solution Using NFC Card Emulation on Android", IEEE 7th International Conference on Service-Oriented Computing and Applications (SOCA), 2014, Page(s): 271 – 278

[5] Binu, Sumitra ; Misbahuddin, Mohammed ; Raj, Pethuru, "A Single Sign on based secure remote user authentication scheme for Multi-Server Environments", International Conference on Computer and Communications Technologies (ICCCT),2014, Page(s): 1 – 6

[6] Kumari, S. ; Om, H.," Remote login password authentication scheme usingtangent theorem on circle in a multi-server environment" First International Conference on Networks & Soft Computing (ICNSC), 2014, Page(s): 76 - 80

[7] Misbahuddin M, Ahmed M.A, Rao A.A, Bindu C.S, Khan M.A.M, "A Novel Dynamic ID-Based Remote User Authentication Scheme", in the proceedings of Annual IEEE Indicon Conference, Delhi, 2006

[8] Mohammed Misbahuddin; Mohammed Aijaz Ahmed; M.H. Shastri, "A Simple and Efficient Solution to Remote User Authentication Using Smart Cards", in the proceedings of IEEE International Conference on Innovations in IT (IIT '06), Dubai, 2006

[9] Omar Cheikrouhou, Manel Boujelben, Anis Koubaa, Mohamed Abid, Attacks and Improvement of "Security Enhancement for a Dynamic ID-based Remote User Authentication Scheme", in the proceedings of IEEE

International Conference on Computer Systems and Applications, 2009.

[10] Bruce Schneier, Applied Cryptography, 2nd edition. John Wiley & Sons, 1996.

[11] Cremers CJF, "Scyther - Semantics and Verification of Security Protocols", Phd Thesis, http://alexandria.tue.nl/extra2/200612074.pdf

[12] Cas Cremers, "The Scyther Tool Verification, Falsification, and Analysis of Security Protocols", Tool Paper, http://people.inf.ethz.ch/cremersc/downloads/papers/The Scyther Tool: Verification, Falsification, and Analysis of Security Protocols.pdf

[13] http://www.internetlivestats.com/internet-users/ Last accessed March 31, 2015

[14] http://www.emarketer.com/Article/Smartphone-Users-Worldwide-Will-Total-175-Billion-2014/1010536, Last accessed March 31, 2015

[15] The World in 2013 - http://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2013-e.pdf Last accessed March 31, 2015

[16] Aadhaar Authentication Overview, Aadhaar Authentication Overview, https://uidai.gov.in/auth.html Last accessed March 31, 2015

[17] http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf, Last accessed March 31, 2015

[18] http://en.wikipedia.org/wiki/National_identity_cards_in_the_European_Economic_Area, Last accessed March 31, 2015

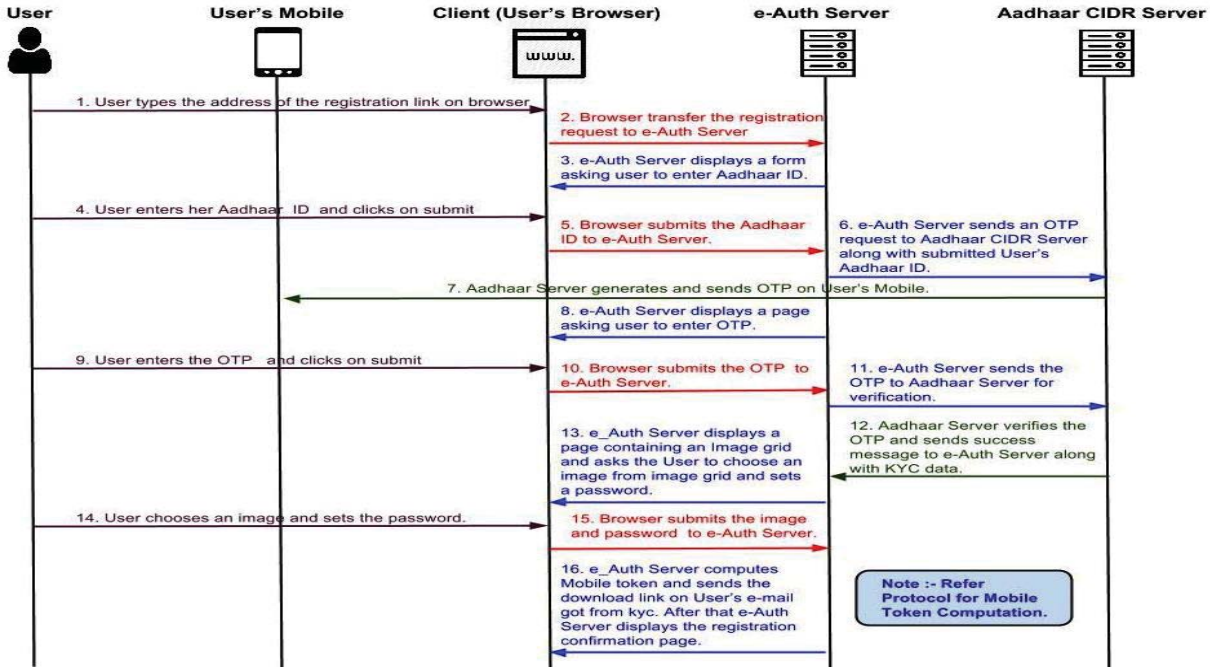


Fig 4: Registration Process of proposed Scheme

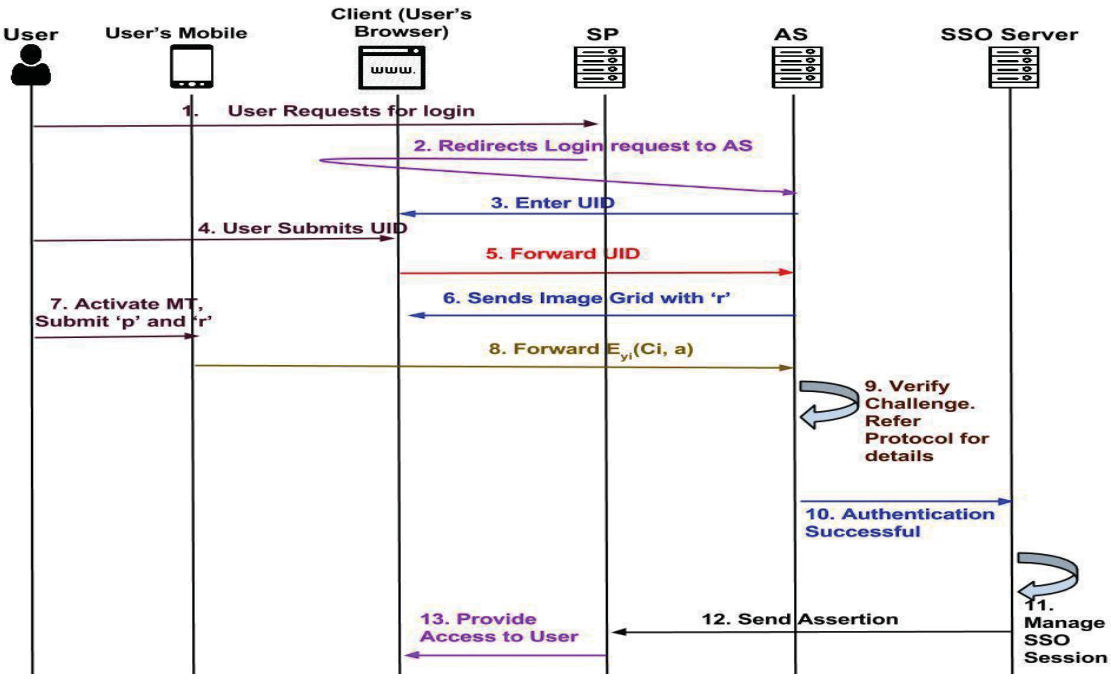


Fig 5: Authentication Process of proposed Scheme

A Strong Single Sign on User Authentication Scheme without Verifier Table for Cloud Based Services

B. Sumitra[1], M. Mohammed[2], and R. Pethuru[3]
[1]Research Scholar, Computer Science, Christ University, Bangalore, Karnataka, India
[2]Senior Technical Officer, C-DAC, Electronic City, Bangalore, Karnataka, India
[3]Infrastructure Architect, IBM India Pvt. Ltd., Bangalore, Karnataka, India

Abstract - *Cloud computing is an emerging computing paradigm that offers computational facilities and storage as services dynamically on demand basis via the Internet. The ability to scale resources and the pay-as-you-go usage model has contributed to its growth. However, Cloud computing inevitably poses various security challenges and majority of prospective customers are worried about who has access to their data. Service providers need to ensure that only authorized users access the resources and for this they need to adopt strong user authentication mechanisms. The mechanism should provide users with the flexibility to access multiple services without repeated registration and authentication at each provider. Considering these requirements, this paper proposes a Single Sign on based two factor authentication protocol for Cloud based services. The proposed scheme uses Password and a crypto-token as authentication factors and does not require a verifier table. The formal verification of the protocol is done using Scyther.*

.

Keywords: Cloud; Two-Factor Authentication; Single Sign-On; Cryptotoken; Scyther

1 Introduction

Revolutionary advances in hardware, networking, middle-ware and virtual machines have led to the emergence of the utility based distributed computing model, viz. Cloud Computing that provides computation facilities and storage as services accessible via the Internet. Cloud computing offers individuals and companies' affordable storage, professional maintenance and adjustable space without much investment in new infrastructure, training or software licensing. The elasticity and scalability of resources, combined with ``pay-as-you-go'' resource usage has heralded the rise of Cloud Computing. Infographic reports that 63% of financial services, 62% of manufacturing, 59% of health care and 51% of transportation industries are using Cloud computing services [1]. Rackspace reports that this pay-as-you-go service saves around 58% of cost [2].

By 2016 more than 50% of global 1000 companies are projected to store their sensitive data in Public Clouds [3]. Anticipating this switch over, many large technology companies such as Amazon and Google have built huge data centers to offer Cloud computing services with self-service interface so that Cloud users can use on-demand resources with location independence. Though the self-service interface provided by Cloud enables users to access the resources without human interaction with the service provider, the indirect control of the physical infrastructure introduces many vulnerabilities unknown in a non-Cloud environment. The Cloud model for delivering computing and processing power has raised many security concerns such as data security, identity and access management, key management and Virtual machine security which could limit the use of Cloud computing. Furthermore, in [4,5] authors have pointed out that identity and access management issues in Cloud requires immediate attention of Cloud Service Provider's (CSP's) to accelerate the adoption of Cloud. A survey by Fujitsu Research Institute [6] reveals that 88% of prospective customers are worried about unauthorized access to their data in the Cloud. To provide secure access to sensitive data, CSP's need to ensure that only valid users are accessing the resources and services hosted in the Cloud and to make this possible they need to adopt strong user authentication mechanisms.

Password authentication is the most commonly used authentication mechanism but irrespective of the strength of the passwords it is found to be susceptible to various attacks [7]. Furthermore, for traditional remote login mechanisms, a user needs to register with different SP's and remember various identities and passwords for ensuring higher security in a multi-server environment (MSE). This may cause user inconvenience, since users can remember only around seven passwords [8]. Therefore, in a MSE, single registration to a trusted registration center is a primary requirement and users can receive desired services from various service providers without repeating registration and by using a single login credential. Single Sign on (SSO) approach satisfies this requirement by allowing users to register once at the Identity provider and thereafter access multiple

Track: Security Applications – Cloud Computing security

services hosted in different domains using the same password.

Relying on a single password to access different accounts can result in account take over at many sites, in case the single password is compromised. Strong authentication mechanisms address this issue by authenticating users based on a combination of two or more factors. Taking into consideration the storage and computational capabilities of smart cards, a number of password based authentication scheme with smart cards have been proposed [9,10]. Most of the proposed schemes assume that the smart card is tamper resistant and recent research results have revealed that the secret information stored in the smart card could be extracted by some means such as monitoring the power consumption [11] and analyzing the leaked information [12]. Therefore such schemes based on the tamper resistance assumption of smart cards are prone to attacks such as impersonation attack, password guessing attack etc. once an adversary has obtained the secret information stored in a smart card. Biometric authentication mechanisms are also quite popular and biometric identifiers are difficult to forge. Biometrics is unique to the individual and non-transferable, but biometric authentication mechanisms have the drawback of being costly as they need additional hardware to read and process the stored data. Hence, there is an immediate requirement to design strong authentication mechanisms that maintains a good level of usability.

This paper discusses a two-factor authentication mechanism for cloud based services. The proposed scheme uses crypto-token and password as the authentication factors. The proposed protocol does not require a Password verifier table at the Server and provides SSO functionality using Security Assertion Mark-up Language (SAML) protocol [13].

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 discusses the authentication architecture & protocol and section 4 analyzes the security of the proposed scheme. Section 5 includes the efficiency analysis of the proposed scheme, section 6 discusses the formal analysis of the protocol using Scyther tool and section 7 concludes the work done.

2 Related Work

This section discusses a few user authentication schemes proposed for Cloud environment.

Hao et al. [14] in 2011 proposed a time-bound ticket based mutual authentication scheme for Cloud computing. In their scheme the authors follow an authentication model similar to that of Kerberos where in a user, to access the services from an Application server should first authenticate himself to and get tickets from a ticket granting server. Authors claimed that their scheme provides mutual authentication and is secure against lost smart card attack, offline password guessing attack, lost ticket attack, masquerade attack and replay attack.

In 2013, Jaidhar et al. [15] proved that the scheme [14] is susceptible to Denial of service (DoS) attack and the password change phase requires the involvement of the Server. Authors proposed an improved mutual authentication scheme which inherited the security measures of Hao et al.'s scheme and was resistant to DoS attack.

Choudhary et al. proposed an user authentication framework for Cloud environment [16] that provides two-step verification using password, Smart Card and out of band (OOB) authentication token. The scheme uses an OTP sent to the user via his e-mail ID as the out of band authentication factor. Authors claimed that their scheme provides identity management, mutual authentication, session key agreement etc. and is resistant to various attacks. Rui Jiang [17] in 2013 proved that their scheme is prone to masquerade user attack, the OOB attack, and has a flaw in the password change phase. They proposed a modified scheme that addresses the security issues of [16], but uses time stamps which can lead to time synchronization problems in a distributed Cloud environment especially when client and server are from two different time zones. Also the protocol requires the server to store a variant of the user password, which can result in a stolen verifier attack.

Sanjeet et al. [18] proposed a user authentication scheme which uses symmetric keys to exchange communication between user and server in which case key distribution may be a challenge. The protocol uses a one-time token which is sent to the registered users e-mail id. In this scenario, the authentication process will require logging into two accounts which may cause user inconvenience.

The above discussed authentication schemes [14-18] do not provide the SSO functionality which is preferable in a multi-server environment as it enhances user convenience. Also the scheme [14-17] uses Smart Cards that require an additional device to read/write which has an additional cost implication.

3 Proposed Scheme

3.1 Architecture of Proposed Scheme

The proposed architecture includes three participants' viz. an Identity Provider (IdP), Cloud SP's and users'. The users and SP's, need to register with the IdP. A user who attempts to access the services of a SP without registering at IdP, will be redirected by the SP to the IdP's registration page. After registration, the user should be authenticated by IdP before accessing the services provided by SP. IdP and SP work in a trust based environment. A user who tries to login to the SP after the registration process will be redirected by the SP to IdP. IdP will authenticate the user and send the

authentication response to SP. The process flows of the registration and authentication stages are depicted in Figure 1.

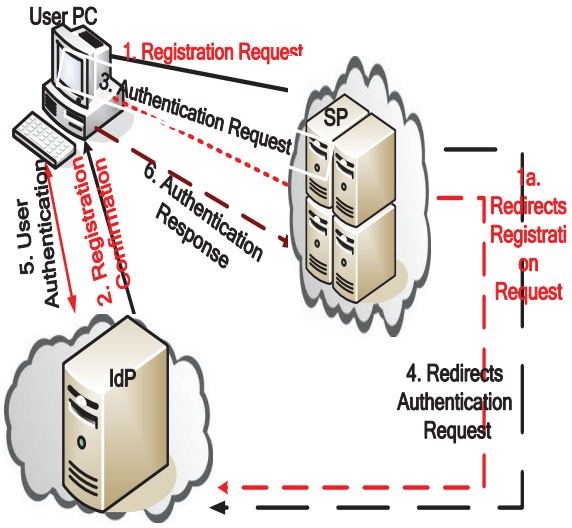


Fig.1 Registration & Authentication Process

3.2 Phases of Proposed Scheme

The proposed scheme consists of Initialization, registration, Login & Authentication and Password change phase. The notations used are listed in Table I.

TABLE 1. NOTATIONS

U_a, IdP, SP	User 'a', Identity Provider, Service provider
ID_a, P_a	Identity, Password of user U_a.
SID, y	Server ID of IdP, Secret key of IdP
G	Additive cyclic group of prime order
g_0	Generator of additive cyclic group
r	Random number generated by Audi Pass unique to each session
h(.) , ⊕ , ||	hash function, XOR operation, Concatenation Operation
⇒	Secure Communication Channel

3.2.1 Initialization Phase

During this phase, U_a generates a finite additive cyclic group 'G' of prime order 'n' with 'g_0' as the generator.

3.2.2 Registration Phase

If user wants to register for the services of a SP, the user U_a clicks the "Create Account" link at SPs web site. SP redirects U_a to the registration page of the IdP. IdP prompts U_a to submit the Identity and Password of the user. U_a chooses her identity ID_a and Password P_a and the phase proceeds as illustrated in figure 2, which can be explained as follows.

UR1: U_a Computes $b = h(P_a)$, $k = g_0^b$ and submits $h(ID_a)$, k to IdP through a secure channel. IdP checks whether the submitted $h(ID_a)$ already exists in its user table and if so prompts U_a to submit a new ID, otherwise IdP proceeds as follows:

IdP computes $B_i = h(h(ID_a)||h(SID))$; $J_i = h(SID||h(y)) \oplus k$; $C_i = h(h(ID_a) || h(SID||h(y))|| k)$; $E_i = B_i \oplus h(SID||h(y))$ where 'y' is a secret key of IdP and h(.) is a one way hash function.

UR2: IdP personalizes the crypto-token with the parameters C_i, E_i, J_i, h(.) and sends the crypto-token to U_a via a secure channel. On receiving the device, U_a stores g_0 into the crypto-token which now contains $\{C_i, E_i, J_i, h(.), g_0\}$.

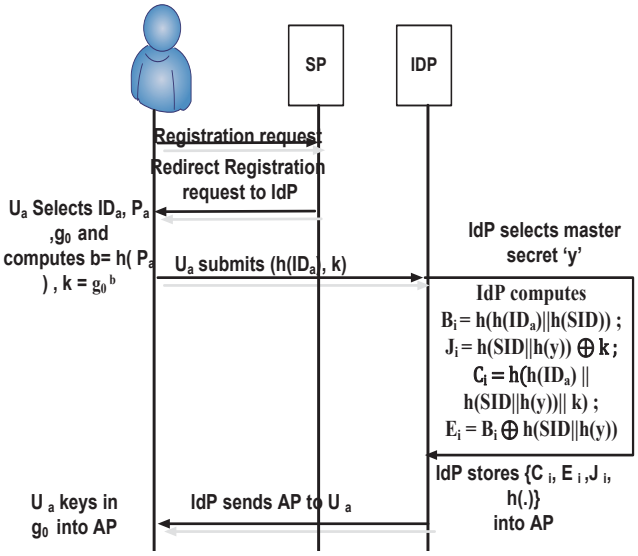


Fig.2 Registration Phase

3.2.3 Login and Authentication Phase

Whenever a registered user wants to login to access the services of the Service Provider 'SP', she attaches the crypto- token to the system and proceeds as follows:

UL1: U_a requests for login to the SP. SP checks for an existing session with U_a and if there is no valid session, SP redirects U_a to IdP with a SAML authentication request.

UL2: U_a keys in her ID_a and P_a. crypto-token computes, $b = h(P_a)$, $k^* = g_0^b$

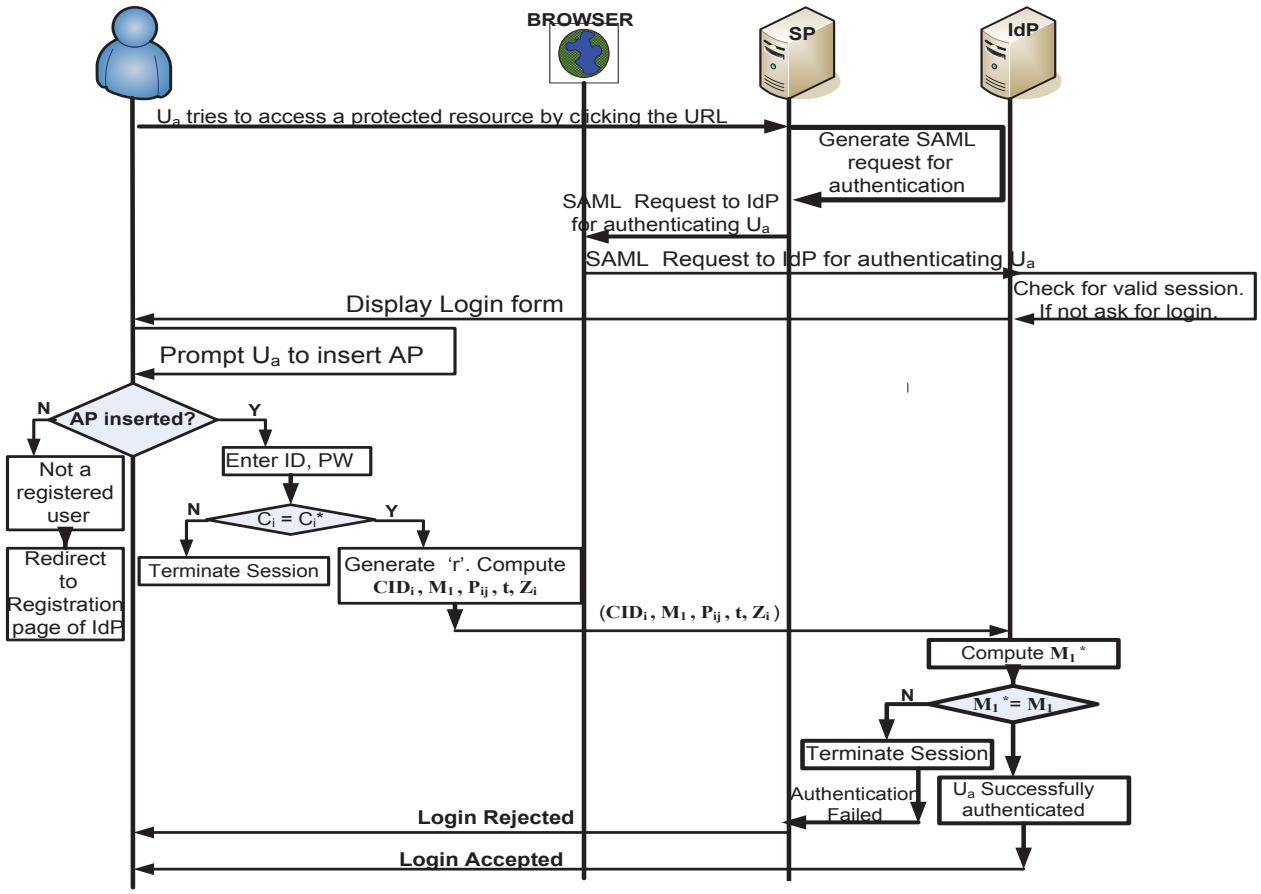


Fig3. Login &Authentication Phase

UL5: crypto- token computes $h(SID||h(y))^* = J_i \oplus k^*$, $C_i^* = h(h(ID_a) \,||\, h(SID||h(y))^* || k^*)$ and compares with C_i stored in the crypto- token. If invalid, AP terminates the session. Otherwise generates the login message as follows:

UL6: crypto- token generates a random number 'r' and computes nonce $n_1 = g_0^r$. crypto- token computes $P_{ij} = E_i \oplus h(h(SID||h(y))||n_1)$; $B_i = E_i \oplus h(SID||h(y))$; $CID_i = C_i \oplus h(B_i||n_1||SID)$; $M_1 = h(Pij \,||C_i \,||B_i||n_1)$; $t = g_0 \oplus h(SID||h(y))$; $Z_i = (r - CID_i) \oplus h(SID||h(y))$ and sends $(CID_i, M_1, P_{ij}, t, Z_i)$ to IdP

UL7: Upon receipt of the login message the IdP performs the authentication process using her own SID and h(y) values

UL8: IdP computes, $r = (Z_i + CID_i) \oplus h(SID||h(y))$; $g_0 = t \oplus h(SID||h(y))$; $n_1^* = g_0^r$, $E_i = P_{ij} \oplus h(h(SID||h(y))||n_1)$; $B_i^* = E_i \oplus h(SID||h(y))$; $C_i^* = CID_i \oplus h(B_i^*||n_1^*||SID)$;

UL9: IdP computes $M_1^* = h(Pij \,||C_i^* \,||B_i^*||n_1^*)$ and compares with the M_1 in the login message received from U_a. If valid, IdP considers the authentication as successful. IdP creates a response message containing the result of the authentication process and redirects it to the SP. The SP permits or denies access to the services after verifying the response from the IdP.

3.2.4 Password Change Phase

U_a attaches his crypto-token into the system and keys in his ID_a and P_a. crypto-token computes C_i^* using ID_a , P_a and compares with C_i stored in the crypto-token. If invalid, crypto-token terminates the session. Otherwise prompts U_a to enter the new password P_{anew}. U_a enters P_{anew}. AP computes b_{new} , k_{new}; J_{inew} , C_{inew} and replaces C_i and J_i in the crypto-token with C_{inew} and J_{inew} respectively.

4 Security Analysis of Proposed Protocol

4.1 Security against Guessing Attack

The proposed protocol is secure against guessing attack as it is impossible within polynomial time, for an adversary to retrieve user's password P_a or IdP's secret key from the intercepted parameters $(CID_i, M_1, P_{ij}, t, Z_i)$.

4.2 Security against Malicious Insider Attack

In the proposed scheme, user submits $k = g_0^{h(Pa)}$ to IdP rather than the plain text form of the password. This guards the password from being revealed to IdP and hence even if the user uses the same password to login to other servers, her credentials will not be susceptible to insider attack

4.3 Security against Replay Attack

The scheme is resistant to replay attack since nonce values used to in each authentication message is unique and varies for each session. Hence the IdP will be able to identify a replayed login message (CID_i, M_1, P_{ij}, t, Z_i) by checking the freshness of nonce, n_1 which is unique to a session.

4.4 Security against Stolen Verifier Attack

The proposed scheme does not require a verifier/password table and hence is resistant to Stolen Verifier attack.

4.5 Security against User Impersonation

If an adversary attempts to impersonate a valid user, he should be able to forge a valid login request on behalf of the user. In the proposed scheme if an adversary intercepts the login message (CID_i, M_1, P_{ij}, t, Z_i) and attempts to generate a similar message, he will fail since the value of nonce 'n_1' as well as the server's secret key 'y' is unknown to him.

4.6 Security against DoS Attack

A DoS attack can be launched by an adversary by creating invalid login request messages and bombarding the server with the same. This attack can also be launched by an adversary who has got control over the server and is able to modify the user information stored in the server's database which in turn prevents the valid user from accessing the resources.

The first scenario will not work in the case of the proposed scheme, since it is impossible for the adversary to create valid login request messages without knowing the password. The validity of the password is checked at the client side before creating a login request. The second scenario is also not applicable in the proposed scheme, since the server does not maintain a verifier/password table.

4.7 Security against crypto-token Lost Attack

If the adversary steals the crypto-token containing the parameters (C_i, E_i, J_i, h(.), g_0), he can neither retrieve the user's password nor the IdP's master secret 'y' from the stored value. To extract the password from $k = g_0^{h(Pa)}$, the adversary needs to solve the discrete logarithm problem . Again the password is used in the hashed form which is irreversible.

5 Efficiency Analysis of Protocol

This section analyzes the efficiency of the proposed protocol scheme in terms of the computational and the communication cost. It is assumed that ID_a , PW_a, nonce values are 128 bits long and the output of hash function(SHA-2) is 256 bits long. Let T_h, T_x and T_E denote the time complexity for hashing and XOR and exponentiation operation respectively. In the protocol, the parameters stored in the crypto-token are C_i, E_i, J_i and g_0 and the memory (E1) needed in the crypto-token is 896 (3*256+128) bits. Communication cost of authentication (E2) includes the capacity of transmitting message involved in the authentication. The capacity of transmitting message (CID_i, M_1, P_{ij}, t, Z_i) is 1280 (3*256) bits. The computation cost of user registration (E3) is the total time of all operations executed in this phase by the user and IdP and is equal to $7T_h+ 2T_x+ 1T_E$ .The computation cost of the user (E4) and the IdP (E5) is the total time of all operations executed by the crypto-token and IdP during login and authentication. During authentication, the crypto-token performs 6 hash functions, 6 XOR and 2 exponentiation making E4 equal to $6T_h+ 6T_x+ 2T_E$. Similarly E5 is $3T_h+ 5T_x+ 1T_E$.

6 Formal Analysis Using Scyther

6.1 Scyther Tool

Automatic tools are preferred in protocol analysis and among the various available tools, Scyther [19] is used for the verification of the proposed protocol. Scyther provides a graphical user interface which incorporates the Scyther command line and python scripting interface. The description of a protocol and the claims in Scyther are written in Security Protocol Description Language (SPDL). The proposed protocol can be written in SPDL as follows.

```
//Login and Authentication Phase
const exp: Function; hashfunction h;
const XOR: Function; const h1:Function; const diff: Function;
protocol ssauth (I,R){
role I {
const ID, x, y,r,g, SID,n1,p,t;
send_1(I,R, (XOR(( h( h(ID), h(SID,h(y)), h1(g,h(p)))),(h((
h(h(ID),h(SID))),h1(g,r ), (SID), //CIDi
(h((XOR((XOR((h(h(ID),h(SID))),(h(SID,h(y))))
),(h(h(SID,h(y)),h1(g,r) ))),(h( h(ID), h(SID,h(y)),
h1(g,h(p)))), (h(h(ID),h(SID))), h1(g,r)), //Mi
(XOR((XOR((h(h(ID),h(SID))),(h(SID,h(y))))
),(h(h(SID,h(y)),h1(g,r) ))), //Pij
(XOR((g),h(SID,h(y)))), //t
(XOR((diff((r),(XOR((h( h(ID), h(SID,h(y)), h1(g,h(p)))),(h((
h(h(ID),h(SID))), h1(g,r ), (SID) ))))),h(SID,h(y))))) )))));
//Zi
claim_i1(I,Secret,
(XOR((XOR((h(h(ID),h(SID))),(h(SID,h(y))))
),(h(h(SID,h(y)),h1(g,r) ))))); //claim for pij
claim_i2(I,Secret,XOR(( h( h(ID), h(SID,h(y)),
h1(g,h(p)))),(h(( h(h(ID),h(SID))),h1(g,r ), (SID) )))); //claim
for CID
claim_i3(I,Secret, XOR((diff((r),(XOR((h( h(ID),
h(SID,h(y)), h1(g,h(p)))),(h(( h(h(ID),h(SID))), h1(g,r ),
(SID) ))))),h(SID,h(y)))); //claim for Zi
```

```
claim_i4(I,Secret,h((XOR((XOR((h(h(ID),h(SID))),(h(SID,h(
y)))) ),(h(h(SID,h(y)),h1(g,r) )))),(h( h(ID), h(y),
h1(g,h(p)))), (h(h(ID),h(SID))), h1(g,r))); //claim for Mi
claim_i5(I,Secret, XOR((g),h(SID,h(y)))); //claim for t
claim_i6(I,Secret, h1(g,r));
claim_i7(I,Niagree);
claim_i8(I,Nisynch);
}
role R{
const ID,x,y,r,g,SID,n1,p,t;
recv_6(I,R,
(XOR(( h( h(ID), h(SID,h(y)), h1(g,h(p)))),
(h((h(h(ID),h(SID))),h1(g,r), (SID),
(h((XOR((XOR((h(h(ID),h(SID))),(h(SID,h(y))))),
(h(h(SID,h(y)),h1(g,r) )))),(h( h(ID), h(SID,h(y)),
h1(g,h(p)))), (h(h(ID),h(SID))), h1(g,r)),
(XOR((XOR((h(h(ID),h(SID))),(h(SID,h(y))))),(h(h(SID,h(y)
),
h1(g,r) )))), (XOR((g),h(SID,h(y)))),
(XOR((diff((r),(XOR((h( h(ID), h(SID,h(y)), h1(g,h(p)))),
(h(( h(h(ID),h(SID))), h1(g,r ), (SID) ))))),h(SID,h(y)))))
)))));

claim_r1(R,Secret,(XOR((XOR((h(h(ID),h(SID))),(h(SID,h(y
)))) ),(h(h(SID,h(y)),h1(g,r) )))));

claim_r2(R,Secret,XOR(( h( h(ID), h(y), h1(g,h(p)))),(h((
h(h(ID),h(SID))),h1(g,r ), (SID) ))));

claim_r3(R,Secret, XOR((diff((r),(XOR((h( h(ID),
h(SID,h(y)), h1(g,h(p)))),(h(( h(h(ID),h(SID))), h1(g,r ),
(SID) ))))),h(SID,h(y))));

claim_r4(R,Secret,h((XOR((XOR((h(h(ID),h(SID))),(h(SID,h
(y)))) ),(h(h(SID,h(y)),h1(g,r)) ))),(h( h(ID), h(y),
h1(g,h(p)))), (h(h(ID),h(SID))), h1(g,r)));

claim_r5(R,Secret, XOR((g),h(SID,h(y))));

claim_r6(R,Niagree);

claim_r7(R,Nisynch);

}

}
```

6.2 Scyther Analysis Results and Interpretation

The protocol analysis model defined in Scyther is role based security model where in roles represent different behaviors. In order to analyze the protocol we assume the existence of an adversary in the communication network. The adversary's capabilities are as defined by Dolev-Yao Network threat model [20] and it is assumed that the network is completely or partially under the control of the adversary. The complete results of the analysis of the proposed protocol are shown in Fig. 4. The output of the verification process is described according to the following Scyther attributes.

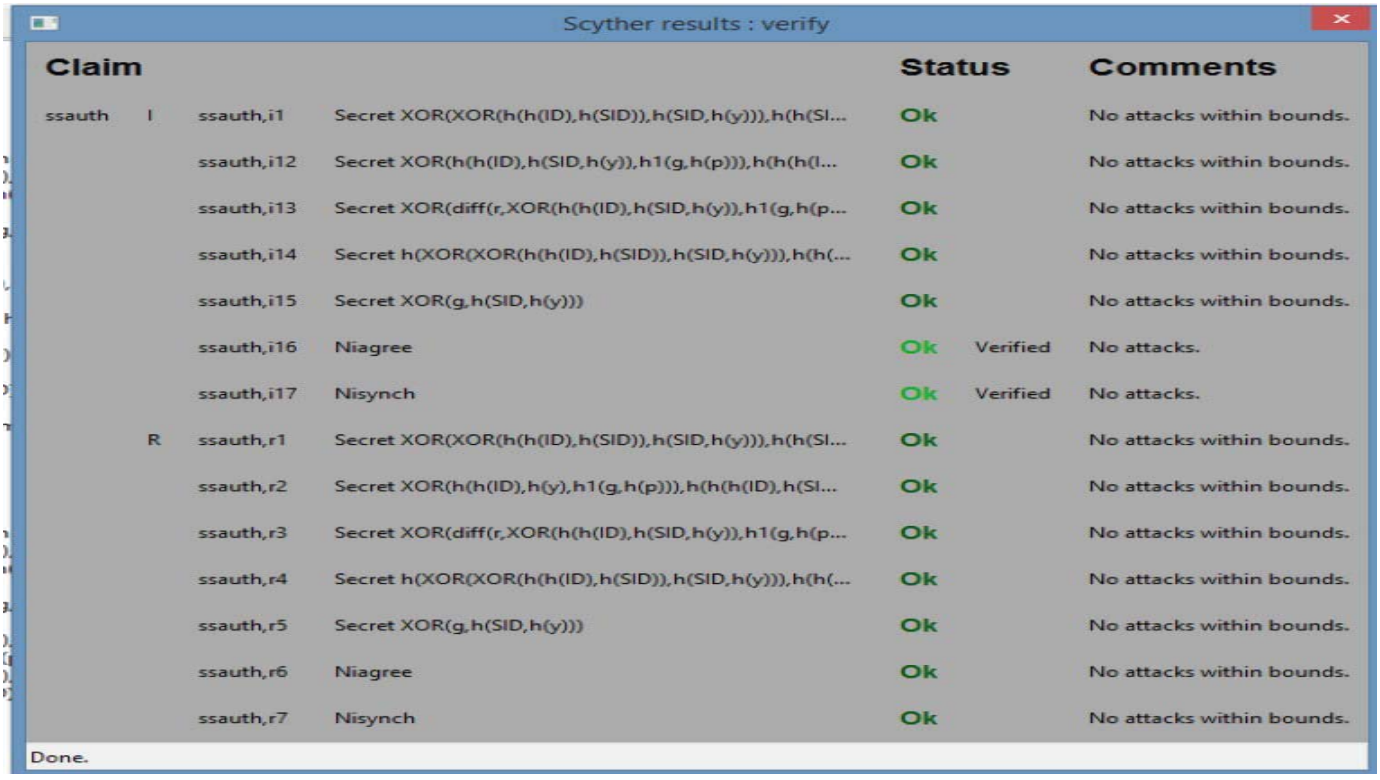


Fig.4 Protocol Verification Results generated by Scyther

Secrecy: The first claim is that the protocol ensures the confidentiality of the user's credentials. After analyzing, it is obvious from the results that the user's credentials are not revealed to the adversary when communicated over an untrusted network. As shown in Fig 4. The authentication parameters {y, g, P_{ij}, t, M_i, CID_i, Z_i} retain the confidentiality during the course of 10 protocol runs.

Non-Injective Agreement (NiAgree): Niagree claim made claims that sender and the receiver agree upon the values of variables exchanged and the analysis results justify the correctness of this claim.

*Synchronisation:*Ni-Synch or Non-Injective Synchronisation property requires that the corresponding send and receive Events (1) are executed by the runs indicated by the cast function, (2) happened in the correct order, and (3) have the Same contents. The proposed protocol satisfies this claim as indicated by the result of Scyther analysis.

7 Conclusions

The proposed authentication scheme provides the user with the flexibility to do single registration at the IdP and Sign on once during a session to access multiple services. The proposed protocol uses a password and a crypto-token as the authentication factors. The scheme uses SAML to provide SSO functionality and the scheme do not require a verifier table at the server. The paper discusses the security analysis of the proposed scheme against common attacks and the automated attacks using theoretical and formal analysis respectively

8 References

[1] G. Meijer, "5 Cloud Computing Statistics", Technical Report, Infographics, 2012

[2] B. Nicholson, A. Owrak, and L.Daly , Cloud Computing Research. Technical Report, Manchester Business School, Commissioned by RackSpace, 2013

[3] D.M. Smith, Y.V Natis, G.Petri, T.J Bittman, E.Knipp, P.Malinverno, and J.Feiman, "Predicts 2012: Cloud Computing is becoming a Reality", Technical Report G00226103, Gartner, 2011

[4] L.Ponemon, "Security of Cloud Computing Users," Ponemon Institute, research report, May 2010. http://www.ca.com/files/industryresearch/security-cloud-computing-users_235659.pdf

[5] F.Gens, "New IDC IT Cloud Services Survey: Top Benefits and Challenges," IDC Exchange, 2009, http://blogs.idc.com/ie/?p=730

[6] Fujitsu, "Personal Data in the Cloud: A Global Survey of Consumer Attitudes", Technical Report, Fujitsu research Institute, 2010

[7] D. Florencio and C. Herley, "A Large-Scale Study of Web Based Password Habits," In Proceedings of the 16th International Conference on World Wide Web (New York, NY, USA, 2007), WWW '07, ACM. Pp. 657-666

[8] J.Yan, A. Blackwell, R. Anderson, A. Grant, "Password Memorability and Security: Empirical Results," Security& Privacy, IEEE vol.2, 2004, pp. 25-31

[9] W.C Ku, S.M Chen, "Weaknesses and Improvements of an Efficient Password Based Remote User Authentication Scheme Using Smart Cards," IEEE Transactions Consumer Electronics 50(1), 204-207, 2004

[10] Y.C Chen, L.Y Yeh, "An Efficient Nonce-Based Authentication Scheme with Key Agreement," Applied Mathematics and Computation, 169(2), 982-994, 2005

[11] P. Kocher, J.Jaffe, B.Jun, "Differential Power Analysis," In: M. Wiener (ed.) CRYPTO 1999. LNCS, vol. 1666, pp.388-397. Springer, Heidelberg, 2010

[12] T.S Messerges, E.A dabbish, R.H Sloan , " Examining Smart Card Security Under the Threat of Power Analysis Attacks," IEEE Transactions on Computers 51(5), 541-552 , 2002

[13] OASIS, Security Assertion Mark Up Language, V2.0, Technical Overview, https://www.oasis-open.org/committees/download.php/20645/sstc-saml-tech-overview-2%200-draft-10.pdf , Online; accessed 29-APRIL-2015

[14] Z. Hao , S.Zhong, N.Yu, " A time-Bound Ticket Based Mutual Authentication Scheme for Cloud Computing," Inernational Journal of Computers, Communications & Control, vol. 6, 2011

[15] C.D Jaidhar, "Enhance Mutual Authentication Scheme for Cloud Architecture," in Proc. 3rd IEEE International Advanced Computing Conference (IACC), 2013

[16] J.C.Amlan, K.Pradeep, S.Mangal, E.L.Hyota, Hoon-Jue-Lee, "A Strong User Authentication Framework for Cloud Computing," *IEEE Asia-Pacific services Computing Conference*, 2011

[17] Rui Jiang, "Advanced Secure User Authentication framework for Cloud Computing", International Journal of Smart Sensing and Intelligent Systems, Vol. 6, No.4, September 2013

[18] Sanjeet Kumar Nayak, Subasish Mohapatra, Bansidhar Majhi, "An improved Mutual Authentication Framework for Cloud Computing", IJCA, vol.52-No.5, Aug.2012

[19] C.Cremers, "Scyther Semantics and Verification of Security Protocols," PhD dissertation: Eindhoven University of Technology, 2006

[20] D.Dolev and A.C Yao, "On the Security of Public-key Protocols," IEEE Transactions on Information Theory, 2(29): pp. 18-208, 1983

Color Image Steganalysis Method for LSB Matching

H. J. Olguin-Garcia, O. U. Juarez-Sandoval, M. Nakano-Miyatake, H. Perez-Meana.
SEPI, ESIME Culhuacan, Instituto Politecnico Nacional, Mexico City, Mexico

Abstract – *This paper presents a new method for color images steganalysis, in which stego-images have been generated by the LSB Matching steganography method, obtaining a database of 15000 images. For this method the Histogram Characteristic Function Center of Mass (HCF-CoM) is used to detect histogram changes in each color channel, and the distance among them is calculated as a form to obtain a unique value between origin ('0') and HCF-CoM values. The Probability Density Function (PDF) is used to find an adequate threshold value to achieve better performance. This paper includes a comparison with another LSB steganalysis method reported previously with the same objective of the proposed one. The obtained results show a better performance with low computational cost and easy implementation.*

Keywords: Stegalalysis, LSB Matching, Histogram Characteristic Function, Center of Mass

1 Introduction

In the XX century, the research about the steganography and steganalysis were not considered as a priority field, however in the XXI century, after the USA-9/11 attack, the study of this field has grown exponentially. During the last decade, large amount of methods have been proposed in the literature. In many cases, steganography is used to hide different types of information, such as a medical, business and personal information, for legal purpose. However steganography is also used for illegal purposes, such as pornography, people trade, counterintelligence, or to share confidential information among different government sectors, which may cause immense damages to society.

The Steganography is the art or science to hide information into digital cover files, such as audio, images, videos, etc. [1]. Its main goal is to achieve a secure undetectable subliminal communication between a transmitter and receiver sides. The principal issues to be considered in this field of study are the embedding capacity or payload, distortion or imperceptibility and robustness [2]. The steganography algorithms are divided into two different categories: spatial domain steganography [3] and frequency domain steganography [2]. Algorithms in the first category generally allow providing a large amount of payload of the hidden information; however they are not robust to different kinds of attacks, such as compression, rotation or noise-adding. Algorithms in the second category have the robustness but the payload is limited. The selection of steganography algorithm depends on its application and necessity. In both cases, the original cover file is not available in the secret message extraction stage.

Steganalysis, a counterpart of the steganography, is the detection study of hidden secret messages in digital files (stegofiles) [4-6]. In the literature several steganalysis methods in the spatial domain have been proposed, especially steganalysis for LSB Replacing steganography (LSB-R), while there are relatively less proposals of steganalysis for LSB Matching steganography method (LSB-M).

The LSB-R is the most popular steganography method due to its simplicity and easy implementation. In the LSB-R, the secret message is replaced by LSB bits of a cover image. This method generates a notable histogram change, which are exploiting by Westfeld-Pfitzmann [7] and Fridrich et al. [8] to design efficient steganalysis algorithms. The LSB-M [9,10] is a variation of LSB-R, where the pixel value increases (+1) or decreases (-1) as random form to match with the secret binary information unlike LSB-R, this method avoids notable histogram changes, therefore the steganalysis for LSB-M is more difficult. Nevertheless this little histogram changes are exploiting by Harmsen & Pearlman in [5], and D. Ker in [6]. They use the Center of Mass (CoM) of the Histogram Characteristic Function (HCF). Friedrich et al. [8] proposed an efficient steganalysis method for color images generated by LSB-R steganography method, using number of close color pairs; however this method cannot detect correctly, if stego-image is generated by LSB-M steganography.

The rest of this paper is organized as follow: section II provides a detailed description of HCF and Section III explains the HCF- CoM. In Section IV, the steganalysis idea proposed by J. Fridrich et al. is described and Section V provides a detailed description of the proposed method, experimental results and performance comparison are shown in Section VI. Finally the conclusion is done in Section VII.

2 Histogram Characteristic Function

Harmsen [5] analyzed the histogram of stego-images generated by LSB-M. Considering that the cover image h_c is modified for the addition of Stego-noise n in the values [-1, 0, +1], the LSB-M can be formulated by

$$h_s = h_c + f_\Delta[n] \tag{1}$$

where $f_\Delta[n]$ represents the probability that a pixel will be altered by n. $f_\Delta[0]$ is the probability that the pixel value is unchanged after embedding, hence $f_\Delta[+1]$ and $f_\Delta[-1]$ are the probabilities that pixel value is changed in 1 or -1, respectively, after embedding. In [5], $f_\Delta[0] = 0.5$, $f_\Delta[-1] = f_\Delta[-1] = 0.25$. Considering that h_c and $f_\Delta[n]$ are

two random continues independent variables, the equation (1) can be rewritten by (2)

$$h_s = h_c * f_\Delta[n] \tag{2}$$

Applying the DFT to (2), we get

$$H_s[k] = H_c[k] * F_\Delta[k] \tag{3}$$

$H_s[k]$ and $H_c[k]$ are histograms of cover image and stego-image, respectively, in the frequency domain.

3 HCF Center of Mass

The Center of Mass (CoM) is defined as (4) [11], which is an energy distribution of a Histogram.

$$C(H[k]) = \left| \frac{\sum_{k \in K} kH[k]}{\sum_{k \in K} H[i]} \right| = \frac{\sum_{k \in K} k|H[k]|}{\sum_{k \in K} |H[i]|} \tag{4}$$

where $K = \left\{0,\ldots,\frac{N}{2}-1\right\}$ and the N will be the DFT length. This equation is called HCF Center of Mass (HCF-CoM). Harmsen et al. demonstrated the following relation of CoMs between cover image and stego-image:

$$C(H_s[k]) \le C(H_c[k]) \tag{5}$$

4 Steganalysis of LSB Encoding in Color Images

Fridrich et al. introduced an efficient steganalysis method for LSB-R in color images [8], in which the relation of number of close colors pairs between natural image and stego-image is used. If two colors (R_1, G_1, B_1) and (R_2, G_2, B_2) are close, then $(R_1 - R_2)^2 + (G_1 - G_2)^2 + (B_1 - B_2)^2 \le 3$ must be satisfied.

Considering that a natural image has *U* unique colors and close color pairs P, when stego-image is generated by LSB-R, the number of unique color *U'* and also number of close color pairs *P'* are increased. However the increasing rate of close color pairs is much higher than that of unique color. The ratio *R* of number of close color respect to total number of color pair of natural image and the ratio *R'* for the stego-image are defined as

$$R = \frac{P}{\left(\frac{U}{2}\right)} \tag{6}$$

$$R' = \frac{P'}{\left(\frac{U'}{2}\right)} \tag{7}$$

The process of steganalysis proposed by [8] is as follows. First, intentionally the LSB-R is applied to an input image under analysis and two ratios R and R' of the input image and the stego-image, respectively, are obtained. Next, R and R' are compared. If $R \cong R'$ is satisfied, the input image is considered as stego-image, otherwise, $R' > R$, the input image can be considered as a natural image. In order to analyze systematically, $V = R'/R$ is compared with a predetermined threshold value *Th*. If $V > Th$ then the input image is natural image, otherwise the input image is considered as stego-image.

The selection of threshold value is important to obtain good performance of this steganalysis. In [8], a threshold value is calculated using the probability density function (PDF) denoted as $f_{\mu,\sigma}$ of the Gaussian distribution $N(\mu,\sigma)$, which are:

$$f_{\mu,\sigma} = \frac{e^{-\frac{(x-\mu)^2}{2\sigma^2}}}{\sqrt{2\sigma^2}} \tag{8}$$

$$f_{\mu(s),\sigma(s)} = \frac{e^{-\frac{(x-\mu(s))^2}{2\sigma(s)^2}}}{\sqrt{2\sigma(s)^2}} \tag{9}$$

where μ, $\mu(s)$ are means, and σ^2, $\sigma(s)^2$ are variances of natural image and stegoimage, respectively. Fig. 1 shows the distributions of two hypotheses: natural image and stego-image. The threshold value *Th* must be selected in order to minimize the false acceptance and false rejection error f_a and f_n at the same time. In [8], the threshold value is determined as $f_a = f_n$., which is given by

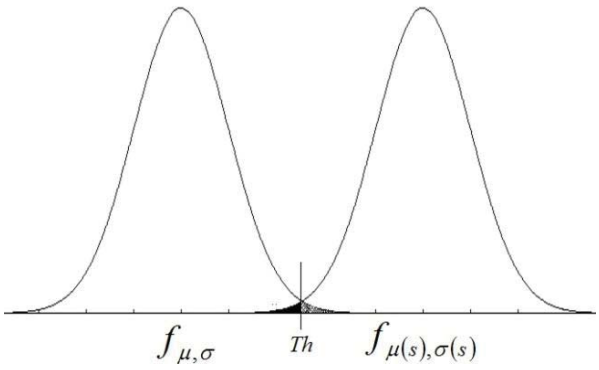


Fig.1.Intersection of PDF of two Gaussian PDFs.

$$f_{\mu,\sigma} = \int_{-\infty}^{Th} \frac{e^{-\frac{(x-\mu)^2}{2\sigma^2}}}{\sqrt{2\sigma^2}}\,dx = \int_{Th}^{\infty} \frac{e^{-\frac{(x-\mu(s))^2}{2\sigma(s)^2}}}{\sqrt{2\sigma(s)^2}}\,dx = f_{\mu(s),\sigma(s)} \tag{10}$$

Solving (10), we get:

$$\frac{Th-\mu(s)}{\sigma(s)} = \frac{Th-\mu}{\sigma} \tag{11}$$

Finally, the threshold value is obtained by

$$Th = \frac{\mu\sigma(s)+\mu(s)}{\sigma+\sigma(s)} \tag{12}$$

5 Proposed Steganalysis Method

The block-diagram of the proposed method is shown in Fig.2. The Test image Tim with color space RGB has a size MxN. This process is given by the following steps.

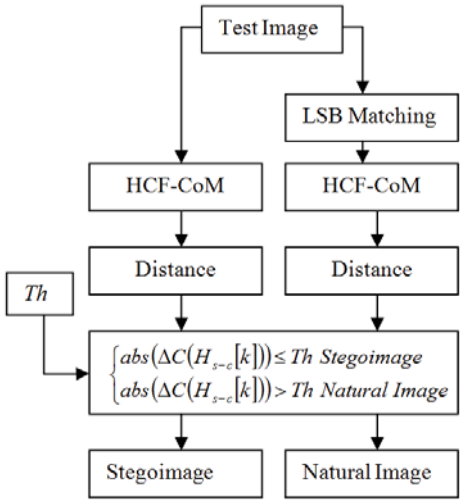


Fig.2. Proposed Method Process.

1. LSB-M steganography method [9, 10], is applied as a random form to Tim, the result is denoted as Tim'.

2. According with (4) the HCF-CoM is rewritten in terms of Tim and Tim' as (13),

$$C(H_{Tim}[k])_j = \left| \frac{\sum_{k\in K} kH_{Tim}[k]_j}{\sum_{k\in K} H_{Tim}[i]_j} \right| = \frac{\sum_{k\in K} k\left|H_{Tim}[k]_j\right|}{\sum_{k\in K}\left|H_{Tim}[i]_j\right|} \quad \forall\ j \in (R,G,B) \tag{13}$$

where $C(H_{Tim}[k])_j$ is the HCF-CoM and $H_{Tim}[k]_j$ is the HCF of every single channel $\forall\ j \in (R,G,B)$, the expression in Tim' is just the substitution of Tim by Tim'.

3. To express $C(H_{Tim}[k])_{R,G,B}$ and $C(H_{Tim'}[k])_{R,G,B}$ in a unique value the distance between the origin '0' and the values of the HCF-COMs is used as a single Cartesian three-dimensional coordinate as follows in (14) and (15),

$$C(H_{Tim}[k])_{R,G,B} = \sqrt{(C(H_{Tim}[k])_R)^2 + (C(H_{Tim}[k])_G)^2 + (C(H_{Tim}[k])_B)^2} \tag{14}$$

$$C(H_{Tim'}[k])_{R,G,B} = \sqrt{(C(H_{Tim'}[k])_R)^2 + (C(H_{Tim'}[k])_G)^2 + (C(H_{Tim'}[k])_B)^2} \tag{15}$$

This equation is called *Histogram Characteristic Function Center of Mass Distance* (HCF-CoM-D).

In terms of (14), (15) and according with (5) and its analysis, (16) and (17) are obtained; for this method and making the same analysis as [8], the PDF is obtained as (8) and (9), finding the same complication as them, those are shown in the Fig.3, as a Gaussian Distribution form. To obtain a better expression of (16) and (17) that helps us to reduce those errors, both equations are rewritten as (18), and $C(H_{Tim'}[k])_{R,G,B}$, Tim' has the 100% hidden information using LSB-M.

$$C(H_{Tim'}[k])_{R,G,B} \cong C(H_{Tim}[k])_{R,G,B} \tag{16}$$

$$C(H_{Tim'}[k])_{R,G,B} \le C(H_{Tim}[k])_{R,G,B} \tag{17}$$

$$abs(C(H_{Tim'}[k])_{R,G,B} - C(H_{Tim}[k])_{R,G,B}) \cong 0 \tag{18}$$

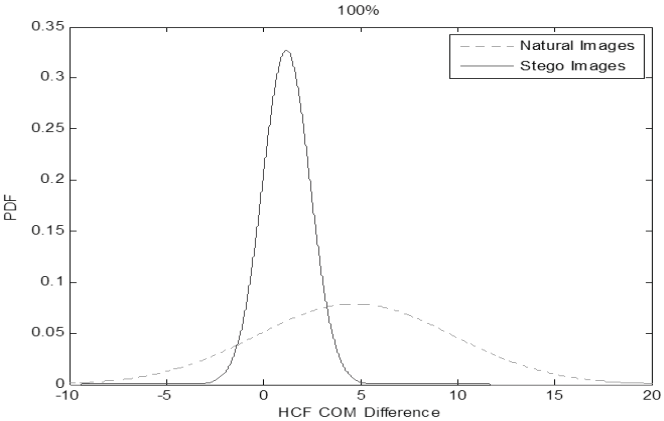


Fig. 3. Distribution of $C(H_{Tim}[k])_{R,G,B}$ and $C(H_{Tim'}[k])_{R,G,B}$

4. To obtain a better detection performance, it is necessary to determinate different threshold values with different percentages of hidden information, which are calculated according with (12) and using the average of 500 natural images and 500 stego-images using the LSB-M steganography method, from 5% to 50 % with increments of 5% of hidden information and from 50% to 100% with increments of 10%; of hidden information obtaining a database of 15000 stego-images, in this sense the new expression of (18) in terms of Th is given by,

$$\begin{cases} abs(\Delta C(H_{s-c}[k])) \le Th \ \ Stegoimage \\ abs(\Delta C(H_{s-c}[k])) > Th \ \ Natural\ Image \end{cases} \quad (19)$$

where

$$\Delta C_{s-c}(H[k]) = C(H_s[k]) - C(H_c[k]) \quad (20)$$

5. To determine if Tim has or not hidden information, it is used the comparison between threshold Th and $\Delta C_{s-c}(H[k])$, which is given by (19).

