

INTERNATIONAL JOURNAL OF

CYBER  
SECURITY  
AND  
DIGITAL  
FORENSICS

(IJCSDF)

ISSN 2305-0012

Volume 5, Issue 4  
2016



[www.sdiwc.net](http://www.sdiwc.net)

**Editor-in-Chief**

Dragan Perakovic, University of Zagreb, Croatia

**Editorial Board**

Ali Sher, American University of Ras Al Khaimah, UAE  
 Altaf Mukati, Bahria University, Pakistan  
 Andre Leon S. Gradvohl, State University of Campinas, Brazil  
 Azizah Abd Manaf, Universiti Teknologi Malaysia, Malaysia  
 Bestoun Ahmed, University Sains Malaysia, Malaysia  
 Carl Latino, Oklahoma State University, USA  
 Dariusz Jacek Jakóbczak, Technical University of Koszalin, Poland  
 Duc T. Pham, University of Birmingham, UK  
 E.George Dharma Prakash Raj, Bharathidasan University, India  
 Elboukhari Mohamed, University Mohamed First, Morocco  
 Eric Atwell, University of Leeds, United Kingdom  
 Eyas El-Qawasmeh, King Saud University, Saudi Arabia  
 Ezendu Ariwa, London Metropolitan University, United Kingdom  
 Fouzi Harrag, UFAS University, Algeria  
 Genge Bela, University of Targu Mures, Romania  
 Guo Bin, Institute Telecom & Management SudParis, France  
 Hadj Hama Tadjine, Technical university of Clausthal, Germany  
 Hassan Moradi, Qualcomm Inc., USA  
 Hocine Cherifi, Universite de Bourgogne, France  
 Isamu Shioya, Hosei University, Japan  
 Jacek Stando, Technical University of Lodz, Poland  
 Jan Platos, VSB-Technical University of Ostrava, Czech Republic  
 Jose Filho, University of Grenoble, France  
 Juan Martinez, Gran Mariscal de Ayacucho University, Venezuela  
 Kaikai Xu, University of Electronic Science and Technology of China, China  
 Khaled A. Mahdi, Kuwait University, Kuwait  
 Ladislav Burita, University of Defence, Czech Republic  
 Maitham Safar, Kuwait University, Kuwait  
 Majid Haghparast, Islamic Azad University, Shahre-Rey Branch, Iran  
 Martin J. Dudziak, Stratford University, USA  
 Mirel Cosulschi, University of Craiova, Romania  
 Monica Vladoiu, PG University of Ploiesti, Romania  
 Nan Zhang, George Washington University, USA  
 Noraziah Ahmad, Universiti Malaysia Pahang, Malaysia  
 Pasquale De Meo, University of Applied Sciences of Porto, Italy  
 Paulino Leite da Silva, ISCAP-IPP University, Portugal  
 Piet Kommers, University of Twente, The Netherlands  
 Radhamani Govindaraju, Damodaran College of Science, India  
 Ramadan Elaies, University of Benghazi, Libya  
 Rasheed Al-Zharni, King Saud University, Saudi Arabia  
 Talib Mohammad, University of Botswana, Botswana  
 Tutut Herawan, University Malaysia Pahang, Malaysia  
 Velayutham Pavanam, Adhiparasakthi Engineering College, India  
 Viacheslav Wolfengagen, JurInfoR-MSU Institute, Russia  
 Waralak V. Siricharoen, University of the Thai Chamber of Commerce, Thailand  
 Wen-Tsai Sung, National Chin-Yi University of Technology, Taiwan  
 Wojciech Zabierowski, Technical University of Lodz, Poland  
 Su Wu-Chen, Kaohsiung Chang Gung Memorial Hospital, Taiwan  
 Yasin Kabalci, Nigde University, Turkey  
 Yoshiro Imai, Kagawa University, Japan  
 Zanifa Omary, Dublin Institute of Technology, Ireland  
 Zuqing Zhu, University of Science and Technology of China, China

**Overview**

The International Journal of Cyber-Security and Digital Forensics (IJCSDF) is a knowledge resource for practitioners, scientists, and researchers among others working in various fields of Cyber Security, Privacy, Trust, Digital Forensics, Hacking, and Cyber Warfare. We welcome original contributions as high quality technical papers (full and short) describing original unpublished results of theoretical, empirical, conceptual or experimental research. All submitted papers will be peer-reviewed by members of the editorial board and selected reviewers and those accepted will be published in the next volume of the journal.

As one of the most important aims of this journal is to increase the usage and impact of knowledge as well as increasing the visibility and ease of use of scientific materials, IJCSDF does NOT CHARGE authors for any publication fee for online publishing of their materials in the journal and does NOT CHARGE readers or their institutions for accessing to the published materials!

**Publisher**

The Society of Digital Information and Wireless Communications  
 Miramar Tower, 13 Nathan Road, Tsim Sha Tsui, Kowloon, Hong Kong

**Further Information**

Website: <http://sdiwc.net/ijsdf>, Email: [ics@sdiwc.net](mailto:ics@sdiwc.net)  
 Tel.: (202)-657-4603 - Inside USA; 001(202)-657-4603 - Outside USA.

**Permissions**

*International Journal of Cyber-Security and Digital Forensics (IJCSDF)* is an open access journal which means that all content is freely available without charge to the user or his/her institution. Users are allowed to read, download, copy, distribute, print, search, or link to the full texts of the articles in this journal without asking prior permission from the publisher or the author. This is in accordance with the BOAI definition of open access.

**Disclaimer**

Statements of fact and opinion in the articles in the *International Journal of Cyber-Security and Digital Forensics (IJCSDF)* are those of the respective authors and contributors and not of the *International Journal of Cyber-Security and Digital Forensics (IJCSDF)* or *The Society of Digital Information and Wireless Communications (SDIWC)*. Neither *The Society of Digital Information and Wireless Communications* nor *International Journal of Cyber-Security and Digital Forensics (IJCSDF)* make any representation, express or implied, in respect of the accuracy of the material in this journal and cannot accept any legal responsibility or liability as to the errors or omissions that may be made. The reader should make his/her own evaluation as to the appropriateness or otherwise of any experimental technique described.

Copyright © 2016 sdiwc.net, All Rights Reserved

The issue date is December 2016.

## Volume 5, Issue 4

## CONTENTS

## ORIGINAL ARTICLES

<b>Method for Detecting a Malicious Domain by using only Well-known Information.....</b>	<b>166</b>
Author(s): Masahiro Kuyama, Yoshio Kakizaki, Ryoichi Sasaki	
<b>Digital Forensic Analysis of Ubuntu File System.....</b>	<b>175</b>
Author(s): Dinesh N. Patil, Bandu B. Meshram	
<b>A Preferential Analysis of Existing Password Managers from End-Users' View Point.....</b>	<b>187</b>
Author(s): S. Agholor, A. S. Sodiya, A. T. Akinwale, O. J. Adeniran and D. O. Aborisade	
<b>Intrusion Detection System with Spectrum Quantification Analysis.....</b>	<b>197</b>
Author(s): Yusuke Tsuge, Hidema Tanaka	
<b>“The Unwitting Danger Within - Detection, Investigation and Mitigation of a Compromised Network” .....</b>	<b>208</b>
Author(s): Emmanuel U Opara, Oredola A. Soluade	

## Method for detecting a malicious domain by using only well-known information

MASAHIRO KUYAMA, YOSHIO KAKIZAKI, RYOICHI SASAKI

Tokyo Denki University  
Tokyo, Japan  
kuyama@isl.im.dendai.ac.jp

### ABSTRACT

Damage caused by targeted attacks is a serious problem. It is not enough to prevent only the initial infections, because techniques for targeted attacks have become more sophisticated every year, especially attacks seeking to illegally acquire confidential information. In a targeted attack, the attacker wants to hide the C&C server so that it cannot be detected. Therefore, the C&C server may not be found by a web search engine. We pay attention to this lack of detection and the results of a web search engine. In this study, we propose a method for identifying the C&C server by using supervised machine learning and feature points obtained from WHOIS, DNS and search sites for domains of C&C servers and normal domains. Moreover, we conduct an experiment that applies real data, and we verify the usefulness of our method by cross-validation. The results indicated that we could obtain a high detection rate of about 99.3%.

### KEYWORDS

Malware, C&C server, Neural network, SVM, Targeted attack

### 1 Introduction

The development of the Internet has contributed to the enrichment of society. In particular, the Internet allows connections to be made very quickly all over the world.

These connections have not only a good side but also a bad side, which is Internet-based crime. In such crimes, damage caused by targeted attacks aimed at a specific organization or company is a serious problem [1]. Many targeted attacks aim at illegal acquisition of

confidential information, such as intellectual property. To achieve their objectives, attackers infect terminals with malware attached to e-mails and use the targeted attack to send information back to the command and control (C&C) server.

In Japan, many organizations, including a leading heavy-industry manufacturer, the House of Representatives, and the Japan Pension Service, have been subject to attacks and have suffered significant damage. Multi-layered countermeasures at the entry and exit points are required because it is very difficult to prevent attacks.

The sequence of targeted attacks consists of four steps, as shown in Figure 1.

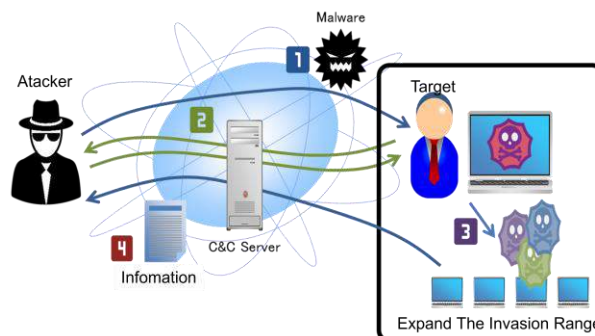


Figure 1 Sequence of targeted attacks

Step 1: A terminal such as a PC in a local area network (LAN) is infected with malware for the targeted attack.

Step 2: This terminal communicates with the C&C server. Then, more malware is downloaded to the terminal.

Step 3: The malware attempts to expand the invasion range to other PCs and servers in the LAN.



Step 4: Important information, confidential information and private information of the organization is transmitted to the C&C server owned by the attacker outside the LAN.

The malware used in the targeted attacks in step 1 is customized for each targeted organization and the malware is difficult for the anti-virus software to detect.

In targeted attacks, the C&C server is essential for the attack to succeed. Therefore, if we can detect the infection of a terminal in the LAN and the communication with the C&C server, we can guard against the expansion of damage. Also, the attacker wants to hide the C&C server so that it cannot be detected. In order to detect the anomalous communication, we must identify the C&C server in advance to detect the infection. New C&C servers are continuously made by attackers, and so their IP addresses are not typically on any blacklists. For this reason, we need to develop a method to find new C&C servers.

In this study, we extract the feature points from well-known information such as WHOIS, DNS, and the results of a web search engine for the C&C server domain, and we try to detect the C&C server by using a neural network.

In a presentation at DigitalSec2016, we showed that a method using WHOIS and DNS information could identify a C&C server and detect it with a 98.5% success rate [2]. In this paper, we report an improved detection rate by using the result of a web search engine.

The C&C server is hidden to avoid detection. This means that it is not typically found on search sites. Therefore, we assume that our search engine is effective to detect the C&C server.

We extract feature points according to their difficulty of spoofing: valid terms, expiration dates, and e-mail addresses from the WHOIS information, number of mail exchanger (MX) records, number of name server (NS) records from the DNS information, and results of the web search engine.

## 2 Related Work

Studies for specifying the C&C server are classified into the following two types.

(1) Studies that focus on communication packets between the bot PC and the C&C server

Jang et al. [3] and Lu et al. [4] proposed methods to detect a C&C server by analyzing the payload of communication packets between the bot PC and the C&C server. Ikuse et al. [5] proposed methods to detect a C&C server in order to identify the falsification of communication data by performing a malware analysis that applies taint analysis technology.

These methods have a high detection rate, because the data body is used for verification and excludes header information, such as the destination address and the source address, which might have a specification change, such as the port number or the proprietary protocol of the transport layer.

However, we have to observe real-time communication. Moreover, inadequacies in the response to unverified issues such as zero-day attacks must be solved.

(2) Studies that focus on domain information for the C&C server

Tsai et al. [6] reported a detection method based on a data mining technique called RIPPER, which uses a combination of information obtained from the domain information and the external repositories of the C&C server. Felegyhazi et al. [7] proposed a method for estimating the identity of an unknown malignant domain from WHOIS and DNS information. Ma et al. [8] proposed a technique using machine learning and DNS, WHOIS, and geographic information for the URL. Invernizzi et al. [9] reported a method for estimating the identity of an unknown malignant domain by using a search engine to

obtain information such as the content of a WHOIS known malignant website.

Although the detection failure rate is high, this method does have an adequate accuracy rate for detecting C&C servers.

In our previous study, we proposed a detection technique that had a 96.5% detection rate in 2009 [10].

Our method used a valid term and reverse lookup of C&C domain information from DNS and WHOIS information. Therefore, acquisition of the data required for analysis was easy. In addition, the method was highly unlikely to be affected by malware because it did not need direct access to the C&C server.

We have been continuing our investigation of the detection rate, which has decreased over time [11]. As shown in Table 1, the detection rate for the data of 2009 was 96.5%. However, the detection rate fell to 85.0% in 2010 and 76.2% in 2011.

Table 1 Detection rates of our method over time

Method year	Detection rate by year (%)				
	2009	2010	2011	2013	2014
2009	96.5	85.0	76.5	-	-
2011	-	-	95.2	42.5	-
2013	-	-	-	80.3	80.8
2014	-	-	-	-	96.7

These results revealed that the 2009 parameters were not suitable for 2010 and 2011[11].

We updated the discriminative model by using recent data to optimize the detection method in each period [12]. Although our results improved, it was necessary to frequently update the discriminative model.

In our 2014 update, we revised our method to use quantification theory and machine learning. The result was still not high enough, although the detection rate improved to 96.7% in 2014 [11].

We improved the detection rate to 98.5% in 2016 [2]. Table 2 shows the changes in the characteristics used in the ongoing investigation.

Table 2 Changes in the characteristics used

Features using		Model year				
		'09	'11	'13	'14	'16
DNS	Reverse resolution	✓	✓	✓		
	TTL				✓	
	minimum	✓	✓		✓	
	A records		✓	✓		
	MX records					✓
	NS records				✓	✓
	CNAME records			✓		
WHOIS	TXT records				✓	
	Valid terms	✓	✓	✓	✓	✓
	e-mail addresses					✓
Total		3	4	4	5	4

We used WHOIS and DNS information for detection in the previous studies. Once we added new features for a search site, the detection rate went up.

### 3 Methods

Our proposed method focuses on the domain of the C&C server.

The method uses well-known information such as WHOIS, DNS, and the result of web search engine for the domain of the C&C server. Our method can identify C&C servers by extracting the feature points for machine learning.

To classify a domain as malignant (C&C) or benign (normal), we use machine learning to construct a training model in advance.

#### 3.1 Detection method

First, we prepare benign domains and malignant domains. Then, feature points are extracted from the WHOIS, DNS, and search information for each domain.

The extracted features are used in machine learning to construct a training model (Figure 2). The training model determines whether an accessed domain is malignant or benign.

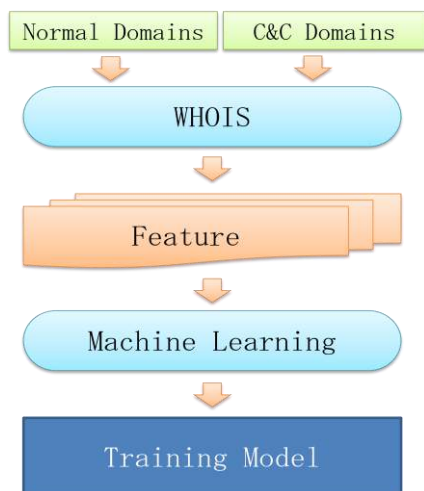


Figure 2 Flow of detection method

### 3.2 Preparing domains

We prepare two types of domains: normal domains and C&C domains.

We choose the normal domains from "the top 500 sites on the web" of Alexa [13], because the top sites have highly secure domains, which best represent normal domains.

The C&C servers are extracted by analyzing Emdivi, PlugX, and PoisonIvy, which are major malwares for targeted attacks [14][15].

We obtain 163 malwares by using VirusTotal [16]. Table 3 shows the breakdown of the collected malwares.

Table 3 Collected malwares from VirusTotal

Malware type	Samples
Emdivi	50
PlugX	63
PoisonIvy	50

The collected malwares are deeply analyzed by using the Sandbox analyzer called LastLine [17]. LastLine extracted 54 domain destinations as the analysis results.

### 3.3 Features of WHOIS

WHOIS is a service that provides management and information for the registration of a

domain. Technical specifications and operational rules of WHOIS are established in RFC812 [18] and RFC3912 [19].

We can obtain the following information from WHOIS.

- Registered domain name
- Registrar name
- DNS server name for the registered domain
- Valid term for the domain
- Expiration date for the domain
- Domain name registrant contact
- Person in charge for technical contact
- Contact for registration personnel
- Contact point for the registrant

It is difficult to tamper with the information from a) to e). The valid period d) for normal servers is long, but that for C&C servers is short, because C&C domains are canceled if their purpose is achieved [7][8][9]. From this viewpoint, we calculate the valid term by subtracting the date in d) from the date in e). Figure 3 shows the valid period for C&C domains and normal domains.

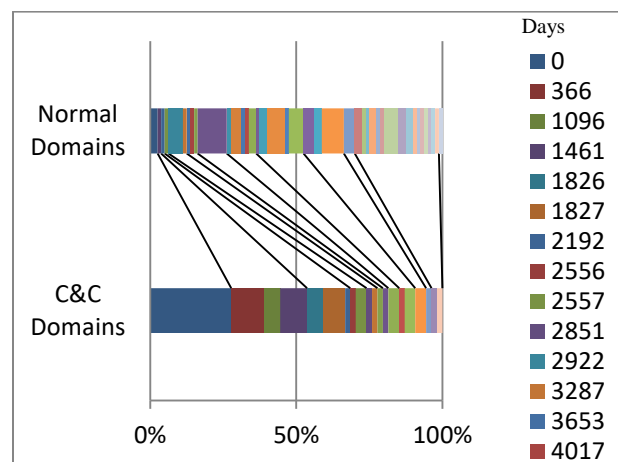


Figure 3 Valid terms for domains

As can be seen in Figure 3, the valid terms for C&C domains are shorter than those for normal domains.

Next, we obtain the following information for each contact person from f) to i).

- a) ID
- b) Name
- c) Organization name
- d) Address
- e) Postal code
- f) Phone number
- g) Country
- h) Fax number
- i) E-mail address

All of the above information can be easily falsified.

Especially, the registration information for most C&C domains is false, because attackers often use WHOIS registration agency services to hide. However, the probability of a true e-mail address is high even if the other information is false, because the e-mail address is required for contact.

