International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:8, No:3, 2014

CVOIP-FRU: Comprehensive VoIP Forensics Report Utility

Alejandro Villegas, Cihan Varol

***Abstract*—**Voice over Internet Protocol (VoIP) products is an emerging technology that can contain forensically important information for a criminal activity. Without having the user name and passwords, this forensically important information can still be gathered by the investigators. Although there are a few VoIP forensic investigative applications available in the literature, most of them are particularly designed to collect evidence from the Skype product. Therefore, in order to assist law enforcement with collecting forensically important information from variety of Betamax VoIP tools, CVOIP-FRU framework is developed. CVOIP-FRU provides a data gathering solution that retrieves usernames, contact lists, as well as call and SMS logs from Betamax VoIP products. It is a scripting utility that searches for data within the registry, logs and the user roaming profiles in Windows and Mac OSX operating systems. Subsequently, it parses the output into readable text and html formats. One superior way of CVOIP-FRU compared to the other applications that due to intelligent data filtering capabilities and cross platform scripting back end of CVOIP-FRU, it is expandable to include other VoIP solutions as well. Overall, this paper reveals the exploratory analysis performed in order to find the key data paths and locations, the development stages of the framework, and the empirical testing and quality assurance of CVOIP-FRU.

***Keywords*—**Betamax, digital forensics, report utility, VoIP, VoIP Buster, VoIPWise.

I. Introduction

TELECOMMUNICATIONS have gone over a dramatic evolution due to technology advances. The days of RJ11 phone lines have become archaic. The VoIP revolution has altered our daily communications. National and international calls via VoIP applications are now the standard and it is expected that 79 percent of U.S. businesses will be using it by 2013 [1].

According to Palaez, Voice over Internet Protocol over Wireless (VoIPoW) has become the most popular mobile communication in the world [2]. However, criminal and terrorist organizations also take advantage of the VoIP innovative invention because of the lack of technology for law enforcement to monitor the calls made by VoIP and the anonymity provided by it. In other words, although VoIP has become a popular communication solution; investigations correlated to attacks on wireless VoIP networks are still in their infancy [3], [4].

VoIP communications are being leveraged by large criminal organizations to facilitate the information exchanges among handlers and other key personnel. VoIP was utilized during the Mumbai terrorist attack by Lashkar-e-Taiba, the Indian authorities had a plethora of difficulties trying to track the members involved in the attack. In mid-October, a draft United States Army intelligence report highlighted the growing interest of terrorists in using VoIP, noting recent news reports of terrorists using Skype to communicate [5]. The fact that terrorists are using VoIP technology is no longer latent, it is a known fact. However, fortunately, setup and initialization of VoIP calls, encapsulated voice components, and contact lists are remain in the physical memory of the computer. Therefore, an investigator can acquire forensically important information without knowing the password of the VoIP application.

Tools designed to track and parse VoIP logs are currently highly demanded as they can become a true asset in the crime investigations field. Most of the available tools in the literature are currently developed to capture Skype data. Skype data can be collected along with different analytical approaches: logs, packets, live memory acquisitions, etc. Nevertheless, if Skype becomes highly traceable, the logical trend will occur. Criminals will opt for less popular solution ones such as Betamax products, namely VoIPBuster, VoIPWise, and etc., since they are less common to be less traceable and also housed in outside of the USA [6].

The main aim of a forensic investigator is to stay ahead of the VoIP tracking and data recollection challenge. The best approach to prevent criminal groups from utilizing a technology is by proactively creating an "antidote". Naturally, we created a solution to assist in neutralizing and mitigating the exploitable vulnerability of untraceable VoIP solutions; by formulating a data gathering utility (CVOIP-FRU) for the Betamax VoIP products, which can also easily expandable to other VoIP solutions as well. Overall, the framework accomplished a VoIP environment that is as traceable as the traditional RJ11 phone line communication. Lowering the threat currently presented by the relatively complicated and extended traceability of VoIP in general.