6 Experimental Analysis and Results

The 500 images from Uncompressed Color Image Data base (UCID) [12] are used to obtain Tim, Tim' and Th ; the Fig.4 shows the comparison of a natural and stego-image with the 100% of hidden information.

a) b)

Fig.4. Comparison of a natural image and stego-image with 100% of hidden information and PSNR 33.472945dB

The Fig.5 shows the HCF-CoM-D of 500 natural images and their stego-images, while in Fig. 6, where the input images correspond to stego-images, which have been obtained using the LSB-M in the 100% of hiding. This Figure has shown a strong relation between the dots and the crosses, according with $\Delta C_{s-c}(H[k])$, which should be near to cero.

Making a detailed analysis of Fig.6, that shows a spread of dots and crosses when both expressions represent the same stego-image, this spread does not exist. To solve this error, the threshold values are obtained according to the analysis in the step 4. The values are shown in Table.1. Fig.7 shows a better effectiveness when the stego-image has the 60%-90% percentages of hidden information.

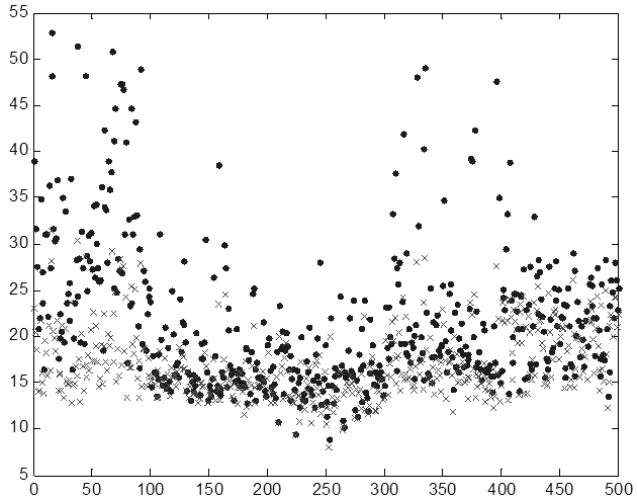


Fig. 5. HCF-CoM-D of 500 input natural image before (dots) and after (crosses) hide information in the 100% by LSB-M.

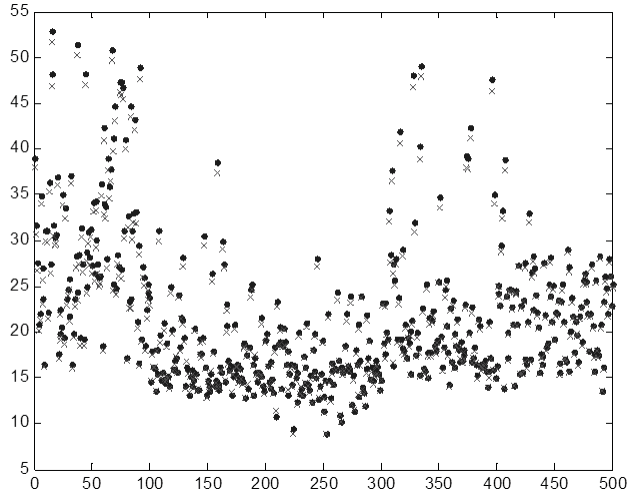


Fig. 6. HCF-CoM-D of 500 input stego-image before (dots) and after (crosses) hide information in the 100% by LSB-M.

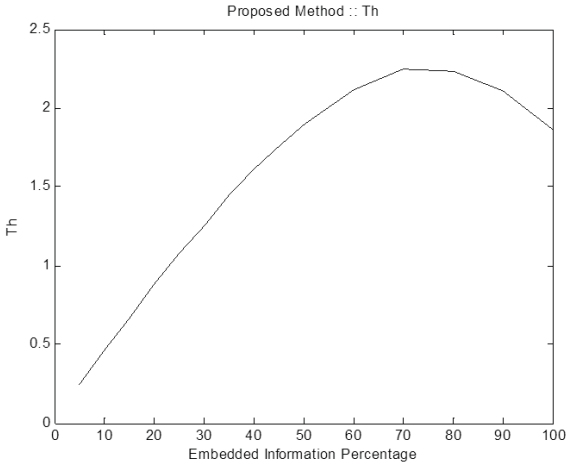


Fig.7. Plot of the Th respect to different percentages of hidden information.

Table.1. *Th* obtained with 1000 images, 500 natural images and 500 stego images for each percentage.

Percentage of hidden Information (%)	*Th*	EDP
5	0.242299	59.6
10	0.46126	64.0
15	0.670235	63.8
20	0.877869	68.4
25	1.075785	68.2
30	1.250817	69.6
35	1.441107	70.8
40	1.611133	72.0
45	1.75855	73.2
50	1.899921	73.2
60	2.114553	75.6
70	2.249342	78.6
80	2.230212	80.8
90	2.105445	83.0
100	1.856936	83.2

The experimental analysis of the 15000 images is given by an Effectiveness Detection Percentage (EDP) contained in the Table 1. Fig. 8 shows the relationship between the percentage of hidden information and the EDP.

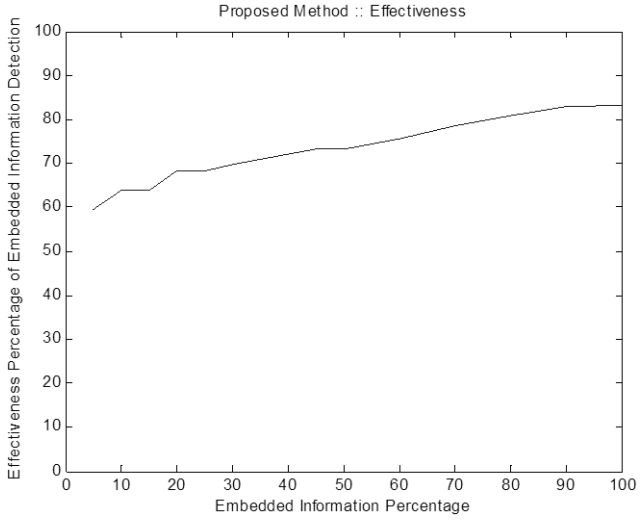


Fig. 8. Relationship between hidden information percentage and the EDP.

Table 2 shows the comparison of the EDP between the proposed algorithm and the Fridrich's method [8] under the same condition.

Table 2: Effectiveness Comparison between proposed method and Fidrich's method [8]

Percentage of embedded information (%)	Proposed algorithm	[8]	Execution Time (sec)	
			Proposed	[8]
5	59.6	5.4	1.01	88.95
10	64	3.2	1.97	89.85
15	63.8	3.2	2.93	90.73
20	68.4	3.6	3.88	91.07
25	68.2	2.4	4.86	91.38
50	73.2	0.0	9.70	94.15
70	78.6	2.2	13.57	95.67
100	83.2	7.2	20.30	97.89

7 Conclusions

In this paper a new method of steganalysis for LSB-M steganography is proposed. The approach, based on the Histogram Characteristic Function-Center of Mass (HCF-CoM) can discriminate successfully stego-images from the natural image.

A selection of an adequate threshold value, depending on the percentage of hidden information, is important issue to improve the performance of the proposed method. The proposed method shows a better performance than the Fridrich's method with much lower computational cost.

The hiding manner by LSB-M used for evaluation is a random hiding, which means randomly selected LSBs are used for LSB-M. Considering that the random hiding is the most common hiding manner and the proposed method provides under random hiding situation a 75% of correct detection in average, we can conclude that the proposed method can be considered as a reliable steganalysis method.

Aknowledgment

The Authors thanks the Instituto Politecnico Nacional and the National Council of Science and Technology of Mexico (CONACyT) for financial support to realize this work.

8 References

[1] S. Katzenbeisser, F.A.P. Petitcolas, "Information Hiding Techniques for Steganography and Digital Watermarking", Comput. Security Ser., Artech House Books, 2000.

[2] O. Juarez-Sandoval, A. Espejel-Trujillo, M. Nakano-Miyatake, H. Perez-Meana, "Robust Steganography Based on QIM Algorithm to Hide Secret Images", International Journal of Computers, Vol.7. N°4, pp145-152, 2013.

[3] K. Shrikants, S. L. Nalbalwar, "Review: Steganography – Bit Plane Complexity Segmentation (BPCS) Technique", International Journal of Engineering Science and Technology, Vol. 2, No. 9, pp. 4860-4868, 2010.

[4] J. Fridrich, "Feature-Based Steganalysis for JPEG Images and Its Implications for Future Design of Steganographic Schemes", Springer-Verlag Berlin Heidelberg, pp. 67-81, 2004.

[5] J. J. Harmsen and W.A.Pearlman, "Steganalysis of additive noise modelable information hiding". MS Thesis, Rensselaer Polytechnic Institute, Troy, New York, 2003.

[6] A. Ker, "Steganalysis of LSB Matching in Grayscale Images". IEEE Signal Processing Letters, vol. 12, No. 6, pp. 441-444, June 2005.

[7] A. Westfeld, A. Pfitzmann, "Attacks on steganographic systems-breaking the steganographic utilities stego. In: Jsteg, Steganos, and S-tools e and some lessons learned", lecture notes in Computer Science. Springer-Verlag; pp. 61-75, 2000.

[8] J. Fridrich, R. Du and M. Long, "Steganalysis of LSB Encoding in Color Images". Multimedia and Expo IEEE International Conference, vol. 3, pp. 1279-1282, Aug. 2000.

[9] T. Sharp "An implementation of key-based digital signal steganography". In: Moskowitz I, editor. Information hiding. of lecture notes in Computer Science. Berlin/ Heidelberg: Springer; vol. 2137 pp. 13-26, Oct. 2001.

[10] J. Mielikainen, "LSB Matching Revisited", IEEE Signal Processing Letters, pp.2-6, October 15 2013

[11] D. Lerch-Hostalot, D. Megías, "LSB Matching steganalysis based on patterns of pixel differences and random embedding", ELSEVIER Computers&Security, Vol.32, pp. 192-206, 2013

[12] Ucid: An uncompressed colour image database [Online]. Available: http://www-users.aston.ac.uk/schaefeg/datasets/UCID/ucid.html

Privacy Risk Metrics for Internal and External Threat Analysis - An Enterprise Perspective

Swaminathan S, Anoop Dobhal, Ankit Kumar, Karthik Sundararaman, Subrahmanya VRK Rao
Global Technology Office, Cognizant Technology Solutions Pvt Ltd, Chennai, Tamil Nadu, India
and Ganesh Srinivasan, Umakantham Rajkumar
Enterprise Risk Security Solutions, Cognizant Technology Solutions Pvt Ltd, Chennai, Tamil Nadu, India

***Abstract* -** Identity and attribute disclosure of enterprise data is hazardous. There are metrics to measure risk involved in a particular data file. However these metrics estimate the maximal risk present in the data file and have rarely focused on multiple sensitive attributes, which is a common feature in enterprise data. In case of an enterprise, insider threat is more hazardous and potentially bigger threat than an external threat. The current paper presents a solution developed by the authors, that could handle big data, multiple sensitive attributes and provide risk profiles for them based on both insider and external threat sources. Hadoop and pig scripts have been used for storing the data and for computing the metrics.

Keywords: Risk Disclosure, k-anonymity, l-diversity, t-closeness, insider threat analysis

1 Introduction

Recent trend of data breach has been a major cause of concern among stake holders. Therefore data security at all levels including computing; storing, transmitting and receiving devices has been gaining prominence. As a preventive technique, enterprises have been enforced to maintain practical/suitable administrative, technical and physical safeguards for protecting data. Such steps should ensure

1. Confidentiality, integrity and availability of all the services involved in data creation, maintenance, transmission and reception
2. Identity and protection of data from reasonably anticipated threat from both internal and external sources
3. Protection against reasonably anticipated, impermissible uses or disclosures; and ensure compliance with the standards such as HIPPA

In simple terms, data at rest and in motion has to be protected either by using IT technologies and/or by following regulations and compliance.

Encryption [1-2], masking [3-4], distortion [5-6] and role based access [7] are among technologies that are being commonly used in enterprises nowadays. While encryption of data is the best practice among all the techniques, its applicability depends on the application, nature of data and costs associated with the practical implementation of the same. Therefore role based data access; masking and distortion techniques do have a role to play in majority of the enterprise data security practices. It must be noted that while role based access, data masking and distortion techniques do provide security, the confidentiality and identity of data could still be breached by re-identification techniques. Background information, access to common reference data from a public data source, homogeneity and skewness present in the data are some of the key components, which might lead to either identity/attribute disclosure or both of them.

2 Related Work and Objective

Metrics to measure privacy content present in masked/distorted data can point to risk involved in data re-identification. Based on the magnitude of such metrics more secure way of handling data could be achieved. Such metrics have been reported frequently [8-14]. Enterprises deal with big data consisting of sensitive attributes, personally identifiable information and quasi identifiers. Quasi-identifier is a set of attributes also present in the publicly available dataset such as age, zip code, gender and ethnicity etc. In majority of its operations enterprises don't differentiate between a sensitive attribute and personally identifiable information, which include names, email, credit card number, social security number etc. As a standard enterprise practice all the data attributes except the quasi identifiers are masked, which is a potential source of risk. This could be leveraged by an internal or external source to breach identity of specific sensitive attributes present in the dataset.

Metrics involving risk disclosure analysis revolve around K-anonymity [8-10], l-diversity [11-13] and t-closeness models [14]. Aforementioned models are based on the analysis of quasi-identifier and sensitive attributes present in the dataset and provide a way for reducing re-identification risk involved in highly confidential data. K-anonymization modeling ensures that the information for each person contained in the release of data cannot be distinguished from at least k-1 individuals whose information also appears in the release [8]. Anonymization/generalization of quasi identifiers such as age, zip code, gender etc. followed by grouping of them into equivalence classes are the two steps involved while finding k. Subsequently, number of tuples/records in each group is computed. Minimal value of these counts is k and re-

identification risk for the dataset would be 1/k. The model works well for most of the datasets; however presence of homogeneity in the sensitive attributes after forming groups/equivalence classes could be a source of potential leak.

To overcome the above limitation of K- Anonymization model, l –Diversity model was designed. This model provides an idea of the distribution of variables of each and every sensitive attribute within the equivalence class and across the whole dataset. Information theory approach such as l-entropy and (c, l) recursive entropy techniques have been used to compute information gain, which could provide an idea on risk involved. However skewness if present in the distribution could lead to higher information gain and therefore a higher risk of attribute disclosure. To overcome the above mentioned problem t-closeness model evolved. Groups/equivalence class is said to have t-closeness if the distance between the distribution of a sensitive attribute in the class and the distribution of the attribute in the whole table is no more than a threshold t. t-Closeness could protect against attribute disclosure but not against identity disclosure and hence k-anonymity with t-closeness should be used to protect against identity disclosure.

Enterprise activities present much more challenging scenarios such as handling of multiple sensitive attributes and insider threat, which have been rarely reported. It must also be noted that providing a risk score for an entire data file as reported in the literature might not be the appropriate method in an enterprise scenario. These scenarios demand profiling of each equivalence class and quasi identifier combination responsible for generation of such a risk score. In other words instead of providing an estimated risk, categorization of risk in each equivalence class present in the data file would be a suitable approach. Hence primary objective of the current study was to develop a tool, which could handle big data and perform a risk disclosure profile for each equivalence class present in the data based on internal and external threat.

3 Proposed Solution instructions

Hadoop set up with 4 nodes (24cores, 64GB ram, 840GB hdd, 64 bit) was used to develop the solution. Pig scripts were used to perform the map reduce programs, which would compute k- anonymity, l- diversity and t-closeness. Architecture of the solution is shown in figure 1. Solution could be accessed through a web interface or through the application running on the machine by auditors/analyst or administrators. Modules that have been shown in figure 1 and its functionalities are explained below.

3.1 Configuration file

Description of quasi identifiers /sensitive attributes and their corresponding data types, file name and location, type of analysis (insider /external threat) to be performed, nature of processing (with/without generalization) are mentioned in this file. This file can only be accessed by either security analyst or the auditor. In certain cases, predefined standards for estimated risk and T closeness value might be provided in the file.

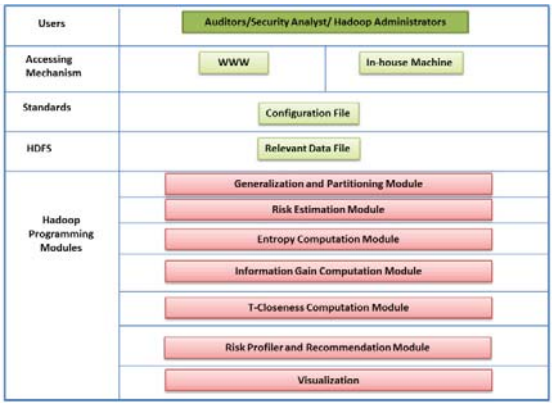


Figure 1 Layered Architecture of the Solution

3.2 Data Access

A pig script was used to read the contents of the configuration file, which contained information of the data source (data files present either in oracle or SQL database). Data extraction was performed using sqoop and relevant data file was stored in hadoop distributed file system (HDFS).

3.3 Generalization and Partition Module

A pig script was used to generalize the quasi identifiers present in the relevant data file. Current study has considered only four quasi identifiers namely age (or any variants of it such as date of birth), zip code, gender and ethnicity. Generalization as performed by the script has been explained by a simple example as illustrated in figure 2.

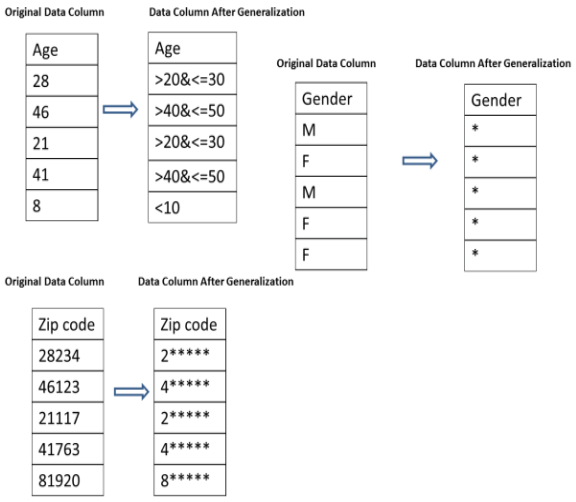


Figure 2 Generalization Examples for Quasi Identifiers

Subsequent to the generalization of quasi identifiers, they would be grouped along identical quasi identifier patterns that exist in the dataset, irrespective of non-identical sensitive variables. Each such group is called an equivalence class and the data file would have many such equivalence classes.

3.4 Risk Estimation Module

This module is a pig script which would compute information loss, estimated risk and also provide quasi identifier combinations that attributed to the same. Number of records in an equivalence class constitutes k and its reciprocal (1/k) is the estimated risk. Information loss is defined as fraction of number of equivalence class with unique records to the number of records present in the data file to the total number of records present in the relevant data file. In some circumstances generalization might not be enforced and in those cases, partitioning is only done based on grouping identical patterns.

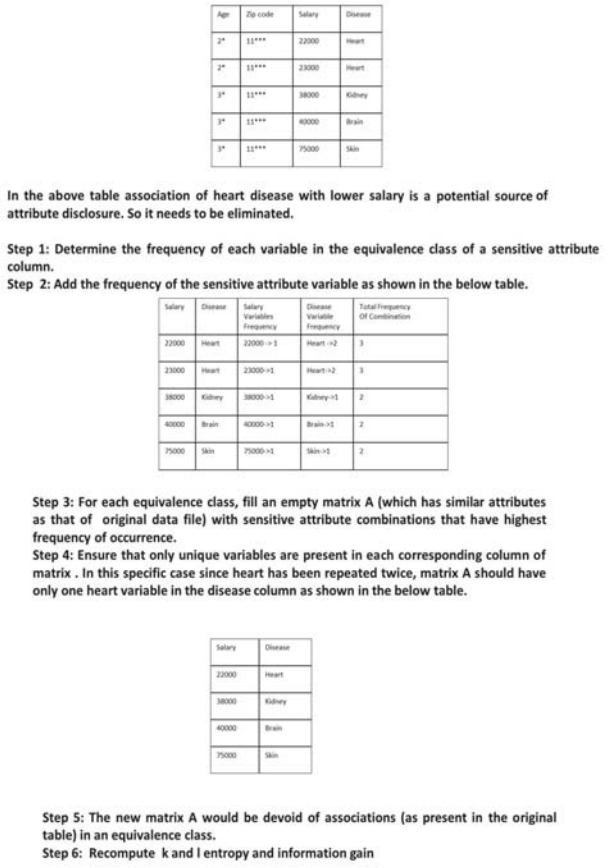
3.5 Entropy Computation and Information Gain Module

This module is a pig script which would determine entropy based on the sensitive attribute of each equivalence class and also that of the entire data table. The procedure is different for computation of l-diversity in case of single sensitive attribute and multiple sensitive attributes. In case of single sensitive attribute l-entropy method as reported in the literature has been adopted in the study [11]. However information gain for each and every equivalence class was computed using the following formula.

Information gain = entropy of the sensitive attribute of the entire column – entropy of the sensitive attributes in equivalence class

3.6 External Threat Analysis and Multiple Sensitive Attributes

The scenario represents data leakage due to an external source, which indicates that the source might not have prior knowledge on sensitive attributes present in the data file. In case of multiple sensitive attributes, association amongst different sensitive attributes (different columns) in a data file would lead to higher information gain and hence such associations must be reduced. Associations were reduced using a pig script, as shown in figure 3. Such a computation would lead to data loss but would ensure minimal risk on attribute disclosure. In some cases when it might not be desirable, records might not be deleted. Such a scenario is usually handled by the auditor/security analyst and according to the context they either keep records intact as-it-is or delete the same. Under circumstances where the data is not deleted in-spite of association being found between sensitive attributes, l-entropy is just simply calculated taking all the sensitive attribute variable present in each row as one string. Subsequently information gain is computed for each equivalence class.



Age	Zip code	Salary	Disease
2*	11***	22000	Heart
2*	11***	23000	Heart
3*	11***	38000	Kidney
3*	11***	40000	Brain
3*	11***	75000	Skin

In the above table association of heart disease with lower salary is a potential source of attribute disclosure. So it needs to be eliminated.

Step 1: Determine the frequency of each variable in the equivalence class of a sensitive attribute column.
Step 2: Add the frequency of the sensitive attribute variable as shown in the below table.

Salary	Disease	Salary Variables Frequency	Disease Variable Frequency	Total Frequency Of Combination
22000	Heart	22000 -> 1	Heart ->2	3
23000	Heart	23000 ->1	Heart->2	3
38000	Kidney	38000->1	Kidney->1	2
40000	Brain	40000->1	Brain->1	2
75000	Skin	75000->1	Skin->1	2

Step 3: For each equivalence class, fill an empty matrix A (which has similar attributes as that of original data file) with sensitive attribute combinations that have highest frequency of occurrence.
Step 4: Ensure that only unique variables are present in each corresponding column of matrix . In this specific case since heart has been repeated twice, matrix A should have only one heart variable in the disease column as shown in the below table.

Salary	Disease
22000	Heart
38000	Kidney
40000	Brain
75000	Skin

Step 5: The new matrix A would be devoid of associations (as present in the original table) in an equivalence class.
Step 6: Recompute k and l entropy and information gain

Figure 3 Algorithm used for computation in case of multiple sensitive attributes

3.7 Internal Threat Analysis and Multiple Sensitive Attributes

Insider threat is the most challenging from an enterprise perspective because of the fact that an insider who is well aware of the data is the source of potential leak. While performing insider threat analysis the study did not expose whole set of sensitive attributes present in the relevant data file. Pig script to handle internal threat was programmed in such a way, that it would only have single sensitive attribute at a time, while all other sensitive attributes would be considered as quasi identifiers. Likewise risk profiles for each sensitive attribute would be computed and based on the measures one would be able to identify the attributes with high risk. Recommendations would be based on auditors security analyst suggestion in this particular case.

3.8 t-closeness

Ontologies were defined for sensitive attributes with string data type present in the data file. Subsequently these were used to develop hierarchies for the same. t-closeness for the same was determined as reported [14]. Similarly for sensitive attributes for numeric data types, earth mover distance measure was used to determine t-closeness [14]. In case of single sensitive attribute, t-closeness was determined as the distance between ordered pairs namely the particular equivalence class and that of entire data file column. Minimal t-closeness value computed was represented as the t-closeness for the data file.

3.9 Risk Profiles, Recommendation and Visualization Module

This module is a pig script which would be able to classify records present in relevant data file under categories such as low, medium, medium to high and high risk as per auditors' suggestion in the configuration file. For example the auditor can set the estimated risk values for risk categories as shown in table 1.

Table 1 : Risk categories and their respective risk values

1/k	Risk Category
<0.1	Low
>=0.1 &<0.3	Medium
>=0.3 &<0.5	Medium to High
>=0.5	High

Depending on the percentage of records falling under each risk category recommendation module might suggest steps to follow to reduce risk. This could be distort data or apply more generalization or vertical/horizontal split of the table. Visualization module would provide a graph indicating the percentage of data records falling under different risk categories.

4 Test Cases

Several data sets were simulated and were tested. Time taken to complete the process along with the respective data file size is shown is shown in table 2.

Table 2 Performance of the hadoop environment for different files

Data File Size (in GB)	Time Taken (in Hours)
0.5	0.30
1.2	0.45
2	0.56
5	0.62
10	0.70
50	1.2

5 Conclusions

Protection of data with multiple sensitive attributes and quasi identifier in an enterprise scenario is a challenging task. The current study has developed solution that could handle big data and perform risk profiling of the relevant data file. The solution handles threat from both internal and external source either by exposing only one sensitive attribute at a time (while keeping the rest as quasi identifiers) or by deleting the associations that exists between multiple sensitive attributes. This way of risk profiling would help the enterprise to focus on portions of the relevant data file; where in the risk of disclosure is high.

6 References

1. Salomon, David. Data privacy and security: encryption and information hiding. Springer Science & Business Media, 2003.
2. Stallings, William, and Lawrie Brown. "Computer Security." Principles and Practice (2008).
3. Wang, Cong, et al. "Privacy-preserving public auditing for data storage security in cloud computing." INFOCOM, 2010 Proceedings IEEE. IEEE, 2010.
4. Wang, Cong, et al. "Toward publicly auditable secure cloud data storage services." Network, IEEE 24.4 (2010): 19-24.
5. Xu, Shuting, et al. "Singular value decomposition based data distortion strategy for privacy protection." Knowledge and Information Systems 10.3 (2006): 383-397.
6. Oliveira, Stanley RM, and Osmar R. Zaiane. "Privacy preserving clustering by data transformation." Journal of Information and Data Management 1.1 (2010): 37.
7. Joshi, James BD, et al. "A generalized temporal role-based access control model." Knowledge and Data Engineering, IEEE Transactions on 17.1 (2005): 4-23.
8. El Emam, Khaled, and Fida Kamal Dankar. "Protecting Privacy Using K-Anonymity." Journal of the American Medical Informatics Association□: JAMIA 15.5 (2008): 627–637. PMC. Web. 27 Feb. 2015.
9. L. Sweeney. k-anonymity: a model for protecting privacy. International Journal on Uncertainty, Fuzziness and Knowledge-based Systems, 10 (5), 2002; 557-570.
10. Shokri, Reza, et al. "Unraveling an old cloak: k-anonymity for location privacy." Proceedings of the 9th annual ACM workshop on Privacy in the electronic society. ACM, 2010.
11. Machanavajjhala, Ashwin, et al. "l-diversity: Privacy beyond k-anonymity." ACM Transactions on Knowledge Discovery from Data (TKDD) 1.1 (2007): 3.

12. Zhou, Bin, and Jian Pei. "The k-anonymity and l-diversity approaches for privacy preservation in social networks against neighborhood attacks." Knowledge and Information Systems 28.1 (2011): 47-77.
13. Xu, Yang, et al. "A survey of privacy preserving data publishing using generalization and suppression." Appl. Math 8.3 (2014): 1103-1116.
14. Li, Ninghui, Tiancheng Li, and Suresh Venkatasubramanian. "t-closeness: Privacy beyond k-anonymity and l-diversity." Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on. IEEE, 2007.

Cloud-based Identity and Access Control for Diagnostic Imaging Systems

Weina Ma and Kamran Sartipi
Department of Electrical, Computer and Software Engineering
University of Ontario Institute of Technology
Oshawa, ON, L1H 7K4, Canada

Abstract - *The evolution of cloud computing is driving the next generation of diagnostic imaging (DI) systems. Migrating DI systems to cloud platform is cost-effective and improves the quality of DI services. However, a major challenge is managing the identity of various participants (users, devices, applications) and ensuring that all service providers offer equivalent access control in cloud ecosystem. In this paper, we propose an access control infrastructure for secure diagnostic image sharing among Diagnostic Imaging Repositories and heterogeneous PACS (Picture Archiving and Communication Systems) in cloud. We utilize an open standard "OpenID Connect" to provide user-centric Single Sign-On solution, and present the extensions for integrating with patient consent directives and system access control policies. Through combining with the dominant access control model XACML in existing DI systems, the extended OpenID Connect authorization server can provide fine-grained access control.*

Keywords*:* diagnostic imaging; cloud; federated identity; access control; OpenID Connect; XACML.

1 Introduction

With the exploding size of electronic medical records and the fast growth of the diagnostic imaging (DI) market, cloud computing is becoming a preferred solution for image sharing over the Internet using external infrastructure, which allows for accessing to applications and data on demand, any time and from anywhere. Medical data contains sensitive information that may affect the lives of people, so security and patient privacy aspects must be primarily issued. In particular, federated identity management and consistent access control are imperative for cross-domain information sharing and is becoming crucially important with the online healthcare services deployed in cloud environment.

OpenID Connect [1] is an open and decentralized authentication standard that provides a way to verify a user for co-operating sites (known as service providers) without sharing user credential or other sensitive information to service providers. OpenID Connect has broad support from major cloud service providers, enterprise companies, and social networking companies (e.g., Google, Yahoo, Microsoft, and Facebook). According to the OpenID Foundation, the department of health and human services of US Government has joined the OpenID Foundation to create a profile of OpenID Connect and associated projects [1].

OpenID Connect authorization server makes an access control decision based on the resource owner's consent in an interactive way. It is suitable for presenting patient consent directives and their impact on access control. However, as an authorization solution for web services, OpenID Connect does not define a method of enforcing fine-grained system access control polices. In contrast, XACML (eXtensible Access Control Markup Language) [2] is the de-facto attribute based access control standard in DI systems, which provides an extreme fine-grained policy language and processing model. Both system access control polices and patient consent directives enforcement are indispensable parts of the access control model in DI systems. Therefore, combining OpenID Connect authorization flow and XACML model would be a valuable attempt to close the gap.

For the ease of applying a consolidated authentication mechanism, we propose delegating the universal identity management including advanced authentication technology (e.g., biometrics and hardware authentication devices) to the OpenID Connect identity provider. The proposed approach enables users to manage their identities, which minimizes the information disclosure to the service providers. We propose XACML policy based extension of OpenID Connect authorization server to enforce patient consent directives and fine-grained access control rules.

The main contributions of this paper can be summarized as follows: i) designing a user-centric decentralized identity management and authentication service for cloud-based DI systems; ii) proposing fine-grained access control model by combining OpenID Connect authorization server with XACML polices; and iii) enforcing patient consent directives in access control flow.

The remaining of this paper is organized as followings. Related work is discussed in Section 2, and the relevant background technologies are presented in Section 3. In Section 4 our proposed OpenID Connect based federated identity management and access control infrastructure is explained. Section 5 is allocated to a case study, and finally conclusion is presented in Section 6.

2 Related work

Identity federation management enables the users in one domain to securely and seamlessly access data in another domain. Maintaining the user identity repository in each individual domain can lead to information inconsistency and synchronization problems. Meanwhile, the industry trend towards cloud computing and Software as a Service (SaaS) are major drives to shift the federated identity solutions from enterprise-centric to user-centric, and from close-world communication to open standard, where account information is persisted and managed by the third party services. The users are authenticated by co-operating sites (e.g., PACS and DI-r services) using these external services. Relying on external identity services allows users manage their own identity and privacy, and offers the healthcare service providers easier and faster access to the advanced identity management and authentication technology with lower investment.

Due to paradigm shift in federated identity solutions towards user-centric authentication some resent researches focus on providing common authentication mechanism and authorization delegation solutions.

Khan et al. [4] introduced a flexible decentralized authentication service "OpenID-authentication-as-a-service" in the open source cloud OpenStack. Ma and Sartipi [5, 6] introduced an agent-based infrastructure for secure medical image sharing between legacy PACS systems which authenticates users against OpenID protocol. Utilizing OpenID Connect as an identity management and authentication service is not our main target. To provide both fine-grained access control and to support patient consent directives, we need to extend OpenID Connect authorization flows. Ardagna et al. [7] presented extensions to the access control standard XACML and SAML (Security Assertion Markup Language) to enable privacy-preserving and credential-based access control.

OpenID Connect increases the security of integrated systems by putting responsibilities for user authentication to the most expert third party service providers. The organizations that contribute to OpenID Connect are leaders in the developing of advanced authentication technologies such as bi-factor and multi-factor authentication. In addition, the integrated systems still have options to manage their own user information and relationship but outsource the expensive, high-risk tasks of identity verification to external professional service providers. Kakizaki and Tsuji [8] proposed a decentralized user attribute information management method using OpenID Connect for identity verification. By assigning a uniform resource identifier (URI) for all attributes, OpenID Connect identity provider only persists the user's unique ID and related attribute URIs. This feature caters to the healthcare organizations that have concerns about exposing some sensitive patient information to an external identity provider.

OASIS cloud authorization technical committee (CloudAuthZ TC) [9] aims at generating profiles for cloud authorization through making the best of existing, well-designed standards (e.g., XACML, OAuth). The principle idea of CloudAuthZ TC is to reduce the load of authorization engine. Client application obtains a contextual entitlement from authorization engine at the first time of sending access request. After that client application is capable of making decision according to this contextual entitlement without calling authorization engine again, which obviously eases the authorization engine. H. Lockhart [10] explores the possibility of expressing the scope of an OAuth access token by using XACML polices to offer self-contained token, which can be interpreted by the resource server without consulting the authorization server. However, both of these approaches are in their initial stages without practical and successfully applied domains and case studies.

3 Background

In this Section, we introduce the key technologies that constitute the proposed cloud-based identity and access control mechanism for diagnostic imaging.

3.1 Diagnostic Imaging Systems

In medical imaging, PACS (Picture Archiving and Communication System) is a complex integrated system equipped with the necessary hardware and software: digital image acquisition devices namely modalities (e.g., CT scanner, MRI system); digital image storage and archive where the acquired images are stored; and workstations where radiologists view the images [11]. With the increasing demand for collaborative work and sharing of medical information, PACS systems in different hospitals or image centers are interconnected across a distributed environment. Diagnostic imaging repository (DI-r) provides a solution for sharing (publishing, discovery, retrieving and reliably storing) of DI documents across affiliated healthcare organizations. According to the status of DI-r projects across Canada [12], provincial DI-r's have been developed to deliver fast and easy access to diagnostic images to all authorized healthcare providers.

3.2 OpenID Connect

OAuth [3] is an open standard for authorization. It defines specific authorization flows for conveying authorization decisions across network for web applications, desktop applications and mobile applications. OAuth is fundamental to securing service APIs in a simplified way, including: delegated access, avoiding password sharing between users and third parties, and revocation of access. OAuth [3] provides client application an access token for granting access to the protected resource on behalf of the resource owner without sharing credentials such as a password.

OpenID Connect [1] provides an identity n,layer on top of the OAuth protocol. OpenID Connect is a token-based

authentication standard that allows applications to verify the identity of the end-user based on the authentication performed by an authorization server, as well as to obtain basic profile information about the end-user. OpenID Connect provides an identity provider discovery protocol, which dynamically discovers the corresponding identity provider once a user unique ID is given. Apart from identity verification, OpenID Connect allows service providers to use more extensible features such as encryption of identity data, dynamic discovery of identity provider, session management, and to obtain user attributes after authentication. Moreover, OpenID Connect protocol is extended to integrate OAuth authorization process. So OpenID Connect can be used for both authentication and authorization.

3.3 XACML

XACML (eXtensible Access Control Markup Language) [13] defines an access control policy language in XML and a processing model to evaluate access requests based on the defined polices. XACML is an implementation of the attribute based access control, where attributes related to users, resources and environment are inputs into the decision engine. According to the access request and input attributes, the decision engine finds applicable defined policies and makes access decision.

4 Proposed approach

Integrating the Healthcare Enterprise (IHE) has developed a collection of profiles for guiding enterprises in using established standards for an existing IT infrastructure to accomplish interoperability. IHE suggests a trust model where each local diagnostic imaging system is responsible for ensuring that personal health information is adequately protected. A key challenge with this trust model is the lack of federated capabilities:

- User authentication is local to each system that imposes a significant administrative burden to ensure that persons are uniformly identified in each system.
- Access Control rules are local to each system, which means consistency of access rules across all systems has to be managed manually.
- Patient consent directives and their impact on access control are not communicated to each local system electronically and automatically.

After moving to cloud computing, the identity and access control policy synchronization can be extremely complicated with plenty of participants (service providers and service consumers). So deploying a common infrastructure for user identity management, universal access control policy and patient consent directives management is highly needed. The intent is to integrate DI systems with this common infrastructure so that: i) legacy system users can be authenticated against the common infrastructure; and ii) access to patient imaging records can be controlled based on patient's consent directives and system access control polices defined in the common infrastructure.

4.1 Architecture overview

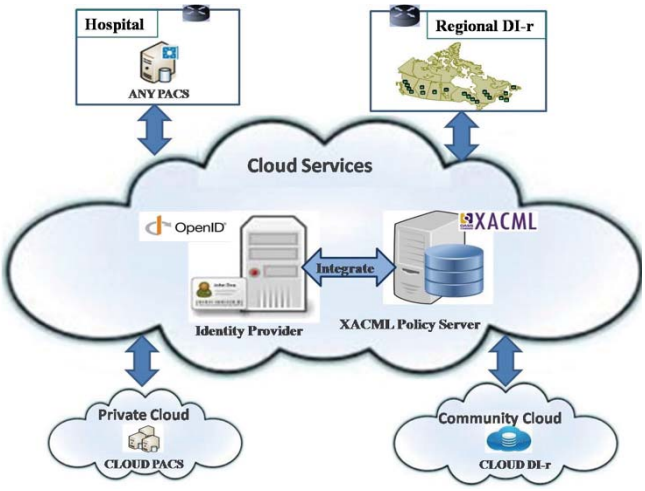


Figure 1. Common identity management and access control method deployed as cloud service.

We propose the design of integrating OpenID Connect identity provider with XACML policy server as cloud service (Figure 1) that: i) provides Single Sign-On user experience to both traditional desktop and mobile users; ii) implements a common service for DI systems to do authentication and authorization; iii) relieves the integrated system administrators from administrative management burden on identity, access control policy, and patient consent directives; and iv) applies ease of utilizing consolidated authentication mechanism to integrated systems, including advanced authentication technology (e.g., biometrics and hardware authentication devices).

Canada Health Infoway stated that the healthcare services deployed in private or community cloud, rather than public cloud, can provide equivalent security level to traditional computing models [14]. So deploying PACS systems and DI-r's to private cloud or community cloud is preferred cloud-based DI solution. We introduce OpenID Connect for creating an identity management and authentication ecosystem for cloud-based DI systems. XACML policy server provides centralized access control policy and patient consent directives management. The existing IT infrastructure in legacy domains is operating based on different technologies, procedures and models. It is not necessary to employ exactly the same access control mechanism across these domains, but it is reasonable that they will agree at the policy level. The integration of OpenID identity provider and XACML policy server will be discussed in next subsection.

4.2 Authentication and authorization flow

OpenID Connect authentication and authorization flow defines six roles as follows: i) "*End User*" is human

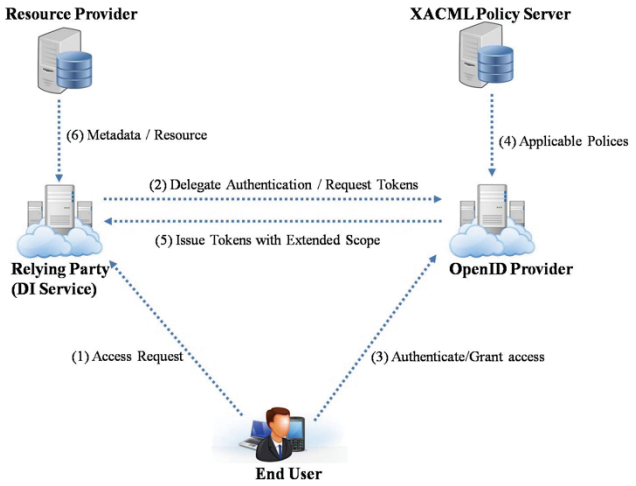


Figure 2. Extended OpenID Connect authentication and authorization flow using XACML Policy Server

participant who wants to access the service (e.g., physician requests to access images from PACS system or DI-r services); ii) "*Resource Owner*" is capable of granting access to a protected resource (e.g., patient may grant a physician or a healthcare organization to access his/her images); iii) "*Relying Party*" is an application (e.g., PACS system and DI-r services) requiring authentication and access grant from OpenID Provider; iv) "*Resource Provider*" manages resources and their metadata; v) "*OpenID Provider*" is an authorization server that is responsible for issuing tokens to Relying Party after successfully authenticating the end user and obtaining granting from Resource Owner; iv) "*XACML Policy Server*" manages access authorization policies and patient consent directives policies.

Suppose that both Resource Owner and End User have already registered to OpenID Provider. In the case that End User and Resource Owner are not the same person, and the Resource Owner is not online to grant or deny the access request, the patient consent directives must be defined and integrated with the OpenID Connect authorization flow.

After successfully authenticated and granted by Resource Owner, Relying Party receives an ID token (which asserts the user identity in a signed and verifiable way), and an access token (credentials used to access protected resource) from OpenID Provider. The access token is typically limited by its "scope" which is issued with access token together. The value of the scope is expressed as a list of space delimited strings, and these strings are defined by the OpenID Provider [1]. For example, a scope of an access control looks like "*profile email address phone*". It means this access token is limited to access Resource Owner's profile, email, address, and phone number. The simple expression of scope is not adequate to describe complicated system access control policies both in syntax and in semantics. So we propose to integrate XACML Policy Server with OpenID Provider, and to represent the scope in the format of XACML policy language. Then the Relying Party may evaluate the scope of access token based on existing XACML infrastructure to make access decisions.

Integration of the OpenID Provider with XACML Policy Server, and the extended authentication and authorization flow is shown in Figure 2.

- OpenID Connect works with any standard Internet browser without any client-software requirement so that the end users, physicians and patients, can set up their devices and applications independently to access DI services from anywhere. Assume that End User has already owned an account with any OpenID Provider. Then he/she initiates an access request and provides his/her OpenID identifier to the diagnostic imaging service (Relying Party). An OpenID identifier for a specific user may look like "myname@example.com" or "http://example.com/myname".
- Relying Party can dynamically discover the location of corresponding OpenID Provider according to the URL "http://example.com". Then Relying Party delegates the OpenID Provider to authenticate End User, and asks for tokens if the access is granted.
- OpenID Provider redirects the End User's browser to a login page to perform authentication. Any authentication method can be used (e.g., password, credentials, information card, and biometrics). After authentication, OpenID Provider checks if End User is also Resource Owner. If they are the same person, OpenID Provider redirects the End User's browser to a granting page and lists the information that will be exposed to Relying Party. The End User can decide to grant or deny this access request. If they are not the same person, OpenID Provider will evaluate this access request against the defined patient (Resource Owner) consent directives.
- XACML Policy Server provides a list of access control polices to OpenID Provider that are applicable to this access request. Such polices can be represented as the scope of access token. As the Relying Party cannot recognize the user related attributes (e.g., user name, role, organization) without consulting OpenID Provider, OpenID Provider should remove or substitute the user related attributes with constant values. Suppose an episode of access control policy is "non-primary physician is allowed to read patient's images from 9:00am to 5:00pm"; if the access session is initialized by a non-primary physician of patient Tom, then the scope of access token is represented as "allowed to read Tom's image from 9:00am to 5:00pm". The method of eliminating variant user attributes from scope reduces the interactions between Relying Party and OpenID Provider.
- The scope of access token specifies what access privileges can be granted to the access token holder. Since the scope does not constitute variant user related attributes, the local decision engine deployed at Relying Party can make access decision by parsing the scope without consulting OpenID Provider.
- Besides of the scope of access token, resource metadata are required to make access control decision. If the access request is granted, Resource Provider returns the demanded resource.

5 Case study

In this section, we describe an end-to-end case study which examines our proposed federated identity and access control architecture. A user account is predefined in OpenID Provider including the following personal information: OpenID identifier "weina@example.com", username "weina" and password, and user attributes such as role (Physician) and organization (Hospital-A). Two access control policies are defined in XACML Policy Server: i) Tom authorizes Hospital-A to access his medical images from Jan 01, 2015 to Dec 31, 2015 (a patient consent directives policy); and ii) physician is allowed to view patient's medical image (a role based access control policy).

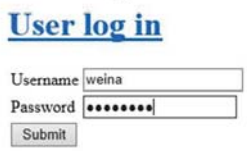
End User named Weina wants to search and view patient Tom's medical images that are created on January 2015. She enters a RESTful request as "GET http://localhost:8080/myregistry/patient/'Tom'/ image/search? 'creationTime'>date'2015-01-01 00:00:00' &'creationTime' < date '2015-02-01 00:00:00'".

The DI-r service receives the access request and delegate OpenID Provider to authenticate the End User. Figure 3-(a) shows redirected page asking for user to enter OpenID identifier. An email address is entered which includes the unique account name "weina" and OpenID Provider host "example.com". Relying party (DI-r service) is able to find the location of OpenID Provider using "example.com". OpenID Provider redirects End User to user login page and needs user input username and password as shown in Figure 3-(b). As End User is not Resource Owner, OpenID Provider queries XACML Policy Server for applicable polices to this access request. Two polices defined above are selected and returned to OpenID Provider. OpenID Provider evaluates the applicable polices according to user related attributes. The End User is a physician and working at Hospital-A, so she is allowed to access patient's medical image from Jan 01, 2015 to Dec 31, 2015. OpenID Provider converts the applicable polices to scope, which constraints the privilege of the issued access token. Figure 3-(c) shows the issued access token, and its scope expressed in JSON (JavaScript Object Notation) rather than a string list. Relying party makes access decision based on the resource and environment related attributes against the scope of access token. As the current date is Feb 2015 and the queried image belongs to patient Tom, the access request is granted. Finally the image is retrieved and displayed in browser as shown in Figure 3-(d).

6 Conclusion

This paper contributes to the domain of diagnostic imaging in cloud computing by providing a solution for federated identity management and access control. We proposed an infrastructure that replaces the existing trust model, which relieves the legacy systems from administrative burdens for identity management and access control policy synchronization. We introduced OpenID Connect as a cloud service to provide user-centric Single Sign-On solution. It allows the user to use one OpenID identifier to sign in to multiple healthcare services, without exposing password or some sensitive information to all these services. OpenID Connect is open to use any modern authentication technology such as smart card and biometrics, which offers the healthcare service providers easier and faster access to the advanced identity management with lower investment. Universal access control policies and patient consent directives are defined in an XACML policy server that is integrated with OpenID Provider.

(a)



(b)

access_token="MXDhGGpKqxZusTu1+Sp9QbSRWaDG/L0IaSWReecSeZHRExbm/+E3nTwfJuYbuYwq6oAEfItW/cAAtgntbKAGn/oH6AQcWU3aIVyrq+GiUSg="
scope=
{
"access": "allow",
"operation": "view",
"resource": "image",
"owner": "Tom",
"time":
{
"from": "Jan 01, 2015",
"to": "Dec 31, 2015"
}
}

(c)

(d)

Figure 3. (a) End User is asked to input an OpenID identifier; (b) User login page for authentication; (c) OpenID Provider issued acces token with extended scope; (d) The pateint's image is displayed in browser.

The applicable access control policies are embedded into the scope that constrains the privilege of the issued access token. The scope can be evaluated by existing XACML decision engine in diagnostic imaging systems without introducing new IT infrastructure change. This research attempts to provide a design for common identity and access control services in cloud-based DI ecosystem and

the implemented prototype proves the feasibility of the design.

7 References

[1] OpenID Connect websit, http://openid.net/

[2] eXtensible Access Control Markup Language (XACML) Version 3.0 (2013), http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html

[3] OAuth Community websit, http://oauth.net/

[4] R. H. Khan, J. Ylitalo, and A. S. Ahmed, "OpenID authentication as a service in OpenStack," In Information Assurance and Security (IAS), 7th International Conference on. IEEE, 2011, pp. 372-377.

[5] W. Ma, and K. Sartipi, "An Agent-Based Infrastructure for Secure Medical Imaging System Integration," Computer-Based Medical Systems (CBMS), IEEE 27th International Symposium on. IEEE, 2014, pp. 72-77

[6] K. Sartipi, K. Kuriakose, and W. Ma, "An Infrastructure for Secure Sharing of Medical Images between PACS and EHR Systems," International Conference on Computer Science and Software Engineering (CASCON), 2013, pp. 245-259

[7] C. A. Ardagna, et al. "Enabling privacy-preserving credential-based access control with XACML and SAML." Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on. IEEE, 2010, pp. 1090-1095.

[8] Y. Kakizaki, and H. Tsuji, "A Decentralized Attribute Management Method and its Implementation," Journal of Information Processing and Management 3.1. 2012, pp. 61-69

[9] OASIS Cloud Authorization (CloudAuthZ) Technical Committee, https://www.oasis-open.org/committees/cloudauthz/charter.php

[10] Using XACML Policies as OAuth Scope (2013), https://www.oasis-open.org/

[11] B. F. Branstetter, "Practical imaging informatics: foundations and applications for PACS professionals", Springer, 2009, pp. 33-47.

[12] A. Gauvin, "Status of Diagnostic Imaging Repository (DI-r) projects across Canada", 2010, http://www.camrt.ca/

[13] eXtensible Access Control Markup Language (XACML) Version 3.0, http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html, 2013

[14] Canada Health Infoway, "Cloud Computing in Health White Paper", 2012, https://www.infoway-inforoute.ca/

A New Edit Distance for Fuzzy Hashing Applications

V. Gayoso Martínez[1], F. Hernández Álvarez[1], L. Hernández Encinas[1], and C. Sánchez Ávila[2]
[1]Information Processing and Cryptography (TIC), Institute of Physical and Information Technologies (ITEFI)
Spanish National Research Council (CSIC), Madrid, Spain
[2]Telecommunication Engineering School (ETSIT), Polytechnic University of Madrid (UPM), Madrid, Spain

Abstract— *Similarity preserving hashing applications, also known as fuzzy hashing functions, help to analyse the content of digital devices by performing a resemblance comparison between different files. In practice, the similarity matching procedure is a two-step process, where first a signature associated to the files under comparison is generated, and then a comparison of the signatures themselves is performed.*

Even though `ssdeep` *is the best-known application in this field, the edit distance algorithm that* `ssdeep` *uses for performing the signature comparison is not well-suited for certain scenarios. In this contribution we present a new edit distance algorithm that better reflects the similarity of two strings, and that can be used by fuzzy hashing applications in order to improve their results.*

Keywords: Edit distance, fuzzy hashing, similarity preserving hashing

1. Introduction

Similarity Preserving Hashing (SPH) functions, also known as fuzzy hashing algorithms, try to detect the resemblance between two files [1]. There are basically four types of SPH functions [2]: Block-Based Hashing (BBH) functions, Context-Triggered Piecewise Hashing (CTPH) functions, Statistically-Improbable Features (SIF) functions, and Block-Based Rebuilding (BBR) functions. In any fuzzy hashing application, files are processed and, as a result of the analysis performed, a code linked to the content of the file is generated, so files can be later compared based on their codes. In this context, the file's code is indistinctly referred to as its digest, hash or signature.

In CTPH functions, the length and content of the signature is determined by the existence of certain special points, called trigger points or distinguished points, within the data object. A point is considered to be a trigger point if it matches a certain property, defined in a way so that the number of expected trigger points falls within a previously specified range. Once a number of trigger points large enough is detected, CTPH applications generate the signature associated to the file by processing the data portions located between consecutive trigger points.

Since its first release, `ssdeep` [3] has been one of the best known fuzzy hashing applications. When comparing files, `ssdeep` generates a matching score after analysing the similarity of the signatures. In order to do that, `ssdeep` implements an edit distance algorithm based on the Damerau-Levenshtein distance between two strings [4], [5]. That edit distance function compares the two strings and counts the minimum number of operations needed to transform one into the other, where the allowed operations are insertions, deletions, and substitutions of a single character, and transpositions of two adjacent characters [6], [7].

Even though the success of `ssdeep` is quite remarkable, its edit distance implementation has important limitations that prevent `ssdeep` from generating a score that reflects the percentage of the bigger file that is also present in the smaller file, which is the definition of similarity better adapted for some real-world scenarios. With the goal to improve the quality of fuzzy hashing applications, in this contribution we present a new edit distance algorithm that can be used as a replacement of `ssdeep`'s edit distance or in new implementations.

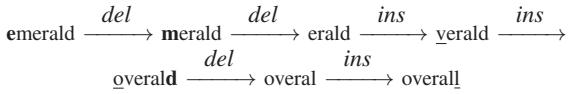
The rest of this paper is organized as follows: Section 2 reviews `ssdeep`'s edit distance. In Section 3, we provide a complete description of our proposed algorithm. Section 4 includes a comparison of both algorithms when working with some special signatures. Finally, Section 5 summarizes our conclusions about this topic.

2. Edit distance in `ssdeep`

In 2006, Jesse Kornblum released `ssdeep` [8], one of the first programs for computing context triggered piecewise hashes, and that soon became very popular. Since that initial release, new versions and updates have not ceased to appear, and the project is still active (at the time of preparing this contribution, the latest version is 2.12, which was released in October 2014 [3]). The core of `ssdeep` is derived from `rsync` [9] and `spamsum` [10], both of them tools developed by Andrew Trigdell.

As mentioned in the previous section, the similarity measurement that `ssdeep` uses is an edit distance algorithm based on the Damerau-Levenshtein distance [4], [5], [6], [7]. In the original Damerau-Levenshtein algorithm, all the operation costs are initially 1, though the substitutions and transpositions decrease their weight to 0 when certain conditions are met [11]. In comparison, `ssdeep` defines the weight of insertions and deletions as 1, the weight of substitutions as 3, and the weight of transpositions as 5.

As an example, using `ssdeep`'s algorithm the distance between the strings "emerald" and "overall" is 6, as it can be checked with the following steps and the computations of Table 1.



		e	m	e	r	a	l	d
	0	1	2	3	4	5	6	7
o	1	2	3	4	5	6	7	8
v	2	3	4	5	6	7	8	9
e	3	2	3	4	5	6	7	8
r	4	3	4	5	4	5	6	7
a	5	4	5	6	5	4	5	6
l	6	5	6	7	6	5	4	5
l	7	6	7	8	7	6	5	**6**

Table 1: `ssdeep` edit distance example.

A consequence of assigning the weights 3 and 5 to the substitution and transposition operations is that, in practice, the edit distance computed by `ssdeep` only takes into consideration insertions and deletions. In this way, a substitution has a cost of 2 (a deletion plus an insertion) instead of 3, and a transposition has also a weight of 2 (again an insertion and a deletion) instead of 5.

One of the limitations derived from this design is that, given a string, a rotated version of the initial string is credited with many insertion and deletion operations, when in its nature it is basically the same string (i.e. the content is the same, although the order of the substrings is different). Consider for example the strings "1234abcd" and "abcd1234".

As the signature comparison algorithm implemented by `ssdeep` is not available in a descriptive way, Algorithm 1 shows our interpretation (made upon inspection of the source code of `ssdeep` [3]) of that functionality, where `A` and `B` are one-dimensional arrays containing, respectively, the m characters of `string1` and the n characters of `string2`, and where `D` is a $(m+1) \times (n+1)$ matrix used in the computations with all its positions initially set to 0. During the set-up phase, the first row (respectively, the first column) of `D` is initialized with the number corresponding to the column (respectively, the row) of the position being processed. The rest of the positions are processed based on the content of the nearby elements and the characters being compared. Once the comparison procedure is finished, the algorithm generates a similarity score in the range 0-100.

The meaning of the functions included in the algorithm is the following:

- `length(string)`: calculates the number of characters of the string.
- `longestCommonSuString(string1,string2)`: provides the longest common substring of two strings.
- `min(param1,param2,param3)`: identifies the minimum value given by the numbers or expressions passed to the function as parameters.
- `floor(value)`: returns the bigger integer whose value is equal to or lower than `value`.