Thus, we pay attention to e-mail addresses obtained from WHOIS. First, we extract e-mail addresses from WHOIS for normal domains and then for C&C domains, and then we conduct data mining.

We extract the features for each domain by using a text mining tool called "UserLocal" [20].

Figures 4 and 5 show the co-occurrence network, which is the appearance pattern for words used in e-mail addresses, for normal domains and C&C domains, respectively.

The co-occurrence network shows relations by structuring the word patterns used in the text. Similar words in the appearance pattern are connected by a line.

We reveal the structures of the e-mail addresses for the domains and extract the features by using the word patterns.

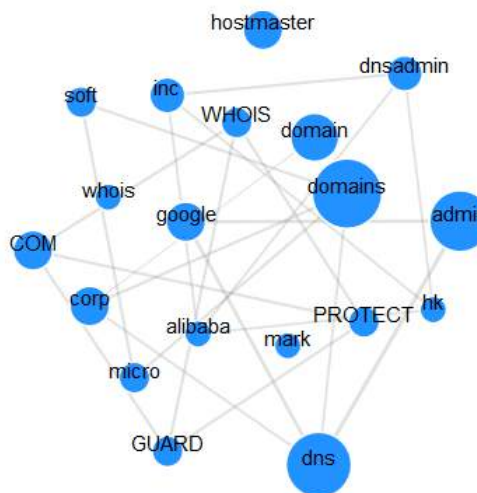


Figure 4 Co-occurrence network for normal domains

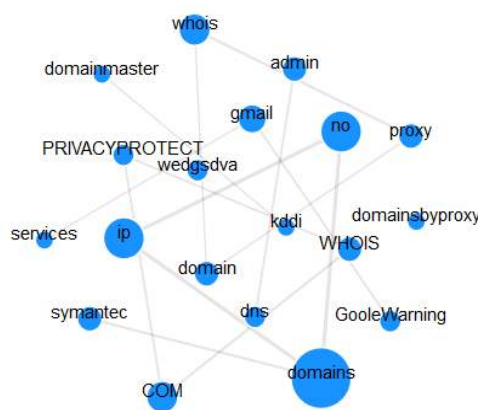


Figure 5 Co-occurrence network for C&C domains

The co-occurrence network for normal domains has a large structure connected with a plurality of words, and two patterns connected with four types of words. On the other hand, the co-occurrence network for C&C domains has three patterns connected with three types of words. When examined closely, "no", "proxy", and "PRIVACYPROTECT", which are usually used by WHOIS registration agency services, are included in Figure 5.

We should point out that the WHOIS registration agency services used for normal domains and C&C domains are different.



Therefore, we choose three features of WHOIS information: domain name, e-mail address, and valid term.

### 3.4 Features of DNS

DNS is a system that translates domain names into IP addresses. Technical specifications and operational rules of DNS in RFC1034 [21] and RFC1035 [22] are determined.

We can obtain the following records from the DNS.

- a) Address (A) record
- b) Start of authority (SOA) record
- c) Host information (HINFO) record
- d) MX record
- e) NS record
- f) Canonical name (CNAME) record
- g) Well-known services (WKS) record
- h) Text (TXT) record

The numbers of registered records for the NS record and the MX record show a remarkable difference. Figure 6 shows the number of NS records for normal domains and C&C domains. Figure 7 shows the number of MX records for normal domains and C&C domains.

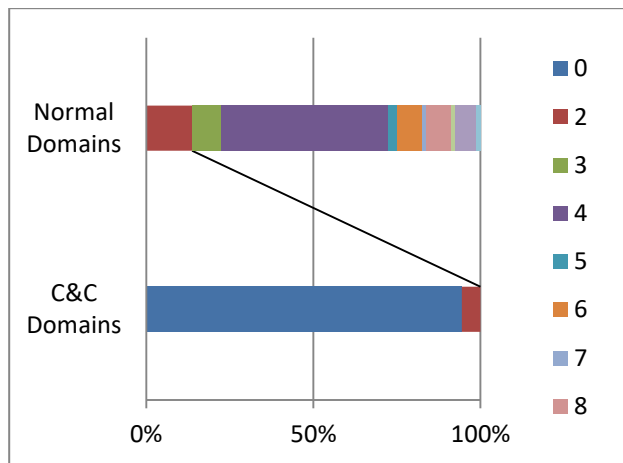


Figure 6 Number of NS records

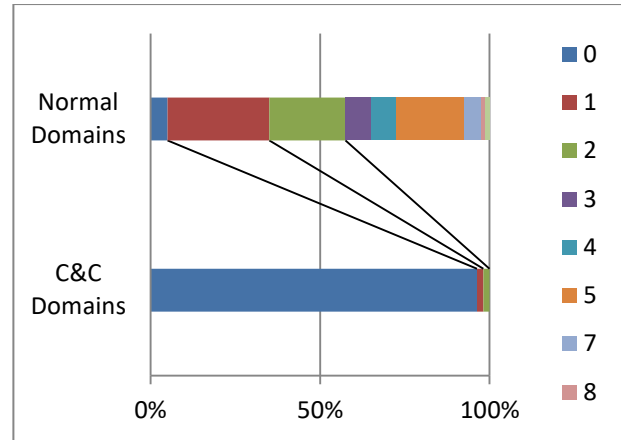


Figure 7 Number of MX records

Almost all records are not registered in the C&C domains, although many records are registered in the normal domains.

Thus, we choose these two features of DNS information: number of NS records and number of MX records.

### 3.5 Features of search site

A related study [9] detected an unknown C&C server by using a search engine to find the characteristics of a known C&C server in a drive-by-download attack.

In drive-by-download attacks, PCs are infected with malware by browsing websites. The malicious websites, which are infected with malware, conduct search engine optimization (SEO) to introduce more malware. It is assumed that the purpose of the SEO is to attract customers.

On the other hand, the website may not be used for malware infection in the targeted attack, because the malware is sent to the target directly by spoofing e-mail, etc. The attacker wants to hide the C&C server for the targeted attack so that it cannot be detected. In addition, a short-lived C&C server cannot be found by the crawler of the web search engine. Therefore, the C&C server may not be found by the web search engine. We note that this feature should receive particular attention.

In the present study, finding an evaluation domain by using the Google search engine was

investigated with regard to hits or non-hits. The results are shown in Figure 8.

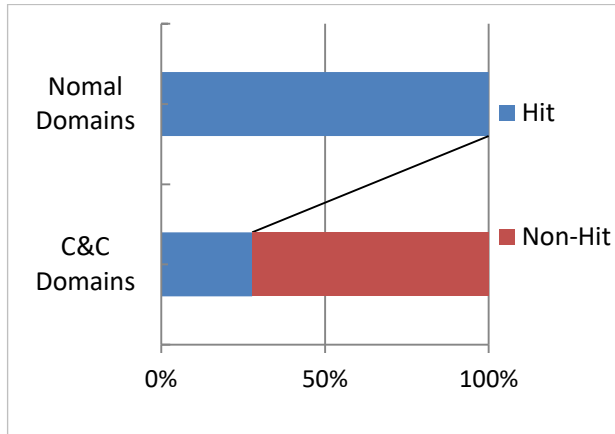


Figure 8 Search sites finding C&C servers

As shown in Figure 8, many C&C domains were not hit by the search site. Some hit C&C domains seemed to be hijacked servers. In targeted attacks, the C&C servers prepared by the attackers themselves are used.

### 3.6 Training model and algorithm

We construct a training model using a support vector machine (SVM) [24] and a neural network [25] as the algorithm for machine learning.

The SVM is a machine learning method that performs classification into two classes by pattern recognition [26].

A neural network is a type of supervised learning method.

It is possible to express the relation between the input and the output by mathematical modeling of some of the features found in human brain functions. The standard method is a hierarchical neural network using two layers.

We construct a training model by using a neural network with e-mail addresses and valid terms from WHOIS, the number of NS records, the number of MX records from the DNS, and the number of hits from a search site.

Table 4 shows the features included in machine learning.

Table 4 Features of machine learning

Input		Type
Label		Normal or C&C
Domain		String
WHOIS	Admin mail address	String
	Registered mail address	String
	Technical mail address	String
	Valid term	Number
DNS	NS record	Number
	MS record	Number
Search site		Hit or Non-Hit

## 4 Results

For evaluation, 80 normal and 54 C&C domains were used.

Because the amount of data was small, the accuracy could be low depending on how we chose the test data.

The amount of provided data for a particular domain used for targeted attacks was small. Thus, we evaluated the data with a cross-validation method, because it can reduce the error margin even for a small amount of data.

The cross-validation method divides the original data into block units [27]. One of the blocks is the test data, and the others are the learning data for evaluation.

The evaluation consisted of calculating the average of each evaluation result as the estimated accuracy (Figure 9).

This evaluation method can increase the estimation accuracy even for a small amount of data. The accuracy is calculated as follows.

Let  $N^{ts}$  be the total number of test data, and  $t^{ts}$  be the total number of data classified accurately, such that  $A^{ts}(d^n) = \frac{t^{ts}}{N^{ts}}$ . The  $n$ -th evaluation accuracy is estimated as follows:

$$A^{CV}(d) = \frac{1}{n} \sum_i^n A^{ts}(d^i) \quad (1)$$



Figure 9 Cross-validation method

The results of the evaluation of each combination of WHOIS, DNS, and search site by the SVM and the neural network using the cross-validation method are shown in Table 5.

Table 5 Detection rates by cross-validation

Combination	SVM	Neural network
WHOIS only	88.8%	88.8%
DNS only	96.3%	95.5%
Search site only	88.8%	88.8%
WHOIS + DNS	97.8%	98.5%
WHOIS + Search site	91.8%	92.5%
DNS + Search site	99.3%	99.3%
WHOIS + DNS + Search site	99.3%	99.3%

As a result, the SVM and neural network achieved a superior detection rate of 99.3%. The WHOIS only or the search site only achieved an 88.8% detection rate. However, the DNS only achieved a higher detection rate. The

results of the WHOIS and the search site indicate that the DNS is an important element. The result of adding the search site to the WHOIS and the DNS also improved the detection rate. Therefore, it is effective to use the search site. Moreover, removing WHOIS from the WHOIS, the DNS and the search site did not change the detection rate. The C&C server constructed by the attacker was not hit by the search site. Therefore, the search site is effective for detection of the C&C server. However, the search is not valid if the attacker hijacks a server, so an attacker hijacking a server is detected by the combination of WHOIS and the DNS.

## 5 Conclusion

In this paper, we collected the feature points of e-mail addresses used for C&C domains and proposed a method to determine C&C servers by using machine learning with well-known information such as WHOIS, DNS, and the result of a web search engine. We clarified the features of WHOIS registration agency services used for C&C domains by illustrating the relation of words in extracted e-mail addresses in co-occurrence networks. Moreover, the use of search sites for detection of C&C servers was found to be effective. Finally, we evaluated domain names and e-mail addresses. The valid terms from WHOIS, the number of NS records, the number of MX records from the DNS, and the number of search sites returned by Google were input for machine learning. As a result, we were able to identify the C&C server at a high detection rate of 99.3%. In future work, we intend to improve the accuracy by revising the machine learning algorithms, input values, and preprocessing.

## REFERENCES

- [1] Cyber GRID View vol.1 English Edition  
[http://www.lac.co.jp/security/report/pdf/apt\\_report\\_vol1\\_en.pdf](http://www.lac.co.jp/security/report/pdf/apt_report_vol1_en.pdf)
- [2] M.Kuyama, Y.Kakizaki, R.Sasaki, "Method for Detecting a Malicious Domain by using WHOIS and DNS features" The Third International Conference on Digital Security and Forensics (DigitalSec2016), pp. 74-80(2016).
- [3] D.I.Jang, M.Kim, H.C. Jung, B.N. Noh, "Analysis of HTTP2P Botnet: Case Study Waledac"2009 IEEE 9th Malaysia International Conference on Communications (Micc), pp. 409-412(2009).
- [4] W.Lu, M.Tavallae, A.A.Ghorbani, "Automatic Discovery of Botnet Communities on Large-Scale Communication Networks" ASIACCS '09 Proceedings of the 4th International Symposium on Information, Computer, and Communications Security(2009).
- [5] T.Ikuse, K.Aoki, T.Yagi, T.Hariu, "Identifying C&C Server Based on Modified Data Descent Analysis" Proceedings of the 2014 IEICE SOCIETY Conference (2014).
- [6] M.H.Tsai, K.C.Chang, C.C.Lin, C.H.Mao, H.M.Lee, "C&C Tracer: Botnet Command and Control Behavior Tracing" IEEE International Conference on Systems, Man and Cybernetics (SMC), Anchorage, AK, pp.1859-1864(2011).
- [7] M.Felegyhazi, C.Kreibich, V.Paxson, "On the Potential of Proactive Domain Blacklisting" USENIX Conference on Large-scale Exploits and Emergent Threats, pp.6 (2010).
- [8] J.Ma, L.K.Saul, S.Savage, G.M.Voelker, "Beyond Blacklists. Learning to Detect Malicious Web Sites from Suspicious URLs" ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp.1245-1254(2009).
- [9] L.Invernizzi, S.Benvenuti, P.M.Comparetti, M.Cova, C.Kruegel, G.Vigna. EvilSeed, A Guided Approach to Finding Malicious Web Pages" IEEE Symposium on Security and Privacy, pp.428-442(2012).
- [10] H.Mihara, R.Sasaki, "Proposal and Evaluation of Technique to Detect C&C Server on Botnet Using Attack Data (CCCDATASET2009) and Quantification Methods Type II" Journal of Information Processing Society of Japan Vol. 51, No. 9, pp. 1579-1590(2010).
- [11] S.Okayasu, R.Sasaki, "Proposal and Evaluation of Methods Using the Quantification Theory and Machine Learning for Detecting C&C Server Used in a Botnet" Computer Software and Applications Conference (COMPSAC) 2015 IEEE 39th Annual Vol. 03, pp. 24-29(2015).
- [12] N.Nakamura, R.Sasaki, "Evaluation of Technique to Detect C&C Server of Botnet Using Accumulated Data" Proceedings of Computer Security Symposium 2011(CSS2011), pp. 456-461(2011).
- [13] Alexa Top 500 Global Sites<http://www.alexa.com/topsites>
- [14] Targeted Attack Trends 2014 Annual Report<https://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/reports/rpt-targeted-attack-trends-annual-2014-report.pdf>
- [15] Trendmicro Press  
<http://www.trendmicro.co.jp/jp/about-us/press-releases/articles/20150409062703.html>
- [16] VirusTotal  
<https://www.virustotal.com/>
- [17] LastLine  
<https://www.lastline.com/>
- [18] RFC954 NICNAME/WHOIS  
<https://www.ietf.org/rfc/rfc954.txt>
- [19] RFC3912WHOIS Protocol Specification  
<http://www.ietf.org/rfc/rfc3912.txt>
- [20] User Local  
<http://textmining.userlocal.jp/>
- [21] DOMAIN NAMES - CONCEPTS AND FACILITIES  
<http://www.ietf.org/rfc/rfc1034.txt>
- [22] DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION  
<http://www.ietf.org/rfc/rfc1035.txt>
- [23] Google  
<https://www.google.com/>
- [24] V.Vapnik, A.Lerner, "Pattern recognition Using Generalized Portrait Method" Automation and Remote Control24, pp.774-780(1963).
- [25] Multilayer Perceptron<http://deeplearning.net/tutorial/mlp.html>
- [26] P.John, "Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines" Technical Report MSR-TR-98-14, pp.1-21(1998).
- [27] R.Kohavi, "A Study of Cross-validation and Bootstrap for Accuracy Estimation and Model Selection" Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence 2 (12), pp.1137-1143 (1995).



## Digital Forensic Analysis of Ubuntu File System

Dinesh N. Patil, Bandu B. Meshram  
Veerмата Jijabai Technological Institute  
Matunga, Mumbai, India  
dinesh9371@gmail.com, bbmeshram@vjti.org.in

### ABSTRACT

A file system of Ubuntu operating system can conserve and manage a lot of configuration information and the information with forensic importance. Mining and analyzing the useful data of the Ubuntu operating system have become essential with the rise of the attack on the computer system. Investigating the File System can help to collect information relevant to the case. After considering existing research and tools, this paper suggests a new evidence collection and analysis methodology and the UbuntuForensic tool to aid in the process of digital forensic investigation of Ubuntu File System.

### KEYWORDS

File System, Digital Forensic, Integrated Analysis, Timeline Analysis, Digital Evidence

### 1 INTRODUCTION

The Ubuntu operating system is one of the distributions of the Linux operating system. Most of the Ubuntu kernels are the default Linux kernel. Ubuntu uses the Linux file system which is usually considered as a tree structure. Ubuntu is having Ext4 as its default file system. Ext4 is an evolution of Ext3, which was the default file system earlier. The evolution of the Ext file system is summarized in table 1. Linux computers are very much prone to attack from the hackers. Linux boxes are often used as servers, essentially for a central control point. In fact, roughly 70% of malware downloaded by hackers to the honeypots is infected with Linux/Rst-B [1]. Linux-based web servers are constantly under attack. At

SophosLabs, an average of 16,000-24,000 websites were compromised in a day in 2013 [2]. Linux systems are indeed attacked by malware.

The Microsoft's operating system design includes some features that make documents able to install executable payloads. The use of a database of software hooks and code stubs (the registry) also simplified things [3]. Linux malware is quite distinct from what it does and how it does it, compared to Windows viruses, but it exists. The crucial operating system directories might be used by the malware to affect the computer system as a whole. In addition, there is always the risk of the malicious insider. Attacks directed at Linux systems tend to aim at exploiting bugs in system services such as web browsers or Java containers. These don't frequently run with elevated privileges either, so an exploit is typically contained to altering the behavior of the targeted service and, possibly, disabling it. The malware uses the various directories in the Linux file system to plant it to run as a service and harm the Computer. Also, the activity of the malicious insider also gets stored in the file system. This raises the need to do the forensic investigation of directories under the Linux file system to find the traces of malicious activities on the system.

The paper is organized as follows: Section 2 discusses the related work and the existing tools on the Linux file system forensics. The potential locations of the digital evidences in the directory structure of the Ubuntu File System are discussed in section 3. Section 4 covers the forensic investigation of the various user activities on the Linux file system. The proposed UbuntuForensic tool is discussed in section 5. Comparative study between the existing Linux tools and the proposed tool is performed in

Table 1. EXT Family features and limitation

Linux File System	Year of Introduction	Features	Limitation
EXT	1992	Virtual File system concept used	No support for separate timestamp for file access
EXT2	1993	File Compression added	No journaling feature
EXT3	1999	Journaling added, online file system growth	Lack feature such as extents, dynamic allocation of inodes and block suballocation
EXT4	2006	Extent-based storage, backward compatibility with EXT2 and EXT3, Online defragmentation	Do not overwrite the file after deletion causing security problem

section 6. The findings are concluded in section 7.