II. Literature Review

Individual and business communications have evolved into VoIP solutions that utilize SIP integrations including IP Multimedia Subsystem [7]. The Next Generation Network (NGN) of the Fixed and Mobile Networks uses both technologies as its prime foundation [7], [8]. VoIP is the industry standard of communications, therefore the vital need to develop forensic applications that adhere to such trend. VoIP's top communication's priority attempts to regulate

A. Villegas was a graduate student at Sam Houston State University, Huntsville, TX 77341, USA (e-mail: axv015@shsu.edu).
C. Varol is with Sam Houston State University, Huntsville, TX 77341, USA (e-mail: cvarol@shsu.edu).

energy consumption within Wi-Fi connections are been made, such as the Power-Saving and Robust Point Coordination function (PSR-PCF) for the transmission of VoIP over 802.11 [9]. The popular codec G.711 is used by headset manufacturers due to its quality, sacrificing bandwidth and energy power [10]. About 24 million US households use a variety of VoIP service [11]. Not to mention that the Internet Engineering Task Force (IETF) has facilitated the PSTN (Public-switched Telephone Network) to VoIP transition by the development of standards for call setup and media transport [11].

Albeit there is a blanket need to support forensics tools for VoIP applications, majority of the previous VoIP work targeted to the most popular application: Skype [12]-[14]. There are a couple of approaches that have been utilized to bypass and/or reverse engineer Skype in order to retrieve evidence data. Skype Fingerprints documents profile information on a Windows system that is resident in the Windows registry and within the hard drive file system [15]. Skype Packets uses a backdoor approach, using Ethereal to capture and analyze Skype network data against the application layer content [2], [16]. Skype Log View parses the logs and provides a GUI interface to access and view Skype activity [17]. Another approach, Skype – Live Memory Acquisition uses a program called Volatility to perform live memory acquisition [18]. Subsequently a hex editor is used to analyze the data gathered. A couple of other tools, such as AESKeyFinder and RSAKeyFinder, are utilized to handle the encryption key recovery [18].

VoIP solutions have been criticized for their lack of security. VoIP is exposed to Denial of Service attacks, invalid access, toll fraud, and eavesdropping [19]. The invalid access concern also validates the need for forensic tools. The Common Vulnerabilities and Exposures (CVE) database reveals around 221 VoIP vulnerabilities, without a doubt a concerning figure [12]. A sample scenario would involve an attacker utilizing a VoIP infrastructure to generate malicious calls with ultimate negative motives. Furthermore, there are VoIP IPv6 implementations currently being tested such as efforts by the Taiwan Academic Network [20]. Overall, CVOIP-FRU's functionality will be instrumental for the VoIP Digital Evidence Forensics Standard Operating Procedure (DEFSOP) by gathering data that can be easily implemented into a report format for a court of law [21].

III. Methodology

A. CVoIP-FRU Architecture

The high level architecture of CVOIP-FRU is given in Fig. 1. The platform collects forensically important information from the Registry Hives and several different resident files located in the User Root Folder. The collected data run through the filter at the CVOIP-FRU engine and presents all findings as an html report to the user. The details of the platform are discussed in the following subsections.

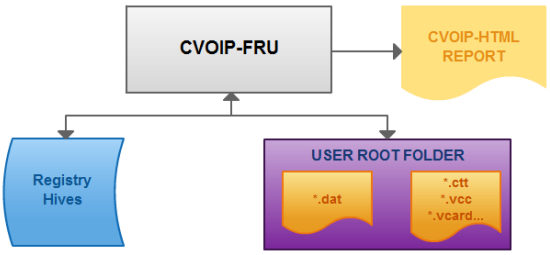

Fig. 1 CVOIP-FRU High Level Architecture

B. CVoIP-FRU Scripting Platform

CVOIP-FRU is developed in PERL and designed for Windows and OSX platforms. Nonetheless, the framework is designed to be adaptable and scalable to other VoIP applications across various OS platforms.

Windows: CVOIP-FRU essentially analyzes the Windows registry and contents of the corresponding AppData folders of each VoIP application in question. The back end programming of choice is PERL since it is resource efficient and able to parse data with minimal effort. Furthermore, PERL can be utilized in any commercial operating system, instrumental for the development of CVOIP-FRU in other platforms. Since the data to be gathered is stored either within the Windows file system or in the registry, the Windows PowerShell is the ultimate co-integration for the PERL framework. The PowerShell is based in the .NET framework and is able to execute practically any Windows native operation. The PowerShell uses cmdlets, which are preconfigured command instructions. The two main cmdlets used for CVOIP-FRU are Get-ItemProperty and Get-Content. Get-ItemProperty retrieves the properties of a given file within the registry. Get-Content retrieves the content (typically text) of an item within the registry given a specific file path. The PowerShell cmdlets can parse data from the registry hives and directories and possess a variety of output format options such as format-list and format-table.