3. Our proposed edit distance

Our edit distance algorithm compares two signature strings, `string1` and `string2`, and produces a similarity score in the range 0-100. Algorithm 2 describes all the steps that must be performed in order to evaluate the similarity of the strings `string1` and `string2`, where the first step consists in identifying as `string1` the shortest string and as `string2` the longest string, swapping the strings if necessary. During the procedure, the algorithm manipulates modified versions of the input strings, using their longest common substring for deciding which modification to perform next and increasing a counter with the differences found so far. The procedure is repeated until there are no more common substrings for the modified versions of the input elements. In the final step, the algorithm compares the resulting strings character by character in order to add to the counter the number of difference elements found for the same positions. It is important to point out that, unlike `ssdeep`, our algorithm does not impose a minimum length for the longest common substring, which allows to compare a wider range of strings.

The meaning of the functions included in Algorithm 2 and not presented in the previous section is the following:

- `longestCommonSuStringNoHyphen(string1, string2)`: returns the longest common substring which does not contain the hyphen (-) character.
- `hyphenString(size)`: creates a new string of length `size` containing only the hyphen character.
- `indexOf(string,substring)`: returns the position where the first character of `substring` is located inside `string`.
- `replace(string,index,size,substring)`: replaces in the element `string` the existing substring of `size` characters starting at `index` with the characters of `substring`.
- `abs(number)`: provides the absolute value of the input number.
- `charAt(string,index)`: returns the character located at position `index` in the element `string`.

In order to illustrate the comparison process performed by Algorithm 2, Table 2 provides an example using two ad-hoc strings, denoted as `string1` and `string2`. In the first row of the table, we have included the two initial strings (renamed as `string1temp` and `string2temp`), the template for the modified version of `string2` (called `string2mod`), and the score, which initially equals 0. Starting with the step 1, the element `substring` identifies the longest common substring of `string1temp` and `string2temp`, which are then updated to show the removal of that substring. Then, we have inserted the common

Algorithm 1 `ssdeep` edit distance algorithm.

if (`length(longestCommonSuString(string1,string2))` < 7) **then**
  return 0
end if
$\lambda_{del} \leftarrow 1$
$\lambda_{ins} \leftarrow 1$
$\lambda_{sub} \leftarrow 3$
`i` $\leftarrow 0$
for all $i \leq m$ **do**
  `D[i,0]` $\leftarrow$ `i`
end for
`j` $\leftarrow 0$
for all $j \leq n$ **do**
  `D[0,j]` $\leftarrow$ `j`
end for
`i` $\leftarrow 1$
`j` $\leftarrow 1$
for all $i \leq m$ **do**
  for all $j \leq n$ **do**
    if (`A[i] = B[j]`) **then**
      $\lambda_{sub} \leftarrow 0$
    else
      $\lambda_{sub} \leftarrow 3$
    end if
    `D[i, j]` = `min(D[i-1,j]` + λ_{ins}, `D[i,j-1]` + λ_{del}, `D[i-1,j-1]` + λ_{sub})
  end for
end for
`score` $\leftarrow$ `D[m,n]`
`score` $\leftarrow$ `floor`$\left(\frac{\texttt{score} \cdot 100}{\texttt{length(string1)} + \texttt{length(string2)}}\right)$
if (`score` > 100) **then**
  `score` $\leftarrow 0$
else
  `score` $\leftarrow 100$ - `score`
end if
return `score`

longest substring obtained in that step into `string2mod`, so the position of that substring in `string2mod` is the same that it occupies in `string1`.

As described in Algorithm 2, we only increase the score if the difference between the initial and final positions of the substring in `string2mod` is greater than the length difference of `string1` and `string2`. With this rule we avoid to penalize the change of positions derived from the different length of the strings under comparison (e.g. this difference could have been produced by the insertion of some characters at the beginning of the string, which would displace the rest of the characters that compose the original string a given number of positions).

The score is increased in one unit if the longest common substring has more than one character, which means that common substrings of different sizes would receive the same penalty (i.e., a penalty of 1.0, but only if they are separated a number of positions bigger than the difference of the string lengths). In this sense, what we penalize is the movement of the string, not its size.

Besides, when the longest common substring has exactly one character, the quantity to be added to the score is 0.5. The reason for doing this is not to penalize in excess the displacement of a unique character. If we do not impose this rule, the displacement of a single character would receive a score of 1.0, which would be the same penalty produced by the substitution of a character by a completely different character. A topic open for future study is the modification of this value in order to obtain better results.

After the rearrangement phase, a pair by pair comparison of the characters elements is performed in the last step of the procedure. As there are eight different characters

Algorithm 2 Our proposed edit distance algorithm.

```
1: if (length(string1) > length(string2)) then
2:   string1 ↔ string2
3: end if
4: string1temp ← string1
5: string2temp ← string2
6: common ← longestCommonSuStringNoHyphen(string1temp,string2temp)
7: string2mod ← hyphenString(length(string2))
8: diff ← 0
9: while (length(common) > 0) do
10:   pos1 ← indexOf(string1temp,common)
11:   pos2 ← indexOf(string2temp,common)
12:   string2mod ← replace(string2mod,pos1,length(common),common)
13:   if (abs(pos1-pos2) > abs(length(string1)-length(string2))) then
14:     if (length(common) > 1) then
15:       diff ← diff + 1
16:     else
17:       diff ← diff + 0.5
18:     end if
19:   end if
20:   string1temp ← replace(string1temp,pos1,length(common),
21:                         hyphenString(length(common)))
22:   string2temp ← replace(string2temp,pos2,length(common),
23:                         hyphenString(length(common)))
24:   common ← longestCommonSuStringNoHyphen(string1temp,string2temp)
25: end while
26: for all i such that 0 ≤ i ≤ length(string2temp) do
27:   char ← charAt(string2temp,i)
28:   if char ≠ "-" then
29:     pos2 ← indexOf(string2mod,"-")
30:     string2mod ← replace(string2mod,pos2,1,char)
31:   end if
32: end for
33: for all i such that 0 ≤ i ≤ length(string2temp) do
34:   if ((i ≥ length(string1)) or (charAt(string1,i) ≠ charAt(string2mod,i)) then
35:     diff ← diff + 1
36:   end if
37: end for
38: return floor(100 − diff · 100 / length(string2))
```

in `string1` and `string2mod`, the score is increased in eight units from 4.5 up to 12.5. In order to facilitate the identification of the dissimilar characters, Table 2 displays in bold font the dissimilar elements of the two strings.

Taking into account that the length of the longest string (`string2`) is 27, the comparison between `string1` and `string2` provides the following output:

$$\text{Result} = 100 - \left\lfloor \frac{12.5 \cdot 100}{27} \right\rfloor = 100 - 46 = 54.$$

A score of 54 implies that 54% of the longest string, `string2`, is also contained in the shorter string, `string1`.

4. Special signatures

When designing this test, our goal was to check the behaviour of `ssdeep`'s algorithm and our proposed algorithm when using some special strings, whose pattern could appear in certain real-world scenarios (for example, when obtaining the signature of files containing lists of elements such as names, file paths, etc.).

Even though we are aware that the tests included below represent extreme cases with ad-hoc strings, we believe it is worthwhile to test both algorithms in this scenario, as it represents different degrees of content rotation and modification.

Step	Element	Content
0	string1temp	ABCDEFGHIJKLMNOPQRSTUVWXYZ
	string2temp	1XYZI2JKL3MNOPQ4BCDEFGH5678
	string2mod	--------------------------
	score	0.0
1	substring	BCDEFGH
	string1temp	A-------IJKLMNOPQRSTUVWXYZ
	string2temp	1XYZI2JKL3MNOPQ4-------5678
	string2mod	-BCDEFGH-------------------
	score	1.0
2	substring	MNOPQ
	string1temp	A-------IJKL-----RSTUVWXYZ
	string2temp	1XYZI2JKL3-----4-------5678
	string2mod	-BCDEFGH----MNOPQ----------
	score	2.0
3	substring	JKL
	string1temp	A-------I--------RSTUVWXYZ
	string2temp	1XYZI2---3-----4-------5678
	string2mod	-BCDEFGH-JKLMNOPQ----------
	score	3.0
4	substring	XYZ
	string1temp	A-------I--------RSTUVW---
	string2temp	1---I2---3-----4-------5678
	string2mod	-BCDEFGH-JKLMNOPQ------XYZ-
	score	4.0
5	substring	I
	string1temp	A----------------RSTUVW---
	string2temp	1----2---3-----4-------5678
	string2mod	-BCDEFGHIJKLMNOPQ------XYZ-
	score	4.5
6	string1	**A**BCDEFGHIJKLMNOPQ**RSTUVW**XYZ
	string2mod	**1**BCDEFGHIJKLMNOPQ**234567**XYZ**8**
	score	12.5

Table 2: String rearrangement example.

The strings included in this test are the following ones:

S01: ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz
S02: ABCDEFGHIJKLMNOPQRSTUVWXYZABCDEFGHIJKLMNOPQRSTUVWXYZ
S03: abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
S04: 12345678901234567890123456ABCDEFGHIJKLMNOPQRSTUVWXYZ
S05: BADCFEHGJILKNMPORQTSVUXWZYbadcfehgjilknmporqtsvuxwzy
S06: CDABGHEFKLIJOPMNSTQRWXUVabYZefcdijghmnklqropuvstyzwx
S07: EFGHABCDMNOPIJKLUVWXQRSTcdefYZabklmnghijstuvopqrwxyz
S08: IJKLMNOPABCDEFGHYZabcdefQRSTUVWXopqrstuvghijklmnwxyz
S09: QRSTUVWXYZabcdefABCDEFGHIJKLMNOPwxyzghijklmnopqrstuv
S10: ghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZabcdef

The first string, S01, can be considered the base element of the set. The second string replaces the second half of S01 with its own first half. String S03 swaps the two blocks that form S01. In addition to the previous change, string S04 replaces the first half of the string with digits. Strings S05 to S10 take as basis the first string and perform transpositions of blocks whose size is 1, 2, 4, 8, 16, and 32 characters, respectively.

The results generated when comparing these special signatures are included in Tables 3, 4, and 5. Table 3 shows the results obtained when using ssdeep with signature files whose content replies the strings of the tests. As the limitation imposed by ssdeep regarding the minimum length for the common substrings produces as a result that several comparisons are not effectively performed (ssdeep directly assigns a score of 0 in those cases), we have implemented the logic of ssdeep's algorithm in Java Standard Edition [12] and have removed that limitation in our code. Thus, Table 4 shows the results that ssdeep would provide if it did not apply the aforementioned minimum length requirement. Finally, Table 5 displays the results obtained with our proposed algorithm once implemented as another Java application.

As it can be observed, our algorithm is able to provide meaningful results in all the comparisons, which is not the case in ssdeep. For example, the comparison between S01 and S07, which renders a score of 0 in ssdeep, is evaluated as having a similarity degree of 77% by our algorithm. Following that example, the modified version of ssdeep without the minimum length requirement generates a score of 55 which, even representing a better result, it still fails to properly reflect the fact that S01 and S07 share far more than half of their content.

When inspecting the tables, it can be stated that the results provided by our algorithm are more realistic according to the similarity definition given in the Introduction. For instance, when comparing S01 to S03 and S04, it is clear that S03 is almost the same string as S01, whilst S04 only shares with S01 half of its string. However, ssdeep is not able to detect that difference and assigns a value of 50% in both cases. In comparison, our algorithm computes the similarity degree as 97% and 49%, respectively.

Even though the modified version of ssdeep provides higher results than our algorithm in some instances (e.g., when comparing S02 and S05 or S04 and S06), those differences are small and do not imply a representative difference. However, when our algorithm provides higher results the difference in some instances is quite important (e.g., when processing S08 and S09). In fact, the average difference in the scores of the test when comparing different strings is 18.96 in favour of our method. We are aware that, in general, a higher value should not imply a better result; however, when comparing the test strings, which clearly share an important part of their contents, a higher result implies a better similarity detection capability.

	S01	S02	S03	S04	S05	S06	S07	S08	S09	S10
S01	100	50	50	50	0	0	0	55	63	63
S02	50	100	50	50	0	0	0	47	50	50
S03	50	50	100	50	0	0	0	36	44	90
S04	50	50	50	100	0	0	0	32	32	50
S05	0	0	0	0	100	0	0	0	0	0
S06	0	0	0	0	0	100	0	0	0	0
S07	0	0	0	0	0	0	100	0	0	0
S08	55	47	36	32	0	0	0	100	32	32
S09	63	50	44	32	0	0	0	32	100	32
S10	63	50	90	50	0	0	0	32	32	100

Table 3: Test results for special cases with `ssdeep`.

	S01	S02	S03	S04	S05	S06	S07	S08	S09	S10
S01	100	50	50	50	50	50	55	55	63	63
S02	50	100	50	50	29	32	36	47	50	50
S03	50	50	100	50	25	29	32	36	44	90
S04	50	50	50	100	25	29	29	32	32	50
S05	50	29	25	25	100	25	29	29	32	32
S06	50	32	29	29	25	100	29	29	32	32
S07	55	36	32	29	29	29	100	32	32	32
S08	55	47	36	32	29	29	32	100	32	32
S09	63	50	44	32	32	32	32	32	100	32
S10	63	50	90	50	32	32	32	32	32	100

Table 4: Test results for special cases with modified version of `ssdeep`.

	S01	S02	S03	S04	S05	S06	S07	S08	S09	S10
S01	100	50	97	49	50	50	77	89	93	97
S02	50	100	49	49	25	25	37	43	47	49
S03	97	49	100	50	50	50	74	85	91	97
S04	49	49	50	100	25	25	37	43	47	49
S05	50	25	50	25	100	50	50	50	50	50
S06	50	25	50	25	50	100	50	50	50	50
S07	77	37	74	37	50	50	100	77	79	75
S08	89	43	85	43	50	50	77	100	87	87
S09	93	47	91	47	50	50	79	87	100	93
S10	97	49	97	49	50	50	75	87	93	100

Table 5: Tests results for special cases with our algorithm.

5. Conclusions

In this contribution we have presented a new edit distance algorithm that can be used in fuzzy hashing applications. Our algorithm provides better results than `ssdeep`'s algorithm according to a definition of similarity useful in computer forensics when comparing two files, and that interprets similarity as the percentage of a file that is also present in another file. We have implemented both our algorithm and a modified version of `ssdeep` in Java, and have used those two applications together with version 2.12 of `ssdeep` in order to test some strings that could represent the signature of files including a list of elements.

The tests performed with the three applications allow us to state that our algorithm provides results better adapted to the aforementioned definition of similarity, so it can be considered as an alternative for the edit distance currently implemented in `ssdeep` and other fuzzy hashing applications.

Acknowledgment

This work has been partially supported by Comunidad de Madrid (Spain) under the project S2013/ICE-3095-CM (CIBERDINE) and by Ministerio de Economía y Competitividad (Spain) under the grant TIN2014-55325-C2-1-R (ProCriCiS).

References

[1] N. Harbour, "Dcfldd. defense computer forensics lab," 2002. [Online]. Available: http://dcfldd.sourceforge.net

[2] V. Gayoso Martínez, F. Hernández Álvarez, and L. Hernández Encinas, "State of the art in similarity preserving hashing functions," in *Proc. of WorldComp 2014 - International Conference on Security & Management - SAM'14*, 2014, pp. 139–145.

[3] A. Tridgell. (2014) Fuzzy hashing and ssdeep. [Online]. Available: http://ssdeep.sourceforge.net/

[4] F. J. Damerau, "A technique for computer detection and correction of spelling errors," *Communications of the ACM*, vol. 7, no. 3, pp. 171–176, 1964.

[5] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Soviet Physics Doklady*, vol. 10, no. 8, pp. 707 – 710, 1966. [Online]. Available: http://profs.sci.univr.it/~liptak/ALBioinfo/files/levenshtein66.pdf

[6] M. Karpinski, "On approximate string matching," *Lecture Notes in Computer Science*, vol. 158, pp. 487–495, 1983.

[7] R. A. Wagner and M. J. Fischer, "The string-to-string correction problem," *Journal of the ACM*, vol. 21, no. 1, pp. 168–173, 1974.

[8] J. Kornblum, "Identifying almost identical files using context trigger piecewise hashing," *Digital Investigation*, vol. 3(S1), pp. 91–97, 2006.

[9] A. Tridgell, "Efficient algorithms for sorting and synchronization," Master's thesis, The Australian National University. Department of Computer Science, Canberra, Australia, 1999.

[10] ——, "Spamsum readme," 1999. [Online]. Available: http://samba.org/ftp/unpacked/junkcode/spamsum/README

[11] Wikipedia. (2014) Damerau-Levenshtein distance. [Online]. Available: http://en.wikipedia.org/wiki/Damerau-Levenshtein_distance

[12] Oracle. (2015) Java SE. [Online]. Available: http://www.oracle.com/technetwork/java/javase/overview/index.html

A Java Implementation of a Multisignature Scheme

V. Gayoso Martínez[1], **L. Hernández Encinas**[1], **A. Martín Muñoz**[1], **and M. A. Álvarez Mariño**[2]
[1]Information Processing and Cryptography (TIC), Institute of Physical and Information Technologies (ITEFI)
Spanish National Research Council (CSIC), Madrid, Spain
[2]SACYL, Gerencia de Atención Primaria, Zamora, Spain

Abstract— *Multisignature protocols are digital signature schemes that allow a group of users to sign a message so that the signature thus produced is valid only if all the members of the group participate in the signature process. In general, these schemes need the collaboration of a Trusted Third Party, which computes and securely stores some of the parameters associated to the scheme.*

In this work, we present our results and conclusions after implementing as a Java application a multisignature scheme based on the Integer Factorization Problem and the Subgroup Discrete Logarithm Problem.

Keywords: Digital Authentication, Java, Multisignatures

1. Introduction

In multisignature schemes, a group of users, typically denoted as G, signs a document such that the signature is valid only if all the members of the group take part in the process and the signature verifies a specific condition. These schemes have a direct application in corporate scenarios for signing contracts, validating agreements, etc.

From a naive point of view, the easiest way to carry out a multisignature consists in computing the individual signatures of all the signers and concatenating them, so the multisignature is composed of the sequence of individual signatures. However, this method is not practical for large groups of users, since the length of the multisignature is proportional to the number of signers.

The first practical multisignature scheme was proposed in [1], where a modification of the RSA cryptosystem was used in such a way that the RSA module consisted in the product of three primes instead of just two. In [2], another scheme was proposed where the signature length is similar to the length of a simple signature and shorter than the signature obtained from the scheme presented in [1]. However, this scheme can be used only if the cryptosystem is bijective, making it difficult to implement. Other proposals based on the RSA cryptosystem are, for example, those described in [3], [4], [5], [6], [7].

Regarding multisignature schemes based on the Discrete Logarithm Problem, in the scheme described in [8] the group of signers must cooperate in order to sign the message and send the signature to a given group of verifiers, but only through the union of all the verifiers it is possible to validate the multisignature. In addition to that, when producing the multisignature the signers not only must use their own private keys, but also the public key of each verifier, which is an important limitation [9], [10]. In the scheme proposed in [11], a multisignature can be performed only if the verifiers of the signature belong to a previously specified group, but apart from that limitation the scheme has some weaknesses [12], [13].

From a more general point of view, a generic public key multisignature scheme is presented in [14]. In that model, each one of the signers must have a certified public key with its corresponding private key, which must be generated by the signer himself. The signers interact completing a number of rounds, where in each round each signer receives a message, performs several calculations and sends the resulting message to the next signer. In this generic model, it should be computationally infeasible to forge a multisignature if there exists at least one honest signer in the group.

In comparison with the previous proposals, the multisignature scheme presented in [15] by one of the authors of this contribution has the advantage that each signer has his own private key, but all of them share the same public key. Besides, the procedure is secure, efficient, and independent of the number of signers. In addition to that, the signature is determined by all the signers in a certain pre-established order and the scheme allows to add new signers at the end of the signing chain, making it easier to update old signatures. Regarding the validation procedure, the scheme requires the verification of a certain property involving the signature itself, the original message, the number of signers, and some of the scheme's public parameters.

This work presents the results obtained when implementing a modified version of the multisignature scheme described in [15] using the Java language, where the modifications introduced have the goal of adapting the scheme to devices with limited resources and making the signing procedure more flexible by allowing the users to operate the scheme in any given order.

The rest of this paper is organized as follows: In Section 2, a detailed description of the mulsignature scheme is included. Section 3 describes the Java application developed in order to test the feasibility of the scheme. Section 4 provides a numerical example of the parameter and signature generation procedures. In Section 5, we offer to the readers the experimental results obtained with our Java application. Finally, our conclusions are presented in Section 6.

2. Description of the scheme

The security of the scheme described in [15] is based on the Integer Factorization Problem (IFP) and the Subgroup Discrete Logarithm Problem (SDLP), and as such it was analysed in [15] (where, in addition to that, interested readers can find a more detailed discussion about other multisignature schemes).

As it is well known, the IFP can be described as follows [16]: Given a positive integer n, find its prime factorization; that is, write $n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$, where p_i are pairwise distinct primes and each $e_i \geq 1$.

Besides, the SDLP is defined as follows [16]: Let p be a prime and q a prime divisor of $p-1$. Let us consider g a generator of the unique subgroup H of $\mathbb{Z}_p^*$ of order q, and y an element in H. The problem is that of computing the integer x, $0 \leq x \leq q-1$, such that $y = g^x \pmod p$.

Let $G = \{U_1, U_2, \ldots, U_t\}$ be the group of t users allowed to perform signatures, and $\mathcal{T}$ the Trusted Third Party (TTP) managing the scheme's parameter generation process. The following subsections include all the details of the multisignature scheme.

2.1 Setup phase

In this phase, $\mathcal{T}$ generates the system parameters, its own private key, and the public key shared by the group. The steps that $\mathcal{T}$ must complete are the following:

1) $\mathcal{T}$ chooses two large primes p and q, such that $p = u_1 r p_1 + 1$ and $q = u_2 r q_1 + 1$, where r, p_1, and q_1 are prime numbers and $u_1, u_2 \in \mathbb{Z}$ with $\gcd(u_1, u_2) = 2$; that is, $u_1 = 2v_1$, $u_2 = 2v_2$, where v_1 and v_2 are prime numbers. In the original version [15], v_1 and v_2 could be composite numbers; we have introduced this modification so that the number of factors of $\lambda(n)$ (see next step) does not depend on v_1 and v_2, which improves the iteration through the divisors of $\lambda(n)$ in the third step.
   In order to guarantee the security of the scheme, the bit length of r must be selected so that the SDLP of order r in $\mathbb{Z}_n^*$ is computationally infeasible.
2) $\mathcal{T}$ computes the values n, the Euler function $\phi(n)$, and the Carmichael function $\lambda(n)$, where $n = p \cdot q$, $\phi(n) = (p-1)(q-1) = u_1 u_2 r^2 p_1 q_1$, and $\lambda(n) = \text{lcm}(p-1, q-1) = 2v_1 v_2 r p_1 q_1$.
3) $\mathcal{T}$ selects an element $\alpha \in \mathbb{Z}_n^*$ with multiplicative order r modulo n, and that fulfils the condition $\gcd(\alpha, \phi(n)) = 1$. The element α can be efficiently computed, as at this point $\mathcal{T}$ knows the factorization of n and consequently it knows $\phi(n)$ and $\lambda(n)$.
   In practice, it is enough to find a random value $g \in \mathbb{Z}_n^*$ such that $g^{\lambda(n)} \equiv 1 \pmod n$ and check that none of the 62 non-trivial divisors of $\lambda(n)$ are the actual order of g [17]. By non-trivial divisor we mean a divisor of $\lambda(n)$ different from 1 or $\lambda(n)$. The number of non-trivial divisors of $\lambda(n)$ is derived from the fact that $\lambda(n) = 2v_1 v_2 r p_1 q_1$ and all the factors are prime numbers. Once the value g is found, the generator is obtained through the following computation [17]:
   $$\alpha = g^{\lambda(n)/r} \pmod n.$$
4) $\mathcal{T}$ generates a secret random number $s \in \mathbb{Z}_r^*$ and determines
   $$\beta = \alpha^s \pmod n.$$
5) $\mathcal{T}$ publishes the values n, r, α, and β, while the elements p, q, and s are kept secret.
6) $\mathcal{T}$ sets up its private key by generating four random numbers $a_0, b_0, c_0, d_0 \in \mathbb{Z}_r^*$.
7) $\mathcal{T}$ determines the shared public key for G by computing the elements
   $$P = \alpha^{a_0} \cdot \beta^{b_0} \pmod n \equiv \alpha^h \pmod n,$$
   $$Q = \alpha^{c_0} \cdot \beta^{d_0} \pmod n \equiv \alpha^m \pmod n,$$
   where $h \equiv (a_0 + s b_0) \pmod r$ and $m \equiv (c_0 + s d_0) \pmod r$.

2.2 User's private key generation

In order to prevent $\mathcal{T}$ from impersonating any member of G, the secret key of each user U_i is composed of four values, two of which are only known to U_i. With that goal in mind, the following steps must be completed:

1) U_i generates two secret integers $b_i, d_i \in \mathbb{Z}_r$ at random and sends the values $\alpha^{b_i} \pmod n$ and $\alpha^{d_i} \pmod n$ to $\mathcal{T}$ using a secure channel.
2) $\mathcal{T}$ computes
   $$A_i = \alpha^h (\alpha^{b_i})^{-s} \pmod n \equiv \alpha^{a_i} \pmod n,$$
   $$C_i = \alpha^m (\alpha^{d_i})^{-s} \pmod n \equiv \alpha^{c_i} \pmod n,$$
   and sends the values A_i and C_i to the user U_i using a secure channel.
3) The private key of U_i is the set (A_i, b_i, C_i, d_i). Note that $\mathcal{T}$ can determine a_i and c_i since it knows h, k, α^{b_i}, and α^{d_i}, but it can compute neither b_i nor d_i because it cannot solve the SDLP. Similarly, U_i cannot compute the values a_i and c_i. As a consequence, both $\mathcal{T}$ and U_i have access to only two out of the four user's secret key parameters.

2.3 Parameter and key verification

Each member of the signer group, U_i, $1 \leq i \leq t$, may check the validity of the system parameters by verifying that $\alpha \not\equiv 1 \pmod n$ and $\alpha^r \equiv 1 \pmod n$.

Then, each signer, U_i, $1 \leq i \leq t$ can verify that their private key is related to the shared public key, by checking

$$P \equiv A_i \cdot \beta^{b_i} \pmod n, \quad Q \equiv C_i \cdot \beta^{d_i} \cdot \beta^{d_i} \pmod n. \quad (1)$$

This verification works because of the following chain of equivalences:

$$A_i \cdot \beta^{b_i} \equiv \alpha^{a_i} \cdot \beta^{b_i} \equiv \alpha^{a_i + s \cdot b_i} \equiv \alpha^h \equiv P \pmod n,$$
$$C_i \cdot \beta^{d_i} \equiv \alpha^{c_i} \cdot \beta^{d_i} \equiv \alpha^{c_i + s \cdot d_i} \equiv \alpha^k \equiv Q \pmod n.$$

2.4 Multisignature generation

Let M be the message to be signed by a member of G. By using, for example, a public hash function of the SHA-2 family [18], either the signing user or $\mathcal{T}$ compute $\mathfrak{h}(M) = m$, where m represents the hash output.

In this contribution we have modified the scheme originally proposed in [15] so, given the set of signing users $G = \{U_1, U_2, \ldots, U_t\}$, they can complete the signature in any order, which reflects better the reality of organizations and the potential temporary (un)availability of the members of G. In order to generate the multisignature, the following steps must be completed:

1) The first signer, U_j, $1 \leq j \leq t$, must obtain the values F_j and g_j that compose his partial signature in the following way:
$$\begin{aligned} F_j &\equiv A_j \cdot C_j^m \pmod{n}, \\ g_j &\equiv b_j + m \cdot d_j \pmod{r}. \end{aligned} \tag{2}$$
Then, U_j sends the partial signature (F_j, g_j) to the next signer, U_k, $1 \leq k \leq t$, $k \neq j$.
2) The second signer, U_k, verifies U_j's signature by checking if the following equivalence holds:
$$P \cdot Q^m \equiv F_j \cdot \beta^{g_j} \pmod{n}.$$
If that is the case, U_k computes his partial signature for the message in the following way:
$$\begin{aligned} F_k &\equiv F_j \cdot A_k \cdot C_k^m \pmod{n} \\ &\equiv \alpha^{a_j + a_k + m(c_j + c_k)} \pmod{n}, \\ g_k &\equiv g_j + b_k + m \cdot d_k \pmod{r} \\ &\equiv b_j + b_k + m(d_j + d_k) \pmod{r}. \end{aligned} \tag{3}$$
3) Then, U_k sends the partial signature (F_k, g_k) to the next signer, U_l, $1 \leq l \leq t$, $l \neq j, k$, and the procedure is repeated until all the group members have signed the message. The signature computed by the last user represents the multisignature for M, denoted as (F, g).

2.5 Multisignature verification

Any verifier knowing the message, M, the hash function, $\mathfrak{h}$, the public key of the group G, (P, Q), the number of members of the group, t, and the group signature, (F, g), can check if the signature is valid through the following computation:

$$P^t \cdot Q^{tm} \equiv F \cdot \beta^g \pmod{n}. \tag{4}$$

Equation (4) can be justified from expressions (1)–(3):

$$\begin{aligned} &F \cdot \beta^g \pmod{n} \equiv \\ &\equiv \alpha^{a_1 + \cdots + a_t + m(c_1 + \cdots + c_t)} \beta^{b_1 + \cdots + b_t + m(d_1 + \cdots + d_t)} \\ &\equiv \prod_{j=1}^{t} \alpha^{a_j} \cdot \beta^{b_j} \left(\alpha^{c_j} \cdot \beta^{d_j}\right)^m \pmod{n} \\ &\equiv \prod_{j=1}^{t} P \cdot Q^m = P^t \cdot Q^{t \cdot m} \pmod{n}. \end{aligned}$$

3. Java implementation of the scheme

The multisignature scheme presented in this contribution has been implemented as a Java application using Java SE 8. The application is composed of three panels which are described in detail in the next subsections. In each panel, the application user has the option of converting the data from decimal (or text, in the case of the message to be sign) to hexadecimal and vice versa.

In all the cases where a random number is needed, the application uses the standard Java classes `BigInteger` [19] and `Random` [20], so the requested number is obtained through the following code:

```
Random random = new Random();
BigInteger number =
    new BigInteger(numBits,random);
```

In the previous code, the element `numBits` indicates that the desired number must be uniformly distributed over the range 0 to $2^{\texttt{numBits}} - 1$. Regarding the `Random` class, it uses a 48-bit seed which is modified using a linear congruential formula according to the method described in Section 3.2.1 of [21].

Whenever a random prime number is needed, the following code is used after obtaining a random number:

```
BigInteger prime =
    number.nextProbablePrime();
```

By calling the method `nextProbablePrime()` over the element `number`, the application obtains the first integer greater than `number` that is probably prime, where the probability that the number returned is composite does not exceed 2^{-100} [19].

Rearding the process of checking if a given value is a prime number, we have used the method `isProbablePrime(int certainty)` implemented by the `BigInteger` class, where `certainty` represents the measure of uncertainty tolerated by the method: if the call returns `true` the probability that the `BigInteger` element is prime exceeds $(1 - (1/2)^{\texttt{certainty}})$ [19], [22]. If the bit length of the number to be analysed is less than 100, the function makes 50 passes of the Miller-Rabin test [23]. On the other hand, if the bit length is higher, it makes a variable number of passes of the Miller-Rabin test (the precise number depends on the actual bit length: 27 for numbers with less than 256 bits, 15 for numbers with less than 512 bits, 8 for numbers with less than 768 bits, 4 for numbers with less than 1024 bits, and 2 for numbers having at least 1024 bits), but in addition to that it runs the Lucas-Lehmer test [23]. An example code would be the following:

```
boolean isprime =
    number.isProbablePrime(10);
```

3.1 *Parameters* panel

This panel includes the general parameters, $\mathcal{T}$'s private key and the group's public key, as it can be seen in Figure 1. More specifically, it includes text boxes for the private elements p, q, s, a_0, b_0, c_0, and d_0, and for the public elements n, r, α, β, P, and Q.

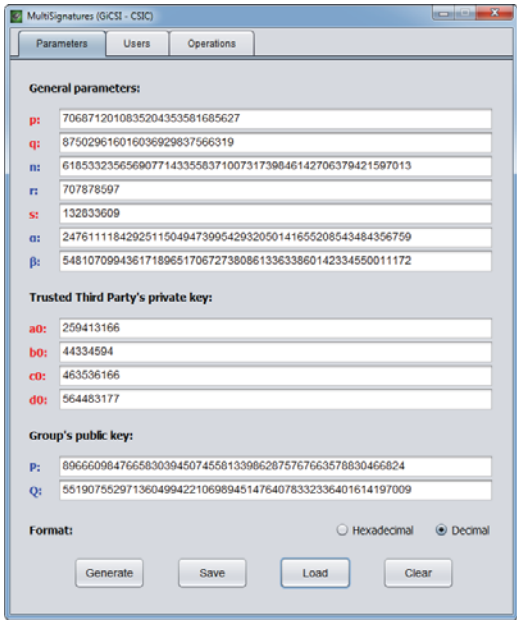


Fig. 1: *Parameters* pannel

There are four buttons available in this panel:

- *Generate*: It computes all the parameters according to the steps 1-7 of the procedure described in §2.1.
- *Save*: It allows the user to save either the public data or all the data included in this panel. The information is stored in a file using an XML structure.
- *Load*: It allows the user to overwrite the data existing in the text boxes with the information stored in the XML file selected by the user.
- *Clear*: It deletes the content of all the text boxes pertaining to this panel.

3.2 *Users* panel

This panel includes the private keys of the four users managed by this application. It is important to point out that the number of users implemented in this version of the application is not a limitation of the scheme, but a figure selected in order to simplify the usage of the application.

For each user from $i = 1$ to 4, a set consisting of the associated values A_i, b_i, C_i, and d_i is displayed, as it can be seen in Figure 2. We remind the reader that the values b_i and d_i are known only to U_i, while only $\mathcal{T}$ knows the value of the elements a_i and c_i.

The four buttons available in this panel implement the following functionality:

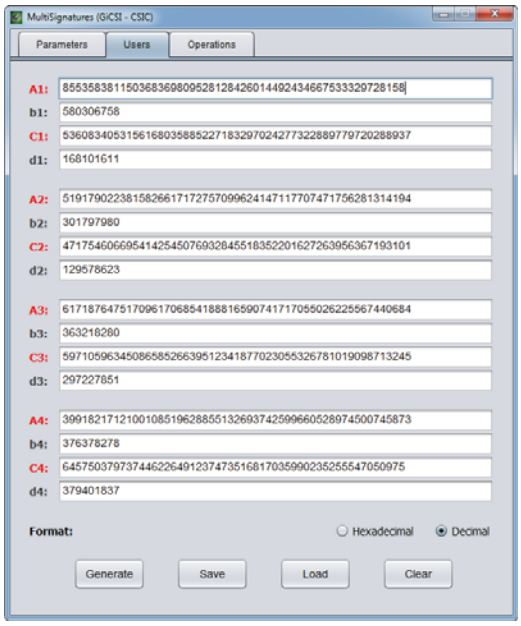


Fig. 2: *Users* panel.

- *Generate*: It generates all the private elements associated to the private keys of the users according to the steps 1 and 2 of the procedure described in §2.2.
- *Save*: It allows the user to save the private elements of the four users in a file using an XML structure.
- *Load*: It allows to overwrite the data existing in the text boxes with the information stored in the XML file selected by the user.
- *Clear*: It deletes the content of all the text boxes displayed in this panel.

3.3 *Operations* panel

This panel includes the operational functionality that can be accessed through the following buttons, as displayed in Figure 3:

- *Generate*: It generates the multisignature of the text message provided manually by the user according to the steps 1-3 described in §2.4. In order to obtain the elements F and g associated to the signature, it is mandatory to select in the panel the hash function and the starting signing user.
- *Order*: By selecting this button, the application changes the order of the users randomly, with the condition that the new order must be different from the previous one. Once a specific order is displayed, the user can select the starting signer by checking the proper element.
- *Verify*: It allows to verify if the multisignature provided by the user corresponds to the text message entered in its text box, as described in Section 2.5.
- *Clear*: It deletes the content of all the text boxes displayed in this panel.

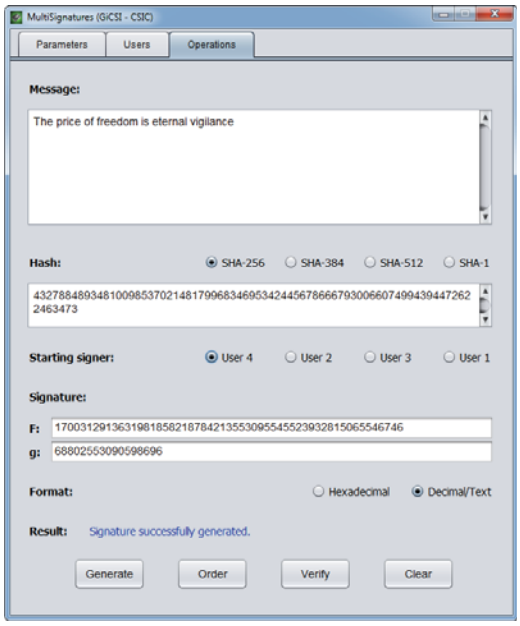


Fig. 3: *Operations* panel.

4. Numerical example

This section provides the details of the signature process depicted in Figures 1, 2, and 3, where the selected bit length for the base elements r, p_1, q_1, v_1, and v_2 is 32. This bit length is clearly inadequate from a security perspective, but making this choice allows us to manage smaller numbers in order to facilitate the comprehension of the example.

After selecting the option *Generate*, the application randomly produces the prime values $r = 707878597$, $p_1 = 3641604649$, and $q_1 = 303316411$. Then, the application enters a loop where it randomly generates the prime values v_1 and v_2 and computes p and q, exiting the loop once it checks that the values p and q are both prime numbers. In the example, the first values that satisfy that condition are $v_1 = 1371067121$, and $v_2 = 2037689777$, producing $p = 706871201083520435358168562 7$ and $q = 875029616016036929837566319$, which can be represented using 93 and 90 bits, respectively.

Then, after computing $\phi(n) = 618533235656907714335583709278799821929146509600234506 8$ (a 183-bit number), $\lambda(n) = 436892172102858277501773167241839791017969222 2$, and the 62 non-trivial divisors of $\lambda(n)$, the application enters a loop for computing a generator α such that it is coprime with $\phi(n)$. In the example, the generator thus calculated is $\alpha = 247611118429251150494739954293205014165520854348435675 9$.

Next, the application randomly generates the value $s = 132833609$, which must be coprime with r. Using α and s, the application computes $\beta = 548107099436171896517067273808613363386014233455001117 2$.

After that, it randomly generates the elements of the $\mathcal{T}$'s private key ($a_0 = 259413166$, $b_0 = 44334594$, $c_0 = 463536166$, and $d_0 = 564483177$) and computes the elements of the public key ($P = 896660984766583039450745581339862875767663578830466824$ and $Q = 5519075529713604994221069894514764078332336401614197009$).

As for the private keys of the signing users, in the example the application generates the following values: $a_1 = 855358381150368369809528128426014492434667533329728158$, $b_1 = 580306758$, $c_1 = 536083405315616803588522718329702427732288977972028 8937$, $d_1 = 168101611$, $a_2 = 519179022381582661717275709962414711770747175628131 4194$, $b_2 = 301797980$, $c_2 = 4717546066954142545076932845518352201627263956367193101$, $d_2 = 129578623$, $a_3 = 617187647517096170685418881659074171705502622556744068 4$, $b_3 = 363218280$, $c_3 = 5971059634508658526639512341877023055326781019098713245$, $d_3 = 297227851$, $a_4 = 399182171210010851962885513269374259966052897450074587 3$, $b_4 = 376378278$, $c_4 = 6457503797374462264912374735168170359902352555470509975$, and $d_4 = 379401837$.

Given the example message (the quote "The price of freedom is eternal vigilance") and the selected hash function (SHA-256), before computing the multisignature the application calculates the message's digest, whose representation in hexadecimal is `09917EFCA9E63C6BE3F5710D4E146146A152B64CE2E1DCDBBAAC3F6EBD6E19F1`. When considered as an integer modulo r, the value associated to the message is 70616700.

In the example, after using the *Order* button the distribution of signing users obtained is 4-2-3-1. If we select User 4 as the starting signer (see Figure 3), the elements forming the subsequent signatures calculated by the application are the following: $F_4 = 503576816705507771847786467994908210469682127707728 1113$, $g_4 = 26792106079256178$, $F_2 = 274673422284569756146097806234524791478777018719584923 2$, $g_2 = 35942521127858258$, $F_3 = 549247081103744449221518307665923446556919615681951584 7$, $g_3 = 56931771476788238$, $F_1 = 17003129136319818582187842135530955455239328150655467 46$, and $g_1 = 68802553090598696$. The multisignature resulting from this process is the signature computed by the last user, so $(F, g) = (F_1, g_1)$.

If, given the initial distribution 4-2-3-1, we had selected User 3 has the starting signer (see detail in Figure 4), the signatures calculated by the application would contain the elements $F_3 = 1525558802257083259276411227735843971228217026108102837$, $g_3 = 20989250348929980$, $F_1 = 426674824760343430582301180781736953299832414511548 3942$, $g_1 = 32860031962740438$, $F_4 = 526968246389121181210959699920776886130175536221988 1607$, $g_4 = 59652138041996616$, $F_2 = 17003129136319818582187842135530955455239328150655467 46$, and $g_2 = 68802553090598696$, where the resulting multisignature is $(F, g) = (F_2, g_2)$. As it can

be observed, the multisignature obtained is the same, as the order does not affect its final result, even though the partial signatures are different in each case. The same multisignature (F, g) would be produced if, for example, the initial distribution of users had been 1-2-3-4 and we had selected any of the four users as the starting signer.

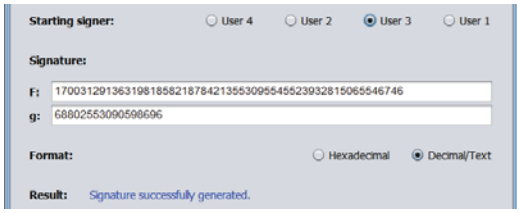


Fig. 4: Signature generation example.

5. Experimental results

The tests whose results are presented in this section were completed using a PC with Windows 7 Professional OS and an Intel Core i7 processor at 3.40 GHz.

Table 1 includes the running time obtained when executing the general parameters generation procedure in the testing computer with the bit lengths indicated in each case, where the bit length represents the maximum length in bits of the parameters r, p_1, q_1, v_1, and v_2. The time displayed for each bit length represents the average time of the generation of 100 sets of parameters.

As expected, the running time has an exponential shape, as it can be seen in Figure 5.

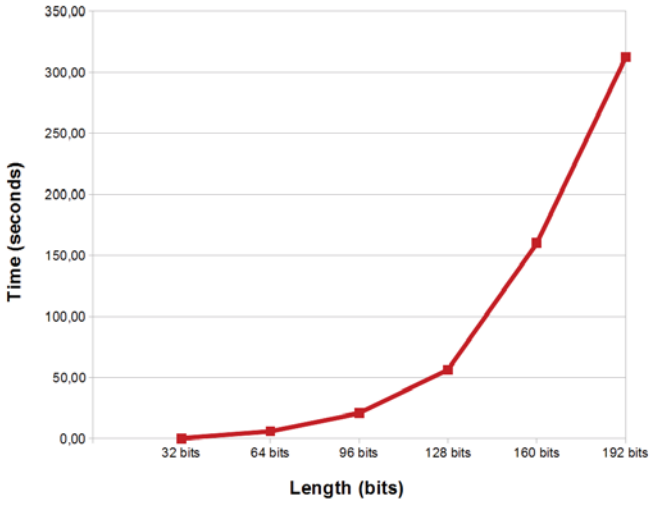


Fig. 5: General parameters' generation running time

Most of the parameter generation running time is due to the operations with prime numbers and `BigInteger` elements: obtaining the first prime number bigger than a certain value (method `nextProbablePrime()`) and checking if a candidate value is a prime number (method `isProbablePrime()`). Table 2 shows the average number of executions of the pieces of code calling those methods.

Figures 6 and 7 show graphically the information contained in Table 2. The main reason for the increase in the execution time in now clear: not only the application spends more time in each call to the methods `nextProbablePrime()` and `isProbablePrime()`, as a result of dealing with bigger numbers, but it also needs to call those methods more times, as the probability of $p = u_1 r p_1 + 1$ and $q = u_2 r q_1 + 1$ being prime numbers is lower as the bit length of those numbers increase.

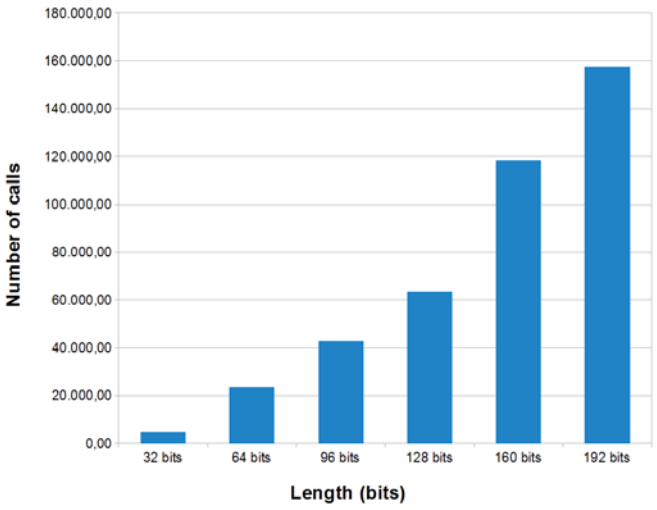


Fig. 6: Number of calls to the method `nextProbablePrime()`

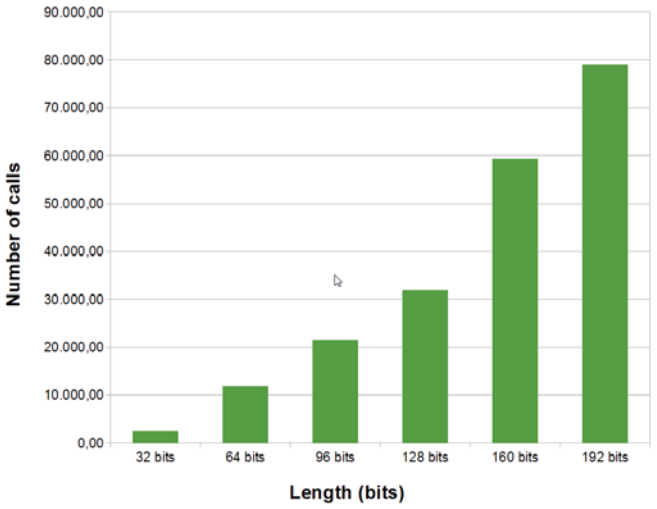


Fig. 7: Number of calls to the method `isProbablePrime()`

Table 1: General parameters generation running time

Length (bits)	32	64	96	128	160	192
Time (seconds)	0.37	6.06	21.03	56.70	160.11	312.36

Table 2: Number of calls to some methods implementd by the `BigInteger` class

Length (bits)	32	64	96	128	160	192
`nextProbablePrime()`	4,583.88	23,370.20	42,538.68	63,448.86	118,089.20	157,435.14
`isProbablePrime()`	2,337.15	11,802.72	21,416.38	31,883.81	59,279.56	78,977.87

6. Conclusions

In this contribution we have presented a modification of the multisignature scheme described in [15]. In order to implement the scheme as a Java application, we have modified the scheme by adding a new requirement which mandates v_1 and v_2 to be both prime numbers, as explained in §2.1. With this modification, we force the number of non-trivial divisors of $\lambda(n)$ to be exactly 62, which facilitates the implementation in devices with limited resources as the application does not need to factor v_1 and v_2 in order to determine the actual number of non-trivial divisors of $\lambda(n)$. In spite of this improvement, further enhancements may be necessary before deploying this scheme in certain platforms.

Regarding its usability, we have modified the scheme so the members of the group can sign a certain message in any given order.

The tests performed with the application allow us to confirm the expected difficulty in generating the system parameters for bit lengths greater than 64 bits. Nevertheless, as the system parameters generation procedure is only executed once by the Trusted Third Party, it is not a limitation for implementing this multisignature scheme in other devices that most of the times will only perform the signature generation and verification procedures.

Acknowledgment

This work has been partially supported by Comunidad de Madrid (Spain) under the project S2013/ICE-3095-CM (CIBERDINE) and by Ministerio de Economía y Competitividad (Spain) under the grant TIN2014-55325-C2-1-R (ProCriCiS).

References

[1] N. K. Itakura, K., "A public-key cryptosystem suitable for digital multisignatures," *NEC Research & Development*, vol. 71, pp. 1–8, 1983.

[2] T. Okamoto, "A digital multisignature scheme using bijective public-key cryptosystems," *ACM Transactions on Computer Systems*, vol. 6, no. 4, pp. 432–441, 1988.

[3] A.-F. M. Aboud, S.J., "A new multisignature scheme using re-encryption technique," *Journal of Applied Sciences*, vol. 7, pp. 1813–1817, 2007.

[4] K.-T. Harn, L., "New scheme for digital multisignature," *Electronics Letters*, vol. 25, pp. 1002–1003, 1989.

[5] H.-L. Kiesler, T., "RSA blocking and multisignature schemes with no bit expansion," *Electronics Letters*, vol. 26, pp. 1490–1491, 1990.

[6] P.-S. K. K. W. D. Park, S., "Two efficient RSA multisignature schemes," *Lecture Notes in Computer Science*, vol. 1334, pp. 217–222, 1997.

[7] L.-E. L. J. Pon, S.F., "Dynamic reblocking RSA-based multisignatures scheme for computer and communication networks," *IEEE Communications Letters*, vol. 6, no. 1, pp. 43–44, 2002.

[8] Y.-S. Laih, C.S., "Multisignature for specified group of verifiers," *Journal of Information Science and Engineering*, vol. 12, no. 1, pp. 143–152, 1996.

[9] W. He, "Weakness in some multisignature schemes for specified group of verifiers," *Information Processing Letters*, vol. 83, no. 2, pp. 95–99, 2002.

[10] S. Yen, "Cryptanalysis and repair of the multi-verifier signature with verifier specification," *Computers & Security*, vol. 15, no. 6, pp. 537–544, 1996.

[11] X.-G. Zhang, Z., "New multisignature scheme for specified group of verifiers," *Applied Mathematics and Computation*, vol. 157, pp. 425–431, 2004.

[12] W.-X. K. K. Lv, J., "Security of a multisignature scheme for specified group of verifiers," *Applied Mathematics and Computation*, vol. 166, pp. 58–63, 2005.

[13] Y.-K. Yoon, E.J., "Cryptanalysis of Zhang-Xiao's multisignature scheme for specified group of verifiers," *Applied Mathematics and Computation*, vol. 170, pp. 226–229, 2005.

[14] N.-G. Bellare, M., "Multi-signatures in the plain public-key model and a general forking lemma," in *13th ACM conference on Computer and Communications Security (CCS'06)*, 2006, pp. 390–399.

[15] R. Durán Díaz, L. Hernández Encinas, and J. Muñoz Masqué, "A multisignature scheme based on the SDLP and on the IFP," *Lecture Notes in Computer Science*, vol. 6694, pp. 135–142, 2011.

[16] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. Boca Raton, FL, USA: CRC Press, Inc., 1996.

[17] W. Susilo, "Short fail-stop signature scheme based on factorization and discrete logarithm assumptions," *Theoretical Computer Science*, vol. 410.

[18] NIST, *Secure Hash Standard*, National Institute of Standard and Technology, Federal Information Processing Standard Publication, FIPS 180-4, 2012.

[19] Oracle Corporation, *BigInteger (Java Platform SE 8)*, http://docs.oracle.com/javase/8/docs/api/java/math/BigInteger.html, 2014.

[20] ——, *Random (Java Platform SE 8)*, http://docs.oracle.com/javase/8/docs/api/java/util/Random.html, 2014.

[21] D. E. Knuth, *The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1997.

[22] Oracle Corporation, *OpenJDK - jdk8 - BigInteger.java)*, http://hg.openjdk.java.net/jdk8/jdk8/jdk/file/00cd9dc3c2b5/src/share/classes/java/math/BigInteger.java, 2015.

[23] R. E. Crandall and C. Pomerance, *Prime Numbers: A computational perspective*. Springer, New York, USA: Springer, 2005.

SESSION

POSTER PAPERS

Chair(s)

TBA

A Synthetic Provable Security Evaluation of Cryptographic Application with Entropy Sources

Nayoung Kim, Ju-Sung Kang*, and Yongjin Yeom
Dept. of Math. / Financial Information Security, Kookmin University, Seoul, Korea
*Corresponding author

Abstract— *On the conventional provable security models it is usually assumed that the secret keys are randomly chosen from the uniformly distributed key space. However in practice we can only use real distributions of key space that are far from uniform. There are several theoretical security bounds to cryptographic applications under the ideal setting in which it is assumed that the key space is uniformly distributed. In this work we examine some provable security bounds for several combined system of a cryptographic application and an entropy source used as the key space under the real setting.*

Keywords: Provable security, Cryptographic application, Entropy source.

1. Introduction

The security of all cryptographic applications is based on the confidentiality of secret keys. In order to analyze the security of cryptographic applications such as message authentication codes (MACs) or authenticated encryptions (AEs), on the conventional provable security models it is usually assumed that the secret keys are randomly chosen from the uniformly distributed key space. However, in practice we can only use real key distribution with entropy sources that are far from uniform. Dodis[1] presented an elementary inequality that upper bounds the expectation of advantage for any adversary attacking a cryptographic application. Bellare-Rogaway[2] investigated the security of encryption schemes such as CBC-MAC under the ideal setting. Meanwhile, Biryukov and Khovratovich[3] proposed a new authenticated encryption scheme PAEQ, which employs a fixed public permutation, and examined confidentiality and ciphertext integrity of the AE scheme. In this work we investigate several cryptographic applications with entropy sources under the real setting and obtain realistic security bounds from the theoretical security bounds to cryptographic applications under the ideal setting. We want to obtain the synthetic security bounds for the combined cryptographic applications with entropy sources under the provable security framework.

2. Cryptographic applications

We examine the security of typical cryptographic applications such as MACs and AEs. The security goal that we seek to achieve with a MAC is to be able to detect any attempt by an adversary to modify the transmitted data. An adversary attacking a MAC obtains a tag for any message that she chooses and tries to find out a valid pair $(M; tag)$. The adversary succeeds to forge if she makes a valid verification query $(M; tag)$ for the message M which is not a message that the adversary already knew a tag for earlier queries. On the other hand the security of an AE scheme is defined by the inability to distinguish between the two worlds, where an adversary has access to some oracles and a permutation. One world consists of the encryption and decryption oracles, and the second world consists of the random-bits and always-invalid oracles.

$\mathbf{Adv}^{(\cdot)}_{\mathrm{app}}(A, \mathrm{K})$ denotes the advantage of an adversary A attacking the cryptographic application under a key $\mathrm{K} \in \mathcal{K}$, where $(\cdot)$ is determined by the kind of attack. The min-entropy of a random variable K on the key space $\mathcal{K}$ is defined as $H_\infty(\mathrm{K}) = -\log(\max_{\mathrm{k}} \Pr[\mathrm{K} = \mathrm{k}])$. In order to analyze the security of cryptographic applications with entropy sources, we consider the following theorem[1].

Theorem 1: For every real-valued function $f : \{0,1\}^k \to \mathbb{R}^+ \cup \{0\}$, for any random variable K and uniform random variable U_k on $\{0,1\}^k$,

$$\mathbb{E}[f(\mathrm{K})] \le 2^{k - H_\infty(\mathrm{K})} \cdot \mathbb{E}[f(U_k)] \,.$$

3. Security under the ideal setting

In this section under the ideal setting, we describe previous provable security results of typical applications such as MACs and AEs.