## 2 RELATED RESEARCH

This section details out the existing research on the Linux file system forensic and the tool developed to carry out the forensic investigation of it.

### 2.1 Existing Research

The logging system is the most important mechanism for Computer forensics on an Operating System. The various logging mechanism in Linux system that can be of forensic importance is discussed in [4]. A comparative study of the various file systems in Ubuntu Linux and Free BSD is performed in [5]. In order to meet the Linux file system analysis applications demand for computer forensics, an object-oriented method of analyzing Linux file system is proposed in [6]. The paper also analyzed different data sources deeply with the inheritance relationship between classes and the encapsulation of class and showed information of Linux file to the users in a friendly interface. The Linux operating system has been used as a server system in plenty of business services worldwide. Unauthorized intrusions on a server are

constantly increasing with a geometric progression. Conversely, the protection and prevention techniques against intrusion accidents are certainly insufficient. A new framework to deal with a compromised Linux system in a digital forensic investigation is developed and implemented in [7]. Issues pertaining to the Linux Forensics and the various forensic tools for the forensic investigation of the Linux system have been discussed in [8].

### 2.2 Existing Tools

**The Sleuth kit(TSK).** It is a collection of Unix-based command line analysis tools. TSK can analyze FAT, NTFS, Ext2/3, and UFS file systems and can list files and directories, recover deleted files, make timelines of file activity, perform keyword searches, and use hash databases.

**Autopsy.** This tool is a graphical interface to the TSK. It also analyzes FAT, NTFS, Ext2/3, and UFS file systems and can list files and directories, recover deleted files, make timelines of file activity, perform keyword searches, and use hash databases.

**Scalpel.** Scalpel is an open source file carver which is also available for Linux. File carvers are used to recover data from disks and to retrieve

files from raw disk images. In some case, file carvers are even able to retrieve data if the metadata of the file system were destroyed. Scalpel is designed to use minimal resources and to perform file carving.

#### **Digital Evidence and Forensic Toolkit (DEFT)**

**Linux.** DEFT is a free computer forensics Linux distribution. DEFT is combined with the Digital Advanced Response Toolkit (DART) which contains a collection of forensics software for Windows.

#### **Computer Aided Investigative Environment (CAINE).**

CAINE is a Linux live distribution which aims to provide a collection of forensics tools with a GUI. It includes open source tools that support the investigator in four phases of the forensic process viz., Information gathering, collection, examination, analysis. It also supports the investigator by providing capabilities to automate the creation of the final report and is completely controlled by a GUI that is organized according to the forensics phases.

**i-Nex.** It is an application that gathers information for hardware components available on the system and displays using user interface [9].

**History.** The history command lists commands that were recently executed. This can help to track the activity of an intruder.

### **3 UBUNTU FILE SYSTEM ANALYSIS**

In Ubuntu Operating System, the information about the actions performed on the system is maintained in the file system. The careful analysis of the file system leads in finding helpful evidence of the user's activity on the system.

The following are some of the files and directories in the file system which can be helpful to the forensic investigator to find the potential digital evidence of the various activity being performed on the system. The evidence identified

in each directory of the Ubuntu File System are discussed as below:

**/etc/rc.d.** In the case of Ubuntu, the information about the programs which are to be executed when the system booted is available in the file stored /etc/rc.d directory. The malicious user might gain an access to the Ubuntu system & will add files in rc.d directory to execute its malicious script. So whenever the Ubuntu System will boot up the malicious script will automatically run. The forensic examiner will have to look into those files to identify if any file contains malicious code which may be causing unauthorized activity on the system.

**/etc/init.d.** To remain running after reboots, malware is usually re-launched using some persistence mechanism available in the various startup methods on a Linux system, including services, drivers, scheduled tasks, and other startup locations. There are several configurations files that Ubuntu uses to automatically launch an executable when a user logs into the system that may contain traces of malware programs. Malware often embeds itself as a new, unauthorized service. Ubuntu has a number of scripts that are used to start the service as the computer boots. The startup scripts are stored in /etc/init.d. Malware program may embed itself in /etc/init.d directory to run as a service. Therefore the forensic examiner will have to look into those files to check for malware incident.

#### **/etc/NetworkManager/system-connections.**

Ubuntu maintains the list of networks connected to the system in /etc/NetworkManager/system-connections. In addition to this, it is possible to know the active network connections which are being used in the system using the command "sudo netstat -tupn".

**/etc/passwd.** The passwd file maintains the details about the users accessing the system. The details include the user name, path to the user's home directory, programs that are generally

started when the users log on. The forensic investigator can come to know about the users working directory, and the program that are executed when the user performs the login.

**/etc/shadow.** The shadow maintains the authentication details of the user. The details included in shadow file are user login name, salted password.

**/etc/profile.** Files and commands to be executed at login or startup time by the Bourne or C shells. These allow the system administrator to set global defaults for all users.

**/etc/networks.** The list of the networks that the system is currently located on is available in this directory.

**/etc/hosts.** The IP address of the machine is available in the hosts file if the machine is connected to the network. The forensic investigator can come to the conclusion whether the system was connected to the network or not.

**/etc/cron.d, /etc/cron.daily, /etc/cron.weekly, /etc/cron.monthly.** These directories contain scripts to be executed on a regular basis by the cron daemon. The investigator has to look into those directories to search for the presence of any malicious code in it.

**/usr/bin.** In Ubuntu, the configuration information about the application is stored in the /usr/bin directory and the library required for these applications is available in the /usr/lib directory. The list of the application installed can be obtained by the command `ls -l /usr/bin/`. The directory /usr/share/ application also provides the graphical view of the application installed Using the information available in the bin directory, analyst can provide the historic view of the application configuration that the user has installed onto the system, date on which a particular application was modified, permissions granted to the user, size of the application.

**/usr/lib.** This directory contains program libraries. Libraries are collections of frequently used program routines. The investigator has to search in the lib directory to search for any malicious file.

**/usr/local/share/recently-used.xbel.** In Ubuntu, the files which have been recently accessed are noted in the file 'recently-used.xbel'. This file is available in the local/share/ directory. The 'cat' command can be used to read the contents of the recently-used.xbel. Recently-used.xbel file provides the detailed information about the files which have been accessed by the user, the application used to access those documents and the timing of accessing & modifying these documents.

**/var/log/syslog.** In Ubuntu, the login time and the logout time can be accessed by using the last command at the terminal. Syslog file in the /var/log maintains the login and shutdown time. The analyst can predict the criminal, if the crime had happened during the duration of the use of the system by the user. Syslog file in /var/log provides the date and time at which a particular network connection was established. Network information enables the forensic examiner to know about the type of network used in order to do malicious activity.

**/var/log/lastlog.** The lastlog file contains the recent login information for all the users. The lastlog command provides the content of this file. The Forensic Investigator can come to know about the user who was logged in at the time of crime.

**/var/log/faillog.** It contains user failed login attempts. The user who was under attack can be identified.

**/var/tmp.** The tmp directory consists of temporary files. These files can provide the details about the files that were accessed by the user.



**/dev.** Hardware devices attached to the system. Also the /dev directory in the file system provides the information about the hardware attached to the system. The syslog also maintains the details of the devices which have been detected. The date and timing at which the device was connected along with device details are recorded in the syslog. The device information provides the knowledge about the kind of devices and the time at which they were used in doing malicious activity.

**/proc/net/netstat.** The netstat file maintains the network statistics about the network connections of the system. The suspicious connections if there are any will be identified by the investigator

**/proc/net/dev\_mcast.** The statistic about the network device connected to access the network is available in the dev\_mcast file.

**/proc/cpuinfo.** The information about the cpu connected to the system is available in the cpuinfo file.

**/proc/PID/exe.** Exe directory contains the Link to the executable of this process with the process identification i.e., PID. If there are any malicious codes running for this process, then it can be detected.

#### 4 EVIDENCE COLLECTION USING PROPOSED TOOL

The forensic investigator should be able to analyze the activities of the user when performing the investigation and in doing so the timing of the activities is needed to be considered to establish the correlation between the time and the activity. As the details of the user's activities are recorded in the various files managed by the file system of the Linux based Computer System. The investigator should be able to investigate the files stored in the seized hard disk of the computer system which was used to commit the crime.

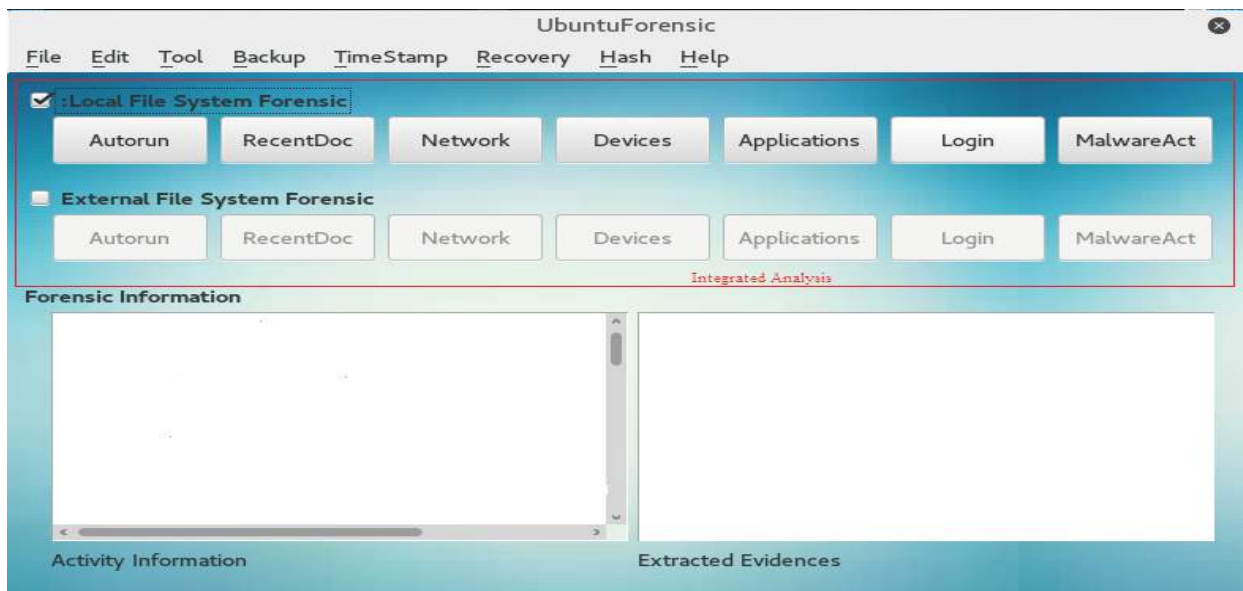


Figure 1. A snapshot of UbuntuForensic tool showing Integrated Analysis

However, the previous forensic tools provided limited facilities for performing the forensic

analysis of Linux file system. For this reason, a new evidence collection and analysis

methodology is required. This methodology performs integrated file system analysis, timeline analysis and extracts the information that is useful for the digital forensic analysis of the file system.

#### 4.1 Integrated Analysis

The cyber crime cell generally used to seize the hard disk of the computer which is used for crime purpose. The forensic investigator has the responsibility to find out the possible traces of evidence against the criminal. The Linux-based computer system maintains the files in the directory structure which begin with root directory '/'.

The proposed UbuntuForensic tool provides the facility for extracting the forensic evidence from the files stored in the external hard disk. This hard disk is needed to be connected to the computer system having a UbuntuForensic tool which mounts the external directory structure in the media directory of the running system to extract the evidence. The proposed tool also performs Local file system forensic which involves extracting the information from the files about the various activity performed by the user on the system, on which the tool is running.

#### 4.2 Analysis of User Activity

The existing tools provide a limited functionality in extracting the forensic information from the file system. This has stimulated the need of having a file system forensic tool which can extract the forensic data from the directory structure based on the various activities being performed by the user and generate a report of the evidence for further use.

The proposed UbuntuForensic tool covers the various activities as discussed in [10], which are performed on the Computer system. These activities include:

- Autorun programs running on the system
- Recently accessed documents/programs,
- Applications installed on the system
- Network connected
- Devices connected to the system
- Last login activity of the user
- Malware activity

The detail of these activities is as follows:

##### **The Autorun programs running on the system**

Many programs are configured in such a way that when the Computer boot and start the operating system, they automatically start running such programs are called as Auto Run program. In the case of Ubuntu, the information about the programs which are to be executed when the system booted is available in the file stored /etc/rc.d directory. The malicious user might gain an access to the Ubuntu system & will add files in rc.d. So whenever the Ubuntu System will boot up the malicious script will automatically run. The forensic examiner will have to look into those files to identify if any file contains malicious code which may be causing unauthorized activity on the system.

##### **Recently Accessed documents and programs**

From the documents that the user has recently accessed, the forensic examiner can know about the documents in which the user has interest. In Ubuntu, the files which have been recently accessed are noted in the file 'recently-used.xbel'. This file is available in the local/share/ directory. The 'cat' command can be used to read the contents of the recently-used.xbel file. Recently-used.xbel file provides the detailed information about the files which have been accessed by the user, the application used to access those documents and the timing of accessing & modifying these documents.

The recently accessed document information helps in understanding the files which may have been read, modified by the user.

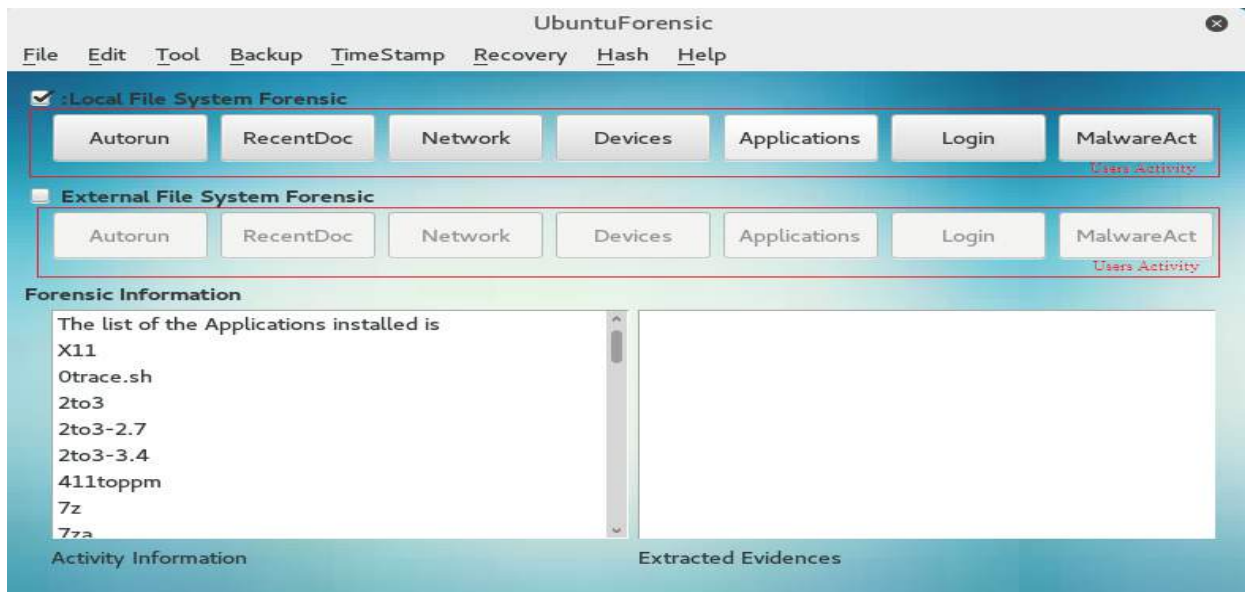


Figure 2. A snapshot of UbuntuForensic tool showing category of User Activities

### Applications installed on the system

In Ubuntu, the configuration information about the application is stored in the `/usr/bin` directory and the library required for these applications is available in the `/usr/lib` directory. The list of the application installed can be obtained by the command `ls -l /usr/bin/`. Using the information available in the bin directory, an analyst can provide the historic view of the application configuration that the user has installed onto the system, date on which a particular application was modified, permissions granted to the user, the size of the application etc.

### Network connected or accessed

Ubuntu maintains the list of networks connected to the system in `/etc/NetworkManager/system-connections`. In addition to this, it is possible to know the active network connections which are being used in the system using the command `“sudo netstat -tupn”`.

Syslog file in `/var/log` provides the date and time at which a particular network connection was established. Network information enables the forensic examiner to know about the type of

network used in order to do the malicious activity.

### Devices connected to the System

In Ubuntu “`lshw`” command provides the list of hardware devices attached to the system. Also, the `/dev` directory in the file system provides the information about the hardware attached to the system. The syslog file also maintains the details of the devices which have been detected.

The date and timing at which the device was connected along with device details are also recorded in the syslog.

### Last Login Activity of the user

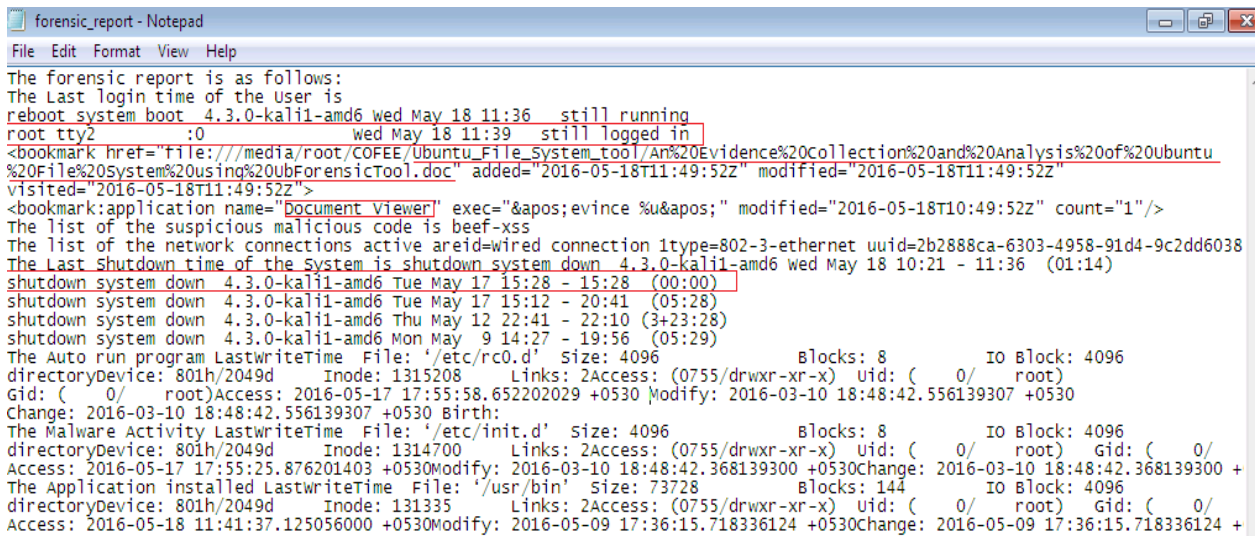
In Ubuntu, the login time and the logout time can be accessed by using the ‘`last`’ command at the terminal. Syslog file in the `/var/log` maintains the login and shutdown time.