Mac OSX: CVOIP-FRU utilizes regex to thoroughly search the *system.log* for VoIP data and activity. Albeit, the scripting platform uses PERL as the preferred coding language, it also leverages UNIX based commands to rapidly retrieve the necessary VoIP information. The UNIX command **bzcat** is used to find information in previously compressed *system.log* files without the need to decompress them. The major search engine relies on the native OSX system command **grep**; along with the corresponding switches that meet the needs of each VoIP query criteria. VoIP applications tend to encrypt the VoIP activity and information. Such encrypted information is then transferred to one of their operating internet facing web servers; eliminating the need to store the configuration settings within the Operating System. The encryption operation was corroborated by the analysis of network traces via Wireshark [22]. Nevertheless, VoIP clients use JavaScript routines to call their functions. Furthermore, the JavaScript subroutines post relevant VoIP information to the system logs such as phone numbers contacted via the phone or SMS functionally as well as new contact list information.

C. CVoIP-FRU Data Gathering

Windows: The PowerShell integration is beneficial reducing the amount of PERL regex that would have been necessary in order to properly handle the script output. Windows 7 stores the VoIP information in the following folder path: C:\Users\$user\AppData\Roaming\VoipApplication\. The folder contains a user folder which contains each user's information. The data is stored in .ctt (.NET Messenger), .vcc (VoIP Contact list) or .vCard (Outlook) file extensions. In addition, the MostUsedContacts is stored as .DAT and contains the most frequent contacts that have been reached either via a call or SMS. The Voip Application data is stored in XML format. The call and SMS log information is stored in the History_$user file saved with a .DAT file extension readable as plain text. It contains the incoming and outgoing VoIP activity. The main registry hive that stores VoIP Application data is: HKEY_CURRENT_USER\Software\VoipApplication.
Majority of the registry VoIP Application values are to enable/disable a particular functionality. Nevertheless, one of the most useful keys is the following one as it contains all the users: HKCU:\software\VoipApplication\Usernames. Fig. 2 displays the log files and registry hives that compose the architecture that contains the pertinent VoIPApplication forensic data.

Mac OSX: CVOIP-FRU leverages the Mac OSX UNIX foundation by utilizing bash shell commands to easily search through the system logs (Fig. 3). The **grep** and **bzcat** commands drive the VoIP find function. The search subroutine attempts to retrieve VoIP data from every system log in the system, including the previously compressed versions in **.bz2** format. The VoIP products store the data on the Voip Application web server while transferring the data in encrypted form. Nonetheless, the Betamax application script that writes to the *system.log* is *voipclientjs*, located in the /Applications/VoipApplication.app/Contents/Resources/
folder. CVOIP-FRU is able to collect VoIP forensic data even if the VoIP application has been removed from the system, or if the contacts and call history have been cleared. The pseudocode of the data gathering algorithms is given in Fig. 4.