- **One time MAC [1]**

Let $\mathcal{H} = \{h_a : \{0,1\}^n \to \{0,1\}^t \mid a \in \{0,1\}^s\}$ be a $\delta - AXU$ function family. We define an one time MAC with key length $k = s + t$ by $\mathbf{Tag}_{\mathrm{K}}(x) = h_a(x) \oplus b$, where parsing $\mathrm{K} = (a, b) \in \{0,1\}^s \times \{0,1\}^t$. Then advantage of an adversary A attacking one time MAC is bounded as

$$\mathbf{Adv}^{uf}_{\mathrm{MAC}}(A, \mathrm{K}) \le \delta \,.$$

- **Basic CBC MAC [2]**

Let $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher, and $m \ge 1$ be an integer. The basic CBC MAC is a deterministic and stateless MAC over the message space $\{0,1\}^{mn}$. Then

for any adversary A making at most q queries and one MAC verification query against the basic CBC MAC, there exists an adversary B making $q+1$ oracle queries against the block cipher E such that

$$\mathbf{Adv}^{uf}_{\mathrm{MAC}}(A,\mathrm{K}) \le \mathbf{Adv}^{prp}_{\mathrm{E}}(B,\mathrm{K}) + \frac{m^2q^2}{2^{n-1}} \ . \qquad (1)$$

- **AE scheme - PAEQ [3]**

In this work we consider PAEQ as an AE scheme. The AE scheme is denoted by Λ. The inputs to the AE scheme are consisted of a key K, a plaintext P, an associated data A, a public message number N, and a tag length t. In [3], the authors proved confidentiality and ciphertext integrity of PAEQ. They proposed a security proof for a fixed key length k, width n of a random permutation π and a block size of ciphertext $c = n - k - 16$. Let A be an adversary making σ_Λ queries and σ_π queries, where σ_Λ is the total number of queries to π made during the calls to the Λ oracle, and σ_π is the total number of queries to π and π^{-1} oracles together. Then Λ(PAEQ) satisfies the following inequalities.

$$\mathbf{Adv}^{con}_{\Lambda}(A,\mathrm{K}) \le \frac{(\sigma_\Lambda+\sigma_\pi)^2}{2^n} + \frac{2\sigma_\Lambda^2}{2^c} + \frac{2{\sigma_\pi}^2}{2^k} + \frac{2\sigma_\Lambda\sigma_\pi}{2^{n-16}} \ ,$$

$$\mathbf{Adv}^{int}_{\Lambda}(A,\mathrm{K}) \le \mathbf{Adv}^{con}_{\Lambda}(A,\mathrm{K}) + \frac{\sigma_\Lambda^2}{2^c} + \frac{q}{2^t} + \frac{\sigma_\Lambda q}{2^c} + \frac{q}{2^k} \ .$$

Fig. 1: Cryptographic applications under the ideal setting

Fig. 2: Cryptographic applications under the real setting

4. Security under the real setting

Now we combine cryptographic applications with an entropy source under the real setting. By applying some previous results, we can straightforwardly obtain upper bounds for the expectation of an adversary A's advantage against these combined cryptographic systems. On the contrary we find out that for the AE scheme PAEQ, we can directly obtain the upper bound of not the expectation but each adversary A's advantage by using the min-entropy of real key distribution. It is possible that by using these provable security bounds for the combined cryptographic systems we synthetically evaluate the cryptographic applications under the real setting.

- **One time MAC**

$\mathbf{Tag}_{\mathrm{K}}(x) = h_a(x) \oplus b$ is a δ-secure one time MAC with key length k and $f_A(\mathrm{K}) = \mathbf{Adv}^{uf}_{\mathrm{MAC}}(A,\mathrm{K}) : \{0,1\}^k \to [0,1]$ is a real-valued function. Then by Theorem 1, the expectation of an adversary A's advantage under the real setting satisfies the following inequality.

$$\mathbb{E}[f_A(\mathrm{K})] \le 2^{k-H_\infty(\mathrm{K})} \cdot \mathbb{E}[f_A(U_k)] \ .$$

- **Basic CBC MAC**

We have obtained the upper bound on advantage of adversary A by (1). Now, let $f_A(\mathrm{K}) = \mathbf{Adv}^{uf}_{\mathrm{MAC}}(A,\mathrm{K})$ and $h_B(\mathrm{K}) = \mathbf{Adv}^{prp}_{\mathrm{E}}(B,\mathrm{K})$. Then by Theorem 1,

$$\begin{aligned}\mathbb{E}[f_A(\mathrm{K})] &\le 2^{k-H_\infty(\mathrm{K})} \cdot \mathbb{E}[f_A(U_k)] \\ &\le 2^{k-H_\infty(\mathrm{K})} \cdot \left(\mathbb{E}[h_B(U_k)] + \frac{m^2q^2}{2^{n-1}}\right).\end{aligned}$$

- **AE scheme - PAEQ**

In the AE scheme case, a key K is used as a part of the inputs and the key length k is associated with the security of the AE. Therefore A's advantage under the real setting is directly influenced by the min-entropy of K such as:

$$\mathbf{Adv}^{con}_{\Lambda}(A,\mathrm{K}) \le \frac{(\sigma_\Lambda+\sigma_\pi)^2}{2^n} + \frac{2\sigma_\Lambda^2}{2^c} + \frac{2{\sigma_\pi}^2}{2^{H_\infty(\mathrm{K})}} + \frac{2\sigma_\Lambda\sigma_\pi}{2^{n-16}} \ ,$$

$$\mathbf{Adv}^{int}_{\Lambda}(A,\mathrm{K}) \le \mathrm{Adv}^{con}_{\Lambda}(A,\mathrm{K}) + \frac{\sigma_\Lambda^2}{2^c} + \frac{q}{2^t} + \frac{\sigma_\Lambda q}{2^c} + \frac{q}{2^{H_\infty(\mathrm{K})}}.$$

5. Conclusion

In this work we have obtained provable security bounds for several combined cryptographic applications with entropy sources under the real setting, where the key distribution isn't the uniform in general. In fact we have obtained upper bounds of the expectations of adversary's advantages for MAC applications by combining two previous results, and we have had directly an adversary's advantage for the AE scheme under the real setting by the min-entropy of key distribution.

Acknowledgment

This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (No. NRF-2014M3C4A7030648).

References

[1] Y. Dodis, Randomness in Cryptography [Online Version], Available at http://cs.nyu.edu/ dodis/randomness-in-crypto/, 2013.
[2] M. Bellare and P. Rogaway, Introduction to Mordern Cryptography [Online Version], Available at http://cseweb.ucsd.edu/ mihir/cse207/classnotes.html, 2005.
[3] A. Biryukov and D. Khovratovich, "PAEQ: Parallelizable Permutation-Based Authenticated Encryption", ISC 2014, Springer-Verlag, LNCS 8783, pp. 72-89, 2014.

A Provably Secure Authenticated Encryption Scheme based on Blockciphers with Large Input Lengths

Wangho Ju, Hojoong Park, Ju-sung Kang*, and Yongjin Yeom
Dept. of Math. / Financial Information Security, Kookmin University, Seoul, Korea
*Corresponding author

Abstract— *Authenticated Encryption scheme is a symmetric encryption which simultaneously provides confidentiality, integrity and authenticity assurances on the data. We propose an AE scheme based on blockciphers having large input and output lengths. Proposed AE scheme is provably secure against* IND_CPA *attack and* INT_CTXT *attack on the viewpoint of confidentiality and integrity, respectively. The distinct features of our AE scheme are that it is based on blockciphers having large input lengths such as 256-bit, and possible to implement each input and plaintext block by parallel computations.*

Keywords: Authenticated Encryption, Provable security, Modes of operation, Blockcipher.

1. Introduction

The need for Authenticated Encryption (AE) scheme emerged from the observation that securely combining an encryption mode with an authentication mode isn't a simple extension of some provable security results. In this work we propose a provably secure AE scheme based on blockciphers having large input lengths such as 256-bit Rijndael[1]. We provide proofs of confidentiality and integrity for our AE scheme by IND_CPA and INT_CTXT, where IND_CPA and INT_CTXT denote indistinguishability under chosen-plaintext attacks and integrity of ciphertext under chosen-message attacks, respectively. These results imply that our AE scheme is secure against chosen-ciphertext attack (CCA). On the computational efficiency view point, it is possible to implement each input and plaintext block of our AE scheme by the parallel processing.

2. The proposed AE scheme

The building block of our AE scheme can be divided into three inner functions using a blockcipher E_K with n-bit input size. The first is encryption function $enc(\cdot,\cdot)$, the second is preprocessing function $prep(\cdot,\cdot)$ and the last is integrity function $mac(\cdot,\cdot)$. The inputs to AE scheme are a plaintext P, an associated data A, a nonce N and a key K. The outputs of the scheme are a ciphertext C and a verification value Tag. The diagram of our scheme is illustrated in Fig.1. For each i, the variables in Fig.1 are as follows:

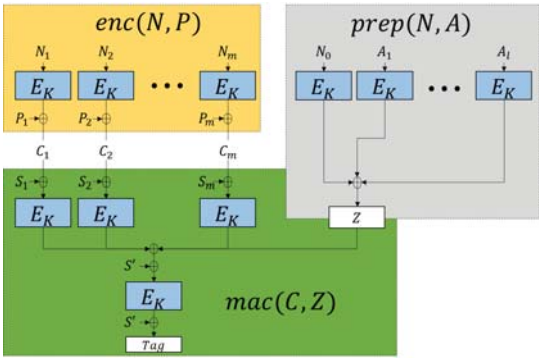


Fig. 1: Overall process of proposed AE scheme

- P_i and C_i: an n-bit block of P and C, respectively.
- N_i and A_i: variables such that $N_i = i\|N$ and $A_i = i\|A[i]$, where $A[i]$ is the i-th block of A.
- S_i and S': inner secret variables that derived by a key K such that $S_i = \alpha^i E_K(0)$ and $S' = \beta E_K(0)$, where α and β are appropriate values of $GF(2^n)^*$ that satisfy "Unique representations" in [2].

3. Provable security of the AE scheme

In order to analyze CCA security of an AE scheme we introduce the following definition[3].

Definition 1(*CCA3-security*). Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an AE scheme, \$ be an random-bit oracle, and $\perp$ be an always-invalid oracle. Then the CCA3-advantage of a computationally bounded adversary $\mathcal{A}$ is defined as

$$Adv_{\Pi}^{CCA3}(\mathcal{A}) = \left| Pr\left[K \xleftarrow{\$} \mathcal{K} : \mathcal{A}^{\mathcal{E}_K, \mathcal{D}_K}\right] - Pr\left[\mathcal{A}^{\$,\perp}\right] \right|.$$

It is well-known that any AE scheme that both IND_CPA and INT_CTXT secure is also secure under the CCA3-advantage context. Hence we prove IND_CPA and INT_CTXT security against any adversary $\mathcal{A}$ that asks at most q_e encryption queries which the total number of plaintext blocks is at most σ_1 and the total number of additional data blocks is at most σ_2 to the encryption oracle. Additionally $\mathcal{A}$ can ask at most q_d queries to decryption oracle. Throughout this paper we use the following notations.

- $Func(mn, n)$ is the family of all functions from mn-bit to n-bit, and $f \xleftarrow{\$} Func(mn, n)$ denotes random choosing of f.
- $Perm(\mathcal{T}, n)$ is the set of all mappings from tweak space $\mathcal{T}$ to random permutations on n-bit, and $\pi \xleftarrow{\$} Perm(\mathcal{T}, n)$ denotes random choosing of π_T for each $T \in \mathcal{T}$.
- $Adv_{\Pi}^{(\cdot)}(\mathcal{A})$ is denoted by $Adv_{\Pi}^{(\cdot)}(r)$, where the resources of $\mathcal{A}$ are r and $(\cdot)$ is determined by the kind of attack.

3.1 Confidentiality analysis

The confidentiality proof of the AE is proceeded through four steps and used five analysis models. For convenience, let proposed AE be M_0 and "random bits" oracle be M_4. M_1, M_2, and M_3 are illustrated in Fig. 2.

- **Step 1**. Calculate IND_CPA advantage for $\mathcal{A}$ that distinguishes M_0 and M_1. In this step E_K in $enc(\cdot,\cdot)$ and $prep(\cdot,\cdot)$ is transformed to a random function $f \xleftarrow{\$} Func(n, n)$. The advantage is bounded by $Adv_E^{prf}(\sigma_1 + \sigma_2)$.
- **Step 2**[2]. Compute the probability of distinguishability between M_1 and M_2. This probability depends on the fact that $E_K(S_i \oplus C_i)$ and $S' \oplus E_K(S' \oplus Y)$ in $mac(\cdot,\cdot)$ are changed to ideal tweakable blockciphers $\pi \xleftarrow{\$} Perm(\mathcal{T}, n)$. The advantage is bounded by $\frac{9.5(\sigma_1+q_e)^2}{2^n} + Adv_E^{prp}(2\sigma_1 + 2q_e)$.
- **Step 3**[2]. In order to generate a Tag uniformly, we change the $mac(\cdot,\cdot)$ in M_2 to a random function $\Phi \xleftarrow{\$} Func(*n, n)$. This process is to transform M_2 into M_3. The advantage is bounded by $\frac{(\sigma_1+q_e)^2}{2^n}$.
- **Step 4**. The difference between M_3 and M_4 comes from some collisions at the instance Z, output of $prep(N, A)$. This collision probability is bounded by $\frac{q_e^2}{2^n}$.

As a result we obtain that

$$\begin{aligned} Adv_{AE}^{\text{IND\_CPA}}(A) \quad \leq \quad & Adv_E^{prf}(\sigma_1 + \sigma_2) \\ & + Adv_E^{prp}(2\sigma_1 + 2q_e) \\ & + \frac{10.5(\sigma_1 + q_e)^2}{2^n} + \frac{q_e^2}{2^n}. \end{aligned}$$

3.2 Integrity analysis

We want to obtain the success probability to forge the verification value where the adversary asks one fresh query to the decryption oracle. The query can be classified by the freshness of three different components N, A, and C. Let τ be the bit-size of Tag.

- *Fresh nonce* N: A fresh N induces a new Z value, and Φ has a fresh input. Thus the probability to forge the tag is upper bounded by $2^{-\tau}$.
- *Fresh additional data* A: A fresh A induces that Φ has a fresh input. Thus the probability to forge the tag is upper bounded by $2^{-\tau}$.
- *Fresh ciphertext* C: In this case Φ has a directly fresh input, and the probability to forge the tag is upper bounded by $2^{-\tau}$.

Therefore the INT_CTXT advantage of $\mathcal{A}$ attacking the proposed AE is bounded as

$$Adv_{AE}^{\text{INT\_CTXT}}(\mathcal{A}) \leq Adv_{AE}^{\text{IND\_CPA}}(\mathcal{A}) + \frac{q_d}{2^\tau} .$$

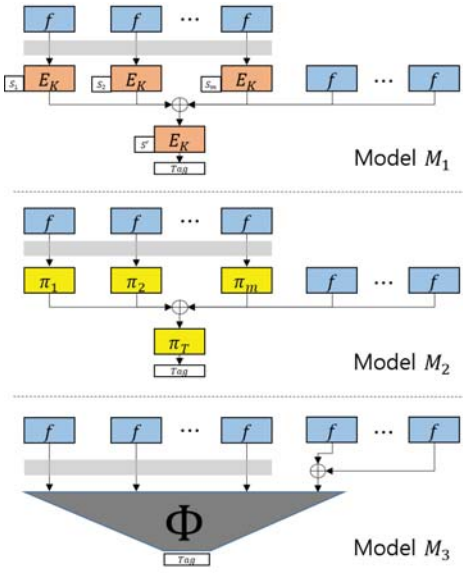


Fig. 2: Three models for the security proof

4. Conclusion

We have proposed an AE scheme based on blockciphers having large input-output lengths. The first feature of our AE scheme is that it is a provably secure against IND_CPA attack and INT_CTXT attack on the viewpoint of confidentiality and integrity, respectively. The second is not based on more primitives such as hash functions, streamciphers but only one primitive that is blockcipher. The third is possible to implement each input and plaintext block by parallel computations.

Acknowledgment

This work was supported by ICT R&D program of MSIP/IITP. [2014-044-014-002, Development of core technologies for quantum cryptography networking].

References

[1] J. Daemen and V. Rijmen, "The Design of Rijndael", Springer-Verlag, Heidelberg, 2001.
[2] P. Rogaway, "Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC", ASIACRYPT 2004, Springer-Verlag, LNCS 3329, pp. 16-31, 2004.
[3] T. Shrimpton, "A Characterization of Authenticated-Encryption as a Form of Chosen-Ciphertext Security", Cryptology ePrint archive, Report 2004/272, 2004.

Multimodal Biometric Methods for Temporary Reception Centres

G.L. Masala[1] and M.L. Ganadu[2],
[1]Department of Political Science, Communication, Engineering and Information Technologies and
[2] Department of Humanities and Social Sciences, University of Sassari, Sassari, ITALY.

Abstract - *The paper describes a project for a complete identification procedure of people with biometrical multimodal approach and data management in a real Italian temporary reception centre for immigrants or asylum seekers. For the entrance to the temporary centre, the authentication is based on the Scale Invariant Feature Transform (SIFT) features extracted by the face and the fingerprints of the users. This secure authentication allows the management of several services of the centre and the control of the access is differentiated for various user profile, and location, in order to protect the database from not controlled actions.*

Keywords: multimodal biometrics, SIFT, clinical records

1 Introduction

Since the beginning of 2014 almost 100 000 people have been arrived on Italian coasts (according to official data of the Italian Ministry of the Interior). Therefore police offices have a great deal of work to record new immigrants typically without official documents. In temporary reception centres for immigrants several facilities are available as refectory, beds, first complimentary objects and additionally staff of the centre have to check daily the presence of registered immigrants.

The complex framework named Individual Admission Plan (PAI) needs a coordination of the emergency centre for immigration with the prefectures to solve the primary identification of people, the daily reports of presence and healthy, and management of all the services.

In this paper we present a method, derived from basic face recognition and fingerprint classification approach, which is a credible candidate of management of the PAI.

2 Main results

In our temporary centre usually about 70 people are daily present and it may occur to accommodate new immigrants in a single group of about 20 individuals once a month, generally without identity documents. The system provides:

-Registration of new applicant with biometrical features and general information.

-Development and management of the electronic clinical records. Management of the administrative and information recording; essential information to establish the person's identity, former employment, status of asylum seeker and family relationships with relatives and people already present in the Italian territory.

-Management of daily access to refectory, complimentary pack and registration of presence.

The first day of arrival, for each person a face photo and fingerprints are taken. In particular the photograph is taken through a CCD camera while the fingerprints using the HI-SCAN PRO BIOMETRIKA scanner (FTIR, 500 dpi, 1" x 1").

Every day immigrants are recognised by means of their fingerprint in order to access to all services: refectory, complimentary pack, healthy treatment and at same time the presence is tracked and call the roll is not more compulsory.

The system is composed by two main modules. The first one ensures the identity of the person while the second one allows to manage the data. It is worth noting that data stored in clinical records are sensitive and require secure access.

To recognize the identity of the person a biometric recognition is used; the relevant features are obtained using the Scale Invariant Feature Transform (SIFT) representation [1-4] either for face and for the fingerprint recognition. SIFT can identify objects even in misperception or when partially hidden, because the descriptor SIFT feature is invariant to scale, orientation and distortion affine and partially invariant to illumination changes [1]. The SIFT key points of objects are first extracted from a set of reference images and stored in a database. So it is not necessary to store the fingerprints in the database but only SIFT models are recorded.

The biometric features are detected in a new image comparing each individual feature of the new image with the database previously obtained and measuring the Euclidean distance of their feature vectors. From the full set of correspondences, subsets of the key points, that are consistent with the object and its location, scale, orientation to filter the best matches, are identified in the new image. The determination of consistent clusters is carried out quickly using an implementation with efficient hash table of the generalized Hough transform [2].

The matching is performed considering the SIFT features located along a regular grid and matching overlapping patches; in particular the approach subdivides the images in different sub-images, using a regular grid with overlapping. The matching between two images is then performed by computing distances between all pairs of corresponding sub-images, and finally averaging them [4]. This is a realistic way because for example all parts of the face could be not matched with all others.

The identity module differentiates between two types of access: admission to a service or entrance to the centre. If a person is already inside the temporary centre, the use of a single fingerprint is required for the access to each service; all fingerprints are enabled and, if the check fails, an alarm system is activated. This is an efficient way to count and guarantee the service preventing multiple requests and to monitoring for the presence during the day. For the entrance to the temporary centre a multimodal authentication is required with the associated achievement of face image and of a fingerprint.

In general template created through the fusion of biometrics can be of considerable size becoming heavy to manage for the system. This strategy in our case is not feasible because it is difficult to merge SIFT features from different biometrics. Therefore the most advantageous choice remains a merge after the matching module through the fusion module .

Merging of the information for the multimodal approach is provided after that the single matching of the separated biometric features has been obtained. This is an additional level of security added to solve problem of camouflage and identity theft. In Figure 1 a simple diagram depicts the access system.

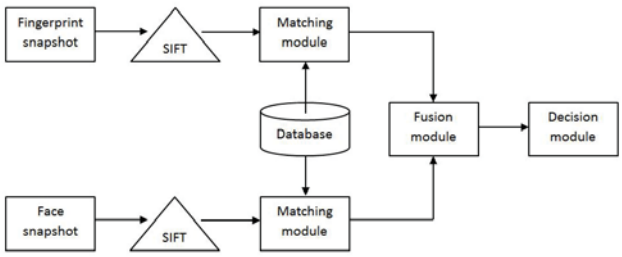


Fig.1 The complete authentication system for entry access based on multimodal biometrics.

The individual biometrics are independently evaluated by the system. In each step the system takes a decision and the fusion module expresses the final response by assessing the congruence results. At the level of decision the success of all biometric tests is required. In addition this approach allows to modify the algorithm of feature extraction without changing the system.

The access to the clinical/administrative records from the operator is made by fingerprint authentication as well. Data are encrypted in the clinical records. The clinical/administrative records is web-based so it is possible to work on data from different locations in the centre. The secure channel of communication with the web server is based on a virtual private connection (VPN). Furthermore the control of the access is differentiated for various user profiles and location in order to protect the database from not controlled actions.

To facilitate learning incrementally by the user, the system allows to configure the interface according to predefined user profiles. Moreover the integration with the most widely used tool for Office Automation as MS Office® (© Microsoft Corporation Inc.) allows:

• to extract in MS Excel® processing and statistical analysis, accompanied synthetic graphics, relative to the data managed by the application, socio-demographic information, detection of needs and nursing care;
• to generate reports both for users and for evaluation boards.
• to handle very rapidly and easily the production of documents containing the stored data in the system and customizing a series of prints directly through models configured with MS Word®.

All clinical data are accessible from tablet through the web server so it is possible to compile clinical records next to the location where medical examinations are carried out or to check history of the immigrants or to update details.

3 Conclusions

A complete system of data management controlled by multimodal biometrics based on SIFT features to be used in Temporary Reception Centres is here presented. The system guarantees the identity of persons and facilitates data management and related services.

4 References

[1] Y. Ke & R. Sukthankar. PCA-SIFT: A more distinctive representation for local image descriptors. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2004.
[2] D. Lowe. Local feature view clustering for 3d object recognition. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 682–688, 2001.
[3] D. Lowe. Distinctive image features from scale-invariant keypoints. *Int. Journal of Computer Vision*, 60, 91–110, 2004.
[4] M. Bicego, A. Lagorio, E. Grosso & M. Tistarelli. On the use of SIFT features for face authentication. In *Computer Vision and Pattern Recognition Workshop*, IEEE Conference 35, 2006.
[5] A.K. Jain, R. Bolle & S. Pankanti (Eds.). Biometrics: personal identification in networked society. Springer Science & Business Media,1999.

802.11 Tracking and Surveillance – A Forensic Perspective

M. Chernyshev, C. Valli, and P. Hannay
Security Research Institute, Edith Cowan University, Perth, Western Australia, Australia

Abstract - *This paper introduces a study on utilizing 802.11 client device traces for forensic purposes. The passive nature in which these traces can be collected presents a number of opportunities to the digital investigator. However, a forensically sound system must exhibit certain attributes not necessarily present in existing open-source tools that employ the associated collection techniques. Additional legal implications and difficulties in linking wireless devices back to their owners present a challenging research area.*

Keywords: Wi-Fi, Surveillance, Digital forensics, Smart devices

1 Introduction

802.11-based wireless signals represent a possible source of digital evidence [2]. The proliferation of the associated 'Wi-Fi' protocol, the omnipresent nature of access hotspots as well as observed and projected continued growth in the numbers and diversity of client devices facilitate an increasing pool of collectable signals [3]. However, only the signals that have been collected and processed in a qualified manner are likely to carry the required evidentiary potential [4]. The aim of the presented study is to ascertain the usefulness of selected 802.11 signals and the associated collection and analysis techniques for the purposes of client device identification and tracking as well as establishing a possible linkage to its owner.

We propose to examine the requirements for and derive a forensically sound and compliant collection and analysis method that is automated, accurate and repeatable. Therefore, our hypotheses are that (1) it is possible to derive a qualified collection technique using low cost hardware and freely available software components; (2) a number of devices can be uniquely identified on the basis of the data collected using this technique, and (3) attributes of potential forensic significance can be processed and extracted from the collected signals in an automated fashion.

2 Background and related work

Extraction of information from 802.11 signals has been subject to wide prior investigation. By design, the wireless access point (AP) discovery mechanism utilized by client devices is prone to leaking sensitive and potentially exploitable information [5]. It is common for Wi-Fi clients to store the list of previously associated networks to facilitate the subsequent automated discovery and AP association in a convenient manner [6]. The list of these networks comprises the Configured Network List (CNL) also known as the Preferred Network List (PNL) of the device [6, 7]. In accordance with the protocol, clients periodically scan for networks in this list by emitting probe requests – unencrypted communication management frames that carry the Media Access Control (MAC) address of the probing client and the Service Set Identifier (SSID) of the preferred network [1]. The standard structure of a probe request frame is presented in Figure 1. The passive and accessible manner in which 802.11 signals can be intercepted provided the basis for a number of research studies and non-academic projects.

2.1 Passive fingerprinting

Signal features of interest can be extracted by monitoring wireless communications without directly interacting with the observed device. Passive extraction can be facilitated at a number of communication layers of the OSI model – most commonly, physical (PHY) and data link (MAC) layer. A number of techniques have been described that are able to identify the type of a wireless Network Interface Controller (NIC) or derive a unique fingerprint for an individual device.

Focusing on the physical layer, signal transient extraction with subsequent classification and filtering was used to accomplish individual wireless transceiver identification [8]. Other signal characteristics such as amplitude and phase angle during the power-up interval were also used to discriminate between individual devices [8]. Leveraging hardware idiosyncrasies common to NIC components, a technique named PARADIS was utilized to identify individual transmitters using low-level signal characteristics in the modulation domain [9]. Furthermore, signal measurements from power amplifiers were found to be easily identifiable in a short period of time even at typical power levels [10].

At the data link layer, the analysis of active discovery implementation behavior was used to differentiate between unique wireless devices [11, 12]. Wireless frame inter-arrival time was used to capture unique fingerprints of both client devices and wireless APs using standard equipment [13]. The contents of the PNL extracted from probe request frames can also be used to derive a unique digital fingerprint for devices

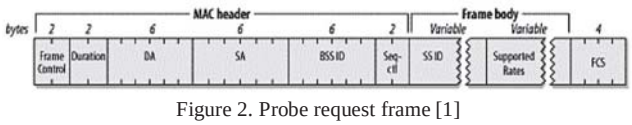


Figure 2. Probe request frame [1]

that probe for an exclusive or highly diverse set of networks [14]. Even relatively short capture sessions can yield unique fingerprints, with one study reporting capture frames below 8 minutes for the majority of analyzed devices [7]. However, the dynamic nature of PNLs implies that fingerprints should not be treated as implicit identifiers. Technique reliability is impacted when devices emit probes with blank SSIDs [15]. Interference and signal loss due to channel cycling can also limit the capture effectiveness [12].

2.2 Profiling and device location inference

The collected PNL-based fingerprints can provide the basis for identifying social relationships between device owners and tracking individuals en masse [16-18]. The SSIDs that make up the fingerprint can indicate the social and economic status of the device owner and timeline of device presence can be used to infer individual daily routines [19]. For instance, consider a fingerprint that contains SSIDs indicating the names of member-only airport lounges or known high-class establishments such as luxury hotels or exclusive clubs.

The locational dimension is introduced by utilizing a GPS-capable sensor network or SSID geolocation techniques [19, 20]. In the latter context, Wigle.net – a freely accessible war-driving database with over 195 million recorded APs can be queried to obtain physical AP locations based on the SSID [6, 7, 16-18, 21, 22]. This process can be used to infer previous device locations with some degree of uniqueness, especially if the SSID being queried represents a single individual AP.

Highly distinct spatiotemporal observations can reveal individual mobility patterns using as little as four points [23]. A number of open and closed-source tools leveraging the described concepts have emerged from the security industry and the hacker community [19, 24-28]. Some of these tools also incorporate additional profiling capabilities based on active fingerprinting techniques such as traffic interception via rogue AP provisioning [19, 24]. However, given the scope of this paper and the associated legal issues these techniques are not being considered. Finally, commercial entities offer visitor and location analytics solutions based on the described passive fingerprinting techniques [29, 30].

2.3 Forensic implications and significance

The captured fingerprints have the potential to be utilized in both ante and post-mortem investigation areas [2, 21]. However, existing tools may not be forensically sound due to differences in the original intent, design or non-compliant architecture. An inspection of some of the existing open-source tools reveals utilization of various software components and libraries to facilitate the collection of wireless signals. For instance, the original proof-of-concept *Snoopy* tool [19] was based on the native command-line interface to *Wireshark*, which is also positioned as a potential forensically applicable instrument with built-in capacity to discard unreliable data frames [31]. The successor of *Snoopy* – *snoopy-ng* – follows a different implementation model taking advantage of the Python packet crafting and dissection library called *scapy*, which at the time of writing does not offer native support for Frame Control Sequence (FCS) verification check [32]. While a frame that does not pass the FCS check could still carry a seemingly valid device MAC address, its forensic value would be rather dubious.

From a legal standpoint, the matter of wireless network signal interception under different legislative frameworks presents additional challenges that need to be considered in a qualified tool [4]. Specifically in [4], the authors describe a scenario where two applicable legislations operate on the basis of two different definitions of a telecommunications network. The subsequent legal issues stem from the fact that under one of these legislations, a telecommunications network comprised purely of wireless components may not necessarily be in scope for capture and storage of network data requiring a warrant. Additionally, the term "data" itself may not be inclusive of any metadata - which are represented by frame headers in 802.11 networks ands may be examined by neighboring devices by the virtue of protocol specification. While this viewpoint requires appropriate legal backing, it does highlight some of the associated challenges.

Through the course of this study, we aim to assess the practicality of the relevant techniques for forensic purposes and establish their place in the digital investigator's toolbox.

3 Our study

3.1 Conceptual framework

The conceptual framework applied to the study is presented in Figure 2. While MAC address may uniquely identify the device, it does not enable the linkage back to its owner. Respecting the associated legislative challenges and privacy issues, the study is concerned with investigating the associated requirements, methods and tools to facilitate this linkage from a forensic standpoint. From a broader perspective, the concepts being explored are anticipated to present implications to the wider surveillance domain areas

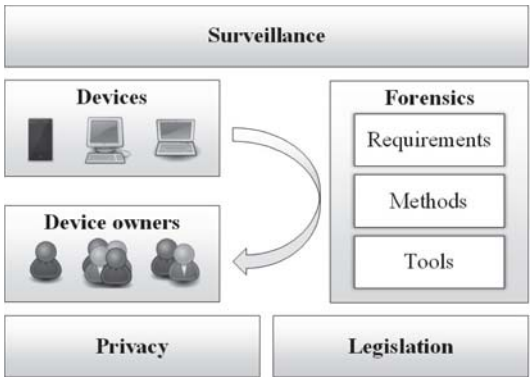


Figure 2. Conceptual framework

such as intelligence.

3.2 Materials and methods

This is an exploratory study that is anticipated to follow a natural progression from exploratory research questions to specific base-rate, relationship or causality inquiries. Initial laboratory experiments followed by field experiments will be utilized to generate the dataset. We will also seek specialized datasets from the Wigle.net administrators to support specific lines of inquiry. In the context of data collection, an isolated test bed such as a Faraday cage will be used to assess and calibrate the selected collection instrument before wider deployment in the field. The architecture of the system to be used during the collection phase is presented in Figure 3. We will utilize a customized version of *snoopy-ng*, to allow us to gauge the volume of both reliable and corrupt signals being collected as well as to facilitate rapid deployment.

In line with the first hypothesis, we will utilize low-cost consumer grade hardware to facilitate the data collection. Specifically, a variety of sensors representing a wide platform mix from single-board computers (Raspberry Pi B / B+ / 2) to commodity devices (Apple Macbook Pro, Google NEXUS 7) will be commissioned for deployment across a number of distinct locations with varying human traffic density. The study will employ both stationary and mobile data collection to enable platform and location-based analyses. The initial collection period is planned for three months. While contextualization to a single metropolitan area is perceived as a potential limitation, it is accepted as a known constraint given the exploratory nature of the study.

In line with the other hypotheses, the collected data will be analyzed in multiple ways. Firstly, the uniqueness of the associated fingerprints will be assessed against published findings to determine any significant deviations and suggest potential causes, where applicable.

Secondly, we will explore quantifiable locatability measures based on the Wigle.net AP location data available for the collected device fingerprints. This exploration will include correlation analysis driven by the PNL size and the corresponding number of locations for a given SSID. Subsequently, the findings are anticipated to indicate the overall practicality of previous or common device location inference using this technique. The associated shortcomings and improvement recommendations are expected to follow.

Finally, we will explore ways of extracting additional information from the collected fingerprints. While SSID string analysis has been visited previously, the process is commonly manual and is unlikely to scale for a large data set [5, 14]. Therefore, we will investigate automated information extraction techniques to support the third hypothesis.

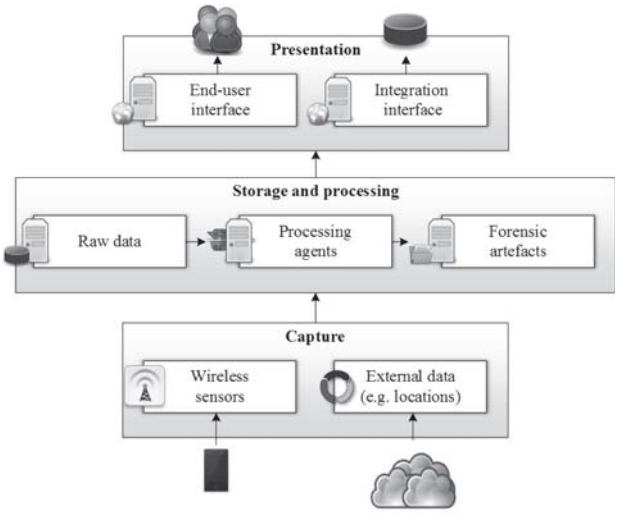


Figure 3. Data collection system architecture

4 Conclusion and ongoing research

While 802.11 wireless signals may carry forensic significance, there is a need for an accepted forensically sound method of establishing the linkage between the device and its owner using the available digital fingerprints. In this paper, we presented a study into 802.11-based tracking and surveillance techniques, which have noteworthy forensic potential and implications. Using the outlined data collection and analysis activities, we aim to examine the usefulness of the relevant techniques for forensic purposes. We also anticipate future scope expansion resulting from the incorporation of additional protocols into subsequent work.

5 Acknowledgements

We would like to thank Robert Hagemann and the entire team behind the Wigle.net service for allowing us extended access to their database and agreeing to provide a specialized data export.

6 References

[1] M. S. Gast, *802.11 Wireless Networks: The Definitive Guide, Second Edition*: O'Reilly Media, Inc., 2005.

[2] B. Turnbull and J. Slay, "Wi-Fi network signals as a source of digital evidence: Wireless network forensics," in *Availability, Reliability and Security, 2008. ARES 08. Third International Conference on*, 2008, pp. 1355-1360.

[3] Cisco, "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2012–2017," ed: Cisco, 2013.

[4] B. Turnbull, G. Osborne, and M. Simon, "Legal and Technical Implications of Collecting Wireless Data as an Evidence Source," in *Forensics in Telecommunications, Information and Multimedia*. vol. 8, M. Sorell, Ed., ed: Springer Berlin Heidelberg, 2009, pp. 36-41.

[5] I. Rose and M. Welsh, "Mapping the urban wireless landscape with Argos," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, 2010, pp. 323-336.
[6] B. Bonné, P. Quax, and W. Lamotte, "Your Mobile Phone is a Traitor!--Raising Awareness on Ubiquitous Privacy Issues with SASQUATCH," 2014.
[7] M. Cunche, "I know your MAC Address: Targeted tracking of individual using Wi-Fi," *Journal of Computer Virology and Hacking Techniques*, vol. 10, pp. 219-227, 2014.
[8] O. Ureten and N. Serinken, "Wireless security through RF fingerprinting," *Electrical and Computer Engineering, Canadian Journal of*, vol. 32, pp. 27-33, 2007.
[9] V. Brik, S. Banerjee, M. Gruteser, and S. Oh, "Wireless device identification with radiometric signatures," in *Proceedings of the 14th ACM international conference on Mobile computing and networking*, 2008, pp. 116-127.
[10] A. C. Polak, S. Dolatshahi, and D. L. Goeckel, "Identifying wireless users via transmitter imperfections," *Selected Areas in Communications, IEEE Journal on*, vol. 29, pp. 1469-1479, 2011.
[11] L. C. C. Desmond, C. C. Yuan, T. C. Pheng, and R. S. Lee, "Identifying unique devices through wireless fingerprinting," in *Proceedings of the first ACM conference on Wireless network security*, 2008, pp. 46-55.
[12] J. Franklin, D. McCoy, P. Tabriz, V. Neagoe, J. V. Randwyk, and D. Sicker, "Passive data link layer 802.11 wireless device driver fingerprinting," in *Proc. 15th USENIX Security Symposium*, 2006, pp. 167-178.
[13] C. Neumann, O. Heen, and S. Onno, "An empirical study of passive 802.11 device fingerprinting," in *Distributed Computing Systems Workshops (ICDCSW), 2012 32nd International Conference on*, 2012, pp. 593-602.
[14] J. Pang, B. Greenstein, R. Gummadi, S. Seshan, and D. Wetherall, "802.11 user fingerprinting," in *Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, 2007, pp. 99-110.
[15] X. Hu, L. Song, D. Van Bruggen, and A. Striegel, "Is There WiFi Yet? How Aggressive WiFi Probe Requests Deteriorate Energy and Throughput," *arXiv preprint arXiv:1502.01222*, 2015.
[16] M. Cunche, M. A. Kaafar, and R. Boreli, "I know who you will meet this evening! Linking wireless devices using Wi-Fi probe requests," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2012 IEEE International Symposium on a*, 2012, pp. 1-9.
[17] M. V. Barbera, A. Epasto, A. Mei, V. C. Perta, and J. Stefa, "Signals from the Crowd: Uncovering Social Relationships through Smartphone Probes," Sapienza University, Rome, Italy2013.
[18] B. Bonné, A. Barzan, P. Quax, and W. Lamotte, "WiFiPi: Involuntary tracking of visitors at mass events," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2013 IEEE 14th International Symposium and Workshops on a*, 2013, pp. 1-6.
[19] G. Wilkinson. (2012). *Snoopy: A distributed tracking and profiling framework*. Available: http://www.sensepost.com/blog/7557.html
[20] T. O'Connor, *Violent Python: A Cookbook for Hackers, Forensic Analysts, Penetration Testers and Security Engineers*: Newnes, 2012.
[21] W. Minnaard, "Out of sight, but not out of mind: Traces of nearby devices' wireless transmissions in volatile memory," *Digital Investigation*, vol. 11, Supplement 1, pp. S104-S111, 5// 2014.
[22] X. Fu, N. Zhang, A. Pingley, W. Yu, J. Wang, and W. Zhao, "The digital marauder's map: A wifi forensic positioning tool," *Mobile Computing, IEEE Transactions on*, vol. 11, pp. 377-389, 2012.
[23] Y.-A. de Montjoye, C. A. Hidalgo, M. Verleysen, and V. D. Blondel, "Unique in the Crowd: The privacy bounds of human mobility," *Sci. Rep.*, vol. 3, 03/25/online 2013.
[24] G. Wilkinson. (2014). *Release the hounds! Snoopy 2.0*. Available: http://www.sensepost.com/blog/10754.html
[25] B. O'Connor, "CreepyDOL," presented at the BlackHat USA 2013, Las Vegas, 2013.
[26] M. Wuergler. (2012, September 10). *STALKER - Analyzing [Your] Wireless Data*. Available: http://immunityproducts.blogspot.fr/2012/08/stalker-analyzing-your-wireless-data.html
[27] H. Seiwert. (2012, September 10). *iSniff GPS: Passive sniffing tool for capturing and visualising WiFi location data disclosed by iOS devices.* Available: https://github.com/hubert3/iSniff-GPS
[28] A. Maxwell. (2013). *Project Watcher - Maltego with a twist of wireless*. Available: https://github.com/catalyst256/Watcher
[29] Cisco, "CMX Analytics," ed, 2015.
[30] Euclid. (2015). *How Euclid location analytics works.* Available: http://euclidanalytics.com/products/technology/
[31] S. Raghavan and S. Raghavan, "A study of forensic & analysis tools," in *Systematic Approaches to Digital Forensic Engineering (SADFE), 2013 Eighth International Workshop on*, 2013, pp. 1-5.
[32] SensePost. (2014). *[sensepost/snoopy-ng] Snoopy v2.0 - modular digital terrestrial tracking framework*. Available: https://github.com/sensepost/snoopy-ng

SESSION

LATE BREAKING PAPERS: SECURITY AND MANAGEMENT AND CYBER SECURITY EDUCATION

Chair(s)

TBA

The Benefits of Hosting the NECCDC at Your Institution

G. Markowsky[1], P. Lutz[2], D. Johnson[2]
W. Stackpole[2], R. Soucy[1], and B. Attaie[3]
[1]School of Computing and Information Science, University of Maine, Orono, Maine, USA
[2]Department of Computing Security, Rochester Institute of Technology, Rochester, NY, USA
[3]School of Information Studies,Syracuse University, Syracuse, NY, USA

Abstract— *The Northeast Collegiate Cyber Defense Competition (NECCDC) [2] is a regional competition that feeds the National Collegiate Cyber Defense Competition (CCDC) [1]. Since RIT organized the first NECCDC in 2008, the NECCDC has selected a representative to compete in the CCDC. It has been relatively successful and has produced the national champion twice and the runner up three times during its eight years of existence. The NECCDC has been hosted on a rotating basis by one of the universities in the northeast and has become a popular event for both the hosting schools and for the students. We feel that the NECCDC has continued to be an exciting event in part because it has been hosted by different universities which have all made important contributions to the event. This paper describes some of the benefits that come from hosting. Our hope is to convince other universities to host the NECCDC and similar competitions.*

Keywords: CCDC, NECCDC, cybersecurity competitions

1. Introduction

Cyber competitions have been found to inspire the students and help faculty put together better courses. One well-established competition is the National Collegiate Cyber Defense Competition, also known as the National CCDC, NCCDC or simply the CCDC [1]. The following material comes from [1]:

> *The mission of the Collegiate Cyber Defense Competition (CCDC) system is to provide institutions with an information assurance or computer security curriculum a controlled, competitive environment to assess their students' depth of understanding and operational competency in managing the challenges inherent in protecting a corporate network infrastructure and business information systems.*

1.1 History of the CCDC

The following material comes from [1]:

> *On February 27 and 28, 2004, a group of educators, students, government and industry representatives gathered in San Antonio, Texas, to discuss the feasibility and desirability of establishing regular cyber security exercises with a uniform structure for post-secondary level students. During their discussions this group suggested the goals of creating a uniform structure for cyber security exercises might include the following:*
>
> 1) *Providing a template from which any educational institution can build a cyber security exercise*
> 2) *Providing enough structure to allow for competition among schools, regardless of size or resources*
> 3) *Motivating more educational institutions to offer students an opportunity to gain practical experience in information assurance*

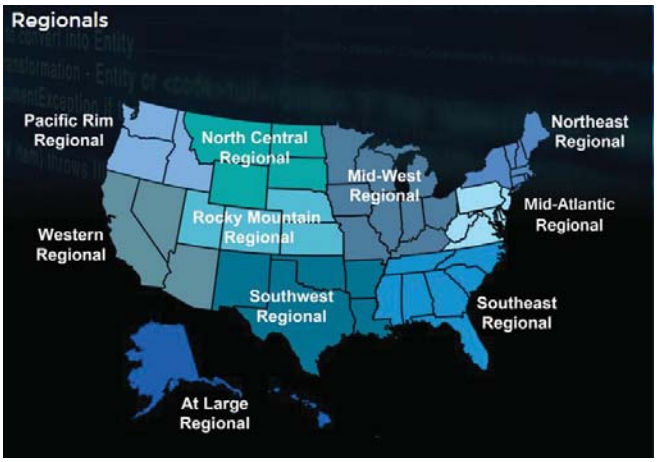


Fig. 1: The Regional Competitions that Feed the CCDC

1.2 Structure of the CCDC

The CCDC is fed by 10 regional competitions. Figure 1 shows these 10 regions. Regional competitions generally take place at a single physical location, but the At Large region is a virtual competition because of the great distances between competitors.

The CCDC is organized as follows. Teams of students from participating schools are called blue teams. Each blue team is treated as a replacement IT department for a company whose IT department was fired for incompetence. Thus,

each blue team inherits a system that has been compromised and will continue to be under constant attack during the competition.

The competition schedule generally looks like the following:

- Friday noon to 7 PM
- Saturday 8AM to 7 PM
- Saturday 7 PM Mixer and Recruiting Evening
- Sunday 8AM to noon
- Awards luncheon and keynote speaker Sunday afternoon

The competition staff is divided into three teams:

- *Red Team*: they provide all the attacks and compromises.
- *Black Team*: they design, assemble, operate and monitor the competition network
- *White Team*: they act as management and design the competitions "injects" (tasks), they judge the competition, and monitors blue teams directly to ensure compliance with the rules

Injects are tasks for the blue teams. For example, a blue team might be asked to scan the network for vulnerabilities, configure new machines, or rebuild a system after a server crash. Blue teams are judged on task performance, reporting and sometimes on the quality of oral presentations.

2. The NECCDC

The Northeast Collegiate Cyber Defense Competition (`http://neccdc.net`) is one of the feeder competitions for the CCDC. It is a very interesting and challenging 3 day competition that follows the format of the CCDC. The northeast region includes Connecticut, Maine, Massachusetts, New Hampshire, New York, Rhode Island and Vermont. Some New Jersey schools have also participated on occasion because of their proximity to New York City.

The NECCDC follows the CCDC very closely. The goal is to help the NECCDC winner to prepare for the CCDC so it follows all the CCDC rules as closely as possible. In particular, the NECCDC schedule is the same as the CCDC. Also, the terminology: Red Team, Blue Team, White Team and Black Team are the same. During the 2013 NECCDC, UMaine made a video of the competition [3] that gives a sense of the competition. The paper by Scaparra [4] gives a feel for the CCDC type of competition. For insight into how the CCDC competitions are scored and ideas on how the scoring can be improved see Markowsky [5].

Over the years, the NECCDC and similar competitions have generated papers on a wide variety of topics. These include such items as red team preparation (Johnson [6], Scaparra and Bullock [7]), blue team preparation (Engebretson, Pauli, and Bosma [8], Gourd [9], Pauli and Engebretson [10], Capalbo, Reed and Arpaia [11], Cavanaugh and Albert [12], [13], Cheung, Cohen, Lo and Elia [14], Glumich and Kropa [15], Casper and Papa [16], Mauer, Stackpole and Johnson [17]), resources for cybersecurity education (L. Markowsky [18]), and even how to run high school and middle school cyber competitions (Albert, Markowsky, Wallingford [19]).

The initial interest in hosting the Northeast regional CCDC started at least as early as 2006, when a group of RIT faculty visited the National CCDC event as observers. This was motivated by the chair of RIT's NSSA (Networking, Security and System Administration) Department, Luther Troell. The goal of the first visit was to better understand the components of the event as well as the commitment required to host the event. A second visit in 2007 was performed in order to obtain a commitment from the CCDC organizers for the first northeast regional competition to be hosted at RIT in Rochester, NY.

RIT's interests in becoming involved in the NECCDC focused mainly on having a venue within which students could compete and display their talents in the field of cyberdefense. As the field of cybersecurity was fairly new at the time, another benefit included promoting the development of cybersecurity programs in the northeast region. After hosting events in 2008 and 2009 it appears that these goals were met. Hosting the NECCDC, especially in the early years, when competitors and sponsors were hard to come by, was a huge effort for which there are no regrets. The effort expended to start this movement appears to have staying power, and has been evidenced by the performance of the teams in the Northeast region.

3. NECCDC I (2008)

There were many lessons learned at the first NECCDC that have influenced the delivery and execution of subsequent NECCDCs. Because of the importance of these lessons we will describe the first NECCDC in more detail than subsequent NECCDCs.

This first competition took place in the early spring of 2008 (February 29 - March 2) which corresponded with spring break at RIT that year. It was sponsored by McAfee, Harris RF Communications, and Cisco in addition to RIT's NSSA (Networking, Security and System Administration) Department. Competitors that year were Champlain College, Northeastern University, Norwich University, NYU Polytechnic, Rochester Institute of Technology, and Syracuse University. RIT was the winner the first year.

The RIT organizer was Peter Lutz, the Black Team leader was Bruce Hartpence, the Red Team leader was Daryl Johnson, the White Team was led by Larry Hill, and the RIT blue team coach role was shared by Sharon Mason and Bill Stackpole. All of these people were faculty in the NSSA Department.

In this first year of the competition, a major problem was simply recruiting competitor institutions. Competing turned out to be a major undertaking for each institution,

and while sponsors helped to cover the costs of running the competition, there was no financial support available to provide competitor institutions any relief. RIT contacted someone at each of the NSA CAEIAE's in the region, as well as announced it at the yearly colloquium. After the competition RIT worked on having a better plan for reaching potential competitors for future NECCDCs. After review, a plan to reach more institutions was devised, and the following year a more organized and robust approach was followed. In particular, mining the web for institutions with coursework in information assurance (IA) was a fruitful way to discover potential participants outside of the CAEIAE community.

Another major problem was in creating an ongoing event without the impression that RIT had an unfair advantage. During the first event, a meeting of all coaches was held on Saturday to discuss this issue. All attending expressed an interest in continuing this competition into the future. However, the other institutions indicated that the resources required would be beyond them. As a result, RIT chose to host it a second time.

In all NECCDCs, team coaches form an oversight panel that is privy to the operations of the White Team and all judging. As questions arise, the White Team captain suggests how particular problems should be resolved and the panel discusses the problems and suggested resolutions. After the discussion, the panel votes (ties to be broken by the White Team Captain) and the vote is binding. The team coaches also decide policy for the competition and decide where future NECCDCs will be held. One policy decision that has been made is to automatically qualify the hosting school for the NECCDC regardless of how they perform in the qualifying round.

3.1 White Team Notes

The White Team experienced myriad issues, questions, decisions during the competition. In particular, the White Team members stationed in blue team rooms tended to make judgments of their own, based on their understanding of the competition rules. The result was that, during the first day, inconsistent rulings were given to the teams. New procedures were adopted on the second day of the competition requiring that White Team members in the blue team rooms refer all questions to the White Team captain, who served as the final judge. This has been the standard operating procedure in all subsequent NECCDCs. It was also discovered that having group White Team meetings and standard forms for injects and incident reports were very helpful to keep the competition moving quickly.

Unlike the NCCDC, the NECCDC allowed and still allows teams to bring alternates to the competition. This is because between the regionals and the nationals the final team composition might need to be altered because of illness, personal emergency, etc. and we did not wish to hamper the NECCDC winner before the NCCDC even began.

At the 2008 NECCDC, blue teams were provided disaster recovery DVDs for each system. These were bootable DVDs with disk images that would restore a system to its pre-competition state. These were in addition to the distribution CDs/DVDs for each OS. The experience at the 2008 NECCDC suggests that these DVDs should be kept by the Black Team who would be the only ones authorized to use them so that any recovery activities would result in a blue team penalty. The blue teams could, of course, create their own disaster recover CDs/DVDs, if they so desired.

During the competition, the blue teams asked frequently about retasking workstations to specific purposes. In general, this was allowed, but in future NECCDC limits were imposed on how many machines could be retasked in this way. At the outset, blue teams had 4 servers and 4 workstations. An inject required the retasking of one workstation. It would be a mistake to lose all workstations in this manner, so it is recommended that at least two workstations be maintained throughout the competition.

The machines at the 2008 NECCDC had three NICs in them, but there were no rules determining whether and how additional NICs could be used. Some blue teams wanted to use the extra NICs to turn servers into a routers/firewalls, while the organizers wanted to force the blue teams to use the PIX firewalls provided. As much as possible we recommend that rules be in place to deal with any extra hardware that might make its way to the blue teams.

One contentious issue that came up was whether reconnaissance is considered an attack. The Red Team did reconnaissance at periods when they were prevented from attacking, and some blue teams questioned this practice. As the NECCDC has evolved we have adopted the position that the NECCDC is not a contest between blue teams and the Red Team - it is an event that tests the abilities of blue teams and to get a good measure of these abilities, it is important to push the blue teams as much as possible during the competition. To make the trial as challenging as possible it is desirable for the White, Black and Red teams to work together.

It was also discovered during the competition that it is a good idea to place staplers and paper clips in each blue team room. The paper clips are ideal for resetting routers and switches.

3.2 Black Team Notes

The Black Team benefited greatly from its detailed knowledge of the the scoring engine. It became clear during the competition that the White Team would benefit greatly from a better understanding of how SLAs and the injects were scored. In subsequent NECCDCs effort has been devoted to improving pre-competition communication between these two teams to minimize doubts about scoring.

An open question is how much detail about the scoring should be communicated to the blue teams. For example, simply telling them that having services down would cost them points, and more points the longer they were down, is one approach. Another is to tell them that they receive points for services being up every 5 minutes of play. If services go down, they stop receiving these points. If services are down for long enough (an hour or more) points are deducted. The goal is to make the scoring system essentially invisible to the blue teams so they can focus on cyber defense.

In this NECCDC we captured only packet headers. In subsequent NECCDCs full packet captures were collected.

3.3 Red Team Notes

The Red Team members were from seven different organizations and for the most part were not familiar with each other before the event. A Google group was created for the red team to get acquainted ahead of time. They used the forum to begin to discuss what tools they were bringing, what strengths they had, what similar experiences they had and to discuss strategies. The Red Team expressed that this interaction was helpful and recommended it for future red teams.

The Red Team members brought their own hardware and software for the competition. We provided a single Linux file server with 750 GB of storage that they could use as needed. They ended up using it for three purposes. First, it was connected to the projection system and projected the overall blue team summary of service availability. Second, it was used to type up the exploit reporting forms and print them to be given to the white team. Third, they installed a wiki that the red team members used to share exploit and strategic information about the blue teams' and the red team's activities. Occasionally, they also used it to display a document or information that they wanted to display to the group at large.

This NECCDC forced the competition organizers to deal with social engineering practiced by the Red Team. The competition staff have extraordinary access to team rooms, and the blue teams are not empowered to prevent this access. One of the staff roamed among all of the team rooms taking pictures of the event and the participants. One of the Red Team members asked the staff member for copies of the pictures and was given a download. The pictures included whiteboard and monitor shots that revealed network and account information. The White Team judge disallowed the attack and the Red Team did not use the information gained. It was decided that social engineering the competition staff gives the Red Team an unrealistic advantage. It is suggested that the rules more explicitly prohibit this kind of attack.

The Red Team made a major phishing attack at the outset of the competition with great success. They were able to get several blue teams to execute a Trojan that installed remote control software on one of their systems. The Red Team then used that access to shut down services. However, this raised the level of awareness among the blue teams who quickly shut down Red Team access. Several blue teams failed to protect their Cisco hardware from remote access. The Red Team was able to take full control of several routers and switches. They proceeded to lock the blue teams out of their own equipment. This tipped off the blue teams and they took corrective action.

3.4 Conclusion

The 2008 NECCDC was conducted much like the nationals, but with some local tweaks. In the future, the tweaks would be greater and lessons learned would be applied. Overall, it was a good experience, and RIT hosted again in 2009.

4. NECCDC II (2009)

The second competition took place in the spring of 2009 (February 27 - March 1). It was sponsored by Harris RF Communications and M&T Bank. The competitors that year were the University of Buffalo, Champlain College, the University of Maine, Northeastern University and Rochester Institute of Technology. The winner was Northeastern University. Many of the lessons learned from the 2008 NECCDC were applied to the 2009 NECCDC which ran quite smoothly. Communication with the blue teams was much improved, but there was still the problem of having teams show up for the competition. At least one team said that they were coming, which caused another team to be turned down, but then the original team backed out of the competition at the last minute. At the competition, the University of Maine indicated a willingness to host the 2010 NECCDC and plans were made for bringing the NECCDC to the University of Maine. Pete Lutz again was director of the competition and White Team Captain, Daryl Johnson was the Captain and organizer of the Red Team and Bo Yuan was the Captain of the Black Team. All were members of the RIT faculty.