### Malware Activity

To remain running after reboots, malware is usually re-launched using some persistence mechanism available in the various startup methods on an Ubuntu system, including services, drivers, scheduled tasks, and other startup locations. There are several configurations

files that Ubuntu uses to automatically launch an executable when a user logs into the system that may contain traces of malware programs. Malware often embeds itself as a new,

unauthorized service. A certain amount of malware use /etc/init.d directory to hide and start their execution on startup of the system.



```
forensic_report - Notepad
File Edit Format View Help
The forensic report is as follows:
The Last login time of the User is
reboot system boot 4.3.0-kali1-amd6 wed May 18 11:36 still running
root tty2 :0 wed May 18 11:39 still logged in
<bookmark href="file:///media/root/COFEE/Ubuntu_File_System_tool/An%20Evidence%20Collection%20and%20Analysis%20of%20Ubuntu
%20File%20System%20using%20UbForensicTool.doc" added="2016-05-18T11:49:52Z" modified="2016-05-18T11:49:52Z"
visited="2016-05-18T11:49:52Z">
<bookmark:application name="Document Viewer" exec="&apos;evince %u&apos;" modified="2016-05-18T10:49:52Z" count="1"/>
The list of the suspicious malicious code is beef-xss
The list of the network connections active areid=wired connection ltype=802-3-ethernet uuid=2b2888ca-6303-4958-91d4-9c2dd6038
The Last Shutdown time of the system is shutdown system down 4.3.0-kali1-amd6 wed May 18 10:21 - 11:36 (01:14)
shutdown system down 4.3.0-kali1-amd6 Tue May 17 15:28 - 15:28 (00:00)
shutdown system down 4.3.0-kali1-amd6 Tue May 17 15:12 - 20:41 (05:28)
shutdown system down 4.3.0-kali1-amd6 Thu May 12 22:41 - 22:10 (3+23:28)
shutdown system down 4.3.0-kali1-amd6 Mon May 9 14:27 - 19:56 (05:29)
The Auto run program LastwriteTime File: '/etc/rc0.d' Size: 4096 Blocks: 8 IO Block: 4096
directoryDevice: 801h/2049d Inode: 1315208 Links: 2Access: (0755/drwxr-xr-x) Uid: ( 0/ root) Gid: ( 0/ root)
Change: 2016-03-10 18:48:42.556139307 +0530 Modify: 2016-03-10 18:48:42.556139307 +0530 Birth:
The Malware Activity LastwriteTime File: '/etc/init.d' Size: 4096 Blocks: 8 IO Block: 4096
directoryDevice: 801h/2049d Inode: 1314700 Links: 2Access: (0755/drwxr-xr-x) Uid: ( 0/ root) Gid: ( 0/ root)
Access: 2016-05-17 17:55:25.876201403 +0530Modify: 2016-03-10 18:48:42.368139300 +0530Change: 2016-03-10 18:48:42.368139300 +
The Application installed LastwriteTime File: '/usr/bin' Size: 73728 Blocks: 144 IO Block: 4096
directoryDevice: 801h/2049d Inode: 131335 Links: 2Access: (0755/drwxr-xr-x) Uid: ( 0/ root) Gid: ( 0/ root)
Access: 2016-05-18 11:41:37.125056000 +0530Modify: 2016-05-09 17:36:15.718336124 +0530Change: 2016-05-09 17:36:15.718336124 +
```

Figure 3. Forensic report using UbuntuForensic tool

### 4.3 Timeline Analysis

The digital forensic investigator should detect the activity being performed by the suspect along with a timeline. By performing the timeline analysis, the investigator can trace the sequence of events that were performed by the suspect. For instance, if the suspect had accessed a word document by logging using a login id, the date and time of these activities can be correlated to convict the suspect. The forensic report obtained as in Figure 3 shows root user had logged in at 11:39AM on 18/05/2016 and accessed the .doc file 'An Evidence Collection and Analysis of Ubuntu File System using UbForensicTool' at 11:49AM using document viewer application. This forensic information can be evidence against the root user for accessing the .doc file as the .doc file was accessed after the login time by root user and before the shutdown of the system. The forensic report thus obtained using the UbuntuForensic tool underlines the importance of performing the timeline analysis of the activities.

### 4.4 Data Security

The UbuntuForensic tool provides the facility for the backup of the files from the hard disk of the running system. The backup of these files is maintained on the external storage media. The content of these files is then hashed one by one and the resulting hashes are then indexed and stored along with file name and the path of the file in a table on the external storage. The md5 algorithm is used to obtain the hashes from the backup data.

In order to detect if any changes have been occurred to the data on the hard disk of the running system by the suspicious criminal, the hashes are obtained from the individual files on the hard disk one by one and these hashes are then compared with the hashes stored on the external storage media. If two hashes which are being compared are found dissimilar then it means that the criminal has caused some modification to the relevant file on the hard disk. A report is prepared about all the files whose



hashes are found dissimilar from that of the hashes in the external storage. In such situation, the affected file can be restored back from the external hard disk. Figure 4 depicts the process for detecting the modification of the data on the hard disk by the criminal.

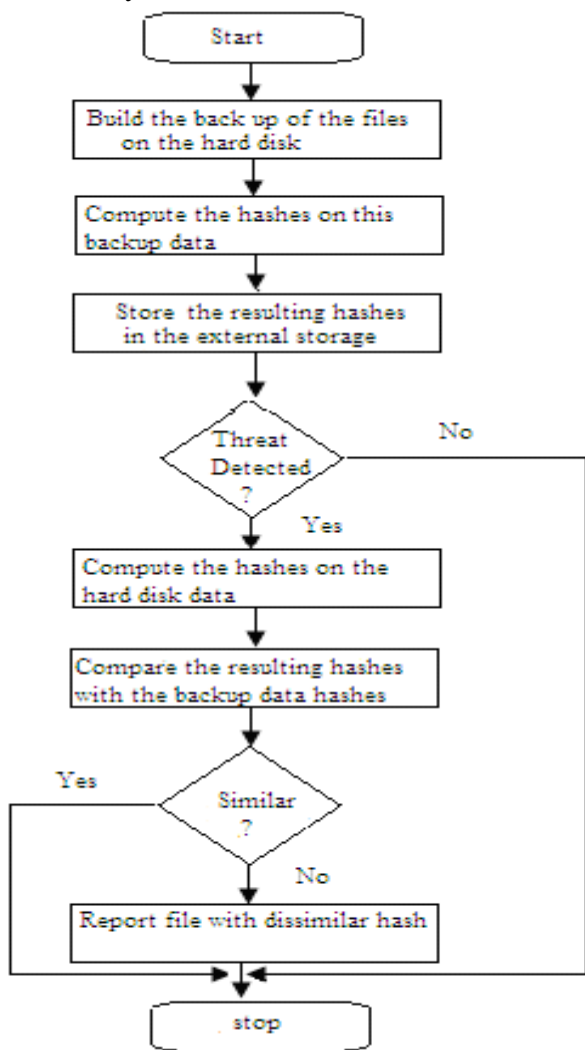


Figure 4. Flowchart depicting operation for identification of modified files using UbuntuForensic tool

## 5 SOFTWARE ARCHITECTURE AND IMPLEMENTATION

The software architecture of the UbuntuForensic tool is illustrated in Figure 5. The analysis of local and the external hard disk directory

structure can be performed using the UbuntuForensic tool. The evidence and time of the activity are extracted and the report is generated for correlating the sequence of events and their timings.

The software architecture consists of following modules: Local File System Forensic, External File System Forensic, Timestamp Generation, Backup File System, Hash Generation and Comparison, and Report Generation. The Local and External File System Forensic deals with extracting forensic evidence for various user activities from the directory structure of the system on which the tool is running and the directory structure available on the external hard disk. The time stamp generation module generates the last modified timestamp for the directory and files associated with the user's activity concerned. The forensic Report based on the forensic evidence obtained and the generated timestamp is obtained using Timestamp Generation module.

The algorithm for the proposed tool is as follows:

**Requires:**  $Activity(i, D(DIR))$  returns the extracted forensic information  $forensic\_info$  for each  $i^{th}$  activity from the DIR directory of the directory structure D.  $Select(forensic\_info(i))$  selects the evidence from the  $forensic\_info$ .  $Timestamp(i, D(DIR))$  returns the timestamp for the directory DIR for the  $i^{th}$  activity.  $Generate\_Report$  generates the report from the selected evidence and the timestamp. MAX indicates the maximum number of user's activity.

**Input:** The directory structure D

**Output:** Report in text format

- 1: For  $i \in (1, MAX)$  do;
- 2:  $forensic\_info(i) \leftarrow Activity(i, D(DIR))$
- 3:  $forensic\_evidence(i) \leftarrow Select(forensic\_info(i))$
- 4:  $timestamp_i \leftarrow Timestamp(i, D(DIR))$
- 5:  $Report \leftarrow Generate\_Report(forensic\_evidence,$

*timestamp*)

The *Activity(i,D(DIR))* function extracts the forensic information from the directory structure for the  $i^{\text{th}}$  activity of the user. Once the forensic information is extracted, the forensic investigator can select the digital evidence from it. The *Timestamp(i, D(DIR))* function generates the

timestamp for the  $i^{\text{th}}$  activity of the user based on the last access and modification timestamp of the directory. As the contents of the directory are accessed or changed, the timestamp of the directory also gets changed. This procedure is repeated for all the users' activity in consideration. Once all the activities are finished, the forensic investigator generates the Forensic report.

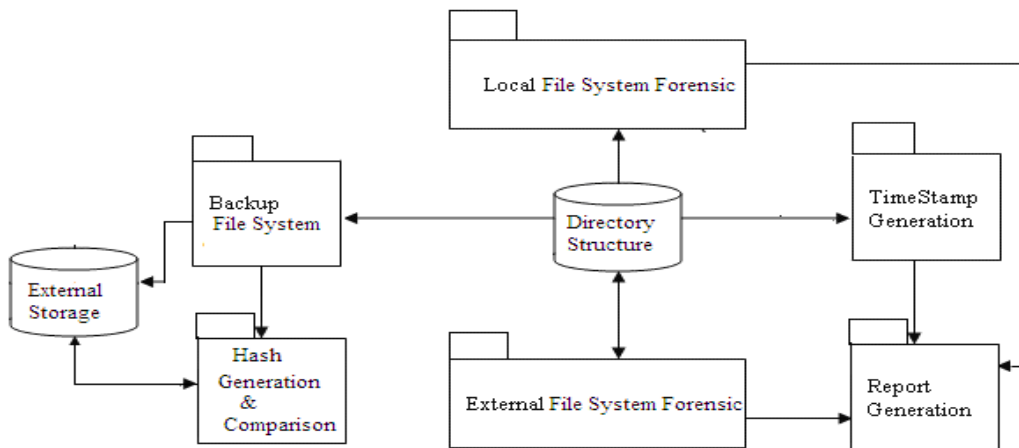


Figure 5. Software Architecture of UbuntuForensic tool

The backup of the files managed by the file system is performed using Backup File System module. The data backed up is then hashed by the hash generation module to generate the md5 hash. The hash so obtained is stored on the external storage in a relational table. Whenever the threat is detected, the hashes are obtained for the hard disk data and these hashes are then compared with the hashes in the external storage. If the mismatch is found then the affected data are restored back from the external storage. The structure definition of the table storing the hashes on the external storage is as follows:

```
typedef struct {
    Number int;
    File_Name string[20];
    Path_Name string[20];
    Hash long int;
} table;
```

The field description is as follows:

- Number: This field is an index for the entry in the relation.
- File\_Name: The name of the backed up file from the hard disk.
- Path\_Name: The path of the file concerned.
- Hash: The md5 hashes obtained on the content of the file.

The UbuntuForensic tool is built using QT4, a cross-platform application frame-work that is widely used for developing application software that can run on various software and hardware platforms with little or no change in the underlying code base while having the power and speed of native applications. Qt uses standard C++ with extensions including signals and slots that simplify handling of events, and this helps in the development of both GUI and server applications which receive their own set of event information and should process them

accordingly. The UbuntuForensic tool uses QSetting class and its methods to extract the information's from the directory structure of the Ubuntu file system.

## 6 EVALUATION

The comparison between the existing widely used Linux forensic tools and the UbuntuForensic tool is performed as in table 2. The tool like TSK, autopsy can list file and directories and perform timeline analysis of file activity. DEFT and CAINE provides GUI based forensic tools. i-Nex and History tools provide information about the hardware connected to the system and the recent command executed on the system recently, respectively. However, it has been observed that none of the Linux tools provides the facility for extracting the evidence for the specific activity of the user. Comparatively, the UbuntuForensic tool performs the extraction of forensic related information about the various users' activity being performed on the system. The UbuntuForensic tool also performs timeline analysis using which the conviction of the criminal can be performed based on the last access, modification dates of the directories and the login time of the suspicious user. The UbuntuForensic tool supports local and external file system forensics. In External file system forensics, the external hard disk with Ubuntu operating system is mounted on the system with the UbuntuForensic tool to extract the forensic evidence. The proposed UbuntuForensic tool also performs the backup of the files and directories. An approach to check the data integrity of all the files managed by the file system is proposed.

Based on the advanced requirements mentioned in the paper, UbuntuForensic tool improves over the shortcoming of the existing tools.

## 7 CONCLUSION

The File System maintains historical information about user activity in its directory structure. All of this information can be extremely valuable to a forensic analyst, particularly when attempting to establish the timeline of activity on a system. It is essential to perform the analysis of file system and use timeline analysis to detect the suspicious activities of the suspect. A wide range of cases would benefit greatly from the information derived or extracted from the file system.

A survey on the existing Linux forensic tools revealed that they extract very little forensic information from the file system. Comparatively, the UbuntuForensic tool provides more evidence from the file system as that of the existing tools; saving the time and effort in searching the evidence. The UbuntuForensic tool also covers forensic analysis of the file system on the external hard disk, thus enabling the forensic investigator to conduct the forensic investigation without changing the setup. The identification of the files which are modified by the criminal can be achieved by computing the hashes on the files from the hard disk.

Table 2. Functional comparison with existing tools

Tool	Function				
	Integrated Analysis	Timeline Analysis	Activity Analysis	GUI support	Any other feature
UbuntuForensicTool (Proposed)	✓	✓	✓	✓	Running process, Hash Generation
The Sleuth kit(TSK)	X	✓	X	X	Recovers deleted files
Autopsy	X	✓	X	✓	Recovers deleted files
Scalpel	X	✓	X	X	Recover data from disks
DEFT	X	✓	✓	✓	Data Recovery and hashing, Process information
CAINE	X	✓	✓	✓	Data Recovery
i-Nex	X	✓	✓	✓	Display device information, generate report
History	X	X	✓	X	Lists only command history

## 8 REFERENCES

- SophosLab: Botnets, a free tool and 6 years of Linux/Rst-B, <https://nakedsecurity.sophos.com/2008/02/13/botnets-a-free-tool-and-6-years-of-linuxrst-b> (2008)
- Sophos: Don't believe these four myths about Linux Security, <http://blogs.sophos.com/2015/03/26/dont-believe-these-four-myths-about-linux-security> (2015)
- McInnes J.: Linux Operating System don't get attacked by viruses,why?, <https://www.quora.com/Linux-Operating-System-dont-get-attacked-by-Viruses-why?> (2015)
- Tang L.: The study of Computer forensics on Linux, International conference on computational and Information Sciences (2013)
- Kuo-pao Y., Wallace K.: File Systems in Linux and Free BSD:A Comparative study, Journal of Emerging Trends in Computing and Information Sciences,2(9) (2011)
- Wei C., Chun-mei L.: The Analysis and Design of Linux File System Based on Computer Forensic, International Conference on Computer Design and Applications (2010)
- Joonah C., Antonio C.,Paolo G., Seokhee L, Sangjin L.: Live Forensic Analysis of a Compromised Linux System using LECT(Linux Evidence Collection Tool), International Conference on Information Security and Assurance (2008)
- Grundy B.: Advanced artifact analysis, European Union Agency for Network and Information Security (2014)
- ArchLinux: [https://wiki.archlinux.org/index.php/List\\_of\\_application/Utilities](https://wiki.archlinux.org/index.php/List_of_application/Utilities) (2016)
- Patil D., Meshram B.: Forensic investigation of user activities on Windows7 and Ubuntu12 operating system, IJNET, 5(3) (2015)



## A Preferential Analysis of Existing Password Managers from End-Users' View Point

<sup>1</sup>S. Agholor, <sup>2</sup>A. S. Sodiya, <sup>2</sup>A. T. Akinwale, <sup>3</sup>O. J. Adeniran and <sup>2</sup>D. O. Aborisade

<sup>1</sup>Department of Computer Science,  
Federal College of Education, Abeokuta, Nigeria

<sup>2</sup>Department of Computer Science,  
<sup>3</sup>Department of Mathematics,  
Federal University of Agriculture, Abeokuta, Nigeria

<sup>1</sup>sunday.agholor@gmail.com

<sup>2</sup>sinaronke@yahoo.co.uk, <sup>2</sup>aatakinwale@yahoo.com, <sup>2</sup>daaborisade@funaab.edu.ng  
<sup>3</sup>ekenedilichineke@yahoo.com

### ABSTRACT

Existing Password Managers which are generally classified into Desktop, Online and Mobile are used for enhancing security and handling memorability of passwords by different categories of users. However, several works toward improving on the three types of Password Managers did not take into consideration the end-users' preference or choice of usage. In this work, an empirical study was conducted to determine which of the three types of Password Managers do end-users' prefer most using the following three attributes-most preferred, most convenient and most trusted. The questionnaire was first pre-tested and its reliability computed. With a reliability correlation coefficient of 0.91, the questionnaire was then administered to capture the end-users' preference and interest among the three types of Password Managers from the four thousand eight hundred and fifty (4850) participants. The results showed that 67.67% of the total participants preferred to use the Mobile Password Manager. This is followed by Online Password Manager with 16.33%, while Desktop Password Manager with 16.00% is the least preferred choice of Password Manager. From the results, the paper recommends that researchers should re-direct their efforts toward improving the Mobile Password Manager.

### KEYWORDS

Desktop, End-Users, Mobile, Online, Password Manager

### 1 INTRODUCTION

A password is a character or sequence of characters used to determine that a device user requesting access to a system is really that particular user. Typically, users of a multiuser or securely single-user system usually have a unique name called a User-ID that can be generally known. In order to verify that someone entering that User-ID really is that person, a second identification, the password, known only to that person and to the system itself is entered by the user before access is granted. While passwords can be fairly secure, the weakness is how users choose and manage them [1, 2]. For instance, by using:

**(i) Simple Passwords:** These are passwords that are short in length, passwords that use words found in the dictionary, passwords created without using different character sets, passwords that are easily guessable or passwords that attackers can easily locate because they are placed on sticky notes pasted on the monitors, in a notepad or in a document stored in a computer or mobile device storage in clear text with the filename sometimes labeled as password, among other negative practices.