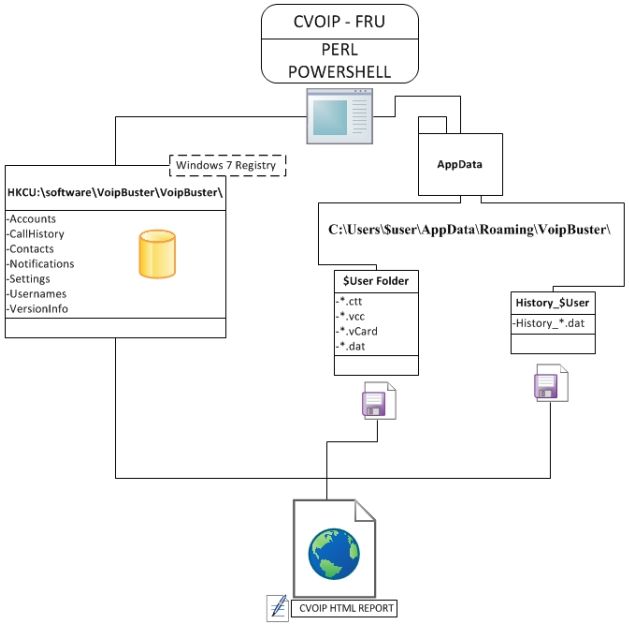

Fig. 2 CVOIP-FRU Detailed Windows Scripting Architecture

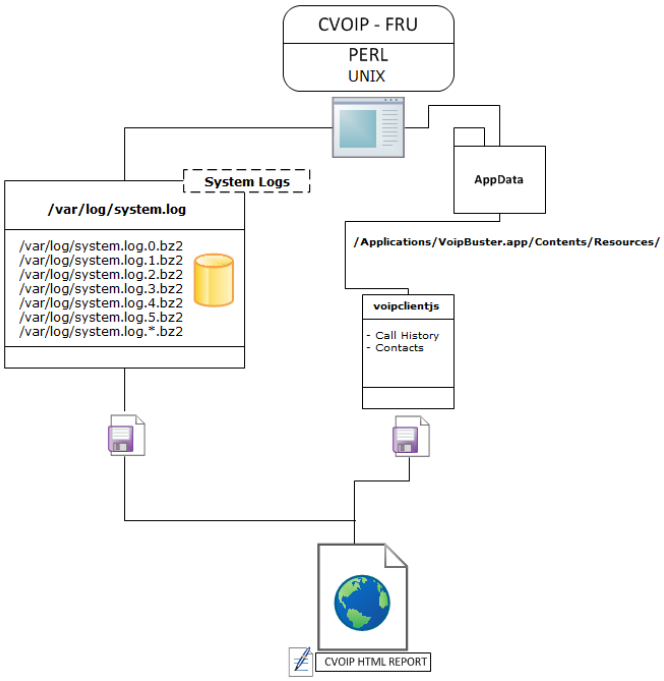

Fig. 3 CVOIP-FRU Detailed OSX Scripting Architecture

```
Microsoft Windows OS
* Launch CVOIP-FRU UI
### Collect Data ###
    #VoIP Usernames
        1 Get-ItemProperty
HKCU:\software\VoipApplication\Usernames | format-list -property
Name* | Out-File voipusernames.txt
    #VoIP Call History
        2 Get-content
C:\Users\*\AppData\Roaming\VoipApplication\*.dat | Out-File
voipcallhistory.txt
    #VoIP Contact List
        3 Get-content
C:\Users\*\AppData\Roaming\VoipApplication\*\*.vcc | Out-File
voipcontactlist.txt
    4 Filter Results
5 Print Single Reports
6 Print Full Report

MacOSX
*Launch CVOIP-FRU UI
### Collect Data ###
    #VoIP Call History
        1 system('bzcat /var/log/system.log.* | grep -A 1 "call rate for
number" >> voipcalls.txt');
        2 system('less /var/log/system.log | grep -A 1 "call rate for
number" >> voipcalls.txt');
    #VoIP Contact List
        3 system('bzcat /var/log/system.log.* | grep -A 4 "save contact
with details" >> voipcontacts.txt');
        4 system('less /var/log/system.log | grep -A 4 "save contact
with details" >> voipcontacts.txt');
    5 Filter Results
6 Print Single Reports
7 Print Full Report
```

Fig. 4. CVOIP-FRU Scripting Language PseudoCode

D.CVoIP-FRU User Interface

The CVOIP-FRU user interface utilizes the Tkx perl module to provide a similar and consistent GUI interface across different OS platforms. Although, each OS provides its own window manager display features, CVOIP-FRU provides a cross-platform unique feel. The end user simply has to launch the application via the CVOIP-FRU executable and click on the corresponding report button to retrieve the collected forensic VoIP data as shown in Figs. 5 and 6.