5. NECCDC III (2010)

The 2010 was held March 5th through 7th at the University of Maine's flagship campus in Orono. There were a total of 9 schools represented: Alfred State College; Champlain College; Harvard University; Northeastern University; Polytechnic Institute of NYU; Rochester Institute of Technology; Stevens Institute of Technology; SUNY Oswego; and The University of Maine. Northeastern University was the 1st place winner, and would continue to win 1st place at the National CCDC for 2010. The University of Maine placed 2nd, and Rochester Institute of Technology 3rd.

The Director of the NECCDC was George Markowsky, Professor of Computer Science of the University of Maine. The Captain of the White Team was former RIT competitor Thomas Vachon, the Red Team captain was Daryl Johnson

of RIT, and the Black Team captains were Ray Soucy and Andrew Moody from the University of Maine.

2010 marked major changes in the NECCDC to raise the profile of the regional, including increased focus on clear rules for competitors and judges, better communication for attendants, and aggressive pursuit of sponsorship from companies like Trustwave, Boeing, Black Hat, Game Logic, and Fairpoint, as well as public sector support from the Department of Homeland Security.

In 2009, the NECCDC suffered from having no-shows. For the 2010 NECCDC it was decided to institute a $750 fee for all schools wishing to participate. The stipulation was that any school that attended the NECCDC would receive a $750 travel assistance grant. Any school not showing up would not receive a travel grant. This mechanism prevented no-shows and has been used since 2010.

Overall, the majority of feedback for the 2010 NECCDC was the lack of information and activities for non-competitors. The request to see team standings or points in real time was very popular, as well as the request to have a non-scored team setup for coaches to gain hands-on experience with the event and better insight into what their teams are exposed to. There was also feedback requesting more information ahead of the competition on how to prepare, particularly for new competitors.

Because of the enlarged scope of the 2010 NECCDC, fund raising became a big concern. Fortunately, we were able to get a $10,000 grant from the Department of Homeland Security. Douglas Maughan, of the Department of Homeland Security attended the 2010 NECCDC and as a result DHS has been funding all the regionals at the rate of $15,000 per event.

Northeastern University won the 2010 NECCDC and went on to win the CCDC. One blue team was disqualified from the competition because of its behavior and using resources of other blue teams. This is the only time in the history of the NECCDC that a team was disqualified during the competition.

6. NECCDC IV (2011)

For 2011, the NECCDC was hosted by Northeastern University at an EMC training facility to accommodate the growing number of competitors.

A total of 11 teams participated: Alfred State College; Champlain College; Harvard University; Northeastern University; Pace University (NY); Polytechnic Institute of NYU; Rochester Institute of Technology; Stevens Institute of Technology; University of Maine; University of Massachusetts Boston; and University of New Hampshire. The winner for 2011 was RIT, with 2nd place going to Stevens Institute of Technology, and 3rd to Champlain College.

The Head Judge was Thomas Vachon, white team co-captain Ray Soucy, Red Team captain Daryl Johnson, and Black Team Captain David LaPorte of Northeastern University,

The majority of the feedback for 2011 centered around the physical security restrictions of the facility, lack of wireless access for non-competitors, and vendor presentations being the wrong choice for the spirit of the event.

7. NECCDC V (2012)

NECCDC V was hosted by Northeastern University at the EMC training facility, with 12 teams representing: Alfred State College; University of Buffalo; Champlain College; Harvard University; The University of Maine; University of Massachusetts Boston; University of New Hampshire; Northeastern University; Pace University; Rochester Institute of Technology; Stevens Institute of Technology; and Worcester Polytechnic Institute. The winner for 2012 was RIT, with 2nd place going to UNH, and 3rd to The University of Maine.

The Head Judge was Marc McLaughlin, and Black Team captain Chris Mills, both from RSA. The Red Team captain was Daryl Johnson.

Despite 2011 feedback on the restrictions of the EMC training facility being the wrong fit, there were no alternatives to accommodate 12 teams. There was also a growing concern expressed that the academic focus of the event was being lost.

8. NECCDC VI (2013)

For 2013 there was an effort to address the concerns of the 2011 and 2012 competition by hosting the event once again at the University of Maine. To make this possible, a virtual qualifier was held for the first time to narrow the competition from 14 interested schools to the top 10.

The 10 teams represented: Alfred State College; Champlain College; Northeastern University; Rochester Institute of Technology; SUNY BUffalo; SUNY IT; Syracuse University; University of Maine; University of New Hampshire; and Worcester Polytechnic Institute.

The winner for 2013 was RIT, with SUNY IT placing 2nd, and both Worchester Polytechnic Institute and Northeastern University placing 3rd.

The Head Judge was Ray Soucy, with Red Team captain Daryl Johnson, and Black Team captain Andrew Moody. For more information about the 2013 NECCDC see [20], [21], [22].

9. NECCDC VII (2014)

The 2014 NECCDC was held at the University of New Hampshire, which was hosting it for the first time. The Director of the competition was Kenneth Graf. Ken Graf developed a good relationship with industry sponsors and is interested in hosting the 2018 NECCDC again at UNH.

10. NECCDC VIII (2015)

Syracuse University (SU) began its participation in NECCDC back in 2008 when a team of graduate students from the School of Engineering traveled to RIT for the inaugural NECCDC. Unfortunately no one had checked the rules which only permitted a maximum of two graduate students per team and as a result the team was not able to compete. It wasnâĂŹt until 2013 when SU was able to field another team this time consisting of a small group of graduate and undergraduate students from the School of Information Studies (iSchool). The team was able to make it through the virtual qualifier and traveled to the University of MaineâĂŹs Orono campus (UMaine) for the 2013 NECCDC. The students had an exceedingly positive experience and when they learned that NECCDCC was looking for other schools as future hosts of the event they urged their coach to investigate it. After careful deliberations it was decided to shadow the 2014âĂŹs host in order to gain more experience so that a determination could be made for 2015.

The iSchool was able to form a full team for the 2014 NECCDC as a result of increased interest generated from the previous yearâĂŹs engagement and the team advanced through the qualifier to the 2014 regional at the University of New Hampshire (UNH) in Durham. Shadowing UNH in 2014 was very helpful as it facilitated a behind the scenes access to the event and seeing how it was organized was a key factor in the iSchoolâĂŹs decision to adventure into hosting the event for 2015.

Several other factors were equally instrumental in finalizing the decision to host the 2015 NECCDC. The sponsorship made available by National CCDC through a grant from Homeland Security and other CCDC sponsors was critical to establishing a sound financial foundation which helped to secure commitments from the DeanâĂŹs office. Equally critical was the never ending encouragement, support and advice from previous hosts UMaine, RIT and UNH making it a family affair.

The list of benefits that can be attributed to hosting the NECCDC is lengthy and includes both long term and short term benefits. Some of the short term gains were enhanced student interest and pride and a nice set of good publicity. Long term gains included the formation of an Information Security Student Club and proposal to add an Ethical Hacking Course to the curriculum as well as research opportunities in cybersecurity and other areas such as the effectiveness of NECCDC as a recruitment tool for cybersecurity talents versus traditional forms of recruitment.

11. NECCDC IX (2016)

At the 2015 NECCDC a new hosting university was selected for the 2016 NECCDC. In July 2015 the university selected indicated that because of budget cuts and staff changes it would no longer be able to host the 2016 NECCDC. George Markowsky indicated that the University of Maine would be willing to host the 2016 NECCDC. After some discussion among the various schools, it was decided to accept the University of Maine's offer. The University of Maine hopes to build on all the successes that the NECCDC has had to this point and is looking to an exciting NECCDC.

12. NECCDC X (2017)

RIT indicated a strong interest in hosting NECCDC X in light of its role in starting the NECCDC. All the schools involved in the NECCDC agreed that the honor of hosting NECCDC X should go to RIT in recognition of their pioneering work in establishing the NECCDC.

13. The NECCDC Red Team

It is not difficult to get volunteers for the Red Team. It is much more difficult to get good Red Team members. By this we mean finding people who see the goal and purpose of the Red Team as "assessing the skills and talents of the blue teams and determining the best team to move on to the NCCDC". Often Red Team members want to see the opportunity to play on the Red Team as a test of their attacking skills. While serving on the Red Team is challenging, Red Team members need to be committed to being fair and launch the exploits against all the teams to provide a thorough test of the blue teams. Slogging through the repetitions requires dedication to the ideal of the Red Team as a challenger of blue teams.

Over the years, one of the greatest changes that has come about is the realization by the NECCDC White Team that the Red Team should not be considered "the bad guys" who need to be keep in the dark, but rather as an integral part and member of the competition. This paradigm shift has allowed the competition to grow and expand its areas of coverage in assessing the blue teams' strengths and weaknesses. The Red Team is now participates in the design, operation, and grading of the event. This change strengthens the competition by making the best utilization of all of the talents available.

Fairness and equity in the attention that each blue team receives from the Red Team is critical to making sure that the winning blue team is indeed the best. Our primary tenant is "no Red Team success can be scored unless it has been tried on every other blue team first." This organizes the Red Team by discipline or skill. A common alternative in cyber competitions is to assign Red Team members to a specific blue team. This is likely to create situations where a very good Red Team member is pitted against a very weak blue team and an exaggerated score is achieved compared to the rest of the blue teams. Potentially, worse is if a weaker Red Team member is pitted against a weaker blue team. This might lead the weak blue team to get a very good score and win the competition, which could cause the NECCDC

to supply a weak team to the NCCDC. Clearly, making sure that every exploit is tried against every team permits us to have some confidence in the results.

14. Conclusions

The benefits to the students of the host institution are that the attention and stature that their institution garners reflects on them and their degree. The benefits for the blue team are obvious but now add the esteem of the host. For those students not on the Blue team there can be opportunities for exercising their skills and learning new ones is aiding and assisting the host in making preparations. This can come if the form of volunteering for the various teams supporting the event. Getting ready involves every aspect of security from building networks, systems, and services, to solving the associated problems with making 10 of duplicate copies, to scoring and monitoring the event, and on to creating and enforcing the rules. All of these pieces can provide growing and enriching opportunities for faculty, staff, and students.

The benefits to an institution for hosting an event such as NECCDC are many. The first impact realized would be the commitment of your administration to your security program through their support, but also the commitment of your faculty and staff to the effort. An event such as this can often galvanize and motivate your faculty to push their limits and bring them together as a team. A common goal is a powerful motivator. The exposure of your programs and curriculum outside of your institution can help expand your visibility and recognition in the field. A fair amount of media coverage can be garnered because of the event. The attention acquired through media and sponsorship efforts can help realize long term relationships with vendors and employers that pay off over the long haul. The preparations for the event can provide a challenge for the host. It is through those challenges that the host institution can grow both their capability and capacity but also find out what hidden talent and capabilities they already have. A host might discover resources, skills, or facilities dormant or hidden at home that they were not aware of.

It can be said of the host institutions over the years, that every one of them has benefited from the experience. They have struggled with various aspects but always persevere and come out of it stronger and more confident of the work they are doing. After the experience, one will often hear comments such as "I wasn't sure we could pull it off but we did!" with pride and a sense that now they are better. It is interesting that another surprise from hosting is the discovery of skills and resources that the host had but did not recognize or fully appreciate. This often opens up new opportunities for the host in terms of curriculum, recognition, support, and associations. For the host it can also spur and encourage the faculty to broaden their course offering by creating new courses supporting the material needed by the blue teams.

References

[1] National Collegiate Cyber Defense Competition Website, http://www.nationalccdc.org/.

[2] Northeast Collegiate Cyber Defense Competition Website, http://neccdc.net/wordpress/.

[3] University of Maine Video of the 2013 NECCDC, https://www.youtube.com/watch?v=V8A7wci81Yo. The video and photos are also available at https://www.flickr.com/photos/geomarkowsky/.

[4] Jeffrey C. Scaparra, "One Individual's Three Perspectives on the Collegiate Cyber Defense Competition," in *Proc. SAM'10*, 2010, pp. 307-313.

[5] George Markowsky, "Toward a More Perfect Scoring System for the NECCDC",*Proc. 2012 Inter- national Conference on Security and Management*, (SAM'12), Las Vegas, NV, July 16-19, 2012, pp. 230-235, http://www.gbv.de/dms/tib-ub-hannover/739342479.pdf.

[6] Daryl Johnson, "The Assembly and Provisioning of a Red Team," in *Proc. SAM'11*, 2011, pp. 530-534.

[7] Jeffrey C. Scaparra and Jeffrey R. Bullock, "Red Teaming for Education," in *Proc. SAM'11*, 2011, pp. 512-517.

[8] Patrick Engebretson, Joshua Pauli, and Joshua Bosma, "Lessons Learned from an Evolving Information Assurance Lab," in *Proc. SAM'10*, 2010, pp. 261-266.

[9] Jean Gourd, "Cyber Storm: The Culmination of an Undergraduate Course in Cyber Security," in *Proc. SAM'10*, 2010, pp. 300-306.

[10] Joshua Pauli and Patrick Engebretson, "A Cradle-to-Grave Approach to Retaining Students in Information Security Programs," in *Proc. SAM'10*, 2010, pp. 255-260.

[11] Nicholas Capalbo, Theodore Reed, and Michael Arpaia, "RTFn - Enabling Cybersecurity Education through a Mobile Capture the Flag Client," in *Proc. SAM'11*, 2011, pp. 500-506.

[12] Cory Cavanagh and Raymond Albert, "Goals, Models, and Progress towards Establishing a Virtual Information Security Lab in Maine," in *Proc. SAM'11*, 2011, pp. 496-499.

[13] Cory Cavanagh and Raymond Albert, "Implementation Progress, Student Perceptions, and Refinement of a Virtual Information Security Laboratory," in *Proc. SAM'12*, 2012, pp. 197-200.

[14] Ronald Cheung, Joseph Cohen, Henry Lo, and Fabio Elia, "Challenge Based Learning in Cybersecurity Education," in *Proc. SAM'11*, 2011, pp. 524-529.

[15] Sonja Glumich and Brian Kropa, "DefEX: Hands-On Cyber Defense Exercises for Undergraduate Students," in *Proc. SAM'11*, 2011, pp. 487-493.

[16] William D. Casper and Stephen M. Papa, "A Multi-Disciplined Security Engineering Education Approach," in *Proc. SAM'12*, 2012, pp. 243-248.

[17] Brandon Mauer, William Stackpole, and Daryl Johnson, "Developing Small Team-based Cyber Security Exercises," in *Proc. SAM'12*, 2012, pp. 213-217.

[18] Linda Markowsky, "An SELinux Sourcebook for Cybersecurity Education," in *Proc. SAM'10*, 2010, pp. 248-254.

[19] Raymond Albert, George Markowsky, and Joanne Wallingford, "High School Cyber Defense Competitions: Lessons from the Trenches," in *Proc. SAM'10*, 2010, pp. 280-285.

[20] George Markowsky, Daryl Johnson, Andrew Moody, Ray Soucy and William Stackpole, "The 2013 NECCDC - Lessons Learned", *Proc. 2013 International Conference on Security and Management*, (SAM'13), Las Vegas, NV, July 22-25, 2013, pp. 433-439, http://umaine.edu/scis/files/2013/07/The-2013-NECCDC-%E2%80%93-Lessons-Learned.pdf.

[21] 2013 NECCD Symposium Schedule. [Online]. Available: http://neccdc.net/wordpress/?page_id=99

[22] 2013 NECCD Media Release. [Online]. Available: http://neccdc.net/wordpress/wp-content/uploads/2013/02/NECCDCPhotoRelease2013.pdf

Mirror Network: A Holistic Approach for Assuring Information Systems

Mahalingam Ramkumar
Department of Computer Science and Engineering
Mississippi State University, Box 9637, Mississippi State, MS 39759
ramkumar@cse.msstate.edu

***Abstract*—A holistic approach presented in this paper for securing any Information System (IS) aims to reduce the complexity of both i) the assurance protocol for any IS, and ii) the platform on which the assurance protocol is executed. The platform for execution of assurance protocols is the mirror network (MN), constrained to be composed solely of *homogeneous, low complexity modules*. While the MN approach can be used to assure *any* IS, for purposes of illustration, we illustrate application of the MN model to secure an important IS, the Domain Name System (DNS).**

I. Introduction

From a broad perspective, assuring the operation of any system is simply a process involving *verification of self-consistency* all critical internal states of the system. For Information Systems (IS), composed of hardware and software components that create, exchange, process and dispose data, the internal states are conveniently available in a format (digital data) that can be easily read and manipulated by software. For such systems, the "assurance protocol" can simply be an additional piece of software that i) verifies consistency of internal states and ii) reports it's findings to entities that desire to be assured of the integrity of the IS.

The confidence in assurances (regarding the IS) offered by the assurance protocol extends only as much as the confidence in the integrity of the assurance software. The latter is affected by a) the complexity of the assurance software, and b) the complexity of the environment in which the assurance software is executed — the higher the complexity of any hardware/software component, the higher the possibility of presence of undesired (malicious or accidental) functionality that can affect it's integrity.

The Mirror Network (MN) framework for assuring any IS is motivated by the maxim [1] that "complexity is the enemy of security." Towards this end, the MN approach seeks to minimize the complexity of both the assurance software, *and* the environment for it's execution. The environment for execution of the assurance software is completely isolated from the IS, to ensure that the assurance software is not affected by undesired functionality in complex IS hardware/software components, or incompetence/malicious intents in personnel who deploy/maintain/operate the IS.

The environment — hereinafter referred to as an *MN module* **T** — is also constrained to possess simple, and completely open functional specification. More specifically, MN modules are constrained to be capable of only executing simple sequences of logical and cryptographic hash operations, and demand only modest (for example, a few KB) memory. By taking cues from the well-known Clark-Wilson [4] system integrity model, the assurance protocols themselves can be designed to possess low complexity.

Notwithstanding such deliberately imposed limitations, the rationale as to why MN modules can indeed execute the assurance protocol for almost any IS (even for complex systems with billions or even trillions of internal states), stems from the versatility of cryptographic hash functions [2]. Furthermore, the environment need not be constrained to a single module **T**; it can be a network consisting of any number of identical MN modules.

The specific contributions of this paper are a) identification of some of the important functional blocks of MN modules **T** capable of organizing themselves into IS-specific mirror networks (MN) for executing IS-specific assurance software, and b) an illustration of the process for the design of simple assurance software for an example IS —the Domain Name System [3].

II. MN Architecture

In the Clark-Wilson system integrity model, those system data items whose integrity is assured are identified, labeled, and considered as **constrained data items** (CDIs). CW Integrity Verification Procedures (IVP) determine if the current state of all CDIs represent a valid (or invalid) state of the IS. Only "well-formed" transformation procedures (TP) can modify create or modify CDIs. TPs that *create* CDIs transform **unconstrained data items** (UDIs), which are typically external inputs to the system, to CDIs. A TP is well-formed if it is guaranteed to always take the system from one correct state to another correct state.

If the correctness of the IS state (the collective state of all CDIs) is initially demonstrated by an IVP, and thereafter, if only well-formed TPs are used to modify/create CDIs, by simple induction, the system is guaranteed to always remain in a correct state. The CW model includes an explicit enumeration of **CW tuples** of the form (user, TP, CDIs/UDIs) that specify which user process is allowed to execute which TP, and which CDIs can be modified by the TP. The correctness of IVPs, TPs and CW-tuples are certified by a "security officer."

A. MN Model

Loosely, the MN model can be seen as a variant of CW model applied not to the IS, but to *the assurance protocol for the IS*.

In the MN model, one-way functions of such IS S data/states whose integrity needs to be assured, are *CDI databases* for the MN S'. A CDI database is a collection of records of the form (a, v) where a is an index and v is the value associated with the index. ρ *types* of CDI databases can exist in MN S' that executes the assurance protocol for IS S. The differences between different CDI database types lie in the IS-specific interpretation of the two values (a, v) in each record, and differences in rules for updating records.

External events (UDIs) that trigger modifications to IS databases are made available to the MN. They may a) directly trigger modifications to CDI databases, and/or b) result in the creation of *internal* events, in the form of MN-messages. Internal events may in turn trigger modifications to CDI databases and/or creation of another MN message.

A parameter μ represents the number of *MN-message types*. A parameter ν represents the number of *types* of (external + internal) events. Each of the ν event types is associated with a TP, characterized by

1) the types of CDI databases (a subset of ρ types) affected by the event;
2) *pre-conditions* for execution of the TP; and
3) post-conditions following execution of the TP.

Most common pre-conditions take the form of i) existence/non-existence of specific records in the CDI database, and/or ii) receipt of an MN message of a specific type. Common post-conditions are i) update to a record in the CDI database and/or ii) creation of an MN message of a specific type.

The type-dependent interpretation of ρ CDI database types, μ message types, along with the characterization (affected database types, pre/post-conditions) of the ν TPs, are explicitly enumerated in a *static MN-rules database*, which is made public. The cryptographic commitment to the entire static MN-rules database is the identity S' assigned to the MN deployed to monitor the IS.

The MN rules database is the "software description" of the assurance protocol for the IS S, designed to guarantee the desired (IS-specific) assurances. This "software" is intended to executed on a platform composed solely of a network of MN modules — the Mirror Network (with identity) S'.

B. Execution of Assurance Software

While ρ, the number of types of CDI databases, is fixed at design time, the total number of CDI databases (say, d) used by MN S' (to "mirror" IS S states) is *dynamic and unbounded*. Any number of databases of a specific type may be used. Each of the $d = \sum_{i=1}^{\rho} n_i$ CDI databases (where n_i is the number of CDI database of type i) is tracked by a dedicated MN module.

All MN modules are identical by design: the only difference between modules is that each has a unique identity. However, the specific TPs executed by a MN module will depend on the type of CDI database tracked by the module. Thus, corresponding to each CDI database type is a MN member (module) *role*.

Apart from d MN members, every MN includes a module with a special role — the *MN creator*, responsible for inducting/ejecting other modules into/from the MN. A module inducted into the MN is assigned a unique (within the MN) role-based member identity. The MN creator maintains a *MN-membership database* indicating correspondence between the original module identities and the MN-specific role-based member identities.

During regular operation of the IS, MN TPs are executed by various MN members. The *only link* between the IS S and the MN S' is that external events that trigger changes IS states are also conveyed to a MN member. A MN member (say) X triggered by an event of type j (where $1 \leq j \leq \nu$) consults the MN-rules database to determine if it's role-type permits it to honor event j, and if the pre-conditions for execution of the TP are satisfied. If so, member X "executes" the TP by imposing the post conditions — by modifying one or more records in the CDI database tracked member X, and/or creating an MN message to trigger an internal event in another member.

C. IOMT and MN Databases

MN databases can take different forms like the static MN-rules database, dynamic CDI databases, or the dynamic MN-membership database. However, all such databases are eventually interpreted as set of two-tuples of the form (a, v).

The index ordered Merkle tree (IOMT) [2], [5]-[7] is a binary hash tree based data-structure, which permits even a severely resource limited module to *virtually* store a database of any size, by *actually* storing only a single cryptographic hash — the root of the IOMT. For reliably performing basic database operations in such a virtually stored database with N records, the module (which stores only the IOMT root) will merely need to evaluate $\mathbb{O}(\log_2 N)$ hashes[1] for each basic database operation like reading, updating, inserting or deleting a record.

Thus, notwithstanding the resource limitations of MN modules, there is no practical restriction on the size of an MN database that may need to be tracked by a MN member. The databases and corresponding IOMT nodes can be stored in any convenient location — for example, by the untrusted IS S that desires to demonstrate it's integrity to the MN S'.

For any database, an important requirement is that of ensuring uniqueness of record indexes. The IOMT is an extension of the Merkle hash tree [8] to support insertion/deletion of indexed records while simultaneously guaranteeing uniqueness of record indexes. The IOMT leaf corresponding to a record (a, v) is of the form $(a, a'v)$, where a' is next higher index in the same database. If a is the highest index then a' is the least index. For a database with a single record (a, v) the corresponding IOMT leaf is $(a, a' = a, v)$.

A value $v = 0$ for any index has a special interpretation: a record $(a, 0)$ (and hence the corresponding IOMT leaf $(a, a', 0)$) is a *place-holder* for index a. Records for currently non existent indexes are always inserted as place-holders; only place-holders can be deleted.

Thus, existence of an IOMT leaf like (b, b', v) (which can be readily ascertained by the module by performing

[1]For e.g., 30 hash operations for performing a read/write/insert/delete operation in a database with a billion records.

$L = \log_2 N$ hash operations) convinces the module of the following:

1) non-existence of leaves (and thus, records) for all indexes *enclosed* by (b, b'). The tuple (b, b') is an enclosure for an index a if $(b < a < b')$ OR $(b' \leq b < a)$ OR $(a < b' \leq b)$; and
2) existence of a record for index b, associated with value $v \neq 0$ or, *if* $v = 0$, *non existence* of a record for index b.

To insert a place-holder for an index a a pre-requisite is that an IOMT leaf (b, b', v) should already exist in the tree such that (b, b') encloses a. Following the insertion, the enclosing leaf will be modified to (b, a, v) and the newly inserted place-holder will be $(a, b', 0)$.

D. IS Assurance Protocols

IS-specific assurance protocols are constructed by appropriately chaining simple *generic* (not specific to any IS) security protocols through IS-specific pre/post-conditions of TPs. The capability to execute generic security protocols are in-built into every module. More importantly, execution of generic protocols will only involve simple hash and logical operations.

In terms of the intended purpose, the generic protocols cater for i) creation and maintenance MNs; and ii) verification of pre-conditions; and iii) imposition of post-conditions. In terms of functionality, generic protocols can be classified into a) IOMT protocols, and b) key distribution and mutual authentication protocols.

IOMT protocols cater for insertion/deletion of place-holders, and reliably reading/updating a record, in a virtually stored database. For example, when an IS-specific TP requires an update to the value v in a CDI record for index a (as a post-condition $(a, v) \rightarrow (a, v')$), a generic IOMT update protocol is used a) verify the existence of (a, v) in the virtually stored database, and to update the IOMT root corresponding to the update mandated by the TP. As insertion/deletion of place-holders have no effect on the application-specific interpretation of the CDI databases, such operations can be completely under the purview of generic functions.

Several key distribution schemes [9] – [11] for mutual authentication exist that have been explicitly designed for scenarios involving resource limited participants. For example, the MLS protocol [11] will need every module to store only a single secret, and evaluate a single hash to compute a pairwise secret with any other entity. Such pair-wise secrets can be used for computing hashed message authentication codes (HMAC) for mutual authentication.

Thus, in the MN model, generic key distribution protocols are used to facilitate mutual authentication of a) message exchanges between modules (potential members and the MN creator) to leave/join an MN; and b) authentication/verification of MN messages between MN-members, and between select MN members and external entities who convey events or query the state of the MN. IOMT protocols are used to a) reliably perform read/write/insert/delete operations in dynamic CDI databases (which is unique for each member) for purposes of verifying pre-conditions and imposing post-conditions; b) reliably perform read/write/insert/delete operations in the dynamic MN-membership databases (tracked by the MN creator module) for inducting/ejecting modules into/from the MN; and c) reliably perform read operations on the static MN-rules databases (the same database referred to by every member of the MN).

E. Mirror Network Design and Deployment

Creation of the MN rules database for an IS S can be seen as a process consisting of the several steps like 1) enumeration of desired IS-specific assurances; 2) identification of CDIs; 3) appropriate choice of ρ types of MN-member roles; 4) identification of ν event types and μ message types; 5) specification of pre-conditions and post-conditions for the TP for each event; 6) specification of any number of application specific constants, and finally, 7) specifying the identity of the MN-creator (by unique module identity, say Π_c). All components of the MN specification become leaves of a static IOMT with root S'.

Before the assurance software (with cryptographic commitment S') can be executed, the platform has to be deployed. This involves induction of modules into the MN, who then become MN members. To induct a module with identity Π_x into the MN S', the MN creator module Π_c creates an authenticated message addressed to Π_x conveying the MN identity S' and the role-based member identity X assigned to module Π_x. The member identity X implicitly specifies the role of the module, using a few MSBs of X. The MN creator also conveys a unique secret to member X, which can be used for authentication of messages created by X. As all members in MN S' are initialized with the same MN identity S', all members execute (TPs in) the same assurance software (or rules database).

The MN creator module Π_c tracks two IOMT roots: one for a database of records of the form (X, Π_x) indexed by member identity to ensure that no member identity is assigned to more than one module; the second for a database of records of the form (Π_x, X) indexed by module identity to ensure that no MN module is assigned more than one member identity.

During regular operation of the MN, a network of CDI databases $\mathbf{D}_1 \cdots \mathbf{D}_d$ (where d is dynamic and unbounded) that capture the state of an IS S is tracked by a MN S' consisting of d MN members, where each member tracks one CDI database, by storing one dynamic IOMT root. When triggered by an event, a member consults the rules database to check if it is required to honor the event (based on MSBs of it's member identity), verify pre-conditions, and satisfy post conditions by utilizing generic IOMT and mutual authentication protocols in-built in each module.

In the next section we proceed to illustrate by example, the process of designing MN-rules for an important IS, the Domain Name System (DNS).

III. MN-Rules for the Domain Name System

The DNS is a hierarchical name space. A level 3 domain name like $x.y.z$ is derived from a level 2 name $y.z$ which is derived from a top-level domain (TLD) name z, which is derived from the *root* of the entire DNS name-space. The

DNS is also a distributed database of DNS resource records (RR) corresponding to domain names. A DNS RR conveys a *value* corresponding to a (domain) *name* and (record) *type*. Frequently used types include *type-A* (values are IP addresses), *type-NS* (values are domain names of name-servers), and *type-MX* (domain names of the mail-servers). An RRSET is a set of RRs with the same name and type.

The *owner* of a name is a DNS *zone-owner*. The zone-owner can derive new names, delegate derived names to other entities (who then become zone-owners), and create different types of RRSETs for retained (not delegated) names. The root zone and TLD zones are special zones as they can only delegate derived names (they can *not* retain ownership of any derived name). The root zone creates new TLD (level 1) names that are delegated to TLD owners. TLD owners can derive level 2 names only when instructed by the TLD *registry*, as potential zone-owners gain ownership of level 2 names from the registry (using DNS Registrars as middle-men).

DNS RRSETs created by owners of various names are hosted by DNS servers that respond to queries by name and type. Ultimately, users of the DNS desire specific assurances regarding integrity of DNS RRSETs they receive from DNS servers. The desired integrity assurances can be classified into two broad categories: a) *process* assurances, regarding the integrity of the process leading to the creation of names (and DNS RRSETs from owned names), and b) *transit* assurances: that the RRSET is not modified in transit.

The process leading to creation/deletion of names, and RRSETs from names, includes various events like acquiring/ceding ownership of a name through delegation, deriving a new name from an owned name, and creating different types of RRSETs corresponding to owned names.

An RRSET created by the owner of a zone, and hosted by one or more zone name servers, may be queried by end-users using a local DNS server (usually operated by the user's ISP) as an intermediary. Thus, both local DNS servers are zone servers are "middle-men" between zone-owners (who create DNS RRSETs) and end users. The desired transit assurance is that the middle-men will not modify RRSETs, or incorrectly deny the presence of explicitly queried records.

It is pertinent to point out here that the current standard for assuring DNS, viz., DNSSEC, [13] does *not* cater for any of the *process* assurances. It caters only for transit assurances. The specific desired assurances regarding DNS can now be summarized as follows:

- A1 Uniqueness of delegation; a name can not be delegated to a plurality of entities.
- A2 Names can be derived only from owned names.
- A3 Only derived names can be delegated.
- A4 Delegated names (that were owned some time in the past) can *not* be used to derive new names or new RRSETs.
- A5 DNS servers (the middle-men) will not illegally modify DNS RRSETs,
- A6 DNS server will not deny the presence of RRSETs that actually exist.
- A7 Registries will not deny available names

A. CDI Databases and Assurances

In the MN S' for the DNS IS S, MN members track one-way functions of DNS data created by different entities like root, TLDs, Registries, zone owners, etc. The CDI databases for such an MN S' can be seen as *name* databases and *RRSET* databases. In name databases the records are indexed by (a one-way function of) the domain name. In RRSET databases the record indexes are one way functions of a name index (in the name database) and RRSET type.

A constant Γ represents the root zone. Indexes corresponding to level-1 names can be derived only by hash-extending Γ. Thus, the index corresponding to a TLD name z is $n_z = h(\Gamma, z)$. The index corresponding to a level-2 name $y.z$ is obtained by hash-extending it's parent index as $h(n_z, y) = h(h(\Gamma, z), y)$. The index for a RRSET record of type t for a name n is also derived by hash extending the name index n with the RRSET type t as $n_t = h(n, t)$.

The implication of the existence of a name record (a, v) in a name CDI database depends on the value v. Specifically, the value can be 0, or a *count* $1 \leq c \leq \kappa$ or a member identity $Y > \kappa$, where κ is a constant defined in the MN rules database that denotes maximum count (all member identities are assumed to be greater than maximum count κ, say, by constraining identities to have MSB 1). Thus, the implications of the existence of a record (a, v) in the database of a member X are as follows:

1) $(a, v = 0)$ implies index a is a place-holder, and thus, provides no information regarding name index a.
2) $(a, v = 1)$ implies name a is owned by X, *and* zero records have been derived by hash-extending a. More generally, $(a, 1 \leq v = c \leq \kappa)$ indicates that a is owned by X, and $c-1$ records have been derived by hash extending the name a.
3) $(a, v = Y > \kappa)$ implies the name a has been delegated by X to a member with identity Y.

Message Types: $\mu = 3$ types of messages are used in the MN for DNS: DLG (delegation), SUR (surrender delegation) and RR (to convey a RRSET). A member X that owns a name m can derive any new name $n = h(m, x)$. A derived name can then be delegated to a member Y by creating a message $DEL(n, X \rightarrow Y)$. A member X that has delegated a name n to Y can request Y to surrender the name by creating a message $(n, X \rightarrow Y)$. A member Y who had received a name n through delegation by X can surrender the name by creating a message $SUR(n, Y \rightarrow X)$. A member X who owns a name n can create a RRSET for any index of the form $n_t = h(n, t)$. The value v for index RRSET index n_t can be conveyed by X to a member Y by creating a message $RR(n_t, v, X \rightarrow Y)$.

Role Types: The DNS MN has $\rho = 4$ member role types, r, g, z and s.

A lone member with role r owns the root name space (index Γ) and can derive level-1 names which are delegated to members with role g.

A member with role g can own any number of level-1 zones. Any number of role-g members may exist. Such members create level-2 names which may be delegated to

TABLE I. MN RULES FOR DNS MN WITH $\rho = 4, \nu = 17, \mu = 3$. PRE/POST-CONDITIONS FOR 17 EVENTS ARE LISTED FOR A MEMBER WITH IDENTITY X. EVENTS 01 TO 10 ARE EXTERNAL (THAT HANDLE UDIS). EVENTS 11-17 ARE INTERNAL EVENTS TRIGGERED BY MN MESSAGES.

Events	UDI / OI	Preconditions	Post-conditions
$01(r)$	x, Y c	Y type $g, (\Gamma, c), (n, 0), n = h(\Gamma, x)$	$(n, 0) \rightarrow (n, Y), (\Gamma, c) \rightarrow (\Gamma, c+1), DLG(n, X \rightarrow Y)$ (//creation of new TLD name)
$02(g)$	x, Y m, c	Y type $g/z, (m, c), (n, 0), n = h(m, x)$	$(m, c) \rightarrow (m, c+1), (n, 0) \rightarrow (n, Y), DLG(n, X \rightarrow Y)$ (//creation of new name under a TLD)
$03(z)$	x m, c	$(m, c), (n, 0), n = h(m, x)$	$(m, c) \rightarrow (m, c+1), (n, 0) \rightarrow (n, 1)$ (//new name derivation by zone-owner)
$04(z)$	Y m, c, x	Y type $z, (n, 1), (m, c), x, n = h(m, x)$	$(n, 1) \rightarrow (n, Y), DLG(n, X \rightarrow Y)$ (//delegation by zone-owner)
$05(z, g, r)$	Y n, Y'	(n, Y')	$SUR(n, X \rightarrow Y)$ (//request a surrender or deny availability)
$06(z)$	x n, c	$(n, c), (n', 1), n' = h(n, x)$	$(n', 1) \rightarrow (n', 0), (n, c) \rightarrow (n, c-1)$ (//removing a name)
$07(z)$	t, v, Y n, c	Y type $s, (n, c), v > 0$	$(n, c) \rightarrow (n, c+1), RR(h(n, t), v, X \rightarrow Y)$ (//RRSET creation by zone owner)
$08(g)$	v, Y n, c	Y type $s, (n, c), v > 0$	$(n, c) \rightarrow (n, c+1), RR(h(n, \gamma), v, X \rightarrow Y)$ (//NS-type RRSET creation by TLD)
$09(z, g)$	t, Y n, c	Y type $s, (n, c),$	$RR(h(n, t), 0, X \rightarrow Y)$ (//Request removal of RRSET)
$10(s)$	U (n_t, v)	(n_t, v)	$RR(n_t, v, X \rightarrow U)$ (//conveying RRSET to a user)
$11(s)$	 n_t, v	$RR(n_t, v > 0, Y \rightarrow X), (n_t, 0)$	$(n_t, 0) \rightarrow (n_t, v)$ (//accepting an RRSET)
$12(s)$	 n_t, v	$RR(n_t, 0, Y \rightarrow X), (n_t, v > 0)$	$(n_t, v) \rightarrow (n_t, 0), RR(n_t, 0, X \rightarrow Y)$ (//removing an RRSET)
$13(z, g)$	t (n, c)	$RR(n_t, 0, Y \rightarrow X), (n, c), n_t = h(n, t)$	$(n, c) \rightarrow (n, c_1)$ (//reducing counter after RRSET removal)
$14(z, g)$		$DLG(n, Y \rightarrow X), (n, 0)$	$(n, 0) \rightarrow (n, 1)$ (//accepting a delegation)
$15(z, g)$		$SUR(n, Y \rightarrow X), (n, 1)$	$(n, 1) \rightarrow (n, 0), SUR(n, X \rightarrow Y)$ (//surrender name)
$16(z, g, r)$		$SUR(n, Y \rightarrow X), (n, 0)$	 (//name not available)
$17(z, g, r)$	 m, c, x	$SUR(n, Y \rightarrow X), (n, X), (m, c), x, n = h(m, x)$	$(n, X) \rightarrow (n, 0), (m, c) \rightarrow (m, c-1)$ (//accepting a surrender)

members with role z or members with role[2] g. Role g members are required to delegate all derived names.

A member with role z gains ownership of names from members with role g or other members with role z. They *may* delegate some derived names. Any number of role-z members may exist. Each member may acquire any number of names through delegation.

Members with roles g and z create RRSETs by hash extending owned names. Role g members can only use a constant γ to hash extend owned TLD names to create records corresponding to NS-type RRSETs. RRSET records created by z and$g$ type members are conveyed to members with role s. Any number of such members exist, each storing different ranges of RRSET indexes. Members with role s may be queried by end users.

B. Events, Messages and TPs

Ultimately, the purpose of the MN S' is to ensure any user who obtains an RRSET for an index n_t (typically from a local DNS server) to be able to query a member s_i (deemed responsible for storing name-and-type records with indexes in the range $n^l_{t_i} \leq n_t \leq n^h_{t_i}$.

[2]A delegation from a TLD to another TLD at a lower level may be necessary for situations where a country-code TLD like 'ca' may delegate 'co.ca' to a lower level TLD.

17 events that trigger modifications to the CDI database of a member X and/or create MN messages are listed in Table I. Each event may be pertinent only to members with specific roles. For example, event $09(z, g)$ will be honored only by members with roles z, g. External events may be required to handle unconstrained data items (UDI). Internal events 11-17 are triggered by MN messages. The second column depicts UDIs at the top and other inputs (OI) necessary to identify preconditions, below UDIs. For example, for Event 05, Y is a UDI (as Y can be *any* member identity), while n and Y' are "other inputs" necessary to identify pre/post-conditions.

The third column shows the pre-conditions for the TP for the event. A precondition represented as (n, c) implies that the record (n, c) should *exist* in the name database with $1 \leq c \leq \kappa$. An upper case value field Y is assumed to be greater than κ.

The fourth column shows the post-conditions and the purpose of the event (in standard comment format). A post-condition $(a, v) \rightarrow (a, v')$ indicates that the value associated with the record index a should be updated from v to v'.

It is assumed that during the induction of the member with role r that the dynamic IOMT root for the name database tracked by r is initialized to a database with a single record $(\Gamma, 1)$, attributing ownership of the root of the name space to r.

Event 01 is a request to the X (member of type r) to derive a new TLD name by hash extending Γ with a UDI x,

and delegate the name to Y (also a UDI), by creating a DLG message addressed to Y — which is only constrained to be a member identity of type g. The counter associated with Γ is incremented. Event 02 is for creation of a new name under a TLD by a member X of type g and delegation to a member Y of type g or z.

Event 03 is for creation of a new name by a zone owner. Note that unlike TLD and root, zone owners do *not* have to delegate all derived names. A new name created using event 03 *can* be delegated using event 04. A member X of type z can create a $DLG(n, X, Y)$ message (event 04) only if record $(n, 1)$ and it's parent (m, c) exists. Following creation of the delegation message the delegated record will be updated to (n, Y).

Event 05 has dual purposes. If the UDI Y does not match the value in record (n, Y') (or if $Y \neq Y'$) the purpose of this event is to convey to member Y that name n is not available. This may be used by members of type g to inform non availability of a name for delegation. On the other hand, if $Y = Y'$ the purpose is to request Y to surrender the name n that was previously delegated to Y.

Event 06 is for removing a name n' whose whose parent n exists. The counter for the parent name is decremented.

Events 07 and 08 are for creation of an RRSET corresponding to an owned name n. Once again the counter associated with n is incremented. Role g members can only hash extend the name with a constant γ (event 08) for creating NS-type RRSETs. Members with role z (event 07) can create *any* type of RRSET. Created RRSETs are conveyed to a member Y with role s.

Event 09 is for requesting a member Y with role s to remove a RRSET. Only RRSET indexes derived from owned names can be included in such a request.

Event 10 is for conveying an RRSET to a user U.

Events 11 to 17 are internal events triggered by MN messages.

Event 11 is for receiving an RRSET by a member with role s, who then adds it to its RRSET database only if no RRSET for the index exists currently.

Event 12 is for removing an RRSET, upon request by the owner Y. An acknowledgement is sent back to owner Y, which triggers event 13 to decrement the counter.

Event 14 is for accepting a delegation. Event 15 is triggered by a request from Y (created using event 05) to surrender a name n that was delegated to X by Y. Event 16 is triggered message from Y (also created using event 05 in when X was *not* delegated the name n previously). As the name should have been delegated to some other member, X can infer that name n is not available. Event 17 accepts the SUR message generated by event 15 in Y to delete the delegated name n and decrement the counter for the parent name m.

At first sight it may appear that the security protocol in Table I completely ignores replay attacks (for example, an old RR message could be replayed to a member with role s to remove/add an RRSET. The reason that the TPs can ignore this is that generic low level protocols for MN message authentication can address such low-level issues (by rejecting replayed messages).

As the IOMT guarantees uniqueness of record indexes, and as the value can indicate only one delegation receiver, and as the name will be deleted (event 17) only on accepting the surrender, assurance A1 is guaranteed. Assurance A2 is guaranteed by TPs for events 01, 02 and 03. Assurance A3 is guaranteed by TPs for events 01,02, and 04. Assurance A4 is guaranteed by keeping track of the number of derived records for each name index. Only if all derived records have been deleted can a name be delegated / deleted.

Assurance A5 is guaranteed as members with role s can only accept or delete RRSET hashes from other members, and convey the RRSET hashes to users who query the MN. Only if a record for an index a does *not* exist can the member with role s confirm the existence of $(a, v = 0)$ (if, necessary, by adding a place-holder for index a). As members with role s can also confirm non-existence of records by conveying a value 0, assurance A6 is guaranteed. Registries, similarly can invoke event 05 to prove unavailability of a requested name.

IV. Discussions and Conclusions

Current approaches to secure systems predominantly rely on a) a variety of ever changing *reactionary* measures to improve the integrity of different subsystems and b) cryptographic strategies for securing interactions between subsystems. The former includes strategies like i) keeping up to date with security fixes, ii) employing intrusion detection systems (IDS), iii) building moats like firewalls to limit access to subsystems etc. Such strategies are plagued by the possibility of new bugs in updates and bugs/incorrect design of IDSes. Often, breaches in the latter strategies [12] result not from incorrect design of the protocol or flaws in cryptographic primitives, but from the lack of integrity of the environment in which the protocol is executed.

The main motivation for the proposed holistic approach for assuring the integrity of any IS stems from the fact that while rules that govern *how* data should be manipulated by ISes tend to be simple and readily understandable (at least to a domain specialist), security breaches in systems result from issues in the process of *translating* the rules into a practical system. The translation process includes numerous tasks performed during design, deployment and maintenance of the system, possibly by numerous personnel. It is far from practical to be able to assure the integrity of such a process. The MN model permits us to short-circuit the translation process to observe if a system is indeed abiding by design rules.

The MN approach is a variant of the well-known Clark-Wilson model for system integrity. While the CW model attempts to directly guarantee the integrity of the IS, the MN model attempts to guarantee only the execution of the "rules-engine" for the IS. CW TPs and IVPs are software modules that are part the IS itself, and executed on a general purpose multi-user computing environment, that read/write CDIs of the IS. MN TPs, on the other hand, are executed by MN modules, where it is easy to rule out undesired functionality due to their simplicity. MN TPs do not read/write IS data. They only manipulate copies of one-way functions of IS data, while following simple rules. The MN rules database specified

as a static IOMT is the assurance software for the IS, designed to be executed on a platform constituted by a network of MN modules.

The MN architecture does *not* obviate the need for measures necessary to root out malicious functionality in IS components, for if such functionality results in illegal modifications to the IS databases, the IS can no longer demonstrate it's integrity to it's users. Deploying a MN to provide assurances regarding the IS mandates only modest additions to the IS: a) a mechanism for relaying external events that necessitate modifications to the IS database to the MN; and b) components (that need not be trusted) for maintaining various MN databases along with their IOMTs, for providing complementary IOMT nodes required by MN modules to execute TPs.

Various steps in MN design were illustrated by designing of an MN to cater for assurances regarding the DNS. The current approach to secure DNS, viz., DNSSEC, has received very poor adoption [15] due to the substantially increased overhead for storing several additional DNS records for signatures for each record, public keys of zones, and delegation certificates, and the need for DNS clients to verify a chain of signatures in order to be verify the integrity of any DNS RRSET. Furthermore, in DNSSEC non existence of records are demonstrated by revealing unsolicited information (names of records that do exist) leading to DNS-walk [14], [15]. In the MN approach modules with role s can assure the absence of records without the need to disclose unsolicited information. Furthermore, DNSSEC does not provide process assurances A1 to A4 regarding the integrity of the process for deriving and delegating names and A7 for authenticated denial of unavailable names by TLD registries.

REFERENCES

[1] B. Schneier, "A plea for simplicity: you can't secure what you dont' understand," *Information Security*, November 1999.

[2] M. Ramkumar, *Symmetric Cryptographic Protocols*, Springer, 2014.

[3] P. V. Mockapetris, "Domain names - concepts and facilities," RFC Editor, 1987

[4] D.D.Clark, D.R. Wilson, "A Comparison of Commercial and Military Computer Security Policies," in Proceedings of the 1987 IEEE Symposium on Research in Security and Privacy (SP'87), May 1987, Oakland, CA; IEEE Press, pp. 184–193.

[5] V. Thotakura, M. Ramkumar, "Minimal TCB For MANET Nodes," 6th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2010), Niagara Falls, ON, Canada, September 2010.

[6] S. D. Mohanty, M. Ramkumar, "Securing File Storage in an Untrusted Server Using a Minimal Trusted Computing Base," First International Conference on Cloud Computing and Services Science, Noordwijkerhout, The Netherlands, May 2011.

[7] A. Velagapalli, S. Mohanty, M. Ramkumar,"An Efficient TCB for a Generic Data Dissemination System," International Conference on Communications in China: Communications Theory and Security (CTS), ICCC12-CTS, 2012.

[8] R.C. Merkle, "A Digital Signature Based on a Conventional Encryption Function," Advances in Cryptology CRYPTO '87. Lecture Notes in Computer Science 293. 1987.

[9] M. Ramkumar, "Trustworthy Computing Under Resource Constraints With the DOWN Policy," IEEE Transactions on Secure and Dependable Computing, pp 49-61, Vol 5, No 1, Jan-Mar 2008.

[10] M. Ramkumar, "The Subset Keys and Identity Tickets (SKIT) Key Distribution Scheme," IEEE Transactions on Information Forensics and Security (TIFS), pp 39-51, Vol 5, No 1, Mar 2010.

[11] M. Ramkumar, "On the Scalability of a "Nonscalable" Key Distribution Scheme," IEEE SPAWN 2008, Newport Beach, CA, June 2008.

[12] Z. Durumeric et. al., "The Matter of Heartbleed," IMC 2014, Vancouver, Canada, Nov 2014.

[13] R. Arends, R. Austein, M. Larson, D. Massey, S. Rose "RFC 4033: DNS Security Introduction and Requirements," March 2005.

[14] S. Weiler, J. Ihren, "RFC 4470: Minimally Covering NSEC Records and DNSSEC On-line Signing," April 2006.

[15] A. Velagapalli, M. Ramkumar, "Trustworthy TCB for DNS Servers," International Journal of Network Security, Vol.14, No.4, PP. 187–205, July 2012.

PROMELA: Introducing Proof into Cybersecurity Education

George Markowsky[1] **and Linda Markowsky**[2]
[1]Cybersecurity Lab, University of Maine, Orono, ME, USA
[2]Maine Cyber Security Cluster, Portland, ME, USA

Abstract— *Much in cyber defense is done reacting to an event, typically perpetrated by attackers who have found some new vulnerability in a system. Ideally, system defenders could find flaws in systems before attackers and shore up their weak spots. We believe that introducing verification tools can be of great benefit to defenders. This paper provides an introduction to PROMELA, a language designed to provide proofs of validity of programs. It is especially geared to providing proofs of the correctness of protocols, but can be used to provide proofs of correctness of many other types of programs. This paper also provides some examples of using PROMELA and SPIN (Simple PROMELA INterpreter) which are available at no cost.*

Keywords: PROMELA, SPIN, computer proofs, verification, protocols, programs, mutual exclusion

1. Introduction

We wish to see cybersecurity develop into more of a science than it currently is. The most distressing thing is that cyber defense tends to be reactive – people fix systems after they have been hacked! Recent serious data breaches are too numerous to discuss in this paper, but it is clear that intruders sometimes hide in systems for months or even years without detection. We need to be more proactive and better understand the vulnerabilities of our systems.

Ideally, we could "prove" that our systems are secure. We do not feel that this is a realistic goal for large, complicated systems. On the other hand, we might be able to subject parts of our systems to rigorous analysis to find weaknesses. To the extent that we can automate our analyses, we might be able to direct resources to other tasks. We've started looking at the applications of proof to cybersecurity. In particular, it appears that a number of protocols have vulnerabilities. Certainly, if a protocol has vulnerabilities, one could hardly expect to build a secure system on top of that protocol.

It might be helpful to recall the definition of a protocol. *Protocols are sets of rules that govern the interaction of concurrent processes in distributed systems.* [1, p. xi]. Let's consider a rather dramatic example of a poor protocol – the "beacon protocol" shown in the movie "The Return of the King" [2]. In the movie this consisted of a string of 13 mountaintop wood pyres that would be lit in sequence to convey a request for aid. The portrayal of this system in the movie was especially problematic since the pyres are located at extreme elevations on very small peaks and it is not clear how these could be staffed. It is interesting to note that even the ancients thought that this was a problematic protocol.

In the 2^{nd} Century BCE, Polybius [1, pp. 1-2] stated:

> *Now in former times, as fire signals were simple beacons, they were for the most part of little use to those who used them. ... For it was quite impossible to have a preconcerted code for things which there was no means of foretelling.*

Most protocols are quite a bit more complicated. For example, Figure 1 is a "simplified" diagram describing the steps for a TCP connection (see [3] for a detailed version). When looking at such a diagram, it seems reasonable to ask whether there are any deficiencies in the definition of the protocol. Furthermore, even if the definition is correct, how can one be sure that any program written to this specification actually implements the details correctly? To understand the meaning of Figure 1 see [4].

Gerard Holtzmann developed PROMELA (Protocol Meta Language or Process Meta Language) to help simulate and verify protocols and programs involving distributed processes. PROMELA (pronounced Pro MEH La) programs can be executed on SPIN (Simple PROMELA INterpreter) which is available for free at `http://spinroot.com`, where you can also get a PDF, corrected version of Holtzmann's pioneering book [1] for $10.

PROMELA is a C-like language so simple programs are relatively easy to read. At the same time, it has unique features that enable it to simulate and verify programs dealing with distributed processes. As Holtzmann notes [1, p. 111] *the primary purpose of PROMELA is validation, not implementation*. Before we get into more details about PROMELA and running PROMELA programs let's consider some of the systems that have been successfully verified by PROMELA and SPIN.

2. PROMELA Success Stories

This section is based on the very interesting case studies described in [5]. In particular, SPIN and PROMELA were used to verify the control algorithms for the new flood control barrier called the Maeslantkering, Figure 2, built in 1997. This work was carried out by the Dutch firm Computer Management Group with the assistance of researchers at the Universite of Twente. For more details see [5] and [6].

Not surprisingly, NASA has made extensive use of PROMELA and SPIN for some of its projects. Gerard Holzmann currently works at NASA and has been involved in

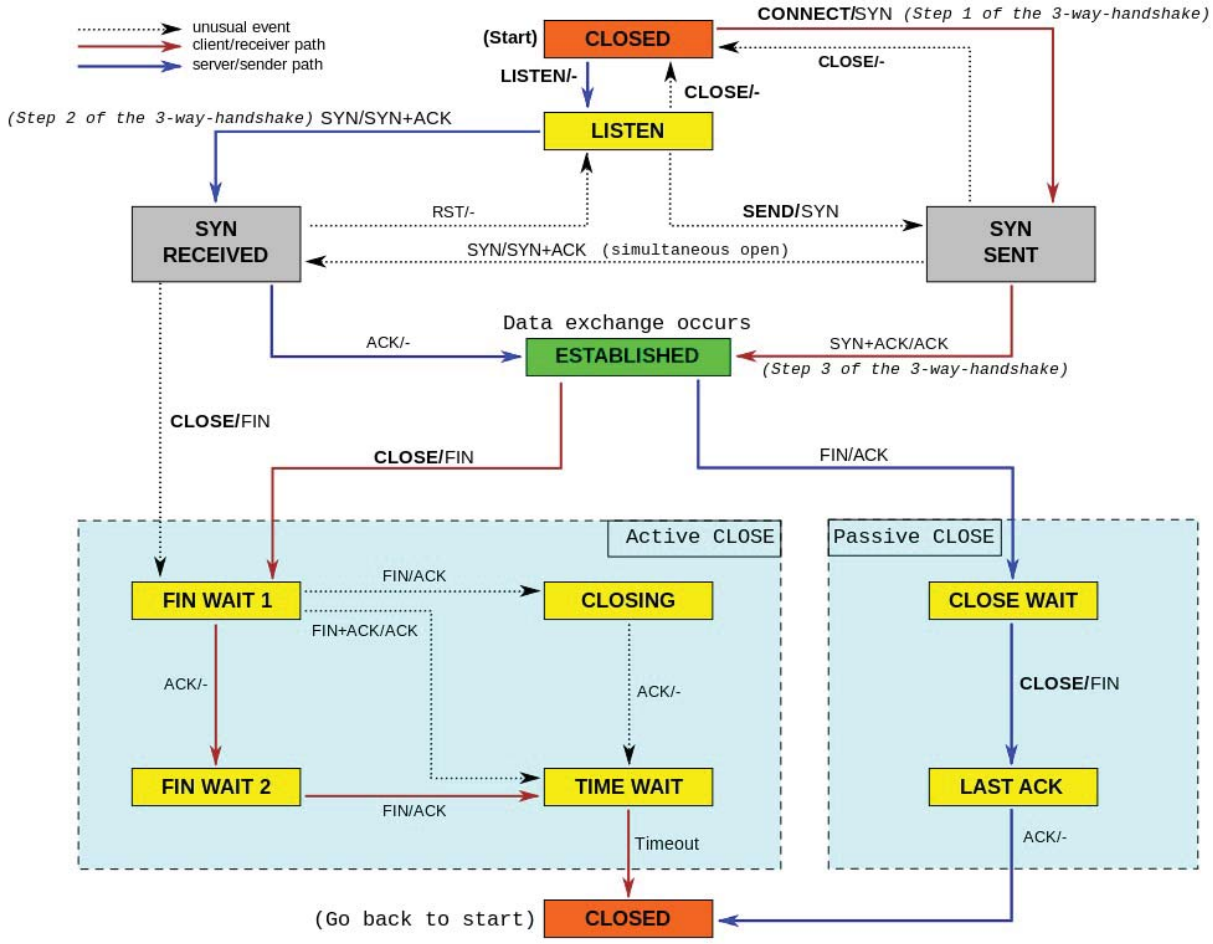


Fig. 1: Simplified Diagram of the TCP Handshake [7]

some of these projects. For a description of additional NASA projects see [5]. A good overview of the role that PROMELA and SPIN played in the Mars Science Laboratory mission is given in [8].