**(ii) The same Password:** This involves using the same password for multiple sites (password

re-use) and never changing the password. A compromised of the password will jeopardize all accounts where the passwords have been used.

**(iii) Shared Passwords:** This involves a situation where users tell others, such as family members, relatives or friends their passwords, sending their password information to their friends or relatives in an unencrypted form through email for keep. This makes the password very vulnerable to the attackers.

Despite the widely circulated accounts' safety rules such as: (i) Never give your PIN, password or token digits to anyone; (ii) Do not write them down or store them on your phone or computer in an unencrypted form; (iii) Your passwords are confidential information and should never be shared with anyone; given by the banks, e-commerce, financial institutions and information security experts, some users still choose and manage their passwords using all or some of the faulty ways earlier highlighted. The reasons for these users' actions are further explained below.

The requirement of creating usernames and passwords to serve as first line of defence against unauthorized access in Web-based services such as online banking, stock trading and e-commerce application is on the increase [3] as most online services providers require users to create a username and password before using their services [3]. This has led to a phenomenal increase in the number of passwords users are expected to memorize, which to a very large extent has overstretched their cognitive abilities [4]. Consequently, users often choose easy to remember passwords which have the potential of being easy to guess by attackers.

It has been observed that as the number of the password increases, users find it more difficult to recollect the appropriate password for a particular account [5], resulting in a phenomenon called password interference. Unfortunately, the system administrator continues to impose very strict password policy requirements for the end-users [6]. These strict password policy requirements made it difficult for end-users to choose randomly

generated passwords which offer high security to their account. Hence, they settle for weak passwords which offer low security. For example, as a result of human memory limitation, users often tend to choose short and low-entropy password that is easy to remember but has a possibility of being too easy to guess [7], or write down their passwords [8, 9] or use the same password at multiple websites [9, 10].

The above description is the bane of password problems: security and memorability. Unfortunately, as human, any attempt to increase one leads to a decrease in the other. But with the aid of a Password Manager, both security and memorability can be enhanced without decreasing either.

The main contribution of this study is to provide a basis for future research direction towards improving on the existing Password Managers based on end-users' choice of preference.

To the best of our knowledge, there are no empirical studies that have examined end-users' preference and interest among the three types of Password Managers. This is the source of our motivation.

The rest of this paper is organized as follows: In Section 2, the Literature Review was discussed, while Research Methodology was discussed in Section 3. In Section 4, the collected data were analyzed and the results discussed, while Conclusion and Recommendation were discussed in Section 5.

## 2 LITERATURE REVIEW

The single most important step that can be taken to improve password security thereby addressing the weakness of a password earlier highlighted, is by increasing its entropy [11], hence [12] recommended a password that is randomly generated from all character sets with an appreciable length. This, no doubt will help address the findings of [13] and [14]. However, increasing the password entropy helps in

increasing the password strength but not without a trade-off, the memorability crisis.

In attempt to solve the twin problems of passwords, that is, security and memorability, highlighted earlier, [7] suggested the use of mnemonics in constructing passwords. The result from this study showed that mnemonic-based passwords offered equal protection as those of randomly-constructed passwords. Furthermore, the finding showed that mnemonic-based passwords are easier to remember than randomly-constructed passwords. However, [15] found that [7] used basic dictionary attacks in their experiment, hence they constructed mnemonic-based dictionary which was used to attack the mnemonic-based passwords. Their findings which contradicted that of [7] revealed that mnemonic-based passwords could be cracked with the use of mnemonic dictionary attack. Thus, to prevent their accounts against hacking, end-users have no option other than to use highly random passwords of appreciable length [12]. Again, this comes with a trade-off, that is, memorability.

To avert memorability problem usually associated with the use of different complex passwords, many end-users resort to self help by using the same password for different online accounts. This is often referred to as password re-use. In a survey conducted by [16], 81.00% of the subjects admitted re-using the same password on many websites, while 68.00% admitted selecting related but not necessarily identical passwords across sites. In a related development, [13] findings showed that 18.75% re-use passwords, while 25.00% admitted using closely related password to access each account. Unfortunately, this action of re-using the same password for many online accounts increases security risk to the end-user whenever the password is breached [17]. Similarly, the investigation carried out by [18] on password usage in companies and that of [19] on the effects of password policies on users' practices, revealed alarming negative password practices caused by memorability problem.

Notwithstanding the wide-held sentiments from the security and usability communities that password should be replaced by other authentication schemes, it is likely to remain the most dominant authentication scheme [13, 20, 21, 22, 23, 24, 25, 26]. The reason for this is as a result of its incumbency, familiarity, and low cost in terms of its implementation, as well as inability of information security experts to reach a consensus on what exactly the alternative should provide [27]. According to [28], the 2005 RSA Conference panel communiqué stated that password has come to stay and will be with us forever. They, therefore, called on information security researchers to come up with measures that will make the use of password simpler and effective. Supporting this assertion is [1] who stated that password authentication is still and will continue to be the working horse of information security. These claims are still valid today as online accounts that require password is on the increase for a particular user. This is why research effort should be channeled towards improving its security and memorability. One way of doing this is through the use of Password Managers.

Password Managers were developed to relieve the end-users the burden of memorability [20]. It only requires the user to create and remember a single Master Password, ideally, a very strong password which grants the user access to their entire password database. According to [29], remembering a single Master Password is much more feasible for users, who still get the security benefits of using a different password for each online service.

Password Managers can also be used as a defence against phishing and pharming attacks. Unlike human beings, a Password Manager can also incorporate an automated login script that first compares the current site's URL to the stored site's URL. If the two did not match, then the Password Manager does not automatically fill in the login fields. This is intended as a safeguard against visual imitations and look-alike websites. With this built-in advantage, the use of a Password Manager is beneficial even if the user only has a few passwords to remember. In addition,

Password Managers can protect against keyloggers or keystroke logging malware. When using a multi-factor authentication scheme, a Password Manager can automatically fill-in the field for login. The user does not have to type any user names or passwords for the keylogger to pick up. However, Password Manager cannot protect against man-in-the-browser attacks, where malware on the user's device performs operations hidden from the user when the user is logged in.

## **2.1 Overview of existing Password Managers**

We present below a brief overview of the Password Managers which according to [30] are generally classified into Desktop, Online and Mobile Password Managers.

### **2.1.1 Desktop Password Manager**

They are used to store multiple passwords on local computers or the user's desktop, that is, on the terminal used for authentication which in turn is protected by a Master Password and can be retrieved when users revisit the websites through that computer. It is often called Offline Password Manager. Users are only required to memorize the Master Password. This Password Manager has the advantages associated with using Password Managers highlighted earlier. However, it has the following disadvantages: It is not portable, it is vulnerable to offline and online attacks. Examples are RoboForm, Mozilla Firefox, Apple MacOS Keychain, Microsoft Internet Explorer etc.

### **2.1.2 Online Password Manager**

Online Password Manager stores the passwords on remote third-party server(s). It is also called Web-based or Cloud-based Password Manager. The passwords are typically protected using a Master Password and at the time of recalling a specific password, the user simply types in his Master Password. The user of this Password Manager enjoys the advantages of its portability, in addition to the general advantages of using Password Manager earlier enumerated. However, the disadvantages include: Vulnerable to offline and online attacks, vulnerable to network failure, and

requires the user to trust the third party server in which the passwords are stored as a disgruntle staff of a third party provider can manipulate the data to his advantage. In other words, the user has no control in the management of his passwords. Examples are LastPass, MozillaWeave Sync, etc.

### **2.1.3 Mobile Password Manager**

A Mobile Password Manager stores passwords on end-users' portable devices such as phones and USB devices. Again, the passwords are typically protected using a Master Password and at the time of recalling a specific password, the user simply types in his Master Password. The user of this Password Manager enjoys all the advantages of a Password Manager earlier highlighted, in addition to the advantages of controlling and managing his passwords locally by himself on his portable device. However, the use of this Password Manager has some disadvantages such as vulnerable to offline and online attacks, vulnerability to lost of mobile devices, in addition to the mobile device becoming a target for thieves. Examples are KeePassmobile, OpenIntents Safe, Roboform2Go etc.

## **3 RESEARCH METHODOLOGY**

The study was conducted to enable us determine the end-users' preferences and interests among the three types of Password Managers. We describe the overview of the procedure used in carrying out this study.

The first stage was the establishment of the population of the study, which consists of all the twenty four (24) tertiary institutions in Ogun State of Nigeria.

In the second stage, the questionnaire was identified as the instrument to be used for data collection. The questionnaire after its construction was first administered to a selected sample from the population and later re-administered to the same sample population. The reliability of the questionnaire was evaluated by computing the correlation coefficient of the results obtained. With a correlation coefficient of 0.91, we

conclude that the research instrument is reliable enough to be used for the field work.

The third stage was the selection and training of the sample population on how to use Password Managers. For the sample population, eighteen (18) schools were selected out of the twenty four (24) tertiary institutions using stratified random sampling. Thus, the sample size for the study when compared to the population is 75%. Next was the training of the sample population on how to use a Password Manager. At the first phase, they were trained on how to use Desktop Password Manager and were allowed to use it for two (2) months. In the second phase, they were trained on how to use Online Password Manager. Again, they were allowed to use it for two (2) months. At the third phase, they were trained on how to use Mobile Password Manager and were allowed to use it for two (2) months.

It should be reported that during the training, we observed that majority of the sample population were already using one form of Password Manager or the other. This made our training easy and simple for the sample population.

The final stage was the random administration of four thousand eight hundred and fifty (4850) questionnaires to the trained sample population which comprised of students, lecturers (faculty staff) and non-academic staff. The number of questionnaires collected back for analyses was four thousand five hundred (4500). This showed that 92.78% of the total questionnaires administered were returned for analyses. In other words, the return rate is high enough to enable us draw meaningful inference from the analyses.

## 4 ANALYSES, RESULTS AND FINDINGS

### 4.1 Demographic Characteristics of the Participants

The study analyzed the demographic variables which comprise of age, sex and educational status of the participants. The result is presented in table 1.

**Table 1.** Demographic Characteristics of the Participants

Parameter	Frequency	Percentage
<b>AGE (YEARS)</b>		
18-30	1423	31.62%
31-40	1173	26.07%
41-50	1045	23.22%
51-60	859	19.09%
<b>Total</b>	<b>4500</b>	<b>100.00%</b>
<b>SEX</b>		
Male	2340	52.00%
Female	2160	48.00%
<b>Total</b>	<b>4500</b>	<b>100.00%</b>
<b>LEVEL OF EDUCATION</b>		
Students (O/L Certificates)	2259	50.20%
ND/NCE/HND/First Degree	1080	24.00%
Masters	981	21.80%
Ph.D.	180	4.00%
<b>Total</b>	<b>4500</b>	<b>100.00%</b>

From table 1, the minimum age of the participants is 18 years, while the maximum age is 60 years. This shows that the sample population is age-centric. The inference that could be drawn from this analysis showed that the participants represent an active age bracket that uses Password Managers.

Furthermore, table 1 shows that a little above half of the participants (52.00%) are male, while 48.00% of the participants are female. Statistically, we conclude that the study used a sample population that is gender-centric.

In the same vein, the distribution of the participants according to their educational status shows that 50.20% are students of the tertiary institutions, which implies that 50.20% of the respondents have Ordinary Level (O/L) Certificates, while 24.00% are in the category of those having qualification that are higher than O/L but not above first degree, 21.80% are holders of Masters degree, while 4.00% are Ph.D. holders. Thus, the educational level of the sample population is high enough and it is a good representation of those that can fill the questionnaire without assistant or guidance.

## 4.2 Participants' use of Passwords and Password Manager Experience

We captured the number of passwords each participant has been managing as well as the years of experience of using all or any of the three different types of Password Managers before the training.

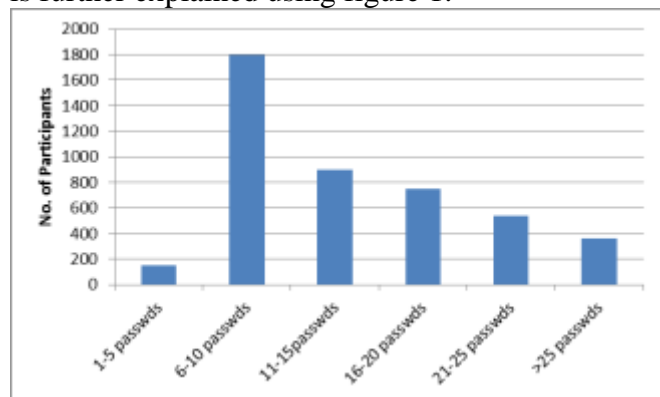
### (a) Number of Passwords in use by the Participants

Participants were asked to indicate the number of different passwords they use to authenticate to their various online accounts. The result is as presented in table 2.

**Table 2.** Number of Passwords own by the participants

No. of Passwords	No. of Participants	Percentage
1-5	150	3.33%
6-10	1800	40.00%
11-15	900	20.00%
16-20	750	16.67%
21-25	540	12.00%
>25	360	8.00%
Total	4500	100.00%

From table 2, one can see that 12.00% of the participants have between 21 and 25 passwords, while 40.00% of the participants have between 6 and 10 passwords. From this analysis, it showed that the participants need Password Manager to manage their numerous online accounts. This data is further explained using figure 1.



**Figure 1.** Bar Graph showing the number of passwords own by the Participants

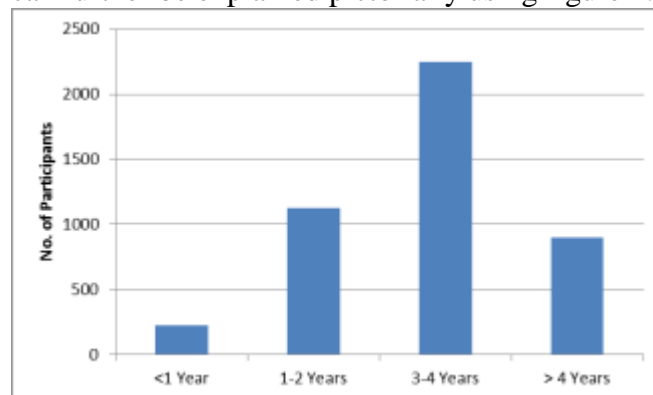
### (b) Participants' Experience in the use of Password Managers

In this section, we captured the participants' experience in the use of Password Managers. The result is presented in table 3.

**Table 3.** Participants' Experience in the use of Password Managers

Years of Experience	No. of Participants	Percentage
Less than 1 year	225	5.00%
1-2 years	1125	25.00%
3-4 years	2250	50.00%
Above 4 years	900	20.00%
Total	4500	100.00%

Table 3 shows the years of experience of using a Password Manager by the participants. From table 3, it shows that 50.00% of the participants have been using Password Managers for a period of 3 to 4 years, while 20.00% have been using Password Managers for more than 4 years. This shows that the participants have enough relevant experience to be used for the conduct of our study. This data can further be explained pictorially using figure 2.



**Figure 2.** Bar Graph showing the Participants' Experience

## 4.3 Preferential Analyses

The preferential analyses for Most Preferred, Most Convenient and Most Trusted Password Manager (PM) are presented in tables 4, 5 and 6.

### (a) Analysis of Most Preferred Password Manager

Table 4 shows the analysis of the Most Preferred Password Manager.



**Table 4.** Analysis of Most Preferred Password Manager

Password Manager	No. of Participants	Percentage
Desktop	900	20.00%
Online	720	16.00%
Mobile	2880	64.00%
Total	4500	100.00%

From table 4, one can see that 64.00% of the participants prefer to use Mobile Password Manager, while 20.00% of the participants prefer to use Desktop Password Manager. Trailing behind is the Online Password Manager with 16.00% of the participants taking it as their preferred choice of Password Manager. From the result, it shows that most end-users prefer to use the Mobile Password Manager. This is further explained pictorially using figure 3.



**Figure 3.** Bar Graph showing the Most Preferred Password Manager

**(b) Analysis of the Most Convenient Password Manager**

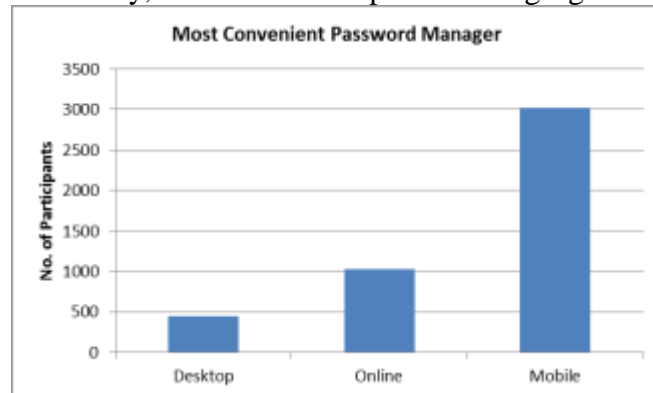
Table 5 shows the analysis of the Most Convenient Password Manager.

**Table 5.** Analysis of Most Convenient Password Manager

Password Manager	No. of Participants	Percentage
Desktop	450	10.00%
Online	1035	23.00%
Mobile	3015	67.00%
Total	4500	100.00%

In table 5, it is seen that 67.00% of the participants affirmed that Mobile Password Manager is the most convenient for them to use, while 23.00% opted for Online Password Manager as the most convenient for them to use and 10.00% said that Desktop Password Manager is the most

convenient for them to use. From this result, we conclude that Mobile Password Manager is the most convenient for the end-users to use. Pictorially, this is further explained using figure 4.



**Figure 4.** Bar Graph showing the Most Convenient Password Manager

**(c) Analysis of the Most Trusted Password Manager**

Table 6 shows the analysis of the Most Trusted Password Manager.

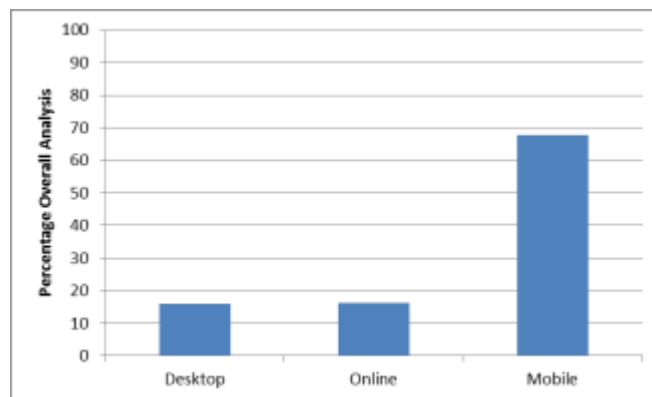
**Table 6.** Analysis of Most Trusted Password Manager

Password Manager	No. of Participants	Percentage
Desktop	810	18.00%
Online	450	10.00%
Mobile	3240	72.00%
Total	4500	100.00%

The result in table 6 shows that a very high percentage, that is, 72.00% of the participants said that the Mobile Password Manager is their most trusted, with 18.00% accepting Desktop Password Manager as their most trusted and only 10.00% agreed that Online Password Manager is their most trusted. Using this result, we conclude that Mobile Password Manager is the most trusted Password Manager that end-users will be willing to use. This is further explained pictorially using figure 5.