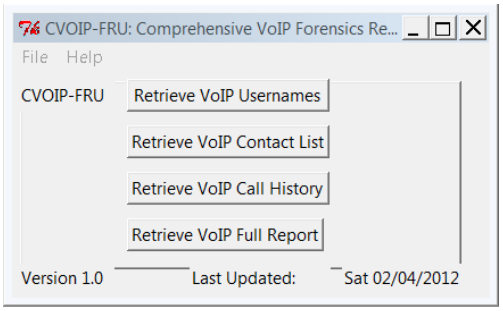

Fig. 5 CVOIP-FRU Windows UI

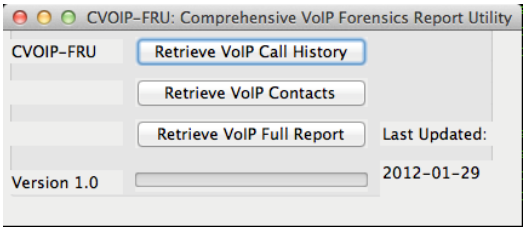

Fig. 6 CVOIP-FRU OSX UI

E. CVoIP-FRU Dat Filtering and Reporting

CVOIP-FRU retrieves the forensic VoIP data in single manageable and portable text files that can be used to formulate a forensic report. Furthermore, CVOIP-FRU generates a comprehensive HTML report that outlines the VoIP data on a categorized manner (shown in Fig. 7): VoIP Accounts, VoIP Contact Lists, VoIP Call and SMS Logs. The CVOIP-FRU report also displays the corresponding registry hive.key, filesystem location as well as other information relevant for forensic purposes. For example, the VoIPApplication Accounts section reveals the registry key displaying the usernames. The VoIPApplication Call History section prints all the call and SMS logs for every user by using a wildcard. Some of the key values for a call log are: Number Called, Call Start Time, Call Connection Time, Call End Time and New Voice Message. The key values for a SMS call are: Time, Other Party Number, GUID and Message.

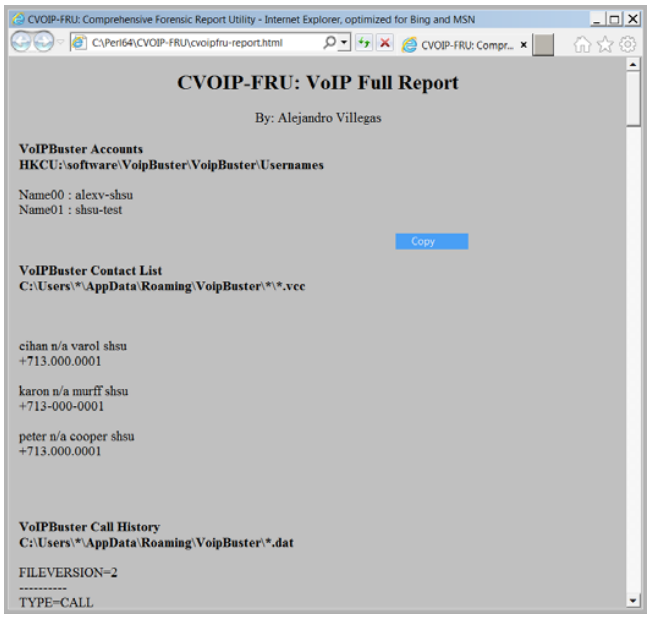

Fig. 7 CVOIP-FRU VoIP Full Report

The VoIP data format varies depending on the VoIP application, Operating System, as well as the source of data collection. In particular, since each VoIP application has its own structure (tags) to store the information in the User Root Folder, the files need to be filtered by the framework. Therefore, the files are compared to a reference table which holds the most frequently used names and common phone

numbering format and a dictionary to differentiate the SMS Logs from the generic tags [23]. At the end, a distance score is calculated to see if there is an exact or close match to the original. The distance between two strings s_1 and s_2 is calculated in time $O(|s_1| \times |s_2|)$ using $O(\min(|s_1|, |s_2|))$ space. The distance is converted into a similarity measure (between 0.0 and 1.0) using

$$sim_{(S1,S2)} = 1 - \frac{dist_{(S1,S2)}}{\#\ of\ Average\ char_{(S1,S2)}}$$

where $dist_{(S1,S2)}$ is the Levenshtein-Edit Distance cost between the two strings [23]. The Levenshtein-Edit Distance distance is symmetric and it always holds that $0 \leq distld(s_1, s_2) \leq \max(|s_1|, |s_2|)$. Based on the distance score obtained, if there is an exact match or close match within the threshold of [0.5-1), the collected data is filtered by removing the tags from the input documents. At the end, a final report is generated. Based on the filtering technique of CVOIP-FRU, the framework was able to generate 100% accuracy, 0% false positives and 0% false negatives for all available Betamax products, such as WebCallDirect, VoIPCheap, VoIPStunt, etc