PROMELA and SPIN have been used to verify call processing software including the PathStar^{TM} switch. This verification was done by 16 CPUs working over several months before the switch was marketed. PROMELA and SPIN were used for other mission critical NASA missions including Deep Space 1, the Cassini mission, and Deep Impact. PROMELA and SPIN were also used to verify several international standards including Firewire IEEE 1394.1 used in many PCs, and ISO/IEEE 11073-20601 used for medical devices. They were even used in NASA's investigation of the control software of the Toyota Camry that was looking into the possibility of unintended acceleration. For more details consult [5].

3. Introduction to PROMELA

This section provides a brief introduction of PROMELA based on [1]. Many features of PROMELA will not be discussed in this paper. PROMELA has three major types of objects:

1) Processes - these are always global objects
2) Message Channels - these can be local or global
3) State Variables - these can be local or global

PROMELA does not deal with transmission details or the environment. In PROMELA there is no difference between conditions and statements. For example, the statement

```
while (a != b) skip /* wait for a == b */
```

is equivalent to

```
(a == b)
```

In both cases, execution is blocked until a == b becomes True.

Fig. 2: The Maeslantkering Closed [9]

Table 1: Basic Data Types in PROMELA

Name	Size (bits)	Usage	Range
bit	1	unsigned	0..1
bool	1	unsigned	0..1
byte	8	unsigned	0..255
short	16	signed	$-2^{15}..2^{15}-1$
int	32	signed	$-2^{31}..2^{31}-1$

PROMELA has six predefined data types for variables:

1) bit	2) bool	3) byte	4) short	5) int	6) chan

The first five are called basic data types and behave as one would expect (see Table 1). The type chan specifies a message channel, which is an object that can store a number of values, grouped in user-defined structures.

The names bit and bool are synonyms for a single bit of information. A byte is an unsigned quantity that can store a value between 0 and 255. The types shorts and ints are signed quantities that differ only in the range of values they can hold. Variables can be declared as arrays. For instance,

```
byte state[N]
```

You can use variables as you expect. For example,

```
state[0] = state[3] + 5 * state[3*2/n]
```

Indexing in arrays runs from 0 to N-1 as in C and Python. Declarations and assignments are executable, but definitions are not executable.

The character ; and character sequence -> are statement separators. PROMELA does not have statement terminators so there are no final ;. The sequence -> is used to indicate a causal relationship and is different from conditional expressions such as (expr1 -> expr2 : expr3) which returns expr3 if expr1 evaluates to 0 and returns expr2 if expr1 evaluates to 1. Note that in this case -> must be used and cannot be replaced by ;.

Every PROMELA program must explicitly declare an init process. The init process is like main in a C program. The smallest possible declaration is

```
proctype A(){byte state; state = 3}
init { run A() }
```

Fig. 3: A Simple PROMELA Program

```
proctype A(byte state; short set)
{ (state == 1) -> state = set}
init { run A(1,3) }
```

Fig. 4: Parameter Passing

```
init { skip }
```

where skip is a null statement. Another simple init is

```
init { printf("hello world\n") }
```

The init process can initialize global variables, create message channels, and instantiate processes.

Figure 3 shows a small example PROMELA program. As one might expect, running this program ends up setting the variable state to 3. A more interesting example is the program shown in Figure 4 which illustrates simple parameter passing. Note that arrays and processes cannot be passed as parameters. Furthermore, run A( ) creates copies of processes of type A.

The program shown in Figure 5 illustrates the use of the global variable *state* to link processes A and B. In particular, the statement (state == 1) pauses the execution of process A until process B changes the value of state to 1. Thus, process B completes before process A even though it is started later. Note that in Figure 5 we could use ; instead of ->, but the latter makes it a bit easier to understand what is happening in process A.

Figure 6 illustrates nondeterministic execution, and the fact that the variable state could have the values 0, 1 or 2 depending on the order of execution of the various

```
byte state = 2;
proctype A() {(state == 1)-> state = 3}
proctype B() {state = state - 1}
init{run A( ); run B( )}
```

Fig. 5: Parallel Execution

```
byte state = 1;
proctype A()
{ (state == 1) -> state = state + 1}
proctype B()
{ (state == 1) -> state = state - 1}
init { run A(); run B() }
```

Fig. 6: Nondeterministic Execution

```
chan a, b; chan c[3]

chan a = [16] of { short}

chan c[3] = [4] of {byte}

chan q = [5] of {byte,int,chan,byte}
```

Fig. 7: Some Channel Declarations

commands in processes A and B. Recall that A and B are independent processes. Each process definition contains two statements: a wait statement followed by an assignment. Let A1, A2, B1 and B2 denote these four statements. The execution order of these statements can vary, but it must be that A1 is executed before A2 and B1 must be executed before B2. Thus there are 6 potential execution sequences: (A1, A2, B1, B2), (A1, B1, A2, B2), (A1, B1, B2, A2), (B1, A1, A2, B2), (B1, A1, B2, A2), (B1, B2, A1, A2).

Let's consider the sequence (A1, A2, B1, B2). Since state is global and equal to 1, A1 completes successfully after which A2 sets state to 2 and process A completes. B1 hangs waiting for state to become 1 and B2 is never executed. PROMELA and SPIN are able to function with these blocked processes and infinite loops and display the results to the user. On the other hand, the sequences (A1, B1, A2, B2), (A1, B1, B2, A2), (B1, A1, A2, B2) and (B1, A1, B2, A2) result in both processes completing and state having the value 1. Finally, the sequence (B1, B2, A1, A2) results in B completing, but A1 hanging and state having the value 0.

Message channels model the transfer of data from one process to another. As noted earlier, channels can be declared globally or locally. Figure 7 shows some examples of channel declarations. The first declaration declares three channel identifiers, with c being an array of 3 messages. The second declaration states that a is a channel with a capacity of 16 messages of type short. The third declaration states that c is an array of three channels each of which can store 4 messages of type byte. The last declaration states that q is a channel that can store 5 messages, each of which has four fields of the types indicated.

Channel communication is indicated by ! (send) and ? (receive). Thus c!expr sends the value of expr to channel c which means that the value is appended to the end of channel c. Similarly, c?msg reads the first value stored in c in the variable msg. PROMELA also supports sending and receiving multiple values in messages. In general, if a program tries to send more data than a channel can handle, the excess data is lost. Similarly, if a program tries to access more data than is available in a channel, then the surplus data is lost. See [1] for more details.

4. SPIN

SPIN runs on Windows, Mac and Linux. We will sketch the installation steps. If you want to install SPIN on your computer be prepared to spend some time getting everything to work properly. SPIN lives at http://spinroot.com, which has lots of documentation and information about SPIN and PROMELA.

You can run SPIN at the command line using the command >spin filename.pml. You can then use any text editor to write and execute your PROMELA programs. Of course your path must be set up appropriately or you may work from the SPIN directory. Alternatively, SPIN has a GUI front end, called iSPIN, based on TCL. If you want to run this program you must install TCL. There is more than enough information available to help with the installation, so we will not say more about it now.

Figure 8 shows iSPIN in action on the program shown in Figure 6. The run shown corresponds to the execution sequence (A1, B1, A2, B2), which ends up with the variable state being equal to 1. For more details see [1], [10], [11], [12], [13] and `http://spinroot.com`.

5. Verification with PROMELA

At this point it might be fair to ask whether PROMELA is a silver bullet that will solve all program verification problems. The answer to this question is NO! Using PROMELA properly involves the creation of accurate models of the protocol or program that is to be verified. If you have a bad model, expect to get bad results.

PROMELA and SPIN can help prove that that certain behaviors are inevitable or that certain behaviors are impossible. Note that these two types of claims are negations of each other, i.e., if it is impossible for a certain behavior to occur, it means that it is inevitable that it does not occur. For example, if it is impossible for a program to halt it will run forever!

Holzmann [1, p. 38] lists three types of conditions that protocols should avoid.

1) Deadlocks - states in which no further protocol execution is possible, for instance because all protocol processes are waiting for conditions that can never be fulfilled.
2) Livelocks - execution sequences that can be repeated indefinitely, often without ever making effective progress.
3) Improper terminations - the completion of a protocol execution without satisfying the proper termination conditions.

It is never enough to just "know" that a design is free of deadlocks. A good design is provably free of deadlocks and other undesirable states.

A key tool in the verification of protocols and processes is the *assert* statement. The expression *assert(expression)*

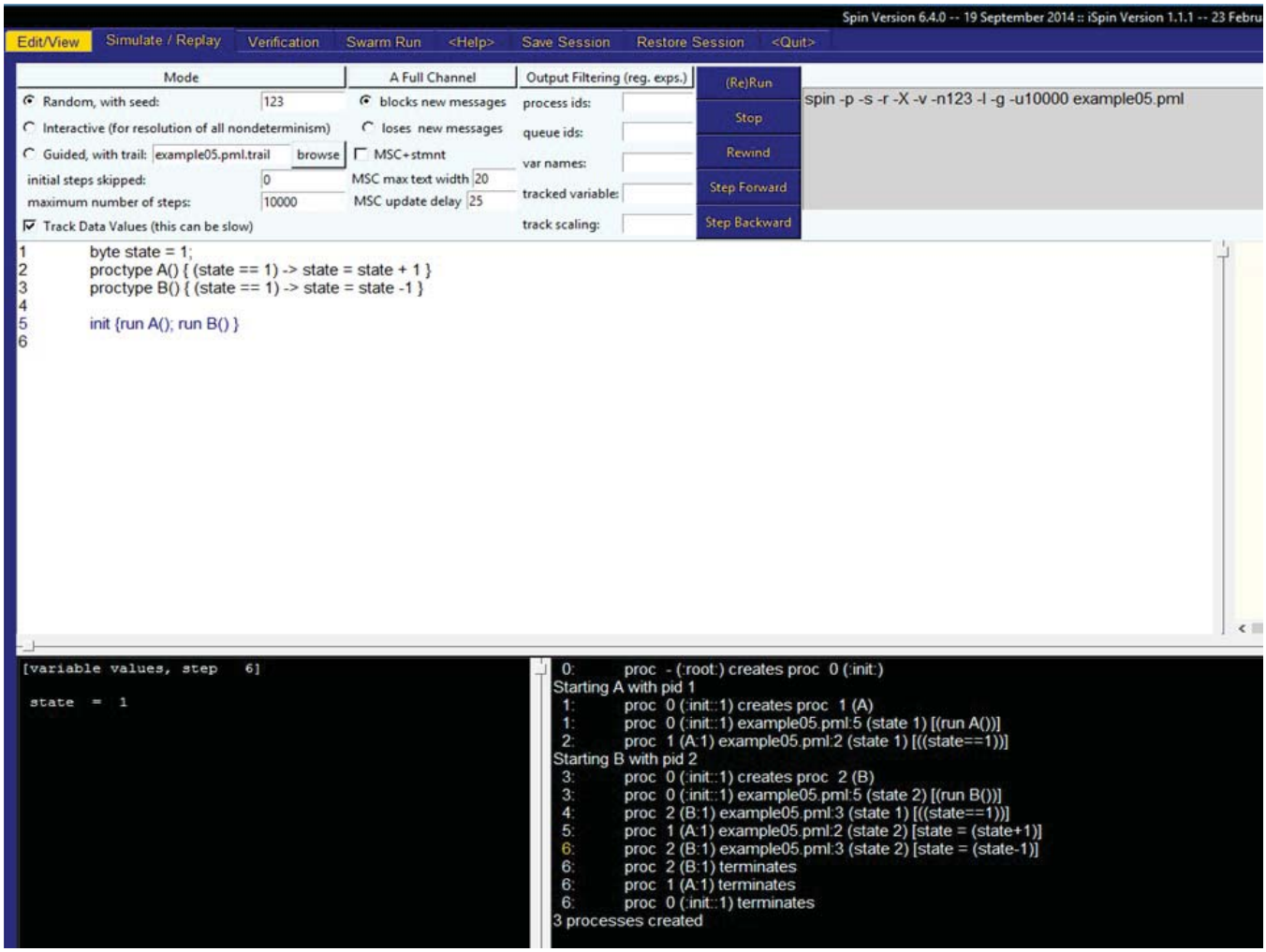


Fig. 8: A Run of the Program from Figure 6

```
byte state = 1;
proctype A()
{ (state == 1) -> state = state + 1
assert(state==2)}
proctype B()
{ (state == 1) -> state = state - 1
assert(state==0)}
init { run A(); run B() }
```

Fig. 9: Using Assert

aborts the program if the expression returns a zero (false), otherwise the statement it is just passed. Thus, if you get to a statement that is past an assert statement you know that the property corresponding to the expression in the assert statement is true. Figure 9 shows the program of Figure 6 modified by the insertion of assert statements. Thus, procedures A and B each have 3 statements which gives us 20 different execution sequences. Figure 10 shows one such sequence that results in an assertion violation because state has the value 1.

6. Mutual Exclusion Algorithms

Dijkstra [14] discussed the problem of designing a correct mutual exclusion algorithm. The idea is to figure out how two or more computers (or threads) can execute the same block of code so that no more than one computer is permitted to be executing in that block at the same time and also there are no situations in which all the computers are locked out of the critical section. This problem is more challenging than one might expect. As Holzmann observes [10, p. 22]:

> *The number of incorrect mutual exclusion algorithms that have been dreamt up over the years, often supported by long and persuasive correctness arguments, is considerably larger than the number of correct ones.*

```
0:        proc  - (:root:) creates proc  0 (:init:)
Starting A with pid 1
  1:        proc  0 (:init::1) creates proc  1 (A)
  1:        proc  0 (:init::1) Example.pml:10 (state 1)        [(run A())]
  2:        proc  1 (A:1) Example.pml:3 (state 1)        [((state==1))]
Starting B with pid 2
  3:        proc  0 (:init::1) creates proc  2 (B)
  3:        proc  0 (:init::1) Example.pml:10 (state 2)        [(run B())]
  4:        proc  2 (B:1) Example.pml:7 (state 1)        [((state==1))]
  5:        proc  1 (A:1) Example.pml:3 (state 2)        [state = (state+1)]
  6:        proc  2 (B:1) Example.pml:7 (state 2)        [state = (state-1)]
spin: Example.pml:4, Error: assertion violated
spin: text of failed assertion: assert((state==2))
#processes: 3
  7:        proc  2 (B:1) Example.pml:8 (state 3)
  7:        proc  1 (A:1) Example.pml:4 (state 3)
  7:        proc  0 (:init::1) Example.pml:10 (state 3)
3 processes created
```

Fig. 10: An Assertion Violation

In this section we will briefly discuss this problem and indicate how SPIN can be used to check on the correctness of a mutual exclusion algorithm. The following discussion is based on the discussions in [1, sec. 5.5], [10, ch. 2], and [15]. Figure 11 shows a simple, incorrect solution to this problem. The key to understanding why the solution in Figure 11 is incorrect is realizing that statements from the two processes A and B can be freely interleaved as long as the relative orders within A and B are preserved. In particular, suppose lines from Figure 11 are executed in the order 7, 8, 9, 10, 16, 17, 18, 19 we will see that both processes are executing in the critical section simultaneously.

We have purposely picked a simple example. To see how Promela can be used to find the fault in this proposed solution we add so "instrumentation" to the program. Figure 12 shows the modified program. Note that we have added a one command monitor process and have modified the init process to also start the monitor process. The monitor process asserts that Acritical and Bcritical cannot be true simultaneously, i.e., the two processes are mutually exclusive.

Figure 13 shows part of the results produced by SPIN. Note that SPIN found a different sequence of instructions that would lead to both processes being in the critical section at the same time. In particular, SPIN found that the sequence of lines 16, 17, 18, 19, 7, 8, 9, 10 in Figure 11 puts both processes into the critical section at the same time. This is the same idea as we discussed earlier, but discovered by SPIN. The true value of SPIN comes from analyzing more complicated algorithms. For more details see [1], [10] and [15]. For a more mathematical treatment of this problem, see [16].

```
1  #define Aturn 1
2  #define Bturn 0
3
4  bool turn, Acritical, Bcritical;
5
6  proctype A()
7   { Acritical = 0;
8     turn = Aturn;
9     (turn != Bturn);
10    Acritical = 1;
11    Acritical = 0;
12    turn = Bturn
13 }
14
15 proctype B()
16  { Bcritical = 0;
17    turn = Bturn;
18    (turn != Aturn);
19    Bcritical = 1;
20    Bcritical = 0;
21    turn = Aturn
22 }
23
24 init { run A(); run B ()}
```

Fig. 11: Incorrect Solution to the Mutual Exclusion Problem

```
1  #define Aturn 1
2  #define Bturn 0
3
4  bool turn, Acritical, Bcritical;
5
6  proctype A()
7   { Acritical = 0;
8     turn = Aturn;
9     (turn != Bturn);
10    Acritical = 1;
11    Acritical = 0;
12    turn = Bturn
13 }
14
15 proctype B()
16  { Bcritical = 0;
17    turn = Bturn;
18    (turn != Aturn);
19    Bcritical = 1;
20    Bcritical = 0;
21    turn = Aturn
22 }
23
24 proctype monitor()
25 {  assert(!(Acritical && Bcritical))}
26
27 init{run monitor(); run A(); run B()}
```

Fig. 12: Verifying With SPIN

```
1: :init:(0):[(run monitor())]
2: :init:(0):[(run A())]
3: :init:(0):[(run B())]
4:         B(3):[Bcritical = 0]
5:         B(3):[turn = 0]
6:         B(3):[((turn!=1))]
7:         B(3):[Bcritical = 1]
8:  A(2):[Acritical = 0]
9:  A(2):[turn = 1]
10: A(2):[((turn!=0))]
11: A(2):[Acritical = 1]
pan:1: assertion violated
!((Acritical&&Bcritical)) (at depth 12)
spin: trail ends after 12 steps
```

Fig. 13: SPIN Finds the Problem

7. Conclusions

Proving correctness is very important, but it is also very hard to do. In general, proving the correctness of an arbitrary program is impossible. This result is not a reason to despair, because in many cases you can prove correctness either totally or partially. Having a proof of partial correctness is valuable because it permits you to focus attention on the parts of your system or program that are not known to be correct.

Program or protocol correctness is not generally taught in the undergraduate curriculum. We feel that enough is known and that high quality tools and documentation are now available so that program correctness can be discussed profitably at the undergraduate level.

References

[1] Gerard J. Holzmann, *Design and Validation of Computer Protocols*, Prentice Hall. Englewood Cliffs, NJ, 1991. An updated version of this book can be obtained inexpensively directly from http://spinroot.com/gerard/popd.html.

[2] J. R. R. Tolkien, *Return of the King*, England, 1955. Movie version Peter Jackson, 2003.

[3] Raid Y. Zaghal and Javed I. Khan, "EFSM/SDL modeling of the original TCP standard (RFC793) and the Congestion Control Mechanism of TCP Reno," Kent State Technical Report, http://www.medianet.kent.edu/techreports/TR2005-07-22-tcp-EFSM.pdf and http://www.medianet.kent.edu/techreports/TR2005-07-22-tcp-EFSM.pdf.

[4] Andrew S. Tanenbaum, *Computer Networks*, 4^{th} Edition, Prentice Hall, Englewood Cliffs, NJ, 2002.

[5] Examples of PROMELA and SPIN Successes, http://spinroot.com/spin/success.html.

[6] Pim Kars, "The Application of Promela and Spin in the BOS Project," http://spinroot.com/spin/Workshops/ws96/Ka.pdf.

[7] "Tcp state diagram fixed new" by Scil100. Licensed under CC BY-SA 3.0 via Wikimedia Commons - https://commons.wikimedia.org/wiki/File:Tcp_state_diagram_fixed_new.svg#/media/File:Tcp_state_diagram_fixed_new.svg

[8] Gerard J. Holzmann, "Mars Code," *Communications of the ACM*, February 2014, Vol. 57, No. 2, pp. 64-73, http://spinroot.com/gerard/pdf/cacm_2014.pdf.

[9] "Maeslantkeringclosed"byWorld66-http://www.world66.com/europe/netherlands/lib/gallery/showimage?pic=europe/netherlands/new_waterway_storm.LicensedunderCCBY-SA1.0viaWikimediaCommons-https://commons.wikimedia.org/wiki/File:Maeslantkering_closed.jpg#/media/File:Maeslantkering_closed.jpg.

[10] Gerard J. Holzmann, *The SPIN Model Checker: Primer and Reference Manual*, Addison-Wesley, 2003. You can find

[11] Karthikeyan Bhargavan and Davor Obradovic, "Formal Verificaiton in SPIN," ftp://ftp.cis.upenn.edu/pub/cis573/public_html/slides/spin2.sept23.ppt

[12] "SPIN Verification Examples and Exercises, http://spinroot.com/spin/Man/Exercises.html.

[13] Rob Gerth, "Concise Promela Reference," http://spinroot.com/spin/Man/Quick.html.

[14] Edsger W. Dijkstra, "Solution of a Problem in Concurrent Programming," *Comm. of the ACM*, Sept. 1965, vol. 8, no. 9, p. 569, http://www.di.ens.fr/~pouzet/cours/systeme/bib/dijkstra.pdf.

[15] Joshua Wise and Greg Hartman, "Proving Dekker with SPIN and PROMELA," Online Lecture, Carnegie Mellon University, http://www.cs.cmu.edu/~410-f08/lectures/L33b_SPIN/.

[16] C. A. R. Hoare, *Communicating Sequential Processes*, Prentice-Hall, Englewood Cliffs, NH, 1985.

Towards Embedded Wearable Security

M. Bouaoud[1], **and A. Bouras**[2,1]
[1]Ministry of Information and Communications Technology MICT, Doha, Qatar
[2]Department of CS and Engineering, College of Engineering, Qatar University, Doha, Qatar

mbouaoud@ict.gov.qa
abdelaziz.bouras@qu.edu.qa

Abstract
Internet of Things (IoT) devices will be counted by the billions in a near future as pervasive technologies are entering everyone's life and private space. From smart cities to smart healthcare or wearable technologies, embedded computing composing the IoTs becomes a new trend with its set of dedicated low powered computing architecture yet delivering powerful information processing capabilities. Sensors collecting information on our behaviors or health status process billions of operations to give everyone the convenience and comfort. However the risks of someone capturing and manipulating the very personal information that wearable technologies are processing will be increasingly true with the scale of deployment. This paper will focus on the existing challenges related to wearable technologies. These technologies will spread with their exposure to more sophisticated cyber threats and threat actors.

Keywords: Wearable technologies, Cyber Security, smart products, Protection/Prevention/Detection, Internet of Things, IoT.

1 Introduction

Today's technical products, such as cell phones and computers are increasingly growing in complexity and therefore need for more interoperability in open environments. This leads to a large amount and great diversity of product-processed data. Internet of Things (IoT) devices will be counted by the billions in a near future as pervasive technologies are entering everyone's life and private space. From smart cities to smart healthcare or wearable technologies, embedded computing becomes a new trend with its set of dedicated low powered computing architecture yet delivering powerful information processing capabilities.
Wearable technologies are derivative from the IoT sticking closer to our very human nature. Intelligent wearable products are hybrid products made of garments, sensor networks and applications. Wearable technologies are interacting with users and the environment while showing capabilities of real time data processing and storage, extending functionalities by communicating with other things [1]. The notion of wearable computer was also defined as a computer that is subsumed into the personal space of the user, controlled by the user, and has both operational and interactional constancy i.e. is always on and always accessible [2].
Sensors collecting information on our behaviours or health status process billions of operations to give everyone the convenience and comfort. However the risks of someone capturing and manipulating the very personal information that wearable technologies are processing will be increasingly true with the scale of deployment. According to Gartner, with more than five billion[1] smart devices, most people have or soon will have access to mobile connectivity while we will count around fifty billion smart devices according to CISCO[2].
The IoT's opportunity and challenge is to deliver truly distributed machine-to-machine (M2M) applications. Every assets such as cars or a simple smartphone for example hold more than hundreds of processors and/or sensors. According to Lux Research, clinical wearable devices should surpass their consumer counterparts in revenue by 2020.
So many other challenges exist today with standard computing technologies while wearable tech are following the same trend of standardization, security vulnerabilities will spread with its exposure to more sophisticated cyber threats and threat actors. Challenges are many fold and there are technical, organizational and environmental.

2 Context

Wearable technology is a computing piece of technology that is attached to the human body directly or indirectly and that allows people to monitor, control or engage with.
The size and spreading adoption of such computing devices create a favorable environment to exploit tons of data and enable new types of services for convenience of the user. The IoT will dwarf the Internet by a factor of 50[3]. It will connect many times as many devices as the mobile revolution.

[1] http://www.gartner.com/newsroom/id/2905717
[2]http://www.cisco.com/web/about/ac79/docs/innov/IoT_IBSG
[2]http://www.cisco.com/web/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf - Internet of Things report: how the Next Evolution of the Internet is Changing Everything, CISCO IBSG, Dave Evans, Apr. 2011.
[3]https://www.rti.com/whitepapers/Right_Middleware_for_IIoT.pdf - DDS: the Right Middleware for the Industrial Internet of Things? RTI.

Wearable technologies are shifting the paradigm from the "Me" to "Us" as data moves away to various platforms in order to accomplishing different objectives: improving health status, recording fitness, access convenience, automation, etc.
That particular aspect of data exchange or data sharing is ubiquitous as no one owning such device would be able to benefit without sharing personal information. Looking closely at the whole flow of information starting from the device itself and where it could end up is by itself a challenging process. Processing information on the computing device, sending it across networks, passing through different routes to end up into a virtualized cloud platform is a typical process that may occur on every day basis by 24/7. Smartphones are already doing it, wearable technologies are just a downsizing of a smartphone including the operating system and I/O options.
The scale of data flow creates several key interception points too, and most of the time it would circulate in plain text over standard TCP/IP based networks.
In the age of the "internet everywhere" and "internet on us", we need to adapt the technology to the transcendence effect of such integration of ubiquitous and seamless technologies without allowing the wave of data to flow to unwanted hands. Sensors, actuators, embedded microprocessors with memory and other technological components are among those which will become the prime actors in delivering information to various service points.
Information could absorb our very notion of privacy, personal information could become volatile and a permanent source for all sorts of analytics platforms. Convenience will transform data privacy regulations into pieces of past obsolete practices that could die if not adapted. The disruption of the harmonized convenience will become more destructive than it may be perceived by many of us.
With all these considerations, it is of utmost importance to understand the stakes now, realize them and anticipate with an adapted response to a future permanent threat. The leakage of our personal data will become officially standard and regular. The trends showing an explosion of data breaches should be put in parallel with the evolution of the technology landscape.
A new era for cyber security to be considered as a plain integrated and extensive actor of the future well-being of societies has started and it should become ubiquitous too and mainstream. No embedded smart technology should be identified as "smart" if it fails to protect what it is processing.
Cybersecurity should be dealt with at micro levels too. Since the beginning of data processing and computation, bits were far from considering data risks. Now it should generate and spread status on information risks as well as encapsulate information properly, obfuscate that stream of data from malicious intent.

3 Current wearable technology Projects

The increased availability of robust sensors and the wide-spread use of the internet as a communication environment lead to the development of many smart products initiatives and wearable projects these past years. Concepts such pervasive computing and ambient intelligence became the trends towards increasingly ubiquitous and connected computing devices. A new vision emerged where distributed services and computing devices, mobile or embedded in almost any type of physical environment (e.g. home, office, cars), all cooperate seamlessly with one another using information and intelligence to improve user experience [3].
Concepts of semantic reality refer to an overarching information space that connects entities in the real world and information from the virtual world [4], and semantic sensor webs leverage current standardization efforts on sensor enablement leading to the accessibility of the sensors via the web. The semantic web activity, in which sensor data is annotated with semantic metadata aims to increase interoperability as well as to provide contextual information essential for situational knowledge [5]. Wearable devices, such as wristbands, smartwatches, eyewear, wearable bio-monitors, and the complementary services that support them have become the focus of much speculation and anticipation, but the success of a wearable depends on its adoption by the market and how well it inspires long-term engagement [6].

Several collaborative research and innovation projects also emerged on smart and wearable technology and information management. We take the opportunity to summarize few of them. *SmartProducts*[4] project for instance focuses on aspects relevant to the acquisition, modelling, reasoning, management, and use of proactive knowledge for smart products. This comprises the technological basis for embedding proactive knowledge into smart products and using it to communicate and co-operate with humans, other products and the environment. The project mainly deals with singular application domains (Smart Home, Smart Offices or small number of specific devices) and mostly on end-users. Proactive knowledge encompasses knowledge about the product itself (features, functions, dependencies, usage, etc.), its environment (physical context, other smart products) and its users (preferences, abilities, intentions, etc.). In addition, proactive knowledge comprises executable workflows and knowledge about interaction, enabling the smart product to proactively engage in multimodal dialogues with the user. Thereby, smart products "talk", "guide", and "assist" designers, workers and consumers dealing with them. Some proactive knowledge is built together with the product, while other parts are gathered during the product lifecycle using embedded sensing and communication capabilities.

The *Welcome*[5] project for example, specifically aims at helping chronic obstructive pulmonary disease (COPD) - Reports estimated that by 2030 COPD will be the fourth largest cause of global Mortality-- patients with comorbidities and to reduce the burden on our health systems, the WELCOME project contributed to create innovative solutions such as an integrated care management tool and a monitoring vest.

[4] http://www.smartproducts-project.eu/
[5] http://www.welcome-project.eu/

Another project called *Easy-Imp*[6] deals with the development of intelligent wearable meta-products in three main pilot applications areas: rehabilitation edutainment games, personal training and cardiac rehabilitation. The meta-product concept consists in smart products with network of sensors that enable them to connect to the cloud, allowing the user to capitalize on the collected data through specific applications. Development of meta-products requires the involvement of teams with interdisciplinary skills (garment designers, sensor networks designers and application developers). Easy-Imp provides a high-level generic methodology to describe an overview of the phases, methods and tools [7].

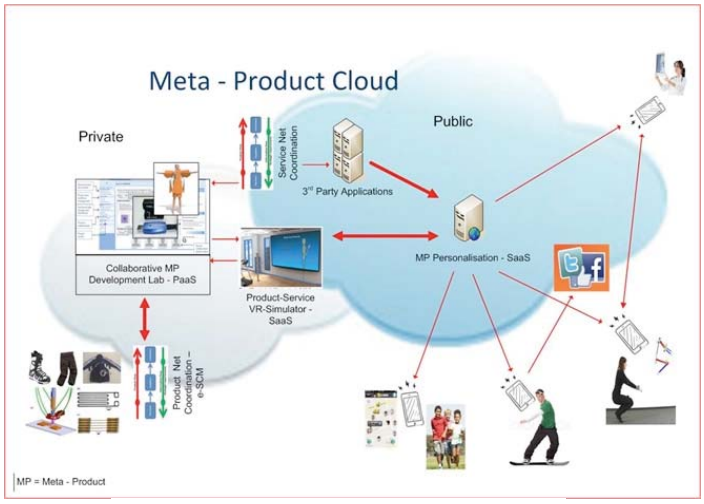


Figure 1. Easy-imp wearable products platform

According to [18] the services of the Meta-Products can be divided into four categories, a) Monitoring: the sensors monitor the internal and external environment according to the specification and the information is then sent for further processing to the other parts; b) Control: could be done automatically by applications or software connected through the cloud, for example responding to the environment once information is processed; c) Optimization, performed either by software or algorithms and d) Automation: the sensors and the software make the meta-product react automatically to certain situations. The level of disruption brought by the development of these smart technologies is so critical, that they should not be ignored as a de-facto trend, as these smart, connected products are transforming competition [8].

4 Wearable Security Challenges

Security challenges are multiple for wearable processing technologies. Whether the devices are conveying regulated or non-regulated data, the security issues remain. While regulations have been put in place to protect consumers, individuals, people in general against market practice abuse, fraud, inconveniences or even manipulation, nowadays technology is perceived as driving the way data will be used. But this is a false perception since technology should serve the needs.

Theft of sensitive information, sabotage, disruption, destruction, theft of identity are risks targeting the user

The types of data that wearable technologies are processing are wide and various from heart rate (ECG), running distance, steps, location, speed, altitude etc.

The data processing activity is not wrong in itself, the issues start when data is kept stored and opened to external accesses or when sharing it (most of the time, transmission of tracking and personal data is in clear text). For example, all wearable activity-tracking devices can be tracked or located through wireless protocol transmissions. According to a security report[7] issued by the security firm Symantec in 2014, fifty-two percent of wearable devices do not have a privacy policy while twenty percent transmitted user credentials in clear text. Some devices would even send data up to fourteen unsecure IP addresses.

Regulations are now trying to play "catch-up" with the evolution of the technology landscape. Data privacy concerns have arisen as a top concern by populations around the globe. Regulators have started several actions towards giant tech companies (Google or Facebook are cases among others) in various part of the world. Technology does not seem to look after user consent before processing user's data nor defining its purpose clearly.

Data privacy laws around the globe are fairly static legislations and require a lot of efforts in defining, registering, exploiting personal data for a definite purpose but also they do not categorize data information explicitly.

HIPAA[8] for example has defined sets of information such as protected health record, while wearable may not be subject to HIPAA, this is due simply to the fact that wearable generated data is not shared with doctors, clinics or hospitals. We are already in the way for ingesting sensors too in order to measure body behaviours and reaction towards particular drugs, thus generating a large amount of sensitive personal data and information. While wearable sensors on personal non-medical devices may still process the same level of sensitivity, still such data will not being considered as regulated data.

Data privacy security issues are related to information/data sharing, data ownership and data/technology lock-in. How to ensure that wearable technology devices are not bound ultimately to the enforcement of these challenges becoming requirements. Today using the latest smartphone technologies is subject to so-often updated terms and conditions embedding a custom tech centric privacy policy. From a privacy regulator statement of "*personal data must be protected and not be shared without user consent*" we entered a world of "*we may share your data with X tier*".

[6] http://www.easy-imp.eu/

[7] "Symantec Quantified-self security report", Web, http://www.symantec.com/connect/blogs/how-safe-your-quantified-self-tracking-monitoring-and-wearable-tech

[8] "U.S. Department of Health & Human Services, Health Information Privacy", Web, http://www.hhs.gov/ocr/privacy/

New regulations must become more specific by drawing clear lines between what is personal data and what is shareable data, moving away from mandatory "opt-in" consent options but giving consumers with better tools to monitor their private life.

Not only regulated data passes through the net of the regulations, but they expose additional risks to users. We mentioned briefly the notion of health records taken from wearable technologies, we can imagine all sorts of development in the area of wearable technologies: from digital preventive care, digital patient experience, bioinformatics to fitness management, athlete performance optimization, data is at risks by those who will take advantages of a vacuum, legally or illegally.

The ingestible sensors are bringing a whole new level of potential exploits, where interception of information and data manipulation could trigger direct harm to humans. The humans become vulnerable through a technology that was supposed to help and support life.

In the case of garment with wearable technologies, the functional clothing design, usually influenced by variables such as the user's own identity, social environment and comfort is organized mostly by the apparel designers not necessarily aware of the technology development. In the other hand the computer expert developing the wearable technology may not integrate the parameters of the needs of users on both the physical or psychological dimensions. The easy exploitation of data could be influenced by the functional designers needs to deliver the right level of comfort, limiting the security efforts needed to adequately protect that information. The role of the technology expert is important in guiding and addressing the functional design needs with a right balance highlighting the risks, until one day functional designers would get notions of risks integrated too.

Cyber-attacks have successfully reached heavily guarded, monitored and secured critical infrastructures. So what about tiny devices that have limited protection mechanisms or none, remaining accessible 24/7 throughout the world? The risks of data interception, corruption and theft of sensitive information are already there, so security mechanisms need to be embedded early.

Looking closely at the flow of information gathered from a wearable technology device gives a blurry picture of where data is taken and ending being processed. The number of network nodes and systems that exist in the data pathway can be tremendous. Routers, switches, wireless access points, sensors, 4G antennas, data centres, servers, there are thousands of potential data access points. This state of data flow from wearable justifies by itself the concerns and the need to cover the security issues in design of this equipment.

Stealing information with sniffing tools targeting wearable devices by proximity or network is largely known and accessible to the many. Hacking tools are also inexpensive and can create massive damages. One can think about the evolution of denial of service attacks which are relayed by "zombie machines" to make these attacks more difficult to apprehend. The distributed nature of computing technologies nowadays is already expanded to wearable technologies as they need a permanent connectivity and access to remote servers, implying a wider access to open networks.

Machines will become so numerous, mobile and pervasive that anyone could potentially become an attack vector relay or simply getting his life endangered by the malicious mind behind an intrusive attack. If one takes control of a mesh of millions of computing nodes in a large area, the same can take control of several key infrastructure or possibly information repositories.

Working on a threefold approach (ref. figure 2 below) with protection, prevention and detection (PPD) processes would help identify the proper security mechanism(s) for data from the sensor to its endpoint access. There are three processes that define the state of data:

- Data is collected through external interactions with sensors, it is a process that can be subject to interferences aiming at either intercepting or manipulating the information.
- Data is stored on device and can be accessed if the information is not controlled by the embedded computer against tempering and unauthorized access.
- Data is transmitted therefore it needs to know where it goes and with which node(s) it will peer to access the destination, in a secure channel established between the origin and the target destination.

Security layers should encompass all the states of data in various ways. For each of these particular states correspond a set of security measures that are aiming at deterring cyber threats.

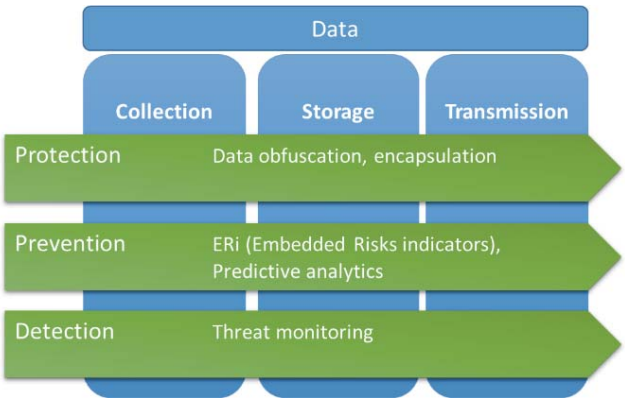


Figure 2. PPD over Data Processing for Embedded Security

There is no doubt that with the evolution of CPUs both in terms of performance and power consumption, the technology landscape is going to evolve toward further smaller yet more powerful components.

Although most of the security functions would be embedded into the system layer, starting with the operating system, it could be worth considering integrating dedicated security components.

Integrated accelerated cryptographic functions for example are intensive consumers of computational power. The integration of security functions within tiny chip components remain a challenge, but security needs to be scaled to the

device too. Low power enabled security functions need to be thought through. After all, these components are already capable of generating and processing an immense amount of data.

Since we can compare the sensitive nature of data processed by wearable technologies to traditional sensitive information operated in large critical organizations the same principles of resiliency, integrity and confidentiality need to apply. These devices are just an extension of existing processing facilities in a way, therefore they could also inherit portion of the security controls too.

Traditionally and for each of the potential threat actors' misbehaviors on systems correspond either one or a set of security control mechanisms. Security standards and policies have been established in order to respond to security issues and challenges, covering most of the time aspects that are not well defined on systems. Which operating system nowadays is fully responding to security issues with smart automated security policies aligned with most of the known controls that can defeat an attacker. Without competing with security dedicated appliances, the application of the security controls as defined in an ISO standard for example (ISO27002) could mitigate a lot of the security challenges.

From defining the right security controls by understanding the security patterns to better mitigate the risks of getting our data in the wrong hands, the computer innovation has still a long way to go in finally considering security as an integral part of the core computational requirements of any processing unit that stores, process and transmit data internally and/or externally. This implies security within the processors, memory, sensors as well as the logical layers. What if my device falls into wrong hands too as my data, would there be mechanisms to wipe out data in case of physical tampering attempt too?

5 Discussion

Embedded security functions become a key essential part of the wearable technology. There can't be trust without security, and the consumers of these technology devices will soon realize that an absence of security is of greater impact on their own physical life than they once thought. Vendors and designers of wearable technology devices will have to understand that customer confidence will fade if security issues spread and become regular daily news.

In order to enable those truly smart functions, we may consider security "by-design" taking advantage of the constant hyper connectivity with functions that will integrate upgradeable:

- Protection mechanisms: with Regular trust controls: devices checks its state of access with a security baseline and security id. Tamper protection through security policies controls is then enabled. Tokenization processes bring some answers to creating the trust control.
- Prevention mechanisms: with lightweight data payload of real time upgradable risk indicators to enable cyber threats predictions and anticipation.
- Detection functions through digital process management on top of the collection of data sensors could help unify data and enable early detection and spread of risk information to security mechanisms and alert recipients.

An additional layer of abstraction with a level of intelligence is needed to correlate and interpret the security information that would trigger security actions based on a particular situation. As an upgradeable mapping of cyber threat actors behaviors to security patterns could be used to determine the existing risk patterns, the level of intelligence needed to confirm and raise the risks can be scaled or distributed via that higher abstraction layer.

The data centric technology evolution is touching everyone's life nowadays, therefore it is worth reconsidering the methods and rules that guide such development.

The human history has witnessed several technological changes or large industrial revolutions. There has never been a time with such a massive amount of disruptive technologies flooding the world. There is a need for actions from different actors of the world: government, technology providers, regulators, citizen to redefine the approach to such invasive technology. We quickly evolved from a world where one computer to many users was a luxury to a world with more than many computers to one person, aiming at facilitating everyone's life. If the aim is definitely a noble objectives, there are too many possibilities to make this dream become a nightmare if not well guided. The connection between the virtual worlds and the physical worlds are blurred that the impact resulting from a technology disruption could be dire.

There is a need to take a strong stance on that wild exploitation of Data. Rather than expecting technology vendors, solution providers to define the trend by themselves, regulators and people shall bring the level of concerns as foundations to build safer and fair technology use towards the market wish to move fast to do business.

Drawing a parallel with Isaac Asimov' laws of robotics, thinking "the laws of Data" could end up as:

1. **Data** shall serve the users without interceding abusively into human privacy,
2. **Data** must be used for a purpose agreed by the user, serving his needs and interests as long as it does not conflict with the first law,
3. **Data** must be protected along its existence as long as this does not conflict with the first and second laws.

The three laws proposed above can ultimately cover the core foundational needs to protect everyone's data, limiting the abuses, by developing detailed charters of rules derived from them. Government should act upon such laws.

Clearly two main paths exist and still options are possible: would there be a set of advanced functions soon implemented that would enable people to trigger choices, validate options, create their own automated responses to clear data processing objectives shared by vendor? In short will I be able to say no to particular processes without being disrupted in my usage of a particular wearable technology?

6 References

[1] Steve Mann. “Wearable computing as means for personal empowerment”; International Conference on Wearable Computing ICWC-98, Fairfax VA, May 1998.

[2] Porter, Michael E., and James E. Heppelmann. "How Smart, Connected Products Are Transforming n." Harvard Business Review, Vol. 92, No. 11 pp: 64–88, 2014.

[3] E. Arts and B. de Ruyter. New research perspectives on ambient intelligence. Journal of Ambient Intelligence and Smart Environments, 1:5–14, 2009.

[4] M. Hauswirth and S. Decker. Semantic Reality - Connecting the Real and the Virtual World. In Proc. of the Microsoft SemGrail Workshop, 2007.

[5] A. Sheth, C. Henson, and S. Sahoo. Semantic Sensor Web. IEEE Internet Computing, 12(4):78– 83, 2008.

[6] D. Ledger, How the Science of Human Behavior Change Offers the Secret to Long-Term Engagement. Endeavour Partners, 2014 (http://endeavourpartners.net/assets/Wearables-and-the-Science-of-Human-Behavior-Change-EP4.pdf)

[7] Essamlali, M.; Sekhari, A.; Bouras, A.; Santiteerakul, S.; Ouzrout, Y., Methodology for collaborative development of intelligent wearable Meta-Products, 8th IEEE SKIMA Int. Conf. on Software, Knowledge, Information Management and Applications, pp.1-8, 18-20 Dec. 2014; doi: 10.1109/SKIMA.2014.7083551

[8] Porter M.E., Heppelmann J.E “How Smart Connected products are transforming competition”. Harvard Business Review. 2014

[9] Jung-Chun Kao; Marculescu, R., "Energy-aware routing for e-textile applications," 2005 Design, Automation and Test in Europe, pp.184-189, doi: 10.1109/DATE.2005.138. 7-11 Mar. 2005.

Recon 2 and the Adversarial Mindset: A Cybersecurity Exercise for Undergraduates

Kian Lutu
Lewis & Clark College
Portland OR, 97219
kianlutu@lclark.edu

Jeanie Mullins
Lewis & Clark College
Portland OR, 97219
jmullins@lclark.edu

Molly Kiefer
Lewis & Clark College
Portland OR, 97219
mollykiefer@lclark.edu

Jens Mache
Lewis & Clark College
Portland OR, 97219
jmache@lclark.edu

Richard Weiss
The Evergreen State College
Olympia WA, 98505
weissr@evergreen.edu

***Abstract* - After a brief overview of the cybersecurity education platform EDURange and its reconnaissance (Recon) scenario, this paper describes a Recon 2 exercise currently being developed: its design, how it will be played, and its learning goals. Like all EDURange scenarios, the purpose of Recon 2 is to provide a hands-on activity to supplement classroom learning. In Recon 2 students will practice scanning to find hosts in a huge network address range, while attempting to avoid an intrusion detection system. Because EDURange is dynamic, the Recon 2 exercise will change each time it is booted. New solutions (when playing again) give students opportunities to refine their skills.**

***Keywords*: Cybersecurity education, scanning, intrusion detection, adversarial mindset**

I. Introduction

Today, there is a GUI for almost everything. But when it comes to cybersecurity, being able to use the command line efficiently is an essential part of network administration. Our goal is to create flexible, accessible, hands-on exercises to teach cybersecurity and the security mindset, also known as the adversarial mindset [1]. EDURange provides a collection of interactive, collaborative cybersecurity exercises, as well as a framework to create new exercises. The activities and framework are cloud-based, providing easy and reliable availability for supplementing lectures, labs, and other classroom activities. Instead of downloading or manually creating virtual machine clients, with the use of a single EDURange account an instructor can create a virtual environment for their entire class to connect to through a Secure Shell (ssh) client [2, 3]. ssh is a network protocol that allows a user to remotely access the command line of another computer.

Recon 2 will help students build an adversarial mindset as they work their way through each level of the exercise. We will go more into depth with both Recon and Recon 2 later in the paper. In the next

section we will discuss related work and how our project extends them.

II. Related Work

Like EDURange, DETERlab and the RAVE Lab are both cloud-based testing environments and both allow experimentation with intrusion detection systems [4, 5]. However, unlike EDURange, they both require each student to have their own personal login which the instructor must apply for in advance. As for the use in the classroom, DETERlab was made for large-scale security experiments and may not be ideal for bursty use associated with large class sizes with homework deadlines. Both DETERlab and The RAVE have a large set of exercises for faculty to use. However, neither seems to provide a framework for instructors to easily create new exercises and modify existing ones to tailor them to their own classes.

EDURange only requires an instructor account, from which scenarios are booted. Students then can ssh to virtual machines located in the cloud. From there the student can complete exercises and experiment. Instructors can utilize the supplied questions available on the EDURange website while students are working through the scenarios. The goal of these questions is to guide students without explicitly telling them the steps.

III. Recon

Recon is a simple exercise, inspired by a PacketWars scenario. The goal of the exercise is to execute reconnaissance techniques to locate hosts in a large network address range. Students must try to identify the IP addresses and the active ports on the live machines. They explore nmap command options and other strategies to find the results within a time limit. The network topology for the Recon exercise can be seen in Figure 1.

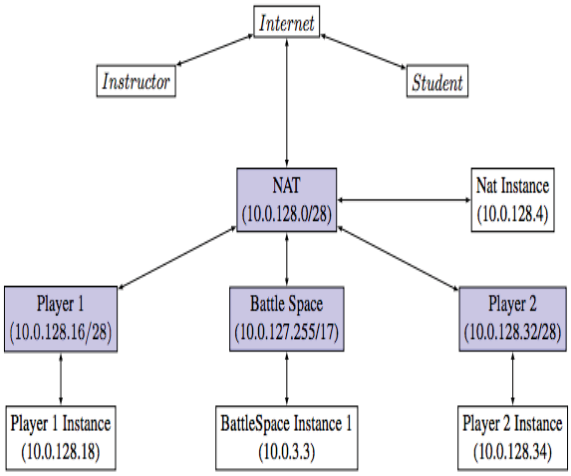


Figure 1: Diagram of Recon Network [6]

Recon has been test-driven by over 120 students and implemented in four classroom curriculums, with the testers ranging from novice to advance. The overall response from these students was that the Recon exercise was engaging, even in varied settings [7]. The next step for the Recon series, i.e. Recon 2, is to challenge the students to understand how an attacker can use stealth when probing the network and to understand how intrusion detection systems operate.

IV. Recon 2

In this exercise, the main focus will be the tradeoff between stealth and speed. Recon 2 will be based in an unknown network with an intrusion detection system (IDS) and network monitoring.

A. Design

The topology for the Recon 2 exercise will be very similar to that of Recon, the main difference being the presence of an active script running on the battle space instance and an added scorer instance. The players will try to locate the hosts in the battle

space, while an IDS will monitor the network traffic. The IDS will output an alert when it has detected a player who wasn't stealthy enough when probing the network. Alerts will be saved in a pcap file and read into a Python script running on the scorer instance. The script will analyze the pcap to determine which player triggered the alert and then decrement that player's score. The topology is shown in Figure 2:

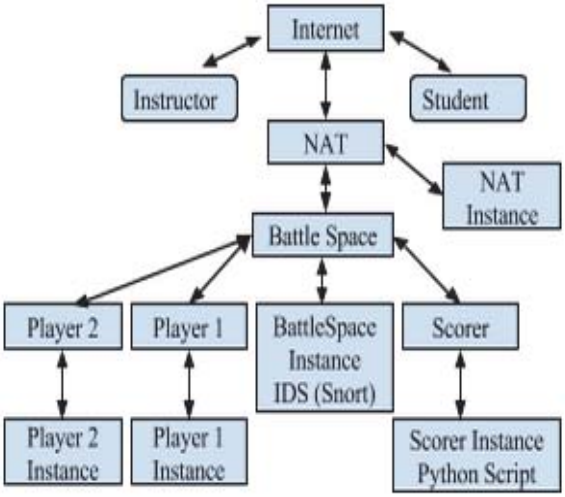


Figure 2: Diagram of Recon 2 Network

B. Playing Recon 2

In Recon students were asked to locate an unknown number of hosts in a large network address range. Students may use any command they choose to find the host, but the primary focus for the exercise is nmap. In Recon 2, students will try to avoid being detected by a network-based intrusion detection system. Intrusion detection is an active area of research, but for this exercise will be focusing on Snort as it is a commonly used open source tool [8, 9]. Snort is capable of analyzing live network traffic and will send alerts when potential malicious traffic is detected [10]. If a player is detected an alert will be sent by the Snort IDS. Then the scorer will dock that player's points. For use in the classroom, the exercise could be split into three levels.

i. Level 1

In the first level, students will be given a list of Snort rules and be asked to figure out nmap commands that won't trigger the rules. Once the students have explored and tested the different stealthy nmap options, they will move on to Level 2. An example of a Snort rule can be seen in Figure 3. This rule sends an alert when someone outside of the Battle Space network runs a TCP scan. In this scenario the Battle Space or $HOME_NET includes the virtual machines that the students are searching for, while the external network includes the student machines, as shown in Figure 2.

```
alert tcp $EXTERNAL_NET any -> $HOME_NET\
(msg:"SNMP request tcp"; flow:stateless;\
reference:bugtraq, 4088; reference: bugtraq, 4089;\
reference:bugtraq, 4132; reference: cve, 2002-0012;\
reference:cve,2002-0013;classtype:attempted-recon;\
sid:1418; rev:11;)
```

Figure 3: Sample Snort Rule

ii. Level 2

In Level 2 students will be asked to test certain nmap commands and to analyze the corresponding Snort alerts. Once the students understand why the rule was triggered, they will then try to write the rule that matches the alert. An example of Snort's alert output can be seen in Figure 4. The alert is in responds to the Snort rule that was shown previously in Figure 3. To make the exercise more realistic, there will be running script that generates background noise. We will utilize tcpreplay to achieve this element of the exercise. Tcpreplay is a collection of utilities that can be used to edit and replay network traffic that was previously captured by a network monitoring tool and saved in a pcap file. [11] After gaining a

working understanding of alerts, students will be prepared for Level 3.

[**] [1:1418:11] SNMP request tcp [**]
[Classification: Attempted Information Leak]
[Priority: 2] 05/29-21:05:50.815139 10.0.128.4:38620 -> 10.0.7.58:161 TCP TTL:64 TOS:0x0 ID:19448 IpLen:20 DgmLen:60 DF ******S* Seq: 0x9DAEC6CO Ack: 0x0 Win: 0x6903 TcpLen: 40 TCP Options (5) => MSS: 8961 Sack0K TS: 172526904 0 NOP WS: 7
[Xref=>http://cve.mitre.org/cgi-bin/cvename.cgi?name=2002-0013]
[Xref=>http://cve.mitre.org/cgi-bin/cvename.cgi?name2002-0012]
[Xref=>http://www.securityfocus.com/bid/4132][Xref=>http://securityfocus.com/bid/4089]
[Xref=>http://www.securityfocus.com/bid/4088]
Figure 4: Sample Snort Alert Output

iii. Level 3

In the last level of Recon 2, students will be assigned roles as either attacker or defender. The defense requires writing snort rules, while the offensive side uses nmap and different members of the team can use different IP addresses. The defense team will be penalized for false positives caused by their Snort rules.

C. Learning Objectives

After this exercise, students will have a firmer grasp on how nmap works and should be able to map networks with a lower probability of being detected by the IDS. Through analyzing the IDS outputs and rules, the students will understand how an IDS detects intrusions. They will also be able to independently comprehend IDS rules and create their own based on an attack's characteristics.

V. Future Work

Because the Recon 2 exercise is currently in development, there is still the opportunity for conceptual and structural changes in the design and gameplay. When the exercise is functional, students at Lewis & Clark College and The Evergreen State College will test it. After the primary testing any necessary changes will be made and then the Recon 2 exercise will be introduced to volunteer test groups outside of these two schools. Once the main goals of the exercise have been ironed out, we will look at possible areas of growth. There is a limit to configuring Snort rules, which means it cannot be completely tailored to our needs. An idea to combat this is to use a program for detecting speedy attacks [12] to make users explore nmap's timing templates.

VI. Conclusion

The EDURange exercise, Recon 2, will allow students to develop scanning skills, learn about intrusion detection, and gain an adversarial mindset. The cloud-based nature of the exercise allows the students to easily access a safe environment to understand and practice these topics. EDURange exercises are dynamic and easily changed; allowing instructors to run the Recon 2 scenario multiple times with their student and better refining their skills. By its nature, Recon 2 is extendable, with providing educators not only the tools listed above, but also the possibility for many different routes towards growth and learning.

VII. Acknowledgements

Funding for this project was provided by The National Science Foundation grants 1141314, 1141341, 1516100, 1516730 and the John S. Rogers Science Research Program of Lewis & Clark College.

VI. References

[1] Hooshangi, Sara, Richard Weiss, and Justin Cappos. "Can the Security Mindset Make Students Better Testers?" In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education* (SIGCSE), ACM, 2015. http://dx.doi.org/10.1145/2676723.2677268

[2] "EDURange: A Cybersecurity Competition Platform to Enhance Undergraduate Security Analysis Skills." The Evergreen State College, n.d. Web. 27 Apr. 2015. http://blogs.evergreen.edu/edurange/

[3] Boesen, Stefan, Richard Weiss, James Sullivan, Michael E. Locasto, Jens Mache and Erik Nilsen. "EDURange: Meeting the Pedagogical Challenges of Student Participation in Cybertraining Environments." In Proceedings of 7th Workshop on Cyber Security Experimentation and Test (CSET), USENIX Association, 2014. https://www.usenix.org/conference/cset14/workshop-program/presentation/boesen

[4] PETERSON, P. A., AND REIHER, P. L. Security exercises for the online classroom with Deter. Proc. of the 3rd USENIX CSET (2010).

[5] Nestler, Vincent J. *Principles of Computer Security Lab Manual*. 4th ed. N.p.: McGraw-Hill Education, 2015. Print.