**Figure 5.** Bar Graph showing the Most Trusted Password Manager



**Figure 6.** Bar Graph showing the overall percentage Analysis

**(d) Overall Analysis of End-users’ Choice among the three categories of Password Managers**

The overall analysis of the end-users’ preference among the three categories of the Password Managers is presented in table 7. This utilizes the aggregate of the three attributes, that is, the most preferred, most convenient and most trusted.

**Table 7.** Overall Analysis of the three categories of Password Managers

Password Manager	Most Preferred		Most Convenient		Most Trusted		Overall % Average
	No.	%	No.	%	No.	%	
Desktop	900	20	450	10	810	18	16.00
Online	720	16	1035	23	450	10	16.33
Mobile	2880	64	3015	67	3240	72	67.67
TOTAL	4500	100	4500	100	4500	100	100

Table 7 shows the percentage of end-users who preferred a particular Password Manager. From the result, the order of preference turned out to be Mobile (67.67%), followed by Online (16.33%) with Desktop (16.00%) being the least preferred choice of Password Manager. From the above findings, it showed that the end-users were not comfortable giving control of their password management to a third party. Hence, they preferred to manage their passwords themselves on their own mobile phones. It is evident from the findings that the Mobile Password Manager gives them high degree of confidence when using it as their Password Management Scheme. Pictorially, the percentage overall analysis can be explained using figure 6.

**5 CONCLUSION AND RECOMMENDATION**

The results of this study showed that the Mobile Password Manager is a more promising password management scheme than the Online and Desktop Password Managers.

From the findings, we recommend that research effort should be directed towards improving the architecture of the Mobile Password Manager. This will enhance the ergonomics of the Password Manager.

Secondly, we recommend that more research work towards protecting the passwords stored in the Mobile Password Manager especially against offline and online attacks should be carried out.

**REFERENCES**

1. Ma, W., Campbell, J., Tran, D., Kleeman, D.: A Conceptual Framework for Assessing Password Quality. In: International Journal of Computer Science and Network Security, vol. 7, no. 1, pp. 179-185 (2007).
2. Gaw, S., Felten, E. W.: Password Management Strategies for Online Accounts. In: Proc. of the 2nd Symposium On Usable Privacy and Security (SOUPS), ACM, pp. 44-55 (2006).
3. Dhananjay, K., Fredrick, C. S.: iPass Framework to Create Secure and Usable Passwords. In: CSS, Chicago, USA (2009).
4. Halderman, A., Waters, B., Felten, E.: A convenient method for securely managing passwords. In: Proc. of World Wide Web Conference, Chiba, Japan, pp. 471-479 (2005).

5. Aborisade, D. O., Alowosile, O. Y., Odunlami, K. O., Odumosu, A.: A Cloud-based Password Manager for Multiple Transactions Accounts. In: Proc. of 11th International Conference of Nigeria Computer Society, Iloko-Ijesa, Nigeria, pp. 3-10 (2013).
6. Schechter, S., Herley, C., Mitzenmacher, M.: Popularity is Everything: A new approach to protecting passwords from statistical-guessing attacks. In: Proc. of the 5th USENIX Conference on Hot Topics in Security, Berkeley, USA, pp. 1-6 (2010).
7. Yan, J. J., Blackwell, A., Anderson, R., Grant, A.: Password Memorability and Security: Some Empirical Results. In: IEEE Security & Privacy, pp. 1-8 (2004). From [www.ieeexplore.ieee.org/ie15/8013/29552/0134406.pdf](http://www.ieeexplore.ieee.org/ie15/8013/29552/0134406.pdf) Accessed on 20/08/2011.
8. Zhang, J., Luo, X., Akkaladevi, S., Ziegelmeier.: Improving Multiple-Password Recall: An Empirical Study. In: European Journal of Information Systems, vol.8, pp. 165-176 (2009).
9. Sodiya, A. S., Agholor, S.: Users' Password Selection and Management Methods: Implications for Nigeria's Cashless Society. In: Proc. of 24th National Conference of the Nigeria Computer Society, Uyo, Nigeria, vol. 23, pp. 39-47 (2012).
10. Moshfeghian, S., Ryu, V. S.: Your Password is Invalid: Improving website Password Practices. In: Science Daily, pp. 1-8 (2012). From [www.sciencedaily.com/release/2012/01...](http://www.sciencedaily.com/release/2012/01...) Accessed on 25/05/2012
11. Gayathiri, C.: Text Password Survey: Transition from First Generation to Second Generation, pp. 1-10 (2013).
12. Agholor, S., Sodiya, A. S., Akinwale, A. T., Adeniran, O. J.: A Secured Mobile-Based Password Manager. In: Proc. of IEEE 6th International Conference on Digital Information Processing and Communications, Beirut, Lebanon, pp. 103-108 (2016).
13. Soluade, O. A., Opara, U. E.: Security Breaches, Network Exploits and Vulnerabilities: A Conundrum and an Analysis. In: International Journal of Cyber-Security and Digital Forensics, vol. 3, no. 4, pp. 246-261, (2014).
14. Ale, J. H., Hussin, J. H., Jose, A. H.: Cyber Warfare Awareness in Lebanon: Exploratory Research. In: International Journal of Cyber-Security and Digital Forensics, vol. 4, no. 4, pp. 482-497, (2015).
15. Kuo, C., Romanosky, S., Cranor, L. F.: Human Selection of Mnemonic Phrase-based Passwords. In: Proc. of SOUPS, New York, NY, USA, pp. 67-78 (2006).
16. Bojinov, H., Bursztein, E., Boyen, X., Boneh, D.: Kamouflage: Loss-Resistant Password Management, pp. 1-16 (2010). From [www.crypto.stanford.edu](http://www.crypto.stanford.edu). Accessed on 11/12/2014
17. Wessels, P. L., Steenkamp, L. P.: Assessment of current practices in creating and using passwords as a control mechanism for Information Access. In: South African Journal of Information Management, vol. 9, no. 2, pp. 1-15 (2007).
18. Inglesant, P., Sasse, M. A.: The True Cost of Usable Password Policies: Password Use in Wild. In: Proc. of ACM Conference on Human-Computer Interaction, New York, USA, pp. 383-392 (2010).
19. Shay, R., Komanduri, S., Kelly, P. G., Mazurek, M. L., Bauer, L., Christin, N., Cranor, L. F.: Encountering Stronger Password Requirements: User Attitudes and Behaviors. In: Proc. of SOUPS, Redmond, pp. 1-15 (2010). From [www.cups.cs.cmu.edu/soups/2010...](http://www.cups.cs.cmu.edu/soups/2010...) Accessed on 23/12/2013.
20. McCarney, D., Barrera, D., Clark, J., Chiasson, S., Van'Oorschot, P. C.: TAPAS: Design, Implementation and Usability Evaluation of a Password Manager. In: Proc. of the 28th Annual Computer Security Applications Conference, Orlando, Florida, USA, pp. 89-98 (2012).
21. Shiva, H. Y., Aggarwal, S.: Building Better Passwords using Probabilistic Techniques. In: Proc. of the 28th Annual Computer Security Applications Conference, Orlando, Florida, USA, pp. 109-118 (2012).
22. Preet, I. S., Gour, S. M. T.: Enhanced Password Based Security System based on User Behavior using Neural Networks. In: International Journal of Information Engineering and Electronic Business, vol. 2, pp. 29-35 (2012).
23. Florencio, D., Herley, C.: Where Do Security Policies Come From? In: Proc. of SOUPS, New York, USA, pp. 1-14 (2010).
24. Dell' Amico, M., Michiardi, P., Roudier, Y.: Password Strength: An Empirical Analysis. In: Proc. of INFOCOM, San Diego, pp. 983-991 (2010).
25. Wiedenbeck, S., Waters, J., Birget, J. C., Brodskiy, A., Memon, N.: Passpoints: Design and Longitudinal Evaluation of Graphical Password System. In: International Journal of Human-Computer Studies, vol. 63, pp. 42-49 (2005).

26. Borneau, J., Herley, C., Van' Oorschot, P. C., Stajano, F.: The quest to replace passwords: a framework for comparative evaluation of web authentication schemes. In: IEEE Symposium on Security and Privacy, vol. 10, no. 1, pp. 37-48 (2012).
27. Herley, C., Van' Oorschot, P. C.: A research agenda acknowledging the persistence of passwords. In: IEEE Security & Privacy, vol. 10, no. 1, pp. 28-36 (2012).
28. Saita, A.: Passwords at the breaking points. In RSA 2005 Conference Panel Communiqué, pp. 1-6 (2005). From [www.searchsecurity.techtarget.com/...](http://www.searchsecurity.techtarget.com/...) Accessed on 22/10/2011.
29. Gasti, P., Rasmussen, K. B.: On the Security of Password Manager Database Formats. In: Computer Security, vol. 7459, pp. 770-787 (2012).
30. Karole, A., Saxena, N., Christin, N.: A Comparative Usability Evaluation of Traditional Password Managers, pp. 1-15 (2010). From [www.cis.uab.edu/saxena...](http://www.cis.uab.edu/saxena...) Accessed on 13/09/2012.

## Intrusion Detection System with Spectrum Quantification Analysis

Yusuke Tsuge and Hidema Tanaka  
National Defense Academy of Japan  
Hashirimizu 1-10-20, Yokosuka, Kanagawa, Japan 239-8686  
Email: {em54027, hidema}@nda.ac.jp

### ABSTRACT

Intrusion Detection System (IDS) is a countermeasure against network attacks. There are mainly two types of detections; signature-based and anomaly based. Signature-based IDS detects intrusion packets by comparing contents of captured packets with the signature which is characteristic of intrusion packets. On the other hand, anomaly-based IDS detects them from normal behavior that is defined to distinguish normal communications from abnormal ones. Since attackers change their technique rapidly, anomaly-based detection draws research interest nowadays. However, since it is difficult to define normal behavior effectively, some anomaly-based IDS depends on visual identification of operator. To solve these problems, we propose a method using Detection-table which can be determined either normal or abnormal sessions. This method uses Discrete Fourier Transform and Shannon-Hartley theorem to analyze spectrum of each session. They assume fluctuation of spectrum in normal sessions as random and abnormal sessions as biased. To quantify difference between each spectrum and the standard one, we can obtain entropy using Shannon-Hartley theorem. Therefore, from the assumption, when entropy is small, we judge the session is normal, and when it is large, we judge the session is abnormal. By spectrum analysis based on such assumption, it is possible to derive the Detection-table. And we also find out that our quantification method will discover unknown abnormal sessions.

### KEYWORDS

Intrusion Detection System (IDS), Discrete Fourier Transform (DFT), Shannon-Hartley theorem, window function, Kyoto2006+ dataset

### 1 INTRODUCTION

Intrusion Detection System (IDS) is a countermeasure against network attacks [1] [2]. Research

of IDS has been conducted in many cases; such as [3] [4] [5] [6] [7] [8] [9] [10] [11]. The detection methods of IDS are divided into two types; signature-based IDS and anomaly-based IDS.

In signature-based IDS, characteristic of intrusion packets are stored as signatures in database. By comparing contents of captured packets with the signatures, intrusion packets can be detected. Snort [12] [13], Bro [14] [15], Swatch [16] and LogSurfer [17] are known as freeware-signature-based IDS. Snort is the most typical freeware of signature-based IDS and have a high detection rate. Bro enables to make signature to suit the purpose by using simple script. Since Swatch and LogSurfer get log-data by using syslog, they can detect intrusion packet by monitoring log-data. This type of IDS can judge recent sessions which are almost known attacks and are already analyzed.

However, this type does not detect unknown attacks. So in general, signature-based IDS has large false negative. And this type needs huge size of database of signature.

In anomaly-based IDS, normal behavior is defined to distinguish normal communications from abnormal ones. Therefore, it may detect unknown attacks. There are some existing methods; Wang et al. method [18], Imai et al. method [19], Sato et al. method [20] and Enkhbold method [21]. Wang et al. method are unsupervised using Mahalanobis distance. Sato et al. and Imai et al. method are also unsupervised using cluster analysis. Enkhbold method are spectrum analysis using Discrete Fourier Transform described in Section 2. This type of IDS is difficult to define normal. So in general, anomaly-based IDS has non-negligible false positive. And this type is difficult to operate. In fact, since almost methods depend on visual identification of operator, it is difficult to compare effec-

tiveness fairly and to quickly determine packets which are whether normal packets or abnormal ones.

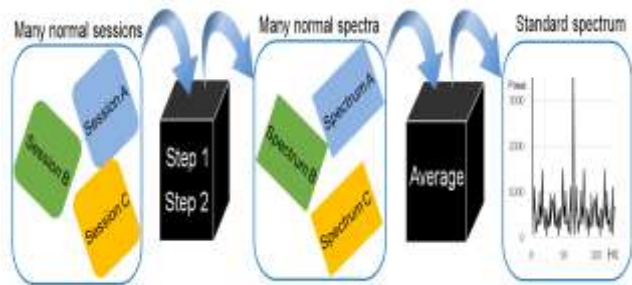


Figure 1. Outline of preparation

standard spectrum which is derived from average of spectra of normal sessions, we can distinguish normal ones from abnormal ones.

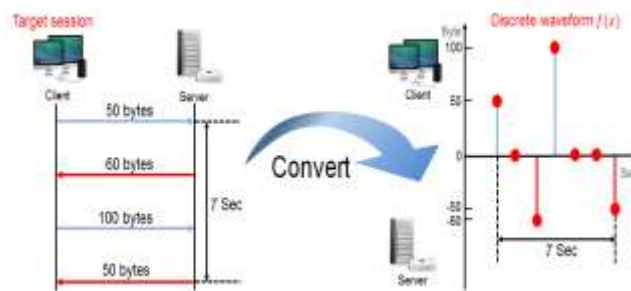


Figure 2. Outline of step 1

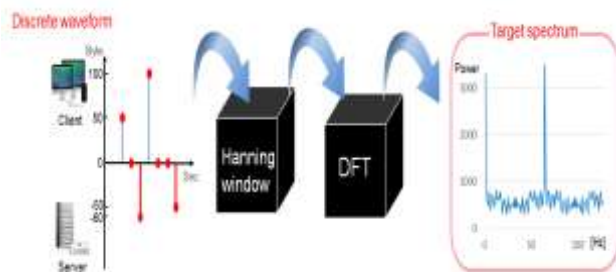


Figure 3. Outline of step 2

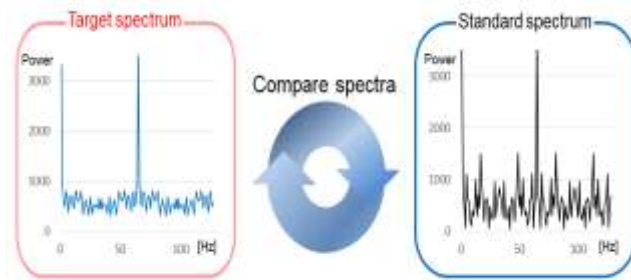


Figure 4. Outline of step 3

Nowadays, the speed of complication and evolution of attack technique is fast, so necessity of anomaly-based IDS is increasing, in especially for critical communication system. Constructing IDS is not to choose signature-based IDS or anomaly-based IDS, we need to combine both of them efficiency. However, as mentioned above, it is possible to operate signature-based IDS automatically but is difficult for anomaly-based IDS. To realize this purpose, we need to solve the problem of anomaly-based IDS depended on decision of operator. In this paper, we solve this problem by proposing quantification method.

As mentioned above, there are many methods on anomaly-based IDS. In this paper, for solving the above problem, we focus on the technique of Enkhbold [21]. This method uses spectrum analysis of sessions by Discrete Fourier Transform (DFT). There are some methods using DFT (e.g., Zhou et al. [22]). They are different in focusing on the features in sessions (“time-interval” or “time-interval and payload”). In Enkhbold method [21], discrete waveforms are made from fluctuation of payloads, then each spectra of session is derived using DFT. By comparing spectra of sessions with

And our previous improvement embedded window function into Enkhbold method to improve the efficiency in visual identification [23] (in the followings we call it previous method). However, since visual identification has no objectivity, we cannot compare it correctly. Also, previous method takes a long time to derive a spectrum (see Section 3.2).

To solve the problems, we propose quantification method using Shannon-Hartley theorem in this paper. In Section 2, we show the outline of previous method. Section 3 shows the basic idea of our proposal method using Shannon-Hartley theorem. In Section 4 and 5, we show our proposal method and example operation. In Section 6, we show our discussion. In Section 7, we discuss the advantageous of our method.

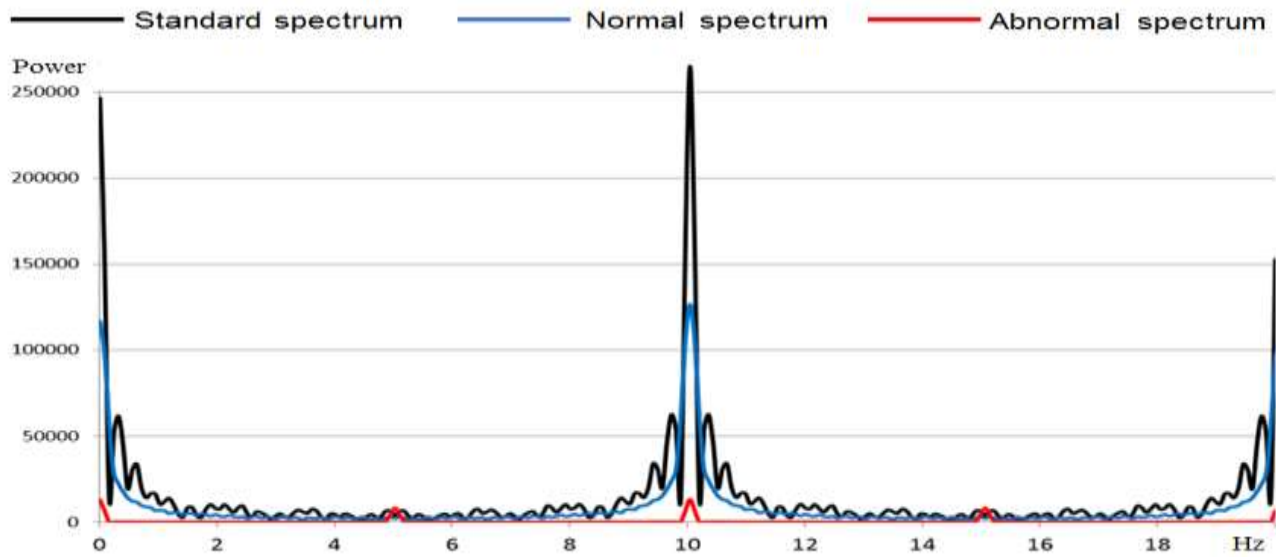
## 2 PREVIOUS METHOD AND PROBLEM

We define “session” the total communication set between one client and the server. Figure 1 ~ 4 shows outline of previous method [23]. It consists of followings.