IV. Discussions and Conclusions

CVOIP-FRU is a utility that will expedite the digital forensic investigations of systems that utilize Betamax products. Moreover, due to its cross platform scripting back end and filtering capabilities, this solution can be expanded to other popular VoIP Applications under various Operating Systems. CVOIP-FRU provides a report that includes usernames, contact lists, in addition to call and SMS logs with detailed information valuable on a court of law. The main purpose of CVOIP-FRU is to facilitate data gathering of the mentioned Betamax VoIP Applications and other VoIP products by reducing forensic analytical time via an automation framework. Providing a quick overview of the information stored along with the specific data containers within the registry and/or the file system. Law enforcement forensic analysts can then opt to save or back up the registry hives and files that contain the data. Presenting a framework that could potentially neutralize VoIP applications threats by making them traceable is a step towards limiting terrorist organization from using this technology to communicate during their malicious operations.

Although this framework provides forensically important information from the VoIP products, there are still some other areas where additional research is needed. One are that needs to be addressed is collecting Voice Messages. Voice Activity detection based on a conjugate structure algebraic code excited linear prediction can be utilized in order to accomplish voice message capture and detection [24]. Moreover, creating a mobile phone version of CVOIP-FRU which can be executed in mobile devices are the next tasks.

Along with the require research, there is a lot of online privacy laws that are currently being crafted by Congress. Hence the evolution of CVOIP-FRU will definitely be decided by the outcome of such regulations. Its usage could be limited to forensic investigators. Additionally, retrieving VoIP voice messages and live calls is currently under development pending legal implications related to privacy.