[6] *EDURange Instructor's Manual* (n.d.): n. pag. *EDURange: A Cybersecurity Competition Platform to Enhance Undergraduate Security Analysis Skills*. The Evergreen State College, 9 Jan. 2015. Web. 27 May 2015. http://blogs.evergreen.edu/edurange/files/2014/03/EDURange_for_faculty.pdf

[7] Weiss, Richard S., Stefan Boesen, James F. Sullivan, Michael E. Locasto, Jens Mache, and Erik Nilsen. "Teaching Cybersecurity Analysis Skills in the Cloud." In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE)*, pp. 332-337. ACM, 2015. http://dl.acm.org/citation.cfm?id=2676723.2677290

[8] Ptacek, Thomas H., and Timothy N. Newsham. *Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection*. N.p.: n.p., 1998. Print.

[9] Wenke Lee and Salvatore J. Stolfo "Data Mining Approaches for Intrusion Detection" In Proceedings of the Seventh USENIX Security Symposium (SECURITY '98), San Antonio, TX, January 1998 https://www.usenix.org/legacy/publications/library/proceedings/sec98/full_papers/lee/lee.pdf

[10] "Snort 2.9.7.3 Has Been Posted!" *Snort.Org*. Cisco, n.d. Web. 03 June 2015. https://www.snort.org/

[11] "Welcome to Tcpreplay." *Tcpreplay*. Trac, n.d. Web. 02 June 2015. http://tcpreplay.synfin.net/

[12] Ali J. Ghandour, Kassem Fawaz, Wassim El Hajj , Hazem Hajj "Slow Port Scanning Detection" Mehiar Dabbagh, *2011 7th International Conference on Information Assurance and Security (IAS)*, pp. 228 - 233, 3 June 2015. http://staff.aub.edu.lb/~we07/Publications/Slow%20Port%20Scanning%20Detection.pdf

Exploring Covert Communication on Google Hangouts

Subash Kumar Saladi[1], Andrew H. Sung[2] and Qingzhong Liu[1,*]
[1]Department of Computer Science, Sam Houston State University
Huntsville, TX 77341, USA,
Email: sks042@shsu.edu; liu@shsu.edu, *correspondence
[2]School of Computing, University of Southern Mississippi
Hattiesburg, MS 39406-0001, USA
Email: andrew.sung@usm.edu

Abstract - *Google+ hangouts is one of the most popular social media where people share information via digital texts, images, video and audio files. While sharing sensitive data such as PII on social media sites, the sensitive data may be intercepted by various third parties through illegal means. Hence there is a need of secret sharing for secure transmission of these data. Taking these parameters into consideration, we develop a Google hangouts application to implement image steganography for the users joining hangouts by hiding data in digital images, providing a real-life application for covert communication on Google Hangouts, one popular social media platform.*

Keywords: Steganography; Google Hangouts; covert communication; image; social media

1 Introduction

In the world of digital technologies, it is very much imaginable that the secret data carrier, was not necessarily an image or Web page source code, but may have been any other file type or organizational unit of data for example, a packet or a frame which occurs in computer networks. However, the process of embedding secret information into an innocent-looking carrier is not some recent invention but was being practiced since ages. This process is called steganography and its origins can be traced back to ancient times. Moreover, its importance has not decreased since its birth.

To protect the security of concerned message data, the information sharing method is developed. A secret message is logically constructed into several shares and they can be distributed to different participants to keep and are rearranged at the receiver to form a meaningful data. To have different participants, these carriers for the shares can be of any digital components like image, video or audio files.

Steganography's applications provide means for conducting clandestine communication. The purpose of establishing such information can fall into the category of legal or illicit activity. Frequently, the illegal aspect of steganography include the criminal communication, through information leakage from guarded systems, cyber weapon exchange. On the other side of the spectrum like legitimate uses, which include surveillance, F5 computer forensics.

The inverse of steganography—steganalysis, detects the covert communication which is started to surface fairly recently, Programs for the embedding of data considerably outnumber those dedicated to the detection and extraction of embedded content.

Images are the most popular cover objects for steganography for given proliferation of digital images in the Internet, and large amount of redundant bits present in the digital representation of an image. This project will focus on hiding information in images. Image steganography techniques can be divided into two groups: Image Domain and Transform Domain. Image domains embed messages in the intensity of the pixels, while for transform domains images are first transformed and then the message is embedded in the image which is more robust making the embedded message survive conversion between loss and lossless compression.

Google Hangouts [3] is a communication platform developed by Google which includes instant messaging, video chat, SMS and VoIP features. It replaces three messaging products that Google had implemented concurrently within its services, including Google Talk, Google+ Messenger, and Hangouts, a video chat system present within Google+, which previously used XMPP protocols that are now replaced by Google's proprietary protocols. Hangouts run natively into many browsers making it evolve into a standards- based cloud video conferencing implemented in client server model to support more than 10 clients to chat at one instance unlike in Skype, which is based on peer to peer network connections. Hangouts use WebRTC interfaces for browser integrations and use VIDYO to facilitate its video chats implemented in H.264/SVC as the primary codec for transmission.

Hangouts provide interface for users to jump into conversations between two or more users with a maximum number of people connecting in a hangout being 10. The service can be accessed online through the Gmail or Google+ websites or through mobile apps available for Android and iOS. Hangouts are distributed as a successor to their existing Google Talk apps by Google. Most third-party applications

which had access to Google Talk do not have access to Google+ Hangouts because these hangouts use proprietary protocol instead of the XMPP open standard protocol previously used in Google Talk.

On the other side, even after support for XMPP has been terminated by Google, the GVJackApp for magicJack and the GVMate Phone Adapter both of which are signaling independent will continue to work for users as normal using the Google Hangouts platform. As Google switched away from the XMPP protocol it used, brought in new challenges like android SMS support in Hangouts doesn't fully integrate with Google Voice for calls or texts. Hangouts work the same way in desktops, mobiles (android and iOS) and so the chat histories are saved online, allowing them to be synced among the multiple devices. Photos can be shared, use color emoji symbols in their messages during conversations. Photos are later uploaded into Google private album. In this paper, we explore the covert communication on Google Hangouts by hiding data in digital images.

2 Proposed method

We explore a new embeddable space with better quality of resulting stego-image and more data hiding capacity, we are using JPEG image's alpha channel plane as cover medium which embeds the secret data to be sent, by combining the advantages of JPEG's high compression rate with the flexibility of having alpha channels like in PNG.

1. JPEG image, which is used as a cover image having set the alpha channel value of each pixel to 255 initially making it a transparent colored one in the beginning of embedding process.

2. Message or the binary data string is transformed into shares using bundles per character based on Shamir's secret sharing method [6], which are then embedded into alpha-channel of JPEG cover image using the HTML5 canvas element.

3. Co-efficient parameters involved in the sharing of data strings are carriers of the hidden data.

4. Prime number used in this method will be taking care of resulting image visual quality and data hiding capacity for the stego-image. Hence the selection of appropriate prime number is very important for quality of transmission.

5. Mapping function is designed in such a way that the alpha-channel values create a uniform transparency for the resulting stego-image.

6. Original image pixels or DCT coefficients are untouched so as to maintain the original image appearance.

A JavaScript coding is used to implement the above technique and embed inside an XML file to be deployed as a Google + hangout application.

2.1 Algorithm for Shamir secret sharing

Input: a secret *d* in the form of an integer, the number *n* of participants, and a threshold *k* not larger than n.

Output: *n* shares in the form of integers for the *n* participants to keep.

Steps:

1. Choose a prime number *p* randomly.

2. Select k - 1 integer values $c_1, c_2, \ldots, c_{k-1}$ within the range of 0 through *p* - 1.

3. Select *n* distinct real values $x_1, x_2, \ldots, x_n$.

4. Use the following (k-1)-degree polynomial to generate *n* equations to compute n function values $f(x_i)$:

$$f(x_i) = (d + c_1 x_i + c_2 x_i^2 + \ldots + c_{k-1} x_i^{k-1}) \bmod p \qquad (1)$$

where i = 1, 2, …, n.

5. Deliver the 2-tuple $(x_i, f(x_i))$ as a share to the i-th participant where i = 1, 2, …, n.

2.2 Algorithm for Shamir secret recovery

Input: *m* shares in the form of $(x_j, f(x_j))$ collected from the *n* participants where 1<= j <= n, k <= m <= n, and *k* is the threshold mentioned in above algorithm.

Output: the secret *d* hidden in the shares.

Steps:

1. Collect any k of the *m* shares, say, $(x_{i1}, f(x_{i1}))$, $(x_{i2}, f(x_{i2}))$, …, $(x_{ik}, f(x_{ik}))$ and use them to set up the following equations:

$$f(x_{ij}) = (d + c_1 x_{ij} + c_2 x_{ij}^2 + \ldots + c_{k-1} x_{ij}^{k-1}) \bmod p \qquad (2)$$

where j = 1, 2, ..., k and 1 <= ij<= n.

$$d = \left(\sum_{j=1}^{k} (-1)^{k-1} \left[f(x_{i_j}) \prod_{m=1(m \neq j)}^{k} \frac{x_{i_m}}{(x_{i_j} - x_{i_m})} \right] \right)_{\bmod p} \qquad (3)$$

2.3 Data embedding algorithm

We utilized the methods in the references [1, 2], described below.

Input: a cover JPEG image *I* and a secret message *M* in the form of a binary data string.

Output: a stego-image *I′* in the JPEG format.

Steps:

1. (Initialization) Divide *M* into t-bit segments with t = 3 and transform each segment into a decimal number, resulting in a decimal-number sequence $M' = d_1\ d_2\ d_3 \ldots$ where $0 <= d_i <= 7$.

2. (Beginning of Looping) Take the first four elements from *M′* as m_1, m_2, m_3, and m_4, starting from the beginning of *M′*.

3. (Partial share creation) Set p, c_i, and x_i in Eqs. (1) of Shamir's Algorithm to be the following values:

(a) $p = 11$ (the smallest prime number larger than 7);

(b) $d = m_1$, $c_1 = m_2$, $c_2 = m_3$, and $c_3 = m_4$;

(c) $x_1 = 1$, $x_2 = 2$, $x_3 = 3$, and $x_4 = 4$, resulting in the following equations:

$$q_1 = f(x_1) = (m_1 + m_2x_1 + m_3x_1^2 + m_4x_1^3) \bmod p,$$

$$q_2 = f(x_2) = (m_1 + m_2x_2 + m_3x_2^2 + m_4x_2^3) \bmod p,$$

$$q_3 = f(x_3) = (m_1 + m_2x_3 + m_3x_3^2 + m_4x_3^3) \bmod p,$$

$$q_4 = f(x_4) = (m_1 + m_2x_4 + m_3x_4^2 + m_4x_4^3) \bmod p.$$

4. (Mapping of partial share values) Add 245 to each of q_1 through q_4 to form q_1', q_2', q_3', and q_4', respectively.

5. (Data embedding) Embed q_1' through q_4' into the alpha-channel plane of I in the following way.

i) Take in a raster-scan order four unprocessed pixels of I and set their alpha-channel values to be q_1' through q_4', respectively.

ii) Remove m_i through m_{i+3} from *M′*.

6. (End of looping) If *M′* is not empty, then go to Step 2 to process the next four decimal numbers in M′; otherwise, take the final I as the desired stego-image I'.

The above algorithm is a (4, 4)-threshold secret sharing method. As the prime value *p* is 11, the possible values of q_1 through q_4 in Step 3 of the above algorithm are between 0 and 10 which are inserted in the alpha channels of I. The values of q1' through q4' form a small range of integers from 245 to 255 which are then embedded into the alpha channels of the cover image I. As the distribution of alpha channel pixels of the image are mostly the similar values, which makes a nearly uniform transparent image, making the intruder not to suspect the image transmission in the network.

To calculate the hiding capacity of algorithm, we know every four 3-bit segments of the secret data string are embedded into the alpha-channel values of four pixels of the cover image I to yield the stego-image I' which means that if the size of the cover image is S, then the data hiding capacity is

$$R = (4*t)*(S/4) = ts \text{ bits.}$$

Above relation shows that hiding capacity is directly proportional to the value t, larger values of t produce larger q1', q2', q3', q4' values which are beyond 255, thereby reducing the picture quality and creating a suspicion to the intruders.

2.4 Data extraction algorithm

Input: a stego-image I' created by embedding algorithm in the JPEG format.

Output: the binary data string M hidden in I'.

Steps:

1. (Initialization) Create an empty string M.

2. (Beginning of looping) Take in a raster-scan order four alpha-channel values q_1', q_2', q_3', and q_4', from I'.

3. Subtract 245 from each of q_1' through q_4' to obtain q_1 through q_4, respectively. Perform the secret recovery process described by Shamir's data extraction algorithm to extract the values m_1 through m_4 of the decimal format hidden in q_1 through q_4.

4. Transform the extracted values of m_1 through m_4 into binary bits and append in order each of them to the end of M.

5. (End of looping) If all shares embedded in I′ are processed, then take the final M as output; otherwise, go to Step 2.

3 Covert communication on Google hangouts

3.1 Create a steganography hangout App

The Hangouts API enables us to develop collaborative applications that run inside of a Google+ Hangout [3].

The following steps show how a hangouts host and run a pre-built steganography application.

1) Add the stegoApp.xml(XML file of any application) file on a server so that it is publicly available. The server should have no firewalls and require no login authentication to access this file.

2) Login to gmail account and go to the Google Developers Console .

3) Create a new project by clicking Create Project - steganoApp:

4) In the Project name field, type in a name for our project -steganoApp.

5) In the Project ID field, optionally type in a project ID for your project or use the one that the console has created for us. This ID must be unique world-wide.

6) Click the Create button and wait for the project to be created.

7) Click on the new project name in the list to start editing the project.

8) In the left sidebar, select the APIs item below "APIs & auth". A list of Google web services appears.

9) Find the Google+ Hangouts API service and set its status to ON.

10) In the sidebar under "APIs & auth", select Consent screen.

11) Choose an Email Address and specify a Product Name.

To the right of the Google+ Hangouts API service name, click on the gear icon. In the Application URL field, enter the URL where you published your Hangout gadget XML file. Click Save.

3.2 Developing application in Google Hangouts

To develop a steganography application which can be used by participants in a googlehangout can share messages only through Google Server. So there should be API's which google should support the communication between the user application and the google server. Since our application is a browser deployed application, we used JavaScript client library to make API requests to interact with Google Services [4].

The general operations to make an API request using the JavaScript client library are described below.

1. The SteganoApp loads the JavaScript client library.

2. The SteganoApp references its API key, which authenticates the SteganoApp with Google services.

3. If the SteganoApp opens a session with a Google auth server. The auth server opens a dialog box which prompts the user to authorize the use of personal information.

4. The SteganoApp loads the API for the Google service.

5. The SteganoApp initializes a request object (also called a service object) that specifies the data to be returned by the API.

6. The SteganoApp executes the request and processes the data returned by the API.

As a client side application, whole application is developed using Javascript. We can use java script to add Google service with a part of code shown below.

```
<script src="//plus.google.com/hangouts/_/api/v1/hangout.js">
</script>
<script>
function showParticipants()
{
        var participants = gapi.hangout.getParticipants();
        var retVal = '<p>Participants: </p><ul>';
        for (var index in participants)
        {
          var participant = participants[index];
          if (!participant.person)
          {
            retVal += '<li>A participant not running this app</li>';
          }
          retVal += '<li>' + participant.person.displayName + '</li>';
        }
        retVal += '</ul>';
        var div = document.getElementById('participantsDiv');
        div.innerHTML = retVal;
}
</script>
```

Here "hangout.js" is the file that imports all the required Google Services API methods in our application. In the above code, we can see gapi.hangout.getParticipants() is used to show different participants who joined the call.

Accessing Google API's is through the Google APIs Console. The steps are listed below:

- Login and visit the Google APIs Console.
- Select Services from the menu for the steganoApp project. The list of accessible Google services appears, enable Google Hangouts' API ON.
- The API access pane appears. For authorized access as we need to run the application between different participants, we should continue as below.
- Click Create an OAuth 2.0 client ID.
- The Create Client ID dialog appears.
- Click the Web application radio button to create a client ID. The Authorized API Access section now displays our steganoApp's OAuth 2.0 credentials [5].

This client ID created is used in the application as follows:

```
gapi.hangout.onApiReady.add(
function(eventObj) {
if (eventObj.isApiReady) {
document.getElementById('showParticipants')
.style.visibility = 'visible';
}
});

var clientId = '989443731264-
dvm1t58ousjfr080sjvksj5jah3jcs4q.apps.googleusercontent.com';
var apiKey = 'AIzaSyDxVGV0-
KTxjNFQ3cASXPeVmC7gP6dyI98';
var scopes =
'https://www.googleapis.com/auth/hangout.participants';
function handleClientLoad() {
gapi.client.setApiKey(apiKey);
window.setTimeout(checkAuth,1);
}

function checkAuth() {
gapi.auth.authorize({client_id: clientId, scope: scopes, immediate:
true}, handleAuthResult);
}
```

```
function handleAuthResult(authResult) {
}
}
```

3.3 Sending and receiving

When participants run steganoApp in a Hangout, they are each running the app in their own separate in-stance of the Hangout client. Hangouts have two major data channels for sharing application-specific data be-tween these instances. They are shared state and sendMessage.

sendMessage( ): When used sendMessage( ), there were problems at the receiver end and then started using Shared State among the participants because sendMessage( ) sends a message to the other application participants. Messages are not retained or stored, and should have lower latency than objects stored via sub-mitDelta method in the API. These messages might be lost, so this method should only be used to send things that can be dropped.

Shared State( ): There is only one shared state object per Hangout. The shared state object contains data that is kept up-to-date with every instance of the Hangout client that is running our steganoApp. The object is a regular JavaScript object with paired key/value strings. This object is our stegano-message sliced into 7000 characters per message with a sequence number. This key/value pair of sequence number and stegano-message string will be distributed to every instance of the Hangout client that is running our application, i.e., all participants of hangout call.

3.4 Encoding and decoding

To decrease the number of HTTP requests to server by decreasing the resources and increasing the overall performance of the website, this application uses base64 data stream of image within the code.

Base64 converts binary data into ASCII string format by translating it into a radix-64 representation .The base64 technique generates ASCII representation for binary data of image file and browser can parse these ASCII values and render an image on the page which makes the application lighter than the cost of an additional external resource link to an image.

4 Experiments

The results of applying the data embedding method used to embed a long sequence of message data into the two images are shown in Figure 1. The original image and the steganogram are visually identical.

Table 1 shows the data hiding capacity and the PSNR values between the cover image and the steganogram. It shows the large capacity with very little distortion to original image.

5 Conclusions

A covert communication on Google Hangouts has been established via image steganography carrying the secret information with the use of Shamir's secret sharing method. The alpha-channel plane of a cover JPEG image is created and utilized to embed the partial shares, resulting in a stego-image with undesirable white noise. The white noise is then eliminated by choosing a small prime number, dividing the input data string into 3-bit segments, and mapping computed share values into a range of alpha-channel values near their maximum value of 255. Deploying this in Google + hangout with two participants were tested, which can be extended to a maximum of 10 people. It is a novel way of sending sensitive information using steganography methods in social media.

(a) Original image (b) steganogram

Figure 1. An example of original image and steganogram

Table 1. Data hiding capacity and PSNR values

t value	DHC (bits) [R= t*size]	PSNR
1	62217	78.80
2	124434	70.15
3	186651	67.48

This project gives a large scope to develop into a large scale commercial software product, thereby enhancing the UI design for more interactive interface for user to do dynamic selection of pictures and also test for other safe possible areas in an image file for data hiding.

Steganography can also be implemented in audio, video sharing files within the hangout but should request more time to processing the data. A further study may be focused on more secure steganographic systems [7, 8] and the covert communication may be expanded to other social media platforms, such as Reddits, Facebook, etc. Additionally, the

advanced steganalysis methods [9, 10, 11] will be conducted to examine the detectability of the steganography.

6 Acknowledgements

Part support for this study under the NSF award No. 1318688 and the support from the SHSU Office of Research and Sponsored Programs are highly appreciated.

7 References

1. C. W. Lee and W. H. Tsai, A secret-sharing-based method for authentication of grayscale document images via the use of the PNG image with a data repair capability, IEEE Trans. on Image Process. 21 (1) (2012) 207-218.
2. C-W. Lee and W-H Tsai, A data hiding method based on information sharing via PNG images for applications of color image authentication and metadata embedding, Signal Processing, 93(7) (2013) 2010-2025.
3. https://developers.google.com/+/hangouts/getting-started
4. https://developers.google.com/api-client-library/javascript/start/start-js
5. https://developers.google.com/api-client-library/javascript/features/authentication
6. A. Shamir, How to share a secret, *Communication of the ACM*, vol. 22, pp. 612-613, 1979.
7. Q. Liu, A. H. Sung, Z. Chen and X. Huang. A JPEG-based statistically invisible steganography. *Proc. 3rd International Conference on Internet Multimedia Computing and Service,* pages 78-81, 2011.
8. J. Fridrich and J. Kodovsky. Multivariate Gaussian model for designing additive distortion for steganography. *Proceedings of 2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (ICASSP), pages 2949-2953.
9. M. Goljan and J. Fridrich. CFA-aware features for steganalysis of color images, *Proc. SPIE 9409,* Media Watermarking, Security, and Forensics 2015, 94090V (March 4, 2015); doi:10.1117/12.207839
10. Q. Liu, A. H., Sung and M. Qiao. Neighboring joint density-based JPEG steganalysis. *ACM Trans. Intelligent Systems and Technology*, 2(2):16. 2011.
11. Q. Liu and Z. Chen, Improved approaches with calibrated neighboring joint density to steganalysis and seam-carved forgery detection in JPEG images. *ACM Trans. Intelligent Systems and Technology*, 5(4):63, 2014.

Development of an Intelligent Audit Trail as a Watch Dog To Monitor Academic Activities in Nigerian Universities

Aru Okereke Eze , Ihekweaba Gozie and Ihekweaba Linda Ogechi
Department of Computer Engineering
Michael Okpara University of Agriculture, Umudike, Umuahia, Abia State, Nigeria

ABSTRACT

The goal of this project is to develop an Audit Trail for automatically monitor students' course results in Nigerian Universities. The Automated Result Software which is being implemented in schools to automate the manual process of student result computation for both first and second semester, course registration and student individual result requires a security tool to keep track of "who" did "what" to "which" data "when" and "how", thereby preventing vulnerability and providing a more complete trace recording of user access and user actions. The Audit Trail System is a real-time web based application that keeps record of logins, activity logs and logout. It captures the user's username, password, time of login and logout, the web browser used and every activity carried out on the Automated Result Software. Microsoft Visual Studio Integrated development provided a single platform that combines powerful tools such as ASP.Net for web development, C# language, Cascading Style Sheet (CSS), HTML and MSSQL as the database backend to build the Audit Trail System application. The application was tested with the Automated Result Software in the Computer Engineering, Michael Okpara University of Agriculture, Umudike, Umuahia, Abia State-Nigeria and it yielded accurate result.

Keywords: Data, Monitor, Security, Password, Activity Log, etc

1.0 INTRODUCTION

For optimal security in Students Data acquisition and processing, an audit trail is needed for managing the logistics facilities. The Audit Trail captures every activity carried out in the Automated Result Software. It analyzes and keeps records of events to provide information about system use and performance in a clear and understandable manner. The goal of an Audit Trail System is to be able to determine if security and other policies are being violated. These policies include general security requirements on student data such as change on the student name and grades. Given an audit log of "who" did "what" to "which" data "when" and "how" and an effective means of processing the log, Audit Trails can answer "why". By answering the question of "why", it provides one of the means to detect intrusions into the system, including privileged users. Another use of the Audit Trail is to provide periodic report of system usage and data modifications.
The Audit Trail System is designed in a way to captures the following:

- Username
- Password
- Login time
- The web browser used
- Successful and failed attempts
- Logout time and
- Every activity log carried out on the Automated Result Software.

It is a real time system which monitors changes as they occur. It is important to note that the Audit Trail System is always active weather it is on the web browser or not, so as to increase individual accountability and deter the user from circumventing security policies. Even if they do, they can be held accountable.
The Audit Trail system for the Automated Result Software allows administrator to investigate suspicious activities carried out by either an authorized or unauthorized user.

1.1 STATEMENT OF PROBLEM

This Research focuses on monitoring every activity carried out on the Automated Result Software. The old system was unable to account for users' actions, notify the administrator of the action of both authorized and unauthorized users, detect problems with an authorization or access control implementation, monitor and gather data about changes made such as who logged into the system, at what time and the grades that were changed. To solve this problem an Audit Trail System is designed to tackle the problem

1.2 OBJECTIVE OF THE STUDY

The main aim of this project research work is to study the existing Automated Result Software in use in the Nigerian Universities and develop an Audit Trail that will serve as a security tool to checkmate vulnerability on the system.

2.0 RELATED WORKS

Prior to regulatory mandates, many IT departments secured corporate data by restricting access to enterprise information to a few privileged users, such as systems and database administrators. However, due to the increasing amounts of data that organizations must manage and protect daily, role-based access alone does not help ensure the security of confidential information. Furthermore, because information may be used in a malicious way, tracking the activities of all users is vital for effective compliance. As a result, many organizations are incorporating Automated Data Audit processes.

Despite the electronic revolution in data processing and information gathering activities, data auditing is essential for regulatory compliance, because it provides a continuous and permanent audit trail of data access and changes, while storing this information in a centralized repository that can be archived easily for long-term retention. The information gathered by a data audit solution can help improve an organization's operational performance by identifying data-use patterns leading to increased IT efficiency and the fine-tuning of existing processes.

Some previous work has actually been carried out in this area which has contributed to the Audit Trail system. Two of the works are audit trail developed for an online examination system and the audit trail developed for result processing system. In online examination system developed, PHP, MySql, CSS and HTML was employs, this is because the following technologies has the advantage of easy development, flexibility and it has the ability of providing the developer or the programmer with possible hints and it a graphical user interface. The system ensures the end-to-end integrity of data activities by identifying when modifications are made, detect and analyze intentional and accidental breaches in user and application behavior. Monitor and provide alerts on the database activities of privileged users that occur outside the application's controls and security measures, Validate policies and controls to protect sensitive data, while monitoring the effectiveness of these polices and controls continually keep track of changes and updates within the firewall. In the system the duties of the administrator is separated from the users to ensure that audit data is not manipulated by those with privileged access to data sources and was made to audit all privileged users to track data access and the activities of those with extended data manipulation capabilities.

The limitation of the system is that it is not real time base. All activities carried out when the audit trail is not on the server are not captured; this exposes the online examination system to vulnerability attack making it possible for either an authorized or unauthorized user to change the grade of a student or possibly change the questions online without being noticed. It does not have the capacity to truncate activity log, logins and logouts. Audit trail captures activities carried out; at a time the display environment becomes clustered, especially the activity log making it difficult to quickly identify instant changes captured by the system. The system does not keep record of logouts. In the case of suspicious activities, the time when the user logged out and username used to log out is not known by the administrator.

To eliminate the problems inherent in the reviewed work the new system captures every activity carried on the automated result software even when it is not the server; the system is a real time base system. The new system has the capability to be truncated by the administrator, and keeps record of logouts details.

The audit trail system developed for the result processing system is similar to the one developed for online examination but has the capability to enable audit trail to capture data definition language (DDL) and data manipulation language (DML) changes, as well as Select statements. Audits of DDL changes should track information on schema changes (i.e., database structure changes, including tables, columns, and their interrelationships); new permissions; successful and failed login attempts; any new data or transaction log devices created; backup executions; and other activity a privileged user conducts on the database. DML allows users to insert, update, or query information from a database and contains features that ease report generation, including the ability to perform simple arithmetic, financial, and statistical calculations. Automated audit solutions should capture all DML changes, including the before and after values of typical DML statements such as insert, update, and delete. Automated solutions also should capture, where appropriate, the value of the row before the data change, the value after the data change, the identity of the individual who executed the change, the date and time of the activity, and the user login username.

The limitation of the system is that individual user activity log cannot be queried. To take cognizant of activity log of a particular user becomes difficult and time consuming. The interface of the system looks clustered. The username, login time, log out time and activity log are combined together. It does not have the capability to query the activities carried out within a time range. As the log approaches its maximum size, it can either overwrite old events or stop logging new events. This makes it susceptible to attacks in which an intruder can flood the log by generating a large number of new events. The system does not have the capability to retrieve the previous grades of a student. The trace does not record old data after it is changed, if an agent or transaction accesses a table more than once in a single unit of recovery, the audit trail records only the first access.

Therefore, this work attempts to eliminate the difficulties and problems in the reviewed work, and achieved that central objective of security and enhancement, cum ease of query of individual user activity. The new system has a neat graphical user interface and has the capability to retrieve previous grades of a particular student to ascertain if changes were made.

2.1 BASIC ARCHITECTURE OF AUDITING TRAIL SYSTEM

The basic architecture of an auditing system is:

Figure 2.1 Anatomy of Auditing System

The logger records information. The information that is stored in the log is determined by the security and the system capabilities. The analyzer takes the log as the input and analyzes it to either determine if an event of interest or a problem has occurred, or if other information needs to be logged. The notifier receives the analysis from the analyzer, and reports the result of the audit to the analyst, auditor, or other entities such as GUI form. In the following sections we will describe features of each component in detail and provide a framework to classify database auditing systems.

2.1.1 LOGGER

Some important issues related to logging include what, when, where, how, and how often a database was logged. More customized auditing procedures can be written for table-oriented logging systems. The mechanisms used to set the type of event and condition to be monitored address the "how" question of logging. An entry in a log must contain sufficient information to find the consequence of a certain action.

2.1.2 ANALYZER

According to Orman (1997) in analyzing there are two major issues: what to analyze for and when to analyze. The question, "what to analyze" considers the goals of auditing, which are usually related to the detection of security violations. Regarding "when to analyze", the analysis can be periodic, based on transaction counting, and/or occur in real-time.

2.1.3 NOTIFIER

There are two issues in the notification stage that are of primary interest. The first is the audit browsing techniques available. The second is the ability of the notification system to sanitize the data based on the policy and the viewer's privilege level. The goal of the audit browsing tools is to be able to present information in a way that it is easy for the security administrators and auditors to identify potential threats or violations of policy. Before a user is allowed to view the log or result of analysis, information that is not relevant or information level that is higher than user's allowed access, is removed. In the context of database auditing, it is salient that users be restricted to search and view entries of a narrow scope so no user is allowed to access all entries.

3.0 DESIGN OF THE CURRENT SYSTEM

System design is the solution to the creation of a new system. This phase focuses on the detailed implementation of the feasible system. System design has two phases of development; logical and physical design.

During logical design phase the inputs (sources), out puts (destinations), databases and procedures (data flows) are describe all in a format that meets the users requirements. The phase also specifies the user needs and at a level that virtually determines the information flow into and out of the system and the data resources. Here the logical design is done through data flow diagrams and database design.

The physical design is followed by physical design or coding. Physical design produces the working systems by defining the design specifications, here the written programs accept input from the user, perform necessary processing on accepted data through call and produce the required report on a hard copy (printed) or display it on the screen.

3.1 LOGICAL DESIGN

Logical design of the system shows the major features and also how they are related to one another. The first step of the system design is to design logical design elements. This is the most creative and challenging phase and important too. Design of proposed system produces the details of how the system will meet the requirements identified during the system analysis that is, in the design phase we have to find how to solve the difficulties faced by the existing system. The logical design of the proposed system should include the details that contain how the solutions can be implemented. It also specifies how the database is to be built for storing and retrieving data, what kind of reports are to be created and what are the inputs to be given to the system.

3.2 INPUT DESIGN

Fig. 3.1 System Login

The input design is the link between the system and the administrator. It comprises the developing specification and procedures for data capturing by the Audit Trail. The design of input focuses on controlling the amount of input required, controlling errors, avoiding delay, avoiding extra steps and keeping the process simple.
One effective prevention attacks is due diligence on the part of the web programmer. All text entered into input fields is validated and stripped of malicious content before it is sent to the database. This is an example of the "Secure by Design" philosophy. A white list of characters where created so that only the administrator has access to the Audit Trail. The MinPasswordLen property sets the minimum length of a user password which is a means of securing the input page to the Audit Trail system. By setting the minimum password length you can prevent using too short passwords, therefore lowering the possibility of a password to be guessed by an intruder. See the authorization input design flow chart below.

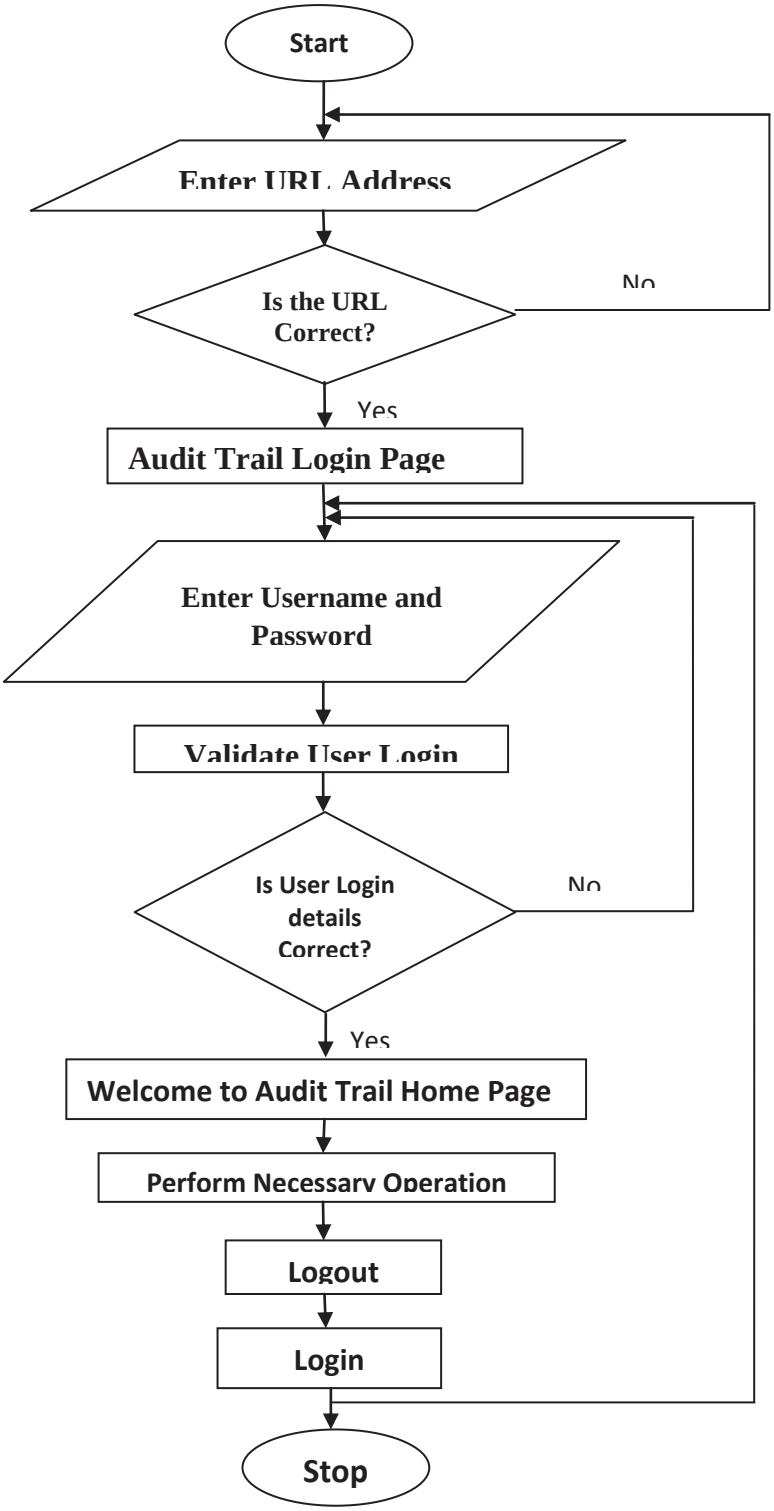


Figure 3.2 System Login Authentication Flowchart

3.3 OUTPUT DESIGN

Computer output is the most important and direct information source to the user. Output design is a process that involves designing necessary outputs in the form of reports that should be given to the users according to the requirements. Efficient, intelligible output design should improve the system's relationship with the user and help in decision making. Since the reports are directly referred by the management for taking decisions and to draw conclusions they must be designed with almost care and the details in the reports must be simple, descriptive and clear to the user. So while designing output the following things are to be considered.

- Determine what information to present
- Arrange the presentation of information in an acceptable format

Depending on the nature and future use of output required, they can be displayed on the monitor for immediate need and for obtaining the hardcopy via this reason a print button is added.

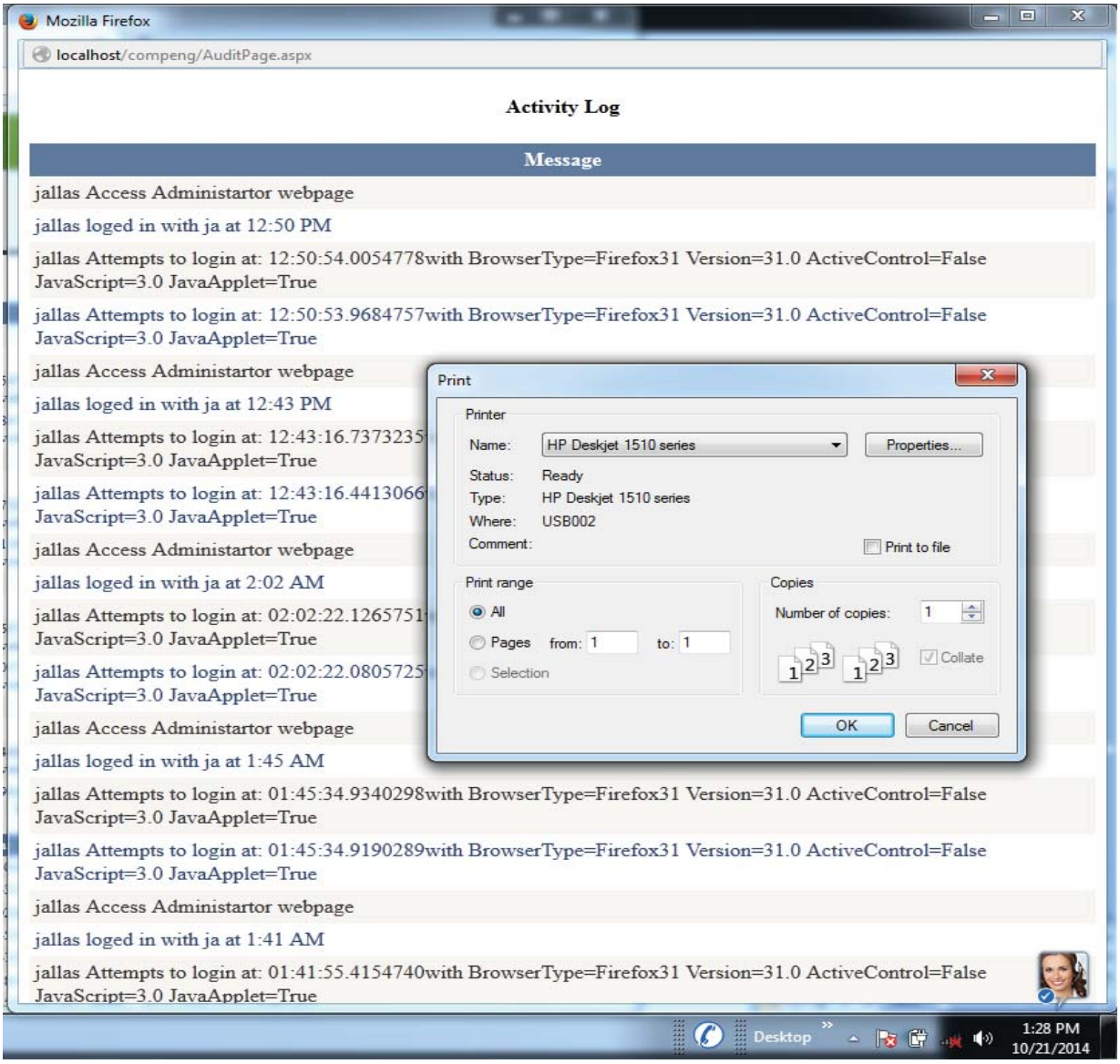


Fig 3.3: Activity Log for a particular user

3.4 PHYSICAL DESIGN

The process of developing the program software is referred to as physical design. We have to design the process by identifying reports and the other outputs the system will produce. Coding the program for each module with its logic is performed in this step. Coding is done at this stage using ASP.Net programming language to coordinate the data flow and control. Proper software specification is also done in this step.

4.0 CONCLUSION

This work shows the use of Audit Trail System as a security watch dog to monitor every activity carried out on the Automated Result Software; thereby increasing the efficiency of the existing system and helps to minimize error in terms of student's data sorting, storage and processing. The Audit Trail system will do the following:

- Enable accountability for actions. These include actions taken in a particular schema, table, or row, or affecting specific content.
- Investigate suspicious activity. For example, if a user is deleting data from tables, then a security administrator can audit all connections to the database and all successful and unsuccessful deletions of rows from all tables in the database.
- Notify an auditor of the actions of an unauthorized user. For example, an unauthorized user could be changing or deleting data, or the user has more privileges than expected, which can lead to reassessing user authorizations.
- Monitor and gather data about specific database activities. For example, the database administrator can gather statistics about which tables are being updated, how many logical I/Os are performed, or how many concurrent users connect at peak times.

This project is designed using ASP.NET platform which offers the functionality of Server-side programming. Its interface is designed with rich Cascading Style Sheet (CSS) for aesthetics in conjunction with HTML controls and tags to offer the rich web interface designs. Also C#.NET programming language is used to make the WebApp dynamic. Microsoft SQL is used at the backend to dynamically manage and control data flow. System test was performed using a test data from the available result and it yielded accurate and reliable results with ease base on the system requirement specification.

REFERENCES

Abel D. J., (1998): Relational Data Management Facilities for Spatial Information Systems. Proceedings of the 3rd International Symposium on Spatial Data Handling International Geographical Union. Columbus Ohio, page 9-18.

Abel D. J., Smith J. L., (1986): A Relational Database Accommodating Independent Partitions of the Region. Proceedings of the 2nd International Symposium on Spatial Data Handling. International Geographical Union. Columbus Ohio, page 23-24.

Anil L. Pereira, VineelaMuppavarapu and Soon M. Chung. (2006): Role-Based Access Control for Grid Database Services Using the Community Authorization Servicel, IEEE Transactions on Dependable and Secure Computing, Vol. 3, No. 2.

Bertino, E., Byun, J., & Kamra, A. (2007). Database security. In M. Petkovic & W. Jonker (Eds.), *security, privacy, and trust in modern data management (Data-centric systems and applications)* (pp. 87-102).

Bertino, E., & Sandhu, R. (2005). Database Security—Concepts, Approaches, and Challenges. *IEEE Transactions on Dependable and Secure Computing,* Vol. 2, No. 1. Page 2-18.

Bishop A. (2003): Anatomy of Auditing System.

Broadbridge, A (1996). "Academic advising--traditional or developmental approaches?". British Journal Of Guidance &Counselling 24 (1): 97–111.

Chen P. (1976): The Entity-Relational Model – Towards A Unified View Of Data. Association for computing machinery transaction on database systems, page 9-36.

Crookston, B.B. (2009). "A Developmental View of Academic Advising as Teaching". NACADA Journal 29 (1): 78–82.

Clark D. M., Hastings D A, Kineman J. J. (1991): Global Database and Their Implications.

Codd E. F., (1970): A Relational Model of Data for Large Shared Data Banks. Communications of the Association for computing machinery, page 377 – 87

Codd E. F. (1979): Extending the Database Relational Model To Capture More Meaning. Association for Computing Machinery Transactions on Database Systems, page 397-434.

Ebube, A. M.(2011). Microcomputer Based Result Processing System, Unpublished Degree Project, Michael Okpara University of Agriculture, Umudike.

Results Processing,Journal of Information Engineering and Applications, ISSN 2224-5782 (print) ISSN 2225-0506 (online),Vol. 2, No.11, 2012. Retrieved online from http:// www.iiste.org.

Frank R. (1979): Requirements for Database Systems Suitable To Manage Large Spatial Database. Proceedings of The 1st International Symposium On Spatial Data Handling, Volume 1, Page 38-60

Frank A U. (1988): Requirements for A Database Management System Based On Linear Quadrees And A Relational Database For Regional Analysis. Page 213-32.

Ugochukwu Ajalla (2014) Design and Implementation of a Result Processing System. Unpublished Degree Project.

Feikis John (1999): Database Security: IEEE Journals.

Guimaraes, M. (2006). New challenges in teaching database security. *Proceedings of the 3rd Annual Conference on Information Security Curriculum Development*, Kennesaw, GA, USA, page 64-67.

Igbajah Abraham (2014) Design and Implementation of an Online Examination System Unpublished Degree Project.

Knox, D. C. (2004). *Effective Oracle database 10g security by design*. New York: McGraw-Hill/Osborne.

Marius ConstantinLeahu, Marc Dobson, and Giuseppe Avolio. (2008): Access Control Design and Implementations in the ATLAS Experiment‖, IEEE Transactions on Nuclear Science, Vol. 55, No. 1.

Ravi S. Sandhu, Edward J. Cope , Hal L. Feinstein, , Charles E. Youman. (1996): Roll Based Access Control Models‖, IEEE Journals.

Yang, L. 2009. Teaching database security and auditing. *Proceedings of the 40th ACM Technical Symposium on Computer Science Education*, Chattanooga, TN, USA.

Jeschke, M.; Johnson, K.E., & Williams, J.R. (2001). "A Comparison of Intrusive and Prescriptive Advising of Psychology Majors at an Urban Comprehensive University.". NACADA Journal 21 (1/2): 46–58.

Authors' Profile

Okereke Eze Aru is a lecturer in the Department of Computer Engineering, Michael Okpara University of Agriculture, Umuahia, Abia State, Nigeria. His research Interests include Computational Intelligence , Security system design, Expert systems, Design of Microcontroller and Microprocessor based system, Electronic and Communication Systems and other computer related subjects. Email: okezearu@yahoo.com

Dr. Ihekweaba, Gozie is an Associate Professor and currently the Head, Department of Computer Engineering, Michael Okpara University of Agriculture, Umuahia, Abia State, Nigeria. His research interests include Computer Hardware design and maintenance, Security system design, Electronic and communication systems, etc.

Dr. Mrs. Linda Ogechi Ihekweaba is an lecturer in Department of Computer Engineering, Michael Okpara University of Agriculture, Umuahia, Abia State, Nigeria. Her research interests include Computational Intelligence , Security system design, Expert systems, Design of Microcontroller and Microprocessor based system, Electronic and Communication Systems and other computer, etc.

Private Lives or Public Property?: The impact of the Internet on data security and privacy in the European Union

J. Bishop

Centre for Research into Online Communities and E-Learning Systems
Ty Morgannwg, PO Box 674, Swansea, SA1 9NN

Abstract – The biggest story in the newspapers of 2012 probably made it into the Leveson Inquiry. This celebrity infested public inquiry intended to be the basis on which the press would be reformed to perform its role as information sources that scrutinise those with power more effectively. This chapter explores the role that European Union law in the areas of property and privacy has on the way the media operates. This is achieved through exploring the issues surrounding the British Royal Family, where such issues came to the forefront following the exposure of explicit photographs of the Duke and Duchess of Cambridge, William Wales and Kate Middleton, and also those of Harry Wales.

1 Introduction

With the publication of the Leveson Inquiry [1], hereinafter referred to as 'Leveson,' issues over media freedom have come to the forefront, including these cases which were looked at by the inquiry in detail. The two polarised fronts are whether there should be state regulation, or whether there should be self-regulation. The third way of self-regulation backed by statute already exists as discussed. That is that the editor's code is enforceable through Section 12 of the Human Rights Act 1998. There therefore needs to be a fourth way that uses the best of both worlds.

The Leveson Inquiry into the culture, practice and ethics of the British press was triggered when the phone-hacking scandal's full scale became clear in July 2011 and closed the News of the World [2]. The Leveson Inquiry was formed in response to the 2011 phone-hacking scandal, and has opened up a dialogue on the culture, practices, and ethics of the British press [3]. Leveson did not however investigate the distribution of images of celebrities illegally obtained by hacking, only the acquisition of answerphone messages through the cracking of celebrity telephones. Three years following Leveson, the iCloud accounts of celebrities were hacked, leading to the illegal distribution of private photographs. Apple was forced to release a statement on the issue. "*We have discovered that certain celebrity accounts were compromised by a very targeted attack on user names, passwords and security questions, a practice that has become all too common on the internet*," they said. "*We are continuing to work with law enforcement to help identify the criminals involved."*

2 Background

Leveson highlights some of the factors that should be considered in trying to balance the rights of privacy with the rights to freedom of expression. The report discusses how some celebrities want extended rights beyond what ordinary members of the public would have. This is no truer than in the British Royal Family. Harry Wales complained about photographs of him being made public as did Kate Middleton about a photograph of her on a balcony being published. Indeed it has been argued that the impacts of Leveson in relation to photographs of "younger members" of the Royal Family taken in France have been recommended an inspection [4]. Furthermore it has been argued that it is important to consider the links between cyberstalking and the growing number of acts of cyberbullying, including by so-called 'Internet trolls' [5]. Table 1 presents a model of understanding this, drawn from the literature [6], which has been adapted to reflect the circumstances of the British Royal Family.

Table 1 The four types of Internet trolling

Type of Internet trolling	Description and examples
Cyber-bantering (Cyber-trolling)	This is tongue in cheek commentary. It may include satirical images online, light-hearted jokes. In the case of the Royals it could be a photograph of the Queen in an unlikely situation or with an unlikely caption, as one will see on satirical television programmes, like Have I Got News For You.
Cyber-trickery (Cyber-trolling)	This type of trolling is where someone becomes a bit more tactical and needs some effort to pull the trolling off. In the case of the Royal Family this can be seen to include the telephone hoaxing they received.
Cyber-bullying (Cyber-stalking)	Cyber-bullying reflects the more strategic side of trolling, where people go out of their way to target an individual in such a way to cause them

	discomfort. This can include the cases of Harry Wales and Kate Middleton, who had photographs of them exposed through the media which they didn't want.
Cyber-hickery (Cyber-stalking)	Those who engage in Cyber-hickery are the most persistent and determined of trollers. They will follow someone around the Internet – stalking, and in the case of the paparazzi, they will pursue celebrities to get photographs often shared online. Had the Diana Spencer tragedy occurred today, then it is likely images of her would be posted on the Internet and accessible by all.

The British Royal Family as an institution has been the subject of media attention since both co-existed. Whilst in the past one might have been beheaded for representing the Royals in a way they disapproved of, today it has become the norm for those Royals to use the courts to force the censorship of unfavourable content. The Royal Family has been the subject of a number of so-called media intrusions. This includes the Princess Diana tragedy in 1997, The Telephone Hoax Affairs in 1995 and 2012, and the Chatroom Bob Fiascos of 2012 and 2013.

2.1 The Princess Diana tragedy (1997)

Diana Spencer, known as Princess Diana, was the first wife of Charles Windsor, the Prince of Wales, and died in Paris as the result of a car crash in August 1997. This resulted in large numbers of British people being drawn to their churches to make some sort of gesture to the late media icon [7]. During this time the British Royal Family were put under significant pressure due to their stance of wanting to grieve in private, of which this stance was pathologised and denounced as cold and inhuman by the media and public [8]. This would be even more difficult today, as the Internet has changed the way people mourn those who have died [9].

2.2 The Telephone Hoax Trolling Affairs (1995, 2012)

The Royal Family was subject to two hoaxes over the telephone. As the telephone is a public communications network then this can be considered to be Internet trolling. The Internet troll has traditionally been a trickster who posts humorous comments online in order to provoke a reaction. Humans, as intelligent beings, seem to need humour in order to get by in a harsh and cruel world. The Royal Family as public figures should expect to be trolled, on Twitter or Facebook, but one might question the ethics of hoax phone calls when the telephone is still seen as part of the privacy of one's home. The first prank call to hit the Royals was in 1995. Canadian radio broadcaster Pierre Brassard, telephoned The Queen's residence, Buckingham Palace, pretending to be Jean Chretien, who was the Canadian Prime Minister at the time. The DJ spoke to the head of the House of Windsor, Elizabeth II, for about 45 minutes, including about the forthcoming referendum on Quebec. The second one was within weeks of the Leveson Report being published when Kate Middleton was at a private hospital being treated for sickness relating to her pregnancy. The radio broadcasters, Mel Greig and Michael Christian, of the Australian station 2Day FM, managed to get through to the nurse on the Duchess of Cambridge's ward and gulled them into revealing personal information about the princess's condition.

2.3 The chatroom bob fiascos (2012-2013)

In 2012, there was mass publication of images of two members of the British Royal Family who were photographed completely unclothed. These included Harry Wales, who is the second son of Charles Windsor and also Kate Middleton, the wife of William Wales, the Duke of Cambridge. A further incident occurred in 2013, where Middleton was photographed on a public beach where she was visibly pregnant.

Chatroom bobs are traditionally known to be defined as; 'a nickname girls give to the kind of guy who uses the Internet primarily to hang out in chat rooms and search for photos of naked women. If he finds a pretty girl's Web site, he will send flirty e-mail messages ad nauseum, even though he would "never in a million years" approach her face-to-face' [10, 11]. More generally they can be seen as someone who takes part in trolling to gain the trust of other members in order to exploit them, for whatever reason [12]. This differentiates them from the current understanding of 'trolls' as 'vile' creatures, in that it is their resulting actions that are vile even if their intervening words are flattering.

The chatroom bob fiasco that engulfed the British Royal Family in 2012, can be seen to resemble a blurring between the traditional concepts of virtual world and the 'real world.' Today, where most people have access to the Internet and being online is not the same as being part of a counter-culture, these terms are obsolete. It is now more common to refer to meeting people face-to-face (F2F) as opposed to meeting them in real-life (IRL), except maybe in imaginary worlds that cross the line of reality, like Second Life and dedicated MMORPGs [13]. The chatroom bob in this case took a close-up shot of the Duke and Duchess of Cambridge, virtually in the nude, but no more obscene than one would find on Page 3 of a Tabloid newspaper like The Sun or The Star. They could be seen as a chatroom bob in the traditional sense, because they made an image of Kate Middleton at a distance, even though they would be unlikely to ask her to do the same face-to-face.

The incident raised a number of ethical and legal questions for the media as a force for good, some of which were answered by the Leveson Inquiry. In the case of Harry Wales, the Leveson Report stated the circumstances around his public outing. "*During the course of the holiday (in Las Vegas), on 21 August, he invited a group into the apartment which he occupied and, in their presence, is said to have played a game of 'strip billiards,'*" the report said. "*However it arose, at least two photographs were taken of him naked, one of which showed him shielding a naked girl and another embracing the girl. The photographs are reported to have been taken on a mobile phone.*" The photographs were then sold to the American

website TMZ.com and then published on the Internet, including prominent weblogs.

The 14 photographs of Kate Middleton and William Wales in 2012 are still available on the Internet, but the Royal couple secured an injunction against the first magazine in France to publish them, called Closer, through a magistrates court in the Nanterre suburb of Paris. The publishers of the magazine, Mondadori, were ordered to handover the photographs within 24 hours or face a €10,000 fine each day. The three magistrates in the court said the photographs "*belong to the Duke and Duchess of Cambridge*". The ones taken in 2012 were taken by a member of the public and then distributed via the Italian magazine Chi.

2.4 Consideration of the Editor's Code in Great Britain

Before the findings of Leveson were considered there was a clear framework under European Human Rights Law in Great Britain for state recognition of the media's desire to self-regulate. Speaking in Parliament following the publication of the Leveson Inquiry, the Home Secretary who brought in the Human Rights Act 1998, Rt Hon Jack Straw, said that his law already gave statutory effect to the Press Complaints Commission's Code, which is what the news media use to regulate themselves (Citation: HC Deb, 3 December 2012, c612). "*The Press Complaints Commission came to me when I was Home Secretary to ask for protection to be written into the Human Rights Act 1998, particularly in respect of the apparent ease with which it felt complainants could otherwise get interlocutory injunctions to stop publication of material,"* he said. "*In other words, it was the press themselves who wanted statutory force—legal force—to be behind their code, because they wanted protection. That was the crossing of the Rubicon, not anything in Leveson."*

In relation to privacy, the Editors Code which is referred to in the Human Rights Act 1998, makes the following clear in relation to what privacy the public should expect, enforceable through the Human Rights Act;

"Everyone is entitled to respect for his or her private and family life, home, health and correspondence including digital communications. ii Editors will be expected to justify intrusions into any individual's private life without consent. Account will be taken of the complainant's own public disclosures of information.iii It is unacceptable to photograph individuals in private places without their consent. Note: Private places are public or private property where there is a reasonable expectation of privacy. There is an exception to this provision where the publication can be demonstrated to be in the public interest. That is defined in this way: 1. The public interest includes, but is not confined to: (i) detecting or exposing crime or serious impropriety. (ii) Protecting public health and safety. (iii) Preventing the public from being misled by an action or statement of an individual or organisation.2. There is a public interest in freedom of expression itself."