**Preparation (Figure 1):** Make a discrete waveform from a payload and time elapsed of normal

session. The standard spectrum derived from an average of the spectra.



**Figure 5.** Example of abnormal detection [23]

**Step-1(Figure 2):** Make discrete waveform from target session.

**Step-2(Figure 3):** Apply window function to the discrete waveform. Perform DFT to the resultant.

**Step-3(Figure 4):** Compare the spectrum with the standard spectrum.

In Preparation, we make the standard spectrum. Its process is the same as the procedure of Step-1 and Step-2. We derive a lot of spectra from normal sessions, and the standard spectrum is derived from an average of the spectra. Note that normal sessions mean the sessions which are already checked as normal by other methods.

In Step-1, we make discrete waveform by regarding positive values as payload from client and negative value as payload from server. We make discrete waveform  $f(x)$  based on time elapsed in transmission as shown in Figure 2. Let  $T$  be the session time from start to end ( $0 \leq x \leq T$ ). Since the value of  $T$  changes for each session, when we perform DFT to any  $f(x)$ s of Step-2, each resultant spectrum has various frequency range. As the result, we cannot compare among spectra in Step-3.

To solve this problem, in previous method, we normalize each session with  $1/T$  and derive discrete waveform. In this process, when we take  $10^{-m}$  of minimum scale ( $m$  decimal places of  $1/T$ ), the discrete waveform has  $N = 10^m$  points.

In Step-2, we perform DFT to discrete waveform  $f(x)$  and make spectrum as follows.

$$|F(k)| = \sum_{n=0}^{N-1} (f(n) \times W_{han}(n)) e^{\frac{-i2\pi kn}{N}} \quad (1)$$

$(k = 0, 1, \dots, N - 1)$

$$W_{han}(n) = 0.5 - 0.5 \cos \frac{2\pi n}{N-1} \quad (2)$$

where  $|F(k)|$  is power of the spectrum. And  $W_{han}(n)$  is Hanning window. Previous method shows the detailed analysis for the reason why Hanning window is effective for IDS using DFT [23].

In Step-3, we compare the spectrum derived in Step-2 with the standard spectrum. Figure 5 shows an example detection. We use visual identification in Figure 5, and focus on status of spectra between 0 [Hz] and 10 [Hz]. The behavior of standard spectrum becomes random in the frequency range.

However, abnormal spectrum which are derived from abnormal session has almost constant comparing with the standard spectrum, and two large peaks



Figure 6. Behaviors of normal sessions

are found around 5[Hz] and 10[Hz]. As the result, we can distinguish normal spectra from abnormal ones.

However, the previous method has two problems. Firstly, the method uses visual identification. Since this scheme has no objectivity, detection results become ambiguous. Therefore, we cannot compare a spectrum correctly. Also, since it requires significant human effort, it is not efficient as a detection method. Secondly, the method has to take a certain time to derive a spectrum (see Section 3.2). This is a critical issue for IDS which is required quick detection. To solve these problems, we show two basic ideas in the following section.

### 3 BASIC IDEA

We show two ideas to solve the problems. The first idea is a solution for the visual identification problem. The second idea is a solution for the problem which takes a lot time to derive a spectrum.

#### 3.1 First idea

To solve the visual identification problem, we propose quantification method using Shannon-Hartley theorem. The principle of previous method is based on the following assumptions.

1. behaviors of normal sessions are various seems and to be random.

2. behaviors of abnormal sessions have some characteristics and biases.

From the viewpoint of spectrum analysis, the spectra of normal sessions become noise spectrum (Figure 6) and ones of abnormal sessions becomes biased spectrum (Figure 7).



Figure 7. Behaviors of abnormal sessions

Therefore when we define the standard spectrum as noise, we can calculate entropy using Shannon-Hartley theorem [24].

Then, if the target spectrum is abnormal, the value of entropy will be large, on the other hand, if it is normal, the value will be small. This assumption is the quantification for IDS using DFT. Shannon-Hartley theorem is shown as follows.

$$C = B \log_2 \left( 1 + \frac{S(f)}{N(f)} \right) \quad (3)$$

where  $B$  denotes bandwidth [Hz] of the channel.  $S(f)$  denotes the average received signal power and  $N(f)$  denotes the one of the noise and interference over the bandwidth. However, since we have discrete values over frequency range, we rewrite above equation as follows.

$$I = \int_{f_1}^{f_2} \log_2 \left( 1 + \frac{S(f)}{N(f)} \right) df \quad (4)$$

where  $S(f)$  and  $N(f)$  denote signal power and noise power of frequency  $f$  respectively. To satisfy the non-negativity of entropy and calculation for discrete waveform, we rewrite above equation as follows.

$$I_s = \sum_{\bar{f}=0}^{N-1} \log_2 \left( 1 + \frac{\max \{S_s(f), S_t(f)\}}{\min \{S_s(f), S_t(f)\}} \right) \times \Delta f \quad (5)$$

Where  $S_s(f)$  and  $S_t(f)$  denote standard spectrum power and target spectrum power at point  $f$ . And where  $\Delta f$  denotes the unit frequency scale which is calculated as follows.

$$\Delta f = \frac{f_s}{N - 1} \quad (6)$$



Figure 8. Outline of P1

communications over initial session time. We need some changes in previous method. Firstly, we do not need normalization procedure in Step-1 because it is done in advance. Secondly, we should not apply window function in Step-2. By applying initial session time, the number of communication in a

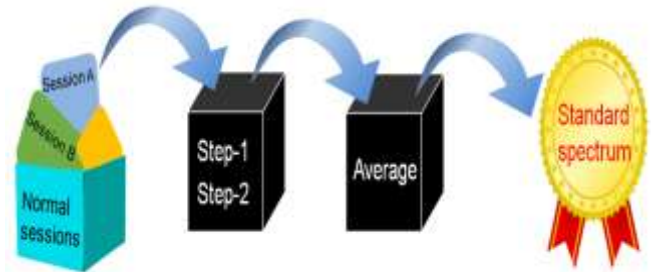


Figure 9. Outline of P2

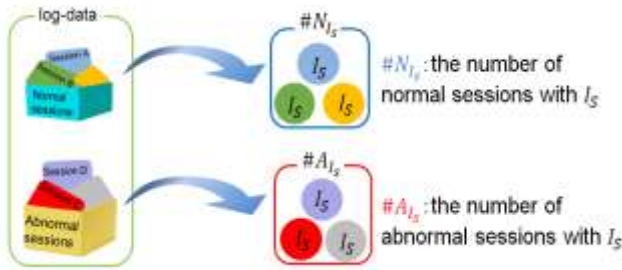


Figure 10. Outline of P3

where  $f_s$  denotes the sampling rate for a real network environment. In this paper, we determine it by the average of total number of sessions per unit time. In the followings, we call  $I_s$  evaluation-value. We can judge whether the target session is normal or abnormal using this evaluation-value, however, the value of normal and one of abnormal may never be different. Therefore we make the Detection-table which shows each range of evaluation-value is normal or abnormal with probability. In this procedure, we need trusted log-data which classify normal sessions and abnormal sessions.

### 3.2 Second idea

Since the previous method derives a spectrum by using the session time  $T$ . If the session time is long, we cannot judge immediately. In IDS which is required quick response, this is critical condition. Therefore we set initial session time in advance to short time for detection. As the result, we omit

Probability of each  $I_s$

$I_s$	Prob. Of normal sessions	Prob. Of abnormal sessions
$I_s(i)$	$P_N(I_s) = \frac{\#N_{I_s}}{\#N_{I_s} + \#A_{I_s}}$	$P_A(I_s) = \frac{\#A_{I_s}}{\#N_{I_s} + \#A_{I_s}}$
$I_s(i+1)$	$P_N(I_s) = \frac{\#N_{I_s}}{\#N_{I_s} + \#A_{I_s}}$	$P_A(I_s) = \frac{\#A_{I_s}}{\#N_{I_s} + \#A_{I_s}}$

Detection-table

$T$	Prob. of normal sessions	Prob. of abnormal sessions
$0 \leq I_s < 49$	0.81	0.19
$49 \leq I_s < 76$	0.20	0.80
⋮	⋮	⋮

Figure 11. Outline of P4

session is decreased. So, we need to make characteristics of session stand out efficiently. On the other hand, since window function regards such characteristic in short time as noise, some significant feature will be lost. Therefore, we should not use window function. As the result, we conclude that our improvements for previous method is effective for visual identification [21]. However, they do not contribute to our quantification method for short time.

## 4 PROPOSAL METHOD

Our proposal method has two phase; Preparation phase and Detection phase. Figure 8~11 shows outline of Preparation phase and Figure 12 shows outline of Detection phase.

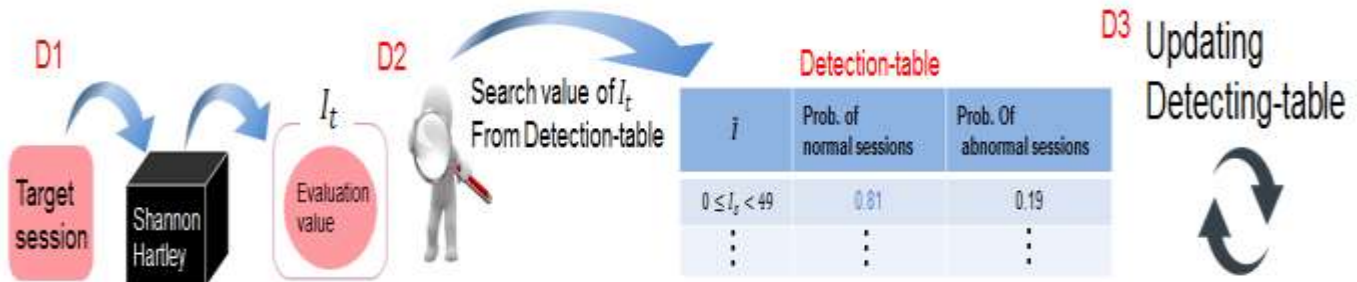
### 4.1 Preparation phase

#### P1: Collecting and classifying log-data

**(Figure 8)**

The purpose of proposal method is to quantify the difference between the standard spectrum and target ones. Therefore we need both normal and abnormal session log-data. And also we need same

methods to classify normal session and abnormal session correctly. For this procedure, it is desirable



**Figure 12.** Outline of Detection phase

**Table 1.** Outline of honey pot

Type	Number of machines
Solaris 8 (Symantec based)	4
Windows XP(full patch)	1
Windows XP(no patch)	5
Windows XP SP2	2
Windows Vista	1
Windows 2000 Server	1
Mac OS X	2(one is mail server)
Printer	2
TV set	1
HDD recorder	1
SGNET honeypots	5
dedicated honeypots	4
Web Crawler	1
Black hole sensor /24	1
Black hole sensor /26	1

to hold log-data in a long term and use some methods such as signature type IDS which can detect definitely.

**P2: Derivation of standard spectrum (Figure 9)**

Among the log-data which is prepared in P1, we use normal sessions to derive the standard spectrum using Step-1 and Step-2 of previous method.

**P3: Calculation of evaluation-value (Figure 10)**

We calculate evaluation-values of all session in log-data using eq. (5) and eq. (6). We reclassify the result of log-data (normal or abnormal session) using each evaluation-value ( $I_s$ ), and count the number of normal session ( $\#N_{I_s}$ ) and the number

of abnormal session ( $\#A_{I_s}$ ) for each  $I_s$ .

**P4: Construction of Detection-table (Figure 11)**

Let  $P_N(I_s)$  be the probability of normal session with evaluation-value which is equals to  $I_s$ . And let  $P_A(I_s)$  be the probability of abnormal session.

$$P_N(I_s) = \frac{\#N_{I_s}}{\#N_{I_s} + \#A_{I_s}} \tag{7}$$

$$P_A(I_s) = \frac{\#A_{I_s}}{\#N_{I_s} + \#A_{I_s}} \tag{8}$$

Let  $Q$ , ( $0.5 < Q \leq 1.0$ ) be the threshold of successful detection probability. We search for the

range of evaluation-value  $\tilde{I}_S$  which satisfies followings.

$$\sum_{I_S \in \tilde{I}_S} P_N(I_S) \geq Q \quad \text{or} \quad \sum_{I_S \in \tilde{I}_S} P_A(I_S) \geq Q \quad (9)$$

**Table 2.** Feature of Kyoto 2006+ dataset

Conventional features	Additional features
Duration	IDS detection
Service	Malware detection
Source bytes	Ashula detection
Destination bytes	Label
Count	Source IP address
Same srv rate	Source Port number
Error rate	Destination IP address
Srv error rate	Destination Port number
Dst host count	Start time
Dst host srv count	Duration
Dst host same src port rate	
Dst host error rate	
Dst host srv error rate	
Flag	

**Table 3.** Specification of computer experiment

Log-data	Kyoto2006+ dataset (15,746,592 sessions)
Sampling rate	$f_s = 7.4$ [Hz]
Threshold	$Q = 0.8$
Initial session time	$T = 2$ [sec]
OS	Windows 7 Professional
CPU	Intel Corei7-3770 3.4 GHz
RAM	16.0 GB
Programming language	Visual Basic for Applications

where

$$\tilde{I}_S = \{I_S | I_S(i) \leq I_S < I_S(j)\} \quad (10)$$

Note that  $I_S(m)$  denotes  $m$ -th ( $0 < m$ ) value of  $I_S$ .

#### 4.1 Detection phase (Figure 12)

##### D1: Calculation of evaluation-value of target session

We derive the spectrum of target session using Step-1 and Step-2 which use initial session time and calculate evaluation-value  $I_t$  using the standard spectrum.

##### D2: Look-up Detection-table

We search for the range  $\tilde{I}_S$  which involves the value of  $I_t$  in Detection-table. Normal session or abnormal session is judged by the probability which exceeds the threshold  $Q$ . Then falsenegative (or false-positive) can be evaluated as  $1 - Q$ .

##### D3: Updating Detection-table

When the result of detection is confirmed true, we update Detection-table to improve successful detection probability.

## 5 EXPERIMENT

### 5.1 Kyoto2006+ dataset

In this paper, we use Kyoto2006+ dataset [25]. This dataset was derived from the actual data traffic during November 2006 to August 2009 by using the honey pot which is installed in Kyoto University. A structure of honey pot is shown in Table 1. This dataset consists of 14 conventional features and 10 additional features (see Table 2). These 14 features were extracted based on KDD Cup 99 data set [26] which is very popular and widely used performance evaluation data for IDS. We use Duration, Source IP address, Destination IP address, Source bytes, Destination bytes and Label

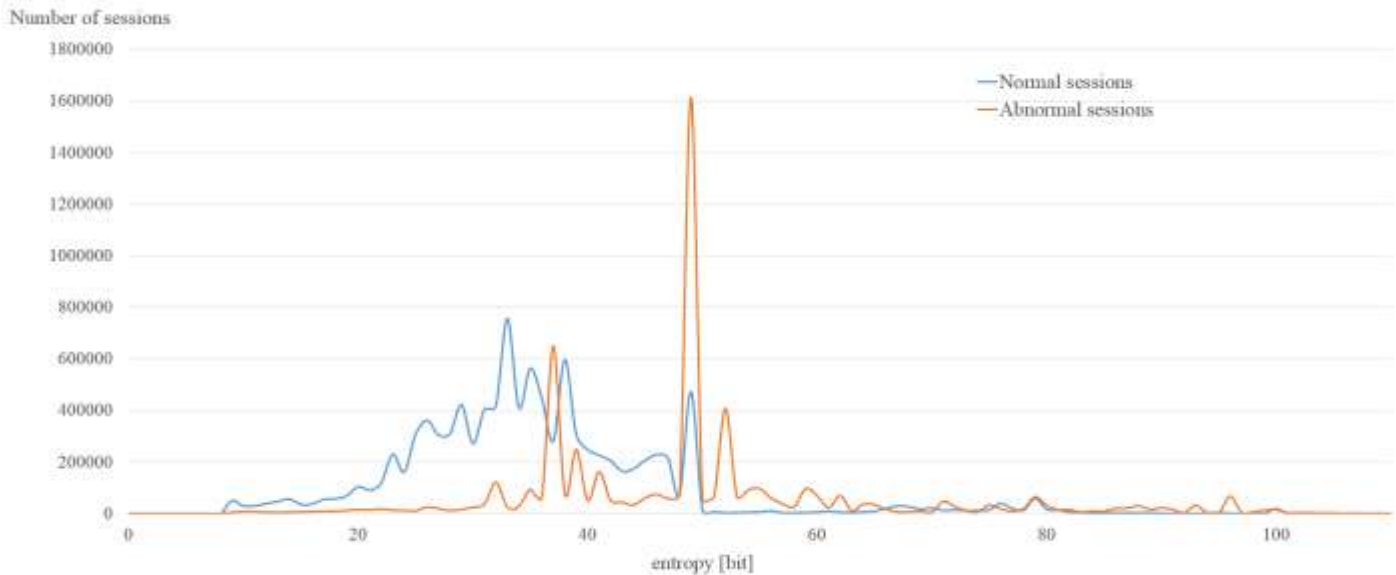


(shaded in Table 2). The label can classify as either normal session or abnormal session for each session. In fact, this dataset is old to use for evaluation of IDS performance. However, since this is

an open public, it is possible for third persons to verify the

**Table 4.** Detection-table

$\tilde{I}_s$	Prob. of normal sessions	Prob. of abnormal sessions	Number of sessions
$0 \leq I_s < 49$	80.6%	19.4%	11,300,021
$49 \leq I_s < 76$	19.6%	80.4%	3,754,637
$76 \leq I_s < 80$	54.1%	45.9%	239,087
$80 \leq I_s$	20.0%	80.0%	452,847



**Figure 13.** Distribution of number of sessions

effectiveness and compare the effectiveness among other methods. In addition, by using the label, we can confirm the successful detections.

### 5.2 Preparation phase

We omit 0 byte of payload and 0 second of session time in Kyoto2006+ dataset because these types of session cannot contribute to derivation of Detection-table. There are total 15,746,592 target sessions. As mentioned in Section 3, we define sampling rate  $f_s$  by the average of total number of sessions per unit time. Table 3 shows the specification of our computer environment.