References

[1] In-Stat. “VoIP Penetration Forecast to Reach 79% of U.S. Businesses by 2012”, Scottsdale, Arizona. http://www.instat.com/newmk.asp?ID=2721
[2] J.C., Pelaez,. "Using Misuse Patterns for VoIP Steganalysis", *20th International Workshop on Database and Expert Systems Application*, pp. 160-164, August 31- September 4, 2009.
[3] J.C. Pelaez, E.B. Fernandez. "VoIP Network Forensic Patterns", *Fourth International Multi-Conference on Computing in the Global Information Technology*, pp. 175-180, August 23-39, 2009
[4] Computing Now. “Researchers Demonstrate that Eavesdropping on Encrypted VoIP is Possible”, http://www.computer.org/portal/web/news/home/-/blogs/researchers-demonstrate-that-eavesdropping-on-encrypted-voip-is-possible;jsessionid=f9d2ca28b6bd88d3649426247ce1?_33_redirect=%2Fportal%2Fweb%2Fnews%2Fhome%2F-%2Fblogs%2F
[5] J. Kahn. “Mumbai Terrorists Relied on New Technology for Attacks”, New York Times. http://www.nytimes.com/2008/12/09/world/asia/09mumbai.html
[6] Betamax. http://www.betamax.com/about.php
[7] W.A. Aziz, S.H. Elramly, and M.M. Ibrahim. "VoIP Quality Optimization in IP-Multimedia Subsystem (IMS)", *Second International Conference on Computational Intelligence, Modelling and Simulation*, pp. 546-552, September 28-30, 2010.
[8] T. Yang, K. Zheng, and Y. Yang. "A Novel VoIP Flooding Detection Method Basing on Call Duration", *2010 First International Conference on Pervasive Computing, Signal Processing and Applications*, pp. 1158-1162, September 17-19, 2010.
[9] K. Ting, F. Kuo, B. Hwang, H.C. Wang, and C. Tseng. "A Power-Saving and Robust Point Coordination Function for the Transmission of VoIP over 802.11", *International Symposium on Parallel and Distributed Processing with Applications*, pp. 283-289, September 6-9, 2010.
[10] M. Naeem, V. Namboodiri, and R. Pendse, R. "Energy implication of various VoIP codecs in portable devices", *2010 IEEE 35th Conference on Local Computer Networks*, pp. 196-199, October 10-24, 2010.
[11] T. Zourzouvillys, E. Rescorla. "An Introduction to Standards-Based VoIP: SIP, RTP, and Friends," *IEEE Internet Computing*, pp. 69-73, 2010.
[12] A.D. Keromytis. "Voice-over-IP Security: Research and Practice", *IEEE Security and Privacy*, pp. 76-78, 2010.
[13] S. Phithakkitnukoon, R. Dantu, and E. Baatarjav, E. "VoIP Security - Attacks and Solutions", *Information Security Journal: A Global Perspective*. Volume 17, no. 3: 114-123. 2008
[14] C. Leung Y. Chan. "Network Forensic on Encrypted Peer-to-Peer VoIP Traffics and the Detection, Blocking, and Prioritization of Skype Traffics", *16th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE 2007)*, pp. 401-408, June 18-20, 2007
[15] R.C. Dodge. "Skype Fingerprint", *41st Annual Hawaii International Conference on System Sciences (HICSS 2008)*, pp. 485, January 07-10, 2008.
[16] Ethical Hacker. http://www.ethicalhacker.net/content/view/127/24/
[17] Ghacks. http://www.ghacks.net/2008/11/22/skype-log-view/
[18] M. Simon, J. Slay. "Recovery of Skype Application Activity Data from Physical Memory", *2010 International Conference on Availability, Reliability and Security*, pp. 283-288, February 15-18, 2010
[19] H. Tienan M. Xiantu and L. Mingjie. "A New Dynamic Self-adaptive Diffluence Algorithm for VoIP Gateway of Intelligrid", *Third International Conference on Measuring Technology and Mechatronics Automation*, pp. 147-149, January 6-7, 2011
[20] W. Chen, T. Wu. "IPv6 VoIP Deployment on Taiwan Academic Network (TANet)", *2011 IEEE Workshops of International Conference on Advanced Information Networking and Applications*, pp. 795-799, March 22-25, 2011
[21] I. Lin, Y. Yen, B. Wu, and H. Wang. "VoIP Digital Evidence Forensics Standard Operating Procedure (DEFSOP)", *International Conference on Broadband, Wireless Computing, Communication and Applications*, pp. 407-412, November 4-6, 2010.

[22] A. Orebaugh, G. Ramirez, J. Beale, and J. Wright. “Wireshark & Ethereal Network Protocol Analyzer Toolkit”, Publisher: Syngress, 448 Pages, ISBN: 1597490733, 2007.
[23] C. Varol, J. Talburt. “Pattern and Phonetic Based Street Name Misspelling Correction”, *8th International Conference on Information Technology: New Generations (ITNG 2011)*, pp. 553-558. April 11-13, 2011, Las Vegas, Nevada, USA
[24] H. Qizhi, C. Qi, Y. Kai, and Y. Zhuping. "Design for New Kind of Low Rate Speech Codec in VoIP", *International Conference on Intelligent System Design and Engineering Application*, pp. 548-551, October 13-14, 2010

Alejandro Villegas was a M.S. student in the Department of Computer Science at Sam Houston State University, Texas, USA. His research interests are in Digital Forensics and Information Assurance.

Cihan Varol is an Assistant Professor of Computer Science at Sam Houston State University. His research interests are in the general area of information (data) quality, VoIP Forensics, and risk management with specific emphasis on personal identity recognition, record linkage, entity resolution, pattern matching techniques, natural language processing, multi-platform VoIP applications, VoIP artifacts data cleansing, and quality of service in business process automation. These studies have led to more than 30 peer-reviewed journal and conference publications, and one book chapter.

He received his Bachelor of Science degree in Computer Science from Firat University, Elazig, Turkey in 2002, Master of Science degree from Lane Department of Computer Science and Electrical Engineering from West Virginia University, Morgantown, WV, USA in 2005, and Doctor of Philosophy in Applied Computing from University of Arkansas at Little Rock, Little Rock, AR, USA in 2009.