France has a long history of protecting privacy, which includes those who are in France from other nations. The Declaration of the Rights of Man and the Citizen of 1789 had direct provisions relating to privacy, long before the ECHR was signed by both the UK and France in 1950. In France, the difference between public and private life are clear to the extent that public figures in France are afforded more privacy than has been allowed by the ECHR in the UK [14]. This security of public figures' privacy in France may not extend to the Internet, however. According to [15], whilst President Mitterrand was entitled to privacy when a book exposing allegations about his private life, called Le Grand, this did not extend to Internet publications it was found. "*The internet cannot be regulated in the way of other mediums [sic] simply because it is not the same as anything else that we have,"* a court said. "*It is a totally new and unique form of communication and deserves to be given a chance to prove itself. Laws of one country cannot hold jurisdiction in another country and holds true on the Internet because it has no borders."*

The European Convention on Human Rights referred to in this case applies in both the UK and France. The UK's ratification of the ECHR, however, meant it was almost inevitable this would be the outcome. Section 12(4) of the Human rights Act 1998 requires that a court when considering a freedom of expression case involving the media should consider the extent to which "*(i) the material has, or is about to, become available to the public; or (ii) it is, or would be, in the public interest for the material to be published,*" and importantly that any privacy code should be considered. In practice 'privacy code' has meant those of the Press Complaints Commission, Ofcom and the Information Commissioner. This would suggest that in order for the British Royal Family to bring any legal action for breach of privacy as a result of the publication of materials about them on the Internet, it is likely that they might have to consider land-law as a basis to do so. In France privacy is protected by Article 9 of the Civil Code, which says, "everyone has the right to respect for his or her private life".

3 Applying land law in UK, French and European Union jurisdictions following the Leveson Inquiry

The Leveson Inquiry found that jurisdictional issues in relation to the publication of information on the Internet were an issue in relation to social media and news reporting generally. "*Witnesses have pointed to the publication of photos of, in particular, Prince Harry and the Duchess of Cambridge, which though different in terms of the surrounding circumstances, highlight issues around the existence of different jurisdictions and regulatory regimes as applied to the press and the Internet,*" the report said. "*The Sun has argued that the ready availability of photographs of Prince Harry on the Internet justified in part its decision to publish those same photographs*."

The effect of the Internet on choice of jurisdiction is becoming clearer. It has been found that a court has jurisdiction (i.e. It is seized) if a sysop has the explicit intention to aim certain content at Internet users in the country that court is in [16]. However this fact was disputed in the Leveson Inquiry where it was stated, "*the suggestion (by the Palace) that the fact that the photographs have appeared in another jurisdiction is*

'meaningless' was to miss the point that the internet transcends jurisdictions." This would suggest that any expression of an idea or concept, even if reproduced in a different jurisdiction could be challenged within that jurisdiction if it was targeted at the Internet users also within it. Indeed, the Court of Justice of the European Union has a rule that where a publication is directly targeted at someone over the Internet that they have the right to bring proceedings within their local jurisdiction and not the one of the person who abused them [17]. It has been argued that basing the jurisdiction on where someone is centred (i.e. permanently resided) is unsustainable. [18] says, "*given that a person's private life (and reputation) may have several centres, which change over time, it does not seem possible to say more than that there might be a strong link between the facts of a particular case and the place where the claimant's centre of interests is held to lie.*" In the British legal system land law cases are usually decided on their facts [19]. It might therefore be considered reasonable under various European laws for the British Royal Family to be allowed to have their cases heard under UK law where their 'centre of interests' lies, even though the alleged offence against William Wales and Kate Middleton took place in France. The British Royal Family did bring legal proceedings in the French Courts.

In the case of Mosley v News Group Newspapers Ltd [2008] EWHC 687 (QB) the judge ruled that the claimant, who had video footage of him performing sexual acts made public, no longer had any reasonable expectation of privacy in respect of that material that was widely accessed. This was because even if he had such an expectation to privacy, as it had entered the public domain to the extent that it had it could be considered that in practical terms there was no longer anything that the law could protect. The Editor's Code says that "*Private places are public or private property where there is a reasonable expectation of privacy."* If one considers that the judiciary did not consider Mosley to be entitled to privacy on private property if the media infringing his privacy was widely available then one can see that in this regard there is little the Royal Family can do, even if they had a reasonable expectation of privacy. The European Convention on Human Rights referred to in this case applies in both the UK and in France. The UK's ratification of the ECHR, however meant it was almost inevitable this would be the outcome. As doiscussed earlier, Section 12(4) of the Human rights Act 1998 requires that a court when considering a freedom of expression case involving the media should consider the extent to which "*(i) the material has, or is about to, become available to the public; or (ii) it is, or would be, in the public interest for the material to be published.*"

Privacy may be an issue where the law of adverse possession could be used. In the case of *Beaulane Properties Ltd v Palmer* [2005] 3 W.L.R. 554, it was found that English law relating to adverse possession was incompatible with the Article 1 Protocol of the European Convention on Human Rights. It was found that the trespasser had to establish "possession" in accordance with the case law in existence at the time of its enactment. The Pennycook's case [2004] 2 W.L.R. 1331 found that the court's duty under section 3 is "*to construe legislation whenever enacted compatibly with Convention rights so far as it is practicable to do so*". It could therefore be that the law of privacy in terms of Article 8 of the ECHR could equally apply to adverse possession. In order for someone to have the right to invade another's privacy on a piece of land, the rules for adverse possession would have to be used. Temporary intrusion onto a land does not give any right to title to it [20], suggesting that one invasion of privacy is not challengeable. It is also clear that intermittent use of land does not constitute possession of it [20], meaning that even if the photographer had regularly invaded the privacy of people at the property that William Wales and Kate Middleton were using, they could have no claim to a right to use the land. A different situation might apply in the case of Harry Wales. The photographs of Harry Wales were taken in a public place, meaning the second son of Charles Windsor had no claim to a right to privacy that his brother and Ms Middleton would likely have had, in terms of land law at least.

The question might then arise that if the photographer had no right to use of the land then did the same apply to William Wales and Kate Middleton who had only temporary access to it? Taking into account *JA Pye (Oxford) Ltd v Graham [2002] 3 W.L.R. 221*, it could be considered that at the point in time that the Duke and Duchess of Cambridge were using the land in France they had the necessary intention to possess the land, even if they did not own it. This was, in applying that case, because they intended to exclude the paper owner from use of the land while they were using it, assuming privacy would be guaranteed. Beaulane Properties Ltd v Palmer taking into account the Land Registration Act 2002 the case of *Beaulane Properties Ltd v Palmer [2005] 3 W.L.R. 554* might appear to over-rule this judgement, but it could be argued that it related only to adverse possession, which was not the case in relation to the first son of Charles Windsor and his wife, which related to 'intended use.'

On this basis, it might be considered that the Duke and Duchess of Cambridge had a reasonable expectation of privacy in theory, as that had been their intended use of the property in France, but that does not affect the public interest test in relation to whether the images should have been republished by the newspapers. In the Royal's legal action brought against the company who published the 14 photographs of them in the Closer magazine, the judges appeared to be of this view. Their statement appears to blur the line between privacy and land law. "*These snapshots which showed the intimacy of a couple, partially naked on the terrace of a private home, surrounded by a park several hundred metres from a public road, and being able to legitimately assume that they are protected from passers-by, are by nature particularly intrusive,"* they concluded.

The Leveson Inquiry was also quite clear about this in relation to the Editors' Code. "Nobody at all has suggested that publishing the photographs of the Duchess of Cambridge would be anything other than a breach of the Code, notwithstanding the widespread availability of the images in other jurisdictions," the report said. "So, at least for the Royal Family, widespread availability of an image on the internet (sic) on its own is not sufficient. There has to be some other

public interest in publication of that image in order to justify it."

The images that were published on the Internet included photographs that would not be out of place on page 3 of the Sun newspaper, or in a lad-mag such as Nuts or Zoo. It has been somewhat expected in the United Kingdom that unknown young women who enter public life – often because of their homogenised looks – should be expected to pose topless for magazines usually purchased by men. In 2013 it was revealed that Conservative Member of Parliament, Esther McVey had posed in raunchy photographs prior to her role as a government minister, which were republished. Whilst the Daily Mail newspaper presented this in a negative light, it actually improved McVey's profile among people normally antoganistic to 'Tories.' Furthermore, [21] spoke of how "bikini-clad" women appear on reality TV, such as 19-year-old twins, Sam and Amanda Merchant and 19-year-old Chanelle Hayes were portrayed as being obsessed with men and lifestyle issues. Chanelle Hayes and others from the Big Brother Reality TV show, such as Michelle Bass and Kate Lawler, have gone on to be regularly featured in the aforementioned lad-mags. One might therefore argue that it is within the standards of decency of contemporary British society for public figures to be subject to the scrutiny of the public through posing for such magazines, as in the case of Chanelle Hayes, or for breakfast television in the case of Esther Mcvey. If the people in Britain have no right to elect their Head of State, or in the case of William Wales and Kate Middleton, their future Head of State and his Consort, then they should at least have the right to assess whether his partner is up to standard, especially if they are chosen from among them, on the same basis of other rags to riches public figures, such as Chanelle Hayes. It has become a part of British life for women to be objects of desire, and many even go so far as wearing T-shirts saying "Porn Queen," or the words, "Pay to touch," printed across the breast part of them [22].

Even though Leveson said that it was a breach of the Editors' Code for the images of the Duke and Duchess of Cambridge to be published without the public interest test being considered, it was clear that the inquiry felt such a judgement was down to publishers. "*Whatever system of press regulation is in force, ultimately, in this country, any editor will be free to publish what he or she believes should be published, What it is about, however, is maintenance of standards and the requirement that an editor is held to account by someone for the decisions which have been made, based on a Code that has attracted the confidence and general approval of editors and commands the confidence of the public."*

The question therefore arises whether there should be an expectation that if someone puts themselves in the public eye for their benefit, whether they should expect any privacy at all when editors publish content which may be unfavourable to their idealised self that they want the media to portray. One might argue that the media have a duty, if not an obligation, to expose the actual selves of celebrities and public figures if they do not match up with the ideal selves they try to portray to the public through their propaganda. In the case of the Duke and Duchess of Cambridge in the French château, where they were in a secluded area, one might consider them to be able to access such lands by virtue of their wealth and status. An ordinary member of the public might not have the chance to access such privacy. For instance, someone who lives in a block of flats which has only one window which is visible to the public, might be considered to be comparable to the château hired by the Duke and Duchess of Cambridge. On that basis, if it were possible for an ordinary citizen to be charged with indecent exposure through revealing themselves from their window, then in real terms the same should apply to the Royal family, who have greater prosperity than most could wish for. If it were possible, then one should expect the law to apply in real terms to the Royal Family, in order for the laws to be interpreted without discrimination. If a member of the public in a remote suburb should be expected to be 'decent' then so should the Royal Family at a secluded location. If a member of the public does not have entitlement to privacy in a public park, then nor should the Royals or any other public figure have that right where that private park that is exclusive to them by virtue of their status. But equally, if the public are entitled to privacy in a fenced off garden adjoining their property, public figures should be entitled to privacy in relative circumstances. The fact is, however, that the Duke and Duchess of Cambridge were on a balcony when they were exposing themselves, so they should not expect any more privacy or exemption from criminal law than a member of the public that is doing the same on their property.

3.1 The extent of data security and privacy in the digital age

One might argue that the difference between public law (i.e. Criminal) and private law (i.e. Torts) is that in the case of the former one has to show intent and in the case of the latter one has to show injury. The test for intent is known as mens rea, and the term for injury is becoming known as malum reus [23-25]. In the case of the Royal Family it could be seen clearly that Harry Wales and Kate Middleton suffered an injury – distress and loss of privacy – and that the person who photographed it intended to make the photographs available when they did. However it is becoming the case that mens reus is redundant in the case of the Internet where things are more fluid [23] as shown in the recent case of Chambers v DPP [2012]. Section 127 of the Communications Act 2003 makes it clear that sending a message that is of a "*grossly offensive or of an indecent, obscene or menacing character,*" is prosecutable and thus the aforementioned case means that mens reus is not necessary. The question therefore arises whether the photographs of the two Royals are obscene or indecent. Such publicised sexualised images are generally considered pornification, and considered indecent and obscene mainly by those Christian in origin, people with unclear politics, some left-leaning, others right-wing, feminist, activist and certain government documents [26].

The preeminent cases within the European locality have been Von Hannover v. Germany (nos. 1 and 2). The cases, heard in

the European Court of Human Rights (ECtHR) have a number of introduced a number of premises that have an effect on property law issues, and indirectly issues to do with free movement within the European Union legal framework. Von Hannover v Germany (2004) 43 EHRR 139, 40 EHRR 1 and Von Hannover v Germany [2012] ECHR 40660/08 (No. 2) both found that where the right to freedom of expression was being balanced against the right to respect for private life, the relevant criteria are set out in Table 2

Table 2 Criteria for loss of privacy

#	Criteria for loss of privacy
1	The contribution made by photographs or articles in the press to a debate of general interest.
2	The role or function of the person concerned and the nature of the activities that were the subject of the report and/or photograph.
3	The conduct of the person concerned prior to publication of the report or the fact that the photograph and the related information had already appeared in an earlier publication.
4	The way in which the photograph or report was published and the manner in which the person concerned was represented in the photograph or report.
5	The context and circumstances in which the published photographs were taken.

The case of Sir Elton John v Associated Newspapers Ltd [2006] EWHC 1611 QB confirmed that "Where an individual is a public figure, he is entitled to have his privacy respected in the appropriate circumstances. The individual, however, should recognise that because of his public position he must expect and accept that his actions will be more closely scrutinised by the media. Even trivial facts relating to a public figure can be of great interest to readers and other observers of the media. Whether you have courted publicity or not, you may be a legitimate subject of public attention."

On this basis it could be thought that things which might be considered trivial in relation to ordinary members of the public, such as their bodily features, it may be that these concerns are less trivial when it comes to celebrities like the Royal Family. In the case of the publication of the photographs of Harry Wales, The Sun newspaper may have passed criterion 1 above, as the article accompanying the photographs discussed the importance of the media in making such imagery available. In the case of the 2012 photographs of Kate Middleton, the Duchess of Cambridge, it might be argued that Closer magazine also passed this test. But with regards to the 2013 photographs this is less clear.

In relation to criterion 2, the case of Elton John v Associated Newspapers Ltd [2006] EWHC 1611 QB also confirmed that there is not a breach of privacy where content does not relate to, among others "*social or personal relationships or, as sometimes happens in these cases, sexual relationships,*" as these are "*all matters in respect of which, to a greater or lesser extent, as with allegations about health, an individual has a reasonable expectation of privacy."*

In relation to Criterion 3, one might argue that because both Kate Middleton and Harry Wales knowingly exposed themselves in places potentially viewable, then they should expect to be photographed in the way they were in 2012. If they were inside a private accommodation, then it might be expected they were entitled to privacy. In the case of *Cartwright v Sunderland City Council* (2009) it was found that an abatement order requesting two persons excessively making sexual noises was not infringing their right to a private life as they had given it up through making their sexual engagement audible outside of their home. It is therefore equally possible that the Duke and Duchess of Cambridge by exposing themselves on their exterior of their building in 2012 gave up their right to privacy. Equally, William Wales could be seen to have given up his rights to privacy by publically exposing himself. In terms of the 2013 photographs taken on a public beach, whilst it may not be entirely appropriate it should be expected by the two Royals that they would be at risk.

With regards to Criterion 4, there is a difference between the cases. In the case of Harry Wales, the photographs of him exposed were published in The Sun newspaper adjacent to an article where the newspaper implied they were the only ones willing to fight for freedom of expression. This may therefore pass a public interest test. The 2012 images of Kate Middleton were published in a magazine called Closer. This magazine is known for publishing images of celebrities and related gossip. The images were therefore not out of place in that magazine, and if it were considered acceptable for some celebrities to be shown in compromising positions in that magazine, then should it not be considered acceptable for a prominent member of the tax-payer funded Royal family to be subject to the same attention as other public figures?

In terms of criterion 5, one might argue that it is easy to justify the publication of Harry Wales's photographs, as the context of the photographs seems to suggest that he was quite deliberately exposing himself in public. The case of Kate Middleton is not as clear. It might therefore be appropriate to compare the situation of the Duchess of Cambridge with a member of the public on a like-for-like basis.

If one considers **Erreur ! Source du renvoi introuvable.** and **Erreur ! Source du renvoi introuvable.** in both cases these 2012 photographs (re-created by a digital artist) were taken without the consent of either party. **Erreur ! Source du renvoi introuvable.** is a representation of Greg Searle, who was a vulnerable member of the public, who posted messages on Facebook that worried friends and family, who raised the alarm. Rather than be sensitive with a man with obvious problems the police launched a standoff which resulted in the man coming to the window without many clothes on pointing an imitation gun at the police officers, which is understandable because they probably seemed to him to be threatening or menacing, which is the norm for them as publically funded humans with power. Searle was photographed and the picture made available to the media. One might question whether if this

man's privacy was invaded by police officers harassing him at his home with the limited privacy it has, whether it should not be considered outside the public interest for Kate Middleton to be photographed in 2012 (as represented in **Erreur ! Source du renvoi introuvable.** when she was in real terms in the same place in her chateau relative to the position Greg Seale was within his home. Should both not be treated equally? Why should Greg Searle be denied privacy, but Kate Middleton not be? Case law appears to support these viewpoints.

Figure 1 Digital Artist's representation of Greg Searle exposed

Figure 2 Artist's impression of Kate Middleton

In Elton John v Associated Newspapers Ltd [2006] EWHC 1611 QB, it was found that even though the photograph taken of Sir Elton John looking bald was not taken with consent, the court stated that there was unlikely to be any doctrine operative in English law whereby it is necessary to demonstrate that to publish a photograph one has to show that the subject of the photograph gave consent. It was concluded that while consent may be a relevant factor, it has little weight when it comes to disclosing information about a public figure when they are in public view.

The Leveson inquiry reported that "Critics said The Sun's public interest arguments were a convenient mask for commercial motives. It is a spurious criticism. Newspapers are fighting for their lives in the toughest of economic climates combined with technological changes that weigh heavily against traditional print. If they are not commercial they will die and they cannot let the internet become the prime forum for communication."

This makes it clear that if The Sun had not printed the images of Harry Wales that everyone wanted to see, they would have lost ground to the Internet where they were freely available. One might ask if this would make a suitable public interest criterion. If the public are interested enough to seek something out on the Internet, such as images of Harry Wales or Kate Middleton, then shouldn't other media outlets make it available to satisfy public demand?

In relation to the 2013 photographs taken of Kate Middleton, the interpretation of condition 5 could be considered quite different if one applies Von Hannover v Germany [2004] ECHR 59320/00 and Campbell v Mirror Group Newspapers Ltd [2004] 2 W.L.R. 1232. In the Von Hannover case it was put before the Court that images of her conveyed personal information and therefore broke rights to privacy under Article 8 and that freedom of expression under Article 11 was no defence. The court ruled that there should not be an expectation of privacy in an '*open space frequented by the general public*,' which in the case of Von Hannover was a swimming pool. However, in the case of Campbell v Mirror Group Newspapers Ltd was ruled that photographs of the claimant, NC, were '*analogous to details about a medical condition or its treatment, and amounted to private information which imported a duty of confidence*'. The court ruled that the '*private nature of those meetings encouraged addicts to attend them in the belief that they could do so anonymously. The assurance of privacy was an essential part of the exercise*.'

Applying both these cases one might argue that the 2013 photographs of Kate Middleton on a beach while pregnant amounted to an invasion of privacy. Even though the beach was an '*open space frequented by the general public*,' it should be quite clear to any reasonable person that imagery of someone pregnant conveys personal information that one should expect to remain private. Even so, Leveson also reported that "*There is a dangerous coalition forming of aggrieved film and television stars, out-of-sorts Labour politicians and bien pensants who would happily bring much greater regulation and censorship to the press. They believe they should decide what is in the public interest and not the millions who buy the red top papers."*

Both Harry Wales and Kate Middleton are known to have not consented to the images of them being made freely available.

But on both occasions it has been known that the public were interested in seeing the images of them. The question therefore is; would it be in the public interest for the State, such as through the court system, to prohibit the reproduction of images the public will be able to access using the Internet in any case? Should there be, as exists at present, the idea that what is interesting to the public is not necessarily in the public interest? In relation to Internet trolling, public pressure was put on the police and court system to prosecute Liam Stacey and Matthew Woods for offensive comments online, which the authorities gladly granted. Liam Stacey had a promising career ahead of him. Was it therefore in the public interest for his entire career to be messed up because he made an error of judgement? Many would say not. With Europe consisting of many different societies with many different value systems, even between people, it is clear that in determining what is in the public interest, we will have to consider these different perspectives across frontiers.

4 The role of free movement laws in protecting privacy and/or free speech

In the European Union, free movement laws are typically related to goods and services, such as freedom of establishment and cross border movement of goods. These provide problems for anyone seeking injunctions against the free movement of goods, such as the photographs in the case of the British Royal Family. The injunction secured by the Duke and Duchess of Cambridge, is pretty much meaningless in a European Union context, especially as France is part of the Schengen Agreement which opens its borders without border checks to most EU nations that have signed the treaty. It is likely that with Title III TFEU of the EU Treaty that there will be even more differences between the different nations. The United Kingdom, however, is not a signatory to Schengen, which means French ban on the publication of the photos of the Royal Family is enforceable against UK citizens trying to import the images into France. However, this may be the case in fact, but not in law. The leading case in this area of free movement is Consten. S.à.R.L. and Grundig-Verkaufs-GmbH v Commission (1966), known as Consten and Grundig. In this case Grundig wanted to restrict the import into France of their audio-visual equipment and so granted Constan an exclusive use of their trademark in France. When there was an attempted import of Grundig audio-visual equipment into France Consten sought an injunction. The subsequent referral to the Court of Justice of the EU resulted in a ruling that any attempt to restrict the free movement of goods between Member States was not compatible with European Union Law. This appeared to be the finding of the Leveson Inquiry more generally. It said, "*Based upon A Woman v Loaded,* it might be thought that substantial dissemination of the material is sufficient to trump any other claim to privacy."
The Consten and Grundig case was based on competition law, namely Article 101 TFEU. If one considers Article 106 also, this indicates that an EU member state has to abide by competition law also. In this regard, the government of a Member State is not allowed to permit the granting of an agreement between two specific undertakings to the exclusion of others. This may mean that a French company wishing to publish photographs of the Royal Family could be disadvantaged if other countries were able to publish and supply French people with publications on Mr and Mrs Windsor. Whilst under EU law France is able to discriminate against French undertakings in France, its ruling trying to prevent the Italian owners of Closer from publishing is not enforceable. The only way that France could ban the import of publications carrying photographs of the Duke and Duchess of Cambridge is through applying the 'rule of reason' under Article 34 TFEU. This article prevents the impositions of indirect discriminatory rules and practices by Member States that has the equivalent effect of direct discrimination against the import of goods or services of other Member States. The 'rule of reason' was devised to provide clarity as to the circumstances where a Member State is able to ban the import of a good or service without it being illegal under Article 34 TFEU. In order to assess the implications of this, the following excerpt from the Leveson Inquiry could be used:
"The Royal Family are, of course, in the public eye and its members will be held to account for what they do. But if society wants them also to mix with the public and in the real world, they have to be given the space to do so and their right to have a degree of privacy (less than that available to ordinary members of the public but more than at a level that is vanishingly small) must also be recognised."
One might argue that the rule of reason relating to morals could be applied through Article 34 TFEU if the images of Kate Middleton would be considered by a member state to be likely to deprave its citizens. In terms of consumer protection these could also falled within Article 34 TFEU for a similar reason. Either way, it could be considered going too far for the State to censor information widely available in non-print publications.

5 Implications and Directions for Future Research

This research study has investigated how land law and human rights law affects the rights of the British Royal Family and other celebrities through considering the Leveson Inquiry's recommendations. By virtue of this all aspects of how the rights of the public to know about public figures are effected by law have not been considered. It is clear that public figures cannot opt in and out of public life and media attention as and when they choose. It is also clear that the rights that celebirities want to enjoy should be equally applied to members of the public and vice versa. Future research could look at how the balance between privacy and free speech can be balanced in the digital age where as soon as something is available online it is available to many. Emerging issues around porn revenge, where a person will distribute images og a former partner online for all to see, will have to be explored further, as there may be greater protection under European laws for members of the public, where otherwise dedicated laws would be needed.

6 Discussion

The paper has explored the role that European Union law in the areas of property and privacy has on the way the media operates. This has been achieved through exploring the issues surrounding the British Royal Family, where such issues came to the forefront following the exposure of explicit photographs of the Duke and Duchess of Cambridge, William Wales and Kate Middleton, and also those of Harry Wales.

With the publication of the Leveson Report issues over media freedom have come to the forefront, including these cases which were looked at by the inquiry in detail. The two polarised fronts are whether there should be state regulation, or whether there should be self-regulation. The third way of self-regulation backed by statute already exists as discussed. That is that the editor's code is enforceable through Section 12 of the Human Rights Act 1998. There therefore needs to be a fourth way that uses the best of both worlds.

One might argue that self-regulation need not remain without state funding, as was the case before Leveson. It is clear that any regulation of the media needs to be both independent of media corporations and the government. This has so far only been effective through the higher courts of the British legal systems, usually in the case of celebrities rather than citizens on more modest incomes. One might also ask whether it is necessary to regulate media corporations differently from any other entity. One might want to answer that regulating the media more than any other type of organisation is in fact counter-productive, and if anything media corporations should be less regulated than other organisations because of their presumed duty to hold those with power to account. This of course is power in itself, and as Leveson has shown can be abused by the press as much as those the media expose for abuse of power.

The solution then it could be argued is to achieve a system that does not rely on expensive court proceedings in order to ensure ethical behaviour by editors and journalists. It would also be necessary to have a separate content regulator which is independent of the media producers, and the government, who in the case of the latter are likely to have a conflict of interest if media content is unfavourable to them as it inevitably will be.

It has been argued by [27] that the British Board of Film Classification (BBFC) is one of the most respected content regulators in the world, and it would therefore seem an ideal body for which to extend assessment of questionable media content in the press to. The BBFC is not paid by government aid or through subscriptions from the film industry, but is only paid by the film media producers when they are required by law to use their services. It might be that this is a suitable model for other news media, where the standards in the content are under question. It might be that where newspapers are provided with photographs that might be in the public interest, such as of alleged abuse by military personnel, before printing them they should have to seek the advice of the BBFC as to their legitimacy and decency. It could be that where certain media content is likely to cause harassment, alarm or distress (as per the Protection from Harassment Act 1997), or where it is could be perceived as grossly offensive, indecent, menacing or threatening, then it should be required to be assessed by the BBFC. Where this is not possible, such as because of the degree of public interest requires immediate broadcast or publication, then the BBFC should be required to indicate whether it should be re-broadcast or published. The fact that media content is widely available on the Internet should not, as is presently the case, give established media corporations the right to republish it. The fact that a member of the public is able to 'pirate' a film by making it available to share on a peer-to-peer service, does not mean the media then has the right to broadcast that film without paying royalties to the producer. This should be the case in regards to new media; images of say Max Mosely in a sex scene made available on the Internet should be considered 'piracy' of his privacy. There will be nothing to stop the spread of this media via peer-to-peer systems, but they should be considered part of the 'black market'. This is not to say that existing copyright rules, such as Section 30 of the Copyright, Designs and Patents Act 1988, should not be extended to 'privacy piracy' so that certain segments of the materials can be used for responsible reporting. However, it should essentially be the BBFC that has the final say on how such materials are used by the established media corporations, such as news and broadcast media. The outcome of this for the British Royal Family might have been that The Sun, which claimed to have been the newspaper standing up for free speech by publishing the photographs of Harry Wales, would not have been allowed to. Publishing the photographs could have been seen as 'privacy piracy,' even though they were widely available on the Internet. The same could have been said of the photographs of Kate Middle and William Wales – even though their photographs were taken on land in a different jurisdiction, and are available on the Internet, it would be for the BBFC to decide whether they should be more widely available or whether their publications falls below the standards of decency and ethics that to allow publication would be 'privacy piracy.'

In considering ethics, one therefore has to turn to editors and journalists. The Leveson Inquiry made clear that in the cases it was asked to look at – such as phone hacking – the editors and journalists involved had already broken the laws of the land. These are ones that apply to everyone, such as the Computer Misuse Act 1990 and the Data Protection Act 1998. It might be argued that one should not subject the public to a derogation of their freedom to thought and conscience from reading news media through because of irresponsible actions of certain editors and journalists. What the public need to know is that if an editor or journalist goes too far then they will be made accountable. One might argue that they best way to do this is to separate the regulation of media content from the regulation of the conduct of media producers.

In the Information Age, where nearly everyone is a content producer the line between what is news media and what is social media will become blurred. Can a journalist for the BBC really be posting on Twitter in a personal capacity for instance? Whether the answer is ‘yes’ or ‘no’, it is clear that regulation should not simply be directed at news outlets, but those who provide the content to them. This can come from many sources, and increasingly today independent opinion websites pull content from a range of sources. It is unfair on the smaller

website owners (i.e. sysops) to be expected to know every detail of the PCC editor's code, or even the ins and outs of the Human Rights Act 1998. There needs to be, therefore, a professionalisation of content producers. Someone in television advertising should have to be a member of the Chartered Institute for Marketing or other Privy Council recognised professional body of their choice. It might be that someone who writes content for the hard-hitting IT news service, The Register, might choose to be a member of BCS – The Chartered Institute for IT, or the Institute for Engineering Technology. Whichever they choose, they should have to follow their code of ethics, and the public should have easy recourse to that professional body should they believe that content producer has fallen below their ethical standards. These professional bodies could then be give statutory powers to issue fines or suspend payment of wages if their members act unethically in their media content production – or indeed, any other aspect of their professional life. This would mean that whether content they handle was produced in another jurisdiction, or obtained through unethical means, such as trespass or other misuse of land resources, they would be directly accountable to their professional body in the case of the latter, and the media corporation the BBFC in the case of the former.

7 References

[1] B. H. Leveson. "An Inquiry into the Culture, Practices and Ethics of the Press". The Stationary Office, 2012.

[2] George Brock. "The Leveson Inquiry: There's a bargain to be struck over media freedom and regulation"; *Journalism,* 13., 4, 519-528, 2012.

[3] Ryan J. Thomas & Teri Finneman. "Who watches the watchdogs? British newspaper metadiscourse on the Leveson Inquiry"; *Journalism Studies,* 15., 2, 172-186, 2014.

[4] Charlotte Waelde, Graeme Laurie, Abbe Brown, Smita Kheria & Jane Cornwell. "Contemporary intellectual property: Law and policy". Oxford University Press, 2013.

[5] Rowland Atkinson. "Stalking and harassment"; *Shades of Deviance: A Primer on Crime, Deviance and Social Harm* (Routledge) Anonymous 1622014.

[6] Jonathan Bishop. "Digital Teens and the 'Antisocial Network': Prevalence of Troublesome Online Youth Groups and Internet trolling in Great Britain"; *International Journal of E-Politics,* 5., 3, 1-15, 2014.

[7] G. Davie. "From Believing without Belonging to Vicarious Religion"; *The role of religion in Modern societies* (Routledge) D. Pollack & D. V. A. Olson (Eds.), 1652012.

[8] F. Furedi. "Drug control and the ascendancy of Britain's therapeutic culture"; *Drug courts in theory and in practice,* 215-233, 2002.

[9] Tony Walter, R. Hourizi, W. Moncur & S. Pitsillides. "Does the internet change how we die and mourn? An overview and analysis"; *Omega: Journal of Death & Dying,* 64., 4, 275-302, 2011.

[10] JANSEN, E., Ed. 2002. NetLingo: The Internet dictionary. Netlingo Inc., Ojai, CA.

[11] JANSEN, E. AND JAMES, V., Eds. 1995. NetLingo: the Internet dictionary. Netlingo Inc., Ojai, CA.

[12] Jonathan Bishop. "Scope and Limitations in the Government of Wales Act 2006 for Tackling Internet Abuses in the Form of 'Flame Trolling'"; *Statute Law Review,* 33., 2, 207-216, 2012.

[13] Jonathan Bishop. "Enhancing the understanding of genres of web-based communities: the role of the ecological cognition framework"; *International Journal of Web Based Communities,* 5., 1, 4-17, 2009.

[14] H. Trouille. "Private life and public image: Privacy legislation in France"; *Int'l & Comp.LQ,* 49., 199, 2000.

[15] E. Grabovszki. "The Impact of Globalization and the New Media on the Notion of World Literature"; *CLCWeb: Comparative Literature and Culture,* 1., 3, 1, 1999.

[16] J. Oster. "Rethinking Shevill. Conceptualising the EU private international law of Internet torts against personality rights"; *International Review of Law, Computers & Technology,* 26., 2-3, 113-128, 2012.

[17] J. Agate. "Jurisdiction in the context of internet publication"; *Journal of Intellectual Property Law & Practice,* 7., 4, 241-243, 2012.

[18] A. Dickinson. "By Royal Appointment: No Closer to an EU Private International Law Settlement?"; *Conflict of Laws.net,* 24., 2012.

[19] M. Wilkie, P. Luxton & R. Malcolm. "Land Law: 2011 and 2012". Oxford University Press, 2011.

[20] R. Card, J. R. Murdoch & S. Murdoch. "Law for Estate Management Students". Butterworths, 1994.

[21] R. Feasey. "Reality Television: Ordinariness, Exhibitionism, and Emotional Intelligence"; *Masculinity and Popular Culture* (Edinburgh University Press) R. Feasey (Ed.), 106-1232008.

[22] A. Ridout. "Lost in Austen: Adaptation and the Feminist Politics of Nostalgia"; *Adaptation,* 4., 1, 14-27, 2011.

[23] Jonathan Bishop. "Tough on data misuse, tough on the causes of data misuse: A review of New Labour's approach to information security and regulating the misuse of digital information (1997–2010)"; *International Review of Law, Computers & Technology,* 24., 3, 299-303, 2010.

[24] Jonathan Bishop. "The art of trolling law enforcement: a review and model for implementing 'flame trolling' legislation enacted in Great Britain (1981–2012)"; *International Review of Law, Computers & Technology,* 27., 3, 301-318, 2013.

[25] Jonathan Bishop. "The effect of deindividuation of the Internet Troller on Criminal Procedure implementation: An interview with a Hater"; *International Journal of Cyber Criminology,* 7., 1, 28-48, 2013.

[26] C. Smith. "Pornographication: A discourse for all seasons"; *International Journal of Media and Cultural Politics,* 6., 1, 103-108, 2010.

[27] U. Smartt. "Media & Entertainment Law". Routledge, 2011.

Exploring Covert Communication in Text Message on Android Smartphones

Damacharla Pavan Chowdary and Qingzhong Liu
Department of Computer Science
Sam Houston State University
Huntsville, USA
Emails: psd005@shsu.edu; liu@shsu.edu

Abstract*—Android mobile phones are one of the most widely used phones. Sharing information in android apps is a very common. However, intruders and hackers may intercept the message which is passing through the network. In this paper, we explore covert communication for hiding secret information in the most commonly used SMS and Contacts (default native apps) for android smartphones. We make use of text messages present in the SMS app that can recognize ASCII and UTF-8 encoding characters and develop steganography algorithms to embed secret information in the carrier message. Along with the character set, we also make use of emoticons and spaces present in the SMS app to hide the information.*

Keyword- Android Smartphones, SMS, Contacts, Emoticons, UTF-8 Encoding, Text Steganography; Communication

I. INTRODUCTION

A. Steganography

Origin from the Greek, steganography is the art of hiding secret information in unremarkable cover media using redundant bits of the cover medium so that the existence of the information is not known to anyone. There were many algorithms to develop steganography in digital images, video clips, and audio streams, as well as the detections [12-23]. Text Steganography is the most difficult to due to relatively lack of redundant data in a text file when compared with any image or sound file [1].

In the field of smart phones, from 1996 with Palm OS followed by windows pocket pc, Symbian, blackberry and android. The present world foresees the usage of android OS will become the platform for any development and research. Android comprises of middleware and key applications along with OS benefits. Developed by Google and Open Handset Alliance for interesting and new applications distributed to thousands of mobile devices across the world. The field of network communications and its rapid development with mobile technology has brought in the necessity for secure data transfer over internet.

Steganography is of can be mainly divided into linguistic and technical steganography. Digital Steganography includes audio, image, video and text as cover mediums for data transfer. In this proposed algorithm, we are using text as the medium of cover for hiding and sharing the secret information. Making use android features, coupled with mobile technology underlined with functionalities steganography makes this application a very robust and highly secured one.

This paper presents two algorithms for android smartphone based text message steganography. The first approach uses the theme hiding the character of the secret message in a corresponding ASCII or an UTF-8 encoding character based on the ASCII character of the SMS. The second approach works by hiding message in a sequence of emoticons and sequence of single and double spaces of the cover message. The message is neither scrambled nor changed by the proposed encipher algorithm but the increase in the size of text message makes it slightly suspicious but it can be overruled as the message tries to be more meaningful as the original one.

B. Text Steganography

Hiding in text has low embedding capacity as text has much less redundant information than image. Some text steganographic systems are available [1, 2, 3, 8, 9, 10, 11]. Approximately, text steganography may be implemented by:

Modifying Spaces: By modifying blank spaces data can be hidden in a cover text. A text message can be modified (1) the inter word spaces in a sentence, (2) the spaces at the end of each line, and (3) the spaces following punctuation marks.

Syntactic Methods: They are based on modifying the text such that its meaning is preserved. This approach is safer but harder to implement, because maintaining the same meaning without modifying the text has very less scope to embed huge chunks of data as every change in carrier alphabet/text can carry one single bit of data. We can also use punctuations in the text to embed data but when it comes to the actual usage of SMS, there are very minimal usage of punctuations.

Semantic Methods: In these methods, data is embedded in text by special word/emoticons usage that are agreed between sender and receiver. Either the sender or receiver can agree upon a set of synonyms for many words to be replaced in the text. We make use of this whole idea to

embed secret message in SMS not by using synonyms but by using a set of emoticons to represent bits 1 and 0. The decoder reads the cover text word by word, space by space, emoticon by emoticons and searches the thesaurus for its occurrence of each symbol and decodes the secret message.

II. Related Works

Thousands of individual papers are available with different algorithms to increase the secrecy in transmission in mobile devices. But the development of steganography application on android platform is considered to be more challenging as stated by authors of paper [1] "Steganography on a phone is more difficult, because it requires access to the device's operating system, but no one should doubt that committed individuals will have no trouble rising to the challenge". White and Martina [2] developed an application that uses steganography to hide a short text message in an audio message recorded by the user and then share that message. MoBiSiS is an application that implements a steganographic algorithm to send the image that covers the secret message via the Multimedia Messaging Service (MMS) [3]. Similar applications with MoBiSiS having the same limitations, are MobiStego [4] and Pixelknot [5] both available on Google Play [6, 7].

All the above related works show different algorithms and techniques for different types of steganography used in mobile phones. There are quite a few applications in android market but they don't interact with native android applications. In this paper, we came up with a steganographic application to be developed using text steganography being deployed in android SMS app and become the pioneer in attributing it to transmit the sensitive data among the people involved in conversation.

III . Proposed Methods

In the proposed method of embedding and extracting secret data in SMS uses two different algorithm approaches. The end user is given the option of selecting the embedding and decoding method as per his choice and needs. User can either choose to hide data using emoticons and spaces or using ASCII/UTF encoding schemes. By providing the user this flexibility, makes this app more user friendly and secure in its own way by sending the desired secret SMS using mobile network. Both the algorithm approaches has communication with native android SMS and Contacts application for the embedding/ decoding/sending/receiving of messages. Small chunks of secret data is being shared across the users internationally as long as your carrier provides the necessary facilities to send it.

A. UTF-8 SMS Steganography

A.1. Data embedding in UTF-8 encoding scheme

Input: a cover plain text ASCII coded SMS message and a secret plain text ASCII coded small data.

Output: a stego-SMS carrier with secret message embedded in it with a combination of ASCII and UTF-8 encodings.

Steps:

1. Consider the secret message in ASCII coding and assign it to a string with the length of x characters.
2. Add a "?" symbol to depict it as the end of the secret message.
3. Convert the secret message string to be in lower case characters.
4. Convert the carrier message into lower case characters without any numbers, special characters
5. Normalize the whole secret message to be in the format as of sender and receiver negotiated much before the transmission of data.
6. Regular ASCII characters are normalized in our proposed algorithm, shown in Table 1.

Table 1. Normalization of ASCII codes

Entity	ASCII Coding	Normalized Coding
Numbers 0-9	48-57	0-9
Small letters a-z	97-122	10-35
Space	32	36

7. Loop through each character of secret message to normalize it in our app ASCII codes.
8. Normalizing from 0-36, we need only 6 bits to represent 0 – 36 decimal numbers.
9. Using the logic of bitwise operations over each character to represent an array of binary digits for its representation in normalized decimal value.
10. Represent our secret message of 6x binary bits.
11. Encrypting our secret message onto the carrier message is done by looping through each character of the carrier message and in place of binary bit true for the secret message, we replace the existing ASCII coded carrier message character with a corresponding UTF-8 similar character, so that the original meaning of the message is unchanged.

Table 2. ASCII – UTF8 mappings

ASCII	UTF-8	ASCII	UTF-8	ASCII	UTF-8
a	uFF41	j	uFF4A	s	uFF53
b	u0253	k	u1D0B	t	u021B
c	uFF43	l	uFF4C	u	u1D1C
d	u217E	m	u217F	v	uFF56
e	uFF45	n	u0578	w	u1D21
f	u1E9D	o	uFF4F	x	uFF58
g	uFF47	p	uFF50	y	uFF59
h	uFF48	q	u1D90	z	u1D22
i	uFF49	r	u1D26		

12. For every space encountered in the carrier message, two spaces are encrypted in the stego message if the corresponding bit at the secret message is true otherwise single space is added in the cipher text.

A.2. Data extraction in UTF-8 encoding scheme

Input: a stego-SMS carrier with secret message embedded in it with a combination of ASCII and UTF-8 encodings.

Output: a secret plain text ASCII coded small data.

1. Consider the stego message in a combination of ASCII and UTF-8 encodings strings is taken.
2. Loop through each character of the stego message to check if the character corresponds to any ASCII code between 97-122, then place a value of false or '0' in decrypt secret message array
3. Loop through each character of the encrypted message to check if the character corresponds to any UTF-8-ASCII mapping code , then place a value of true or '1' in decrypt secret message array
4. Now consider the whole array of binary bits to divide them into 6 digits at once to represent a decimal number corresponding to a value in normalized original ASCII codes to represent the hidden secret character.

B. Emoticons an Space-based Text Steganography

B.1. Data embedding in emoticons and spaces of SMS

Input: a cover plain text ASCII coded SMS message and a secret plain text ASCII coded small data.

Output: a stego-SMS carrier with secret message embedded in it at every space between words of carrier message replaced with a sequence of emoticons and spaces.

Steps:
1. Consider the secret message in ASCII coding and assign it to a string
2. Add a "?" symbol to depict it as the end of the secret message.
3. Convert the secret message string to be in lower case characters.
4. Normalize the whole secret message to be in the format as of sender and receiver negotiated much before the transmission of data.
5. Regular ASCII characters are normalized in our proposed algorithm, as shown by Table 1.
6. Loop through each character of secret message to normalize it in our app ASCII codes.
7. As we are normalizing from 0-36, we need only 6 bits to represent 0 – 36 decimal numbers.
8. Using the logic of bitwise operations over each character to represent an array of binary digits for its representation in normalized decimal value.
9. We embed our secret message in our carrier message spaces where every space between two words carry a single character of six binary digits.
10. Each of these binary alternatives are represented by emoticons and spaces
11. Two sets of emoticon mappings are created to represent 1 and 0, for example, Table 3 shows the two sets.

Table 3. Emoticons mapping for binary digits

Binary digit	Emoticons
1	56842; 56833; 56834; 56835; 56836
0	56837; 56838; 56843; 56861; 56860

12. Single space represents 1 and double space represents 0
13. Encrypting our secret message onto the carrier message is done by looping through each space of the carrier message and in place of binary bit true for the secret message, we randomly select an emoticon from one emoticon mapping.
14. Alternative secret bit represent a single or double space based on existing secret binary digit.
15. After placing six binary digits in one single word space, the carrier message is continued in the encrypted message and steps from 9 – 15 steps.
16. Embed a secret message of length 'x' characters or 6x length in binary into the text message.

B .2. Data extraction in emoticons and spaces of SMS

Input: a stego-SMS carrier with secret message embedded in it at every space between words of carrier message replaced with a sequence of emoticons and spaces.

Output: a secret plain text ASCII coded small data.

1. Consider the encrypted cipher message in a combination of ASCII encoding strings with sequence of emoticons and spaces embedded in carrier message word spaces.
2. Loop through each character of the encrypted message to check for spaces, if the character in the space corresponds to any emoticon of two mappings representing either 0 or 1 is decoded and added to binary string
3. If the character in the space corresponds to any single or double space , representing either 0 or 1 is decoded and added to binary string
4. Now consider the whole array of binary bits to divide them into 6 digits at once to represent a decimal number corresponding to a value in normalized original ASCII codes to represent the hidden secret character.

V. DEPLOYMENT AND APPLICATION USE CASES

The application is portable with the signed and unsigned .apk file that is deployed across any android platform. By

enabling the device to run application from unknown sources in the developer settings, any smartphone can run this application. To decode the secret message, the receiver should also run this application in his device. This application can be deployed in any android OS device but can be used only in smartphones which has SMS and Contacts application. Application is developed with target android as API 21, but it supports all the lower versions of android OS.

A . Data embedding workflow

The following figure 1(a) through (d) shows the workflow while we embed secret information to the text message on Android smartphone.

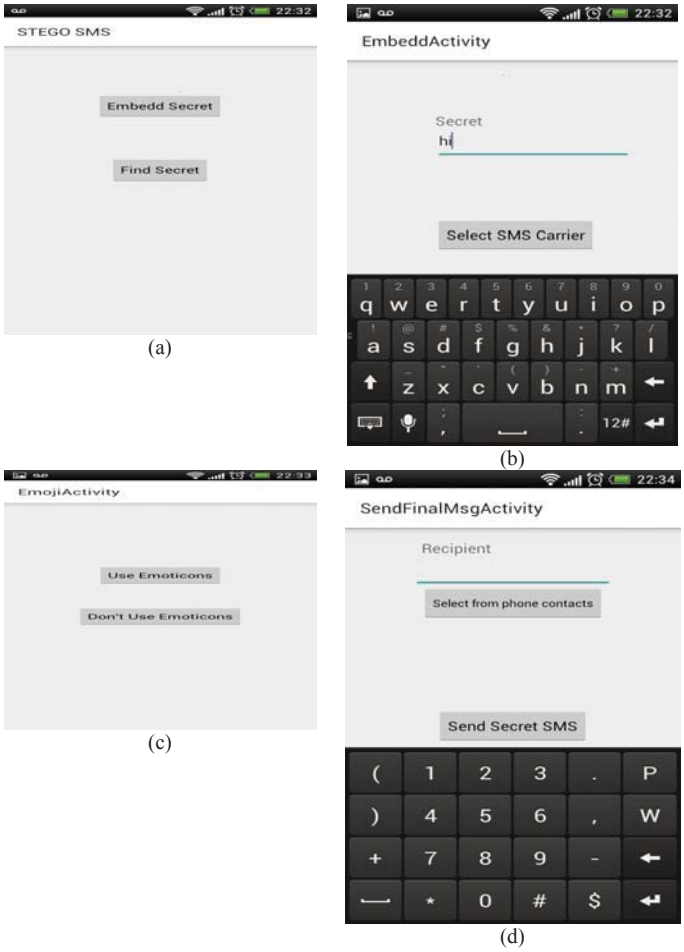


Figure 1. Working flow for secret data embedding. (a) Selecting embedding or decoding activity from menu; (b)Writing secret message and selecting SMS carrier message; (c) Selecting either of two algorithms to embed secret data into SMS carrier message; and (d) Selecting recipient contact from mobile contacts or writing your own number, click on SEND

B . Date extraction workflow

The following figure 2 (a) through (c) shows the workflow while we extract secret information from the text message wherein the secret information is carried.

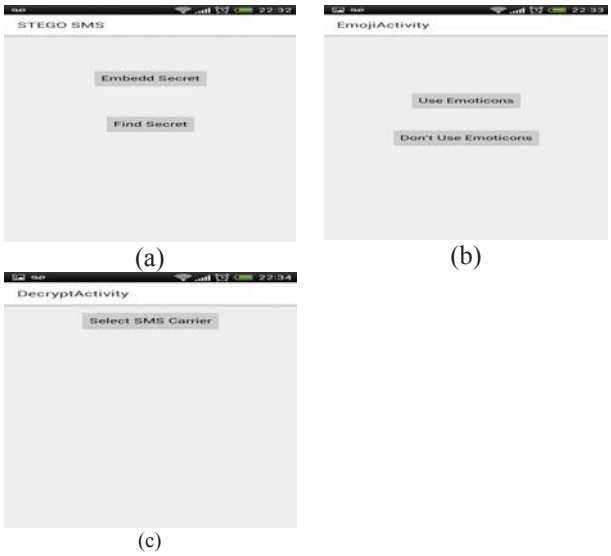


Figure 2. Working flow for secret information extraction. (a) Select find secret button to decode; (b) Selecting either of two algorithms to decode secret data from SMS carrier; (c) Selecting encoded SMS carrier from SMS app to decode the secret message.

VI. APPLICATION SECURITY AND LIMITATIOIN

Data embedding and extraction algorithms use normalization of regualar ASCII codes into application defined encoding standards which confuses teh intruder while decoding the actual message. From the very large character set of UTF-8 encodings, we chose only 26 small lettered alphabets which are very similar to our regular ASCII standard small lettered alphabets. Unless the mappings were found , it is very difficult for the hackers to decrypt the secret message in the carrier. From the very large set of emoticons, we grouped set of emoticons to represent 1 and others to represent 0. So for any man in the middle attack, it will be a tedious job for to find which group of emoticons return 1 and others return 0.

On top of all these security aspects taken care of the application , the actual secret message is being hidden in a SMS carrier (unlike any carrier of our own text message from the developed application). This secret carrier message is being transmitted to recipient only through the mobile network carrier which provides an high official security around our application security layer to gaurd our secret message from any intruders/hackers.

Secret message can be only of lower case alphabets and the carrier message should be only of lower case/upper case characters without any punctuation marks and special characters. As the SMS carrier is of short size, we cannot send huge chunks of paragraphs as the message carrier might break down the message while sending and then the receiver might receive it in various messages where the concatenation of these messages doesn't happen, then the receiver cannot decode the message.

VII. Testing Examples

This application is tested on API 21 of android version smartphone devices. It is also tested on emulators but the character set sending of an emulator has a limit which will actually break our message into small data chunks, hence it cannot be decrypted. Various android versions of mobile phones read UTF-8 encodings and some doesn't recognize, therefore the message can be viewed differently in different devices. The results of embedding secret information using the options"with " and " without emoticons " will encrypt the carrier message look as follows:

Secret Message: goodnight sir

Original Message: Hi dr liu please consider my final project demo This project is about developing an android application using characters and spaces in text making this a text steganography I have put in great and sincere efforts to make this project happen in a very short span of time I embedded data in spaces and some selected characters as spaces alone are not sufficient to hide the whole data

Stego Message using UTF-8 encoding : hi dr liu p leas e c ons i der my □ i n al pro j e c t dem o ṭhis pro j e c t is aɓouṭ d ev e lo p ing a n d ro i d ap p lic a tion us i ng c h aract e rs and sp a c e s in te x ṭ m aĸing ṭh i s a te x t stegano g raphy i h a ve p uṭ in greaṭ a n d s i ncer e e□□orṭs to make t h i s proj e c ṭ happen i n a v e ry s hort sp a n of time i emb e d d ed daṭa i n sp a c e s and so me s e lect e d char a cters as spaces al o ne a r e n o t suffici e nt to hide ṭhe w h ole data

Stego Message using Emoticons and Spaces : hi  dr liu please consider my final project demo this project is about developing an android application using characters and spaces in text making this a text steganography.

V. Conclusions

We has designed algorithms to implement text message based steganography on Android smartphones, which have been tested and validated by our experiments. The new application has limitations where the secret message and carrier messages don't support special characters and future work on this will improve the robustness, quality and security of the application. In future development work, implementing a stego key which is shared only between the sender and the receiver will increase the security aspect of the application. As android is an open source platform, once should be very carefully in developing android applications so as to provide high end security across the network.

Acknowledgment

The support for this study from Sam Houston State University research office under ERG grant and part support from the NSF award No. 1318688 are highly appreciated.

References

1. Viraj Sharadbhai Gandhi "Steganography using cone insertion algorithm and mobile based stealth steganography". A Thesis Presented to the Faculty of San Diego State University, 2010. http://sdsu-dspace.calstate.edu/bitstream/handle/10211.10/483/Gandhi_Viraj.pdf?sequence=1
2. Thomas F. M. White and Jean E. Martina, "Mobile Steganography Embedder". 11 SBSeg Simposio Brasileiro Em Seguranca Da Informacao E De Sistemas Computacionais, Bsalia-DF, 6 a 11 de Novembro de 2011, http://www.peotta.com/sbseg2011/resources/downloads/wticg/91964.pdf
3. Rosziati, L. C. Kee, "MoBiSiS: an android-based application for sending stego image through MMS", ICCGI 2012. The Seventh International Multi-Conference on Computing in the Global Information Technology, 115–120, 2012.
4. https://www.openhub.net/p/mobistego
5. https://guardianproject.info/apps/pixelknot/
6. https://play.google.com/store/apps/details?id=info.guardianproject.pixelknot&hl=en
7. https://play.google.com/store/apps/details?id=it.mobistego&hl=en
8. M. S. Shahreza, and M. H. S. Shahreza, "Text steganography in SMS," 2007 Int. Conf. on Convergence Information Technology, 2007, pp. 2260-2265.
9. M. Khairullah, "A novel text steganography system in cricket match scorecard," Int. Journal of Computer Applications, vol.21, pp. 43-47, 2011.
10. H. Kabetta, B. Y. Dwiandiyanta, and Suyoto, "Information hiding in CSS: a secure scheme text steganography using public key cryptosystem," Int. Journal on Cryptography and Information Security, vol.1, pp. 13-22, 2011.
11. A. Chandragiri, P. A. Cooper, Y. Liu, Q. Liu, "Implementing secure communication on short text messaging". Proc. 2nd International Symposium on Digital Forensics and Security (ISDFS'14), 12-13 May 2014, Houston, TX, pages 77-80
12. Q. Liu and Z. Chen (2014). Improved approaches with calibrated neighboring joint density to steganalysis and seam-carved forgery detection in JPEG images. *ACM Trans. on Intelligent Systems and Technology*, 5(4):63.
13. Q. Liu (2011). Steganalysis of JPEG-based adaptive steganography and YASS. *Proc. ACM Multimedia & Security* 2011, pages 77-86.
14. Q. Liu, A. H. Sung, M. Qiao, Z. Chen and B . Ribeiro (2010). An improved approach to steganaysis of JPEG images. *Information Sciences* 180(9):1643-1655.
15. Q. Liu, A. H. Sung and M. Qiao (2011). Neighboring joint density-based JPEG steganalysis. *ACM Trans. on Intelligent Systems and Technology*, 2(2):16.

16. Q. Liu, A. H. Sung and M. Qiao (2011). A method to detect JPEG-based double compression. In *Proc. 8th International Symposium on Neural Networks*, pp 466-476.

17. Q. Liu, A. H. Sung and M. Qiao (2011). Derivative-based audio steganalysis. *ACM Transactions on Multimedia Computing, Communications and Applications*, 7(3):18.

18. Q. Liu, A. H. Sung, Z. Chen and X. Huang (2011). A JPEG-based statistically invisible steganography. *Proc. 3rd International Conference on Internet Multimedia Computing and Service*, pages 78-81.

19. Q. Liu, A. H. Sung, Ribeiro BM, Wei M, Z. Chen and J. Xu (2008). Image complexity and feature mining for steganalysis of least significant bit matching steganography. *Information Sciences* 178(1): 21-36.

20. Q. Liu, A. H. Sung, Z. Chen and J. Xu (2008). Feature mining and pattern classification for steganalysis of LSB matching steganography in grayscale images. *Pattern Recognition* 41 (1): 56-66.

21. J. Kodovsky and J. Fridrich (2011). Steganalysis in high dimensions: fusing classifiers built on random subspaces, *Proc. SPIE* 7880, 78800L, 2011; doi:10.1117/12.872279

22. J. Kodovsky, J. Fridrich, and V. Holub (2012). Ensemble classifiers for steganalysis of digital media, *IEEE Transactions on Information Forensics and Security*, **7**(2):432-444, 2012.

23. J. Kodovsky and J. Fridrich (2012). Steganalysis of JPEG images using rich models. *Proc. SPIE 8303, Media Watermarking, Security, and Forensics 2012, 83030A* (February 9, 2012); doi:10.1117/12.907495.