### 5.3 Detection phase

When we use the Detection-table shown in Table 4, we can find following facts.

1) **The assumption about evaluation-value is**

**true.**

In Section 3, we described our assumption that

when evaluation-value is small, the session will be normal, on the other hand, when it is large, the session will be abnormal. We can confirm that this assumption in Table 4; the range  $\tilde{I}_s: 0 \leq I_s < 49$  is normal with probability of 80.6%, the range  $\tilde{I}_s: 49 \leq I_s < 76$  is abnormal with probability of 80.4% and the range  $\tilde{I}_s: 80 \leq I_s$  is abnormal with probability of 80.0%. In particular, our assumption can be confirmed clearly in detection of abnormal sessions.

2) **Ranges of evaluation-value which are under the threshold  $Q$  exist.**

The ranges  $\tilde{I}_s: 76 \leq I_s < 80$  are under the threshold. In fact, we cannot solve this problem, and



compromised this result. For these ranges, where under the threshold, our detection results will become unstable, however, these cases are only 1.52% of the whole of log-data (Figure 13). From

this fact, we conclude that the sessions whose evaluation-values are included in these range are too small in Kyoto

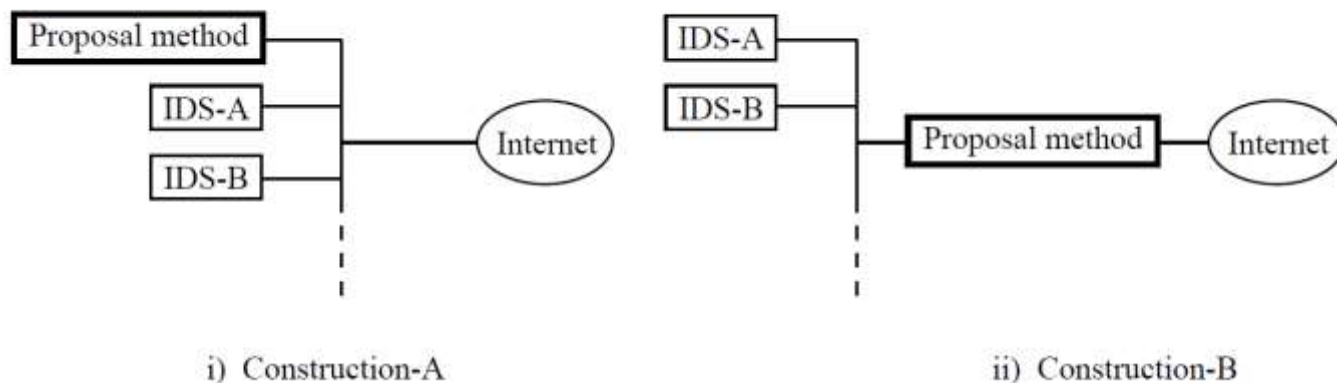


Figure 14. Position of proposal method in IDS construction

2006+ dataset. Therefore, we expect that this problem will be solved by updating Detection-table using more sessions.

### 3) Where is unknown abnormal sessions?

A possibility that we can find out the unknown abnormal sessions is low in the ranges which the threshold of successful detection is satisfied. Because these ranges have enough sample sessions to analyze in details. Therefore, we have to pay attention and analyze in the range where the threshold is not satisfied. As the result, we can expect that our proposal method will contribute the effective detection and analysis of unknown abnormal session. As mentioned in Section 4.2, Detection-table is updated using detection results, the probability of successful detection always keeps being improved. Therefore, the result shown in Table 4 is the initial state of our proposal method for the communication environment of Kyoto 2006+ dataset. At the same time, by resetting the threshold  $Q$  high, suspicious sessions will be found easily.

## 6 DISCUSSION

In Section 3, we assume that the spectra of normal sessions become random noise and ones of abnormal sessions have peaks. Based on this assumption, the previous method obtained result of Figure 5. In the previous method, operators con-

duct spectrum analysis by using visual identification. It is ambiguous to depend on visual identification of operator. Also, the previous method takes a long time to derive a spectrum. On the other hand, the proposal method defines the standard spectrum as noise, we can calculate entropy using Shannon-Hartley theorem. We can expect that normal spectrum will be small and abnormal one will be large. And we can make the Detection-table by using assumption. We can conduct spectrum analysis by using Detection-table without using visual identification of operator. Also, since the proposal method uses initial session time, deriving a spectrum finishes in short. From these improvements, proposal method solves problems of the previous method. Therefore, we conclude that our proposal method is more excellent than the previous one.

## 7 CONCLUSION

In this paper, we propose a quantification method for IDS using DFT and define the Detection-table. Using the Detection-table, we can operate IDS using DFT by deterministic algorithm. There are some concepts to construct IDS shown in Figure 14. Construction-A is the majority decision type. Obviously, our proposal method is not effective in this position. Construction-B is adequate position for our proposal method and it will work as proac-

tive detection as mentioned in Section 5.3. The possibility which the target session is unknown abnormal session is high when its evaluation-value is involved in the range where threshold Q is not satisfied. Unfortunately, Kyoto 2006+ dataset does not include any unknown abnormal sessions, we cannot confirm the effectiveness of our proposal method concerning to this feature. And as mentioned in Section 4.2 (D3), we can update the Detection-table. However, the restrictions in time and on the computer environments did not enable us to execute this procedure in this experiment. These are our future works.

In the operation of our proposal method, we need only the Detection-table as the dataset. Therefore we can conclude that our proposal method is very low-cost IDS and very fast computational method. These feature will enable to construct real-time detection. This is also our future work.

Obviously, it does not need to point out, the Detection-table is various in the communication environment. We need to more experiments in various communication systems.

## ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant Number 24560491.

## REFERENCES

1. G. Bruneau, "The history and evaluation of intrusion detection," <https://www.sans.org/reading-room/whitepapers/detection/history-evolution-intrusion-detection-344>.
2. J. Anderson, "Computer security technology planning study volume ii," *Electronic Systems Division Technical Report*, pp. 73–51, 1972.
3. D. E. Denning, "An intrusion-detection model," *IEEE Transactions on software engineering*, no. 2, pp. 222–232, 1987.
4. V. V. K. Labib and V. R. Vemuri, "Anomaly detection using a language framework: Clustering and visualization of intrusive attacks on computer systems," in *Fourth Conference on Security and Network Architectures, SAR '05*. Citeseer, 2005.
5. J. Hochberg, K. Jackson, C. Stallings, J. McClary, D. DuBois, and J. Ford, "Nadir: An automated system for detecting network intrusion and misuse," *Computers & Security*, vol. 12, no. 3, pp. 235–248, 1993.
6. L. Portnoy, E. Eskin, and S. Stolfo, "Intrusion detection with unlabeled data using clustering," in *In Proceedings of ACM CSS Workshop on Data Mining Applied to Security (DMSA-2001)*. Citeseer, 2001.
7. S. Kumar and E. H. Spafford, "A pattern matching model for misuse intrusion detection," 1994. [8] S. Staniford-Chen, S. Cheung, R. Crawford, M. Dilger, J. Frank, J. Hoagland, K. Levitt, C. Wee, R. Yip, and D. Zerkle, "Grids-a graph
8. S. Staniford-Chen, S. Cheung, R. Crawford, M. Dilger, J. Frank, J. Hoagland, K. Levitt, C. Wee, R. Yip, and D. Zerkle, "Grids-a graph based intrusion detection system for large networks," in *Proceedings of the 19th national information systems security conference*, vol. 1. Baltimore, 1996, pp. 361–370.
9. U. Lindqvist and P. A. Porras, "expert-ism: A host-based intrusion detection solution for sun solaris," in *Computer Security Applications Conference, 2001. ACSAC 2001. Proceedings 17th Annual*. IEEE, 2001, pp. 240–251.
10. D. Curry, H. Debar, and B. Feinstein, "Intrusion detection message exchange format data model and extensible markup language (xml) document type definition," *IDWG, February*, 2002.
11. E. Eskin, "Anomaly detection over noisy data using learned probability distributions," in *In Proceedings of the International Conference on Machine Learning*. Citeseer, 2000.
12. "Snort," <https://www.snort.org/>.
13. M. Roesch *et al.*, "Snort: Lightweight intrusion detection for networks." in *LISA*, vol. 99, no. 1, 1999, pp. 229–238.
14. "The bro network security monitor," <https://www.bro.org/>.
15. V. Paxson, "Bro: a system for detecting network intruders in real-time," *Computer networks*, vol. 31, no. 23, pp. 2435–2463, 1999.
16. "Swatch," <http://swatch.sourceforge.net/>.
17. "Logsurfer," <https://www.cert.dfn.de/eng/logsurf/>.
18. K. Wang and S. J. Stolfo, "Anomalous payload-based network intrusion detection," in *International Workshop on Recent Advances in Intrusion Detection*. Springer, 2004, pp. 203–222.
19. K. Imai, S. Aoki, and T. Miyamoto, "Anomaly detection

based on clustering of network traffic characteristics considering results of signature based ids evaluation,” *ICISS Technical Report*, vol. 489, no. 114, pp. 7–12, 2015.

20. M. Sato, H. Yamaki, and H. Takakura, “Unknown attacks detection using feature extraction from anomaly-based ids alerts,” in *Applications and the Internet (SAINT), 2012 IEEE/IPSJ 12th International Symposium on*. IEEE, 2012, pp. 273–277.
21. E. Chimedtseren, K. Iwai, H. Tanaka, and T. Kurokawa, “Intrusion detection system using discrete fourier transform,” *Proceedings on the 7<sup>th</sup> IEEE Symposium on Computational Intelligence for Security and Defense Applications(CISDA)*, no. CS3-1, pp. 1–5, 2014.
22. M. Zhou and S.-D. Lang, “A frequency-based approach to intrusion detection,” in *Proc. of the Workshop on Network Security Threats and Countermeasures*, 2003.
23. Y. Tsuge and H. Tanaka, “Intrusion detection system using discrete fourier transform with window function,” *International Journal of Network Security & Its Applications (IJNSA)*, vol. 8, no. 2, pp. 23–34, 2016.
24. C. E. Shannon, “A mathematical theory of communication,” *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 1, pp. 3–55, 2001.
25. J. Song, 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*. ACM, 2011, pp.29–36.
26. KDDCup1999Data, “The third international knowledge discovery and data mining tools competition dataset kdd99-cup,” <https://kdd.ics.uci.edu/dataases/kddcup99/kddcup99.html>.

## “The Unwitting Danger Within - Detection, Investigation and Mitigation of a Compromised Network”

**Emmanuel U Opara,**  
**College of Business,**  
**Prairie View A&M University,**  
**Prairie View - Texas U.S.A.**  
**Email: euopara@pvamu.edu**

**Oredola A. Soluade**  
**Iona College,**  
**Hagan School of Business,**  
**New Rochelle - New York**  
**email: osoluade@iona.edu**

### ABSTRACT

The war on cyber security issues has exploded exponentially. Persistent attacks are on the rise routinely penetrating perimeter defenses and bypassing antivirus technologies to successfully launch attacks against endpoints and servers. The Internet of Things [IoTs] have motivated hackers to compromise networks but a massive data breach does not have to be. Businesses, hospitality, travel, healthcare, insurance, financial institutions, retails and other big enterprise systems succumbed to lingering, multistage attacks that siphoned sensitive, and valuable data out of the respective networks. This study will generate next generation end-point security systems that will identify evil, or unusual and abnormal patterns in an intrusion scheme. The outcome will know abnormal by finding Evil. Recommendation for best practices will be provided.

**Keyword:** Breaches, Exploits, Network Security, Threats, Vulnerabilities,

### 1. INTRODUCTION

Enterprise systems are witnessing the frequency of cyber-attacks as they are becoming very severe and complex to manage. Report in 2016 shows massive data breaches, major cyber-attack and excruciating stream of new vulnerabilities across the cyber world [5]. In 2015, cyber actors inflicted physical atrocities by stealing intellectual properties, hijacking personal and customer confidential information for espionage [4].

The most threats come from the signature-less oriented vulnerabilities known as Shellshock which were responsible for several attacks in

2015. Shellshock as the name suggests, is a flaw in the Bash shell that is used in Mac OS, Linux, and Solaris, etc. Other signature less oriented threats include the Zero-day, APT Tactis, Zeus Trojan [Zbot], Stuxnet, Malicious Computer Worm, Duqu, Flame, RATs [Romte Access Trojan], GhOst RAT, Ransom-ware, threat kits, Spear-phishing, Rombertik, CryptLocker, CryptonWall, Armored, Sparse-infector, Multi-partite, Macro, Polymorphic, FakeAV, MacDefender, W32/Netshy-P, the Sobig virus, Mimail, Bagle, Regin, etc. [5, 6,12]. These are examples of anomalies that are very difficult to detect by the signature detection tools.

Another dangerous malware is the *Regin* spyware. This malware is used by hackers for intelligence-gathering and monitoring of targeted networks. Hacker's employ this tool to attack individuals, small businesses, private companies, governmental entities, research institutions, telecommunication companies, etc. [19].

Unwarranted targeted cyberattacks are on the rise, often crippling networks, resulting in noteworthy loss of time. The cyber world is witnessing ever increasing amounts of distributed denial of service (DDoS) attacks [10, 17].

Hactivists and Cyberterrorists are terrorizing international affairs, not only because the borders implications, but because they have become mechanisms of foreign and statehood powers. These actors are using various arrays of signature-less tools to exploit access into enterprise

network, layer 7 targeted attacks, SLL-based and bulk-volumetric attacks [18].

Administrators and cyber professionals are faced with the problem of escalating attacks as they are occurring on a regular basis. Is it lack of adequate funding or that the type of defenses that many agencies have in place are ill-equipped to combat today's sophisticated and advanced persistent attacks? Most of the signature-based defensive tools such as firewalls, next generation firewalls, IPS, AV and gateways have remained important security tools, but they have not been able to prevent signature-less based attacks from occurring. The fact remains that traditional defenses such as URL blacklist and signatures [IPS or AV] programs stop "known" attacks, however, they are rendered defenseless against "unknown" advanced targeted attacks [15].

Understanding and manipulating a signature-less analysis is critical to the detection and prevention of polymorphic malware that could be present in the network. When the analyzer engine flags a malicious code, the system blocks its communication ports, IP addresses, and protocols in order to isolate any dangerous transmission. A thorough analysis exposes the intension and activities of the cyber actors, while identifying real threats. It also prevents a false positive and false negatives report.

As network breaches and attacks continue to rise, understanding and managing the risks have become major concern for leaders in business and government. This study will focus on how organizations are responding to threats and recommend mitigation processes.

## 2. LITERATURE REVIEW

[13,14] in their report, noted that a Milan-based information technology Hacking Team company that sells offensive intrusion and surveillance capabilities to governments, law enforcement agencies and corporations was hacked. This company sells spyware to governmental agencies all around the world including agencies in Ethiopia, Morocco, the United Arab Emirates as well as the US Drug Enforcement Administration. Hackers made 400GB of client files, contracts,

financial documents and e-mails available for download to the general public.

[5] In another study noted that Russian hackers gained access to unclassified White House systems in 2014. Their goal was to access President's schedules and emails that revealed personnel moves and policy matters.

[6,7] among others, cited that Multiple breaches at the U.S. government's Office of Personnel Management in 2015 led to theft of data on 22 million current and former federal employees that included the fingerprints of about 5 million people. Among those affected were members of law enforcement and intelligence communities. The agency was cited as having lots of problems, including the lack of a comprehensive inventory of its IT assets.

[10, 12] in their report cited that two major health insurers, Anthem and Premera, were hacked, likely by the same actor, resulting in the largest theft of medical records in history. However, both break-ins were discovered on the same day, leading some to think law enforcement had discovered the attacks and tipped off the victims. The perpetrators seemed to be after intelligence as opposed to data they could sell for cash, indicating that a nation might be behind it. The breaches involved methods and tactics linked to a Chinese group known as Deep Panda.

According to reports in [7, 14], it was revealed that in 2015, Ashley Madison data was compromised that contained over 37 million of customer records. These include millions of accounts and passwords that became vulnerable by a bad MD5 hash implementation. The study could not ascertain how the actors got into the systems but summarized that the attackers posted personal information of customers seeking extramarital affairs with other married persons, which led to embarrassment, and in two cases, possible suicides.

A recent study found that multiple breaches at the U.S. government's Office of Personnel Management led to the theft of data on 22 million current and former federal employees that included the fingerprints of about 5 million persons. Among those affected were members of

law enforcement and intelligence communities. The studies summarized that the agency had lots of problems, including the lack of a comprehensive inventory of its IT assets.” [3, 7, 17].

In other reports [8,9], it was stated that the Credit Bureau and consumer data broker Experian North America disclosed that a breach of its computer systems exposed nearly 15 million Social Security and other data on people who applied for financing from wireless provider T-Mobile USA Inc. Experian reported that the compromise of its internal server exposed the names, dates of birth, addresses, Social Security numbers and/or drivers’ license numbers, as well as additional information of its patrons used in T-Mobile’s own credit assessment. However, the Costa Mesa, California-based data broker issued the statement that no payment card or banking details were stolen, and that the intruders were not able to steal its consumer credit database.

Early studies as reported by [16], [18], found evidence that in 2013, the watering hole Internet Explorer 8 zero-day attack on the US Department of Labor website attack, spread to nine global websites, including those run by a big European company operating in the aerospace, defense, and security industries. The report found evidence that IP addresses that were compromised included addresses from thirty-seven countries of which seventy-one percent of those were in the USA, while eleven-percent were from South/Southeast Asia and ten-percent from Europe.

According to [1,2], Black Hat Hacker Survey report showed evidence that hackers were able to penetrate the Security of Thycotic Secret Server and compromise patrons privileged accounts captioned as [ the “keys to the kingdom”]. The Thycotic Secret Server was designed to deliver an indispensable, comprehensive Privileged Account Management (PAM) solution and protect its patrons from cyber-attacks and insider threats. The study concludes that the cyber-actors believe that it is very easy now with all the new technology available to break into any system and steal Patrons privileged account credentials compared to the past years.

[10,] In their report, summarized that the IRS in 2015 released information about a massive data breach when hackers stole information from 330,000 taxpayers to successfully fill bogus tax refunds and obtain over 450 million in federal funds. The actors used stolen credentials and knowledge-based authentication information technology to enter the IRS filing and refund systems.

### **3. METHODOLOGY**

In order to pilot-test the network-security concerns, the authors constructed, distributed and collected responses from survey questionnaires at a network-security business professional conference in Oct 2015 at Orlando Florida.

The survey population comprises of professionals who publish research findings and work in their respective fields. These are experts with extensive history in teaching and in the business world. Survey data was distributed to senior IT professionals from midmarket (100 to 999 employees) and enterprise-class (1000 employees or more) organizations.

The survey questionnaires were distributed to 366 attendees. The number completed and returned was 250. Overall, we consider these as an equitable representative random population. Most of the survey items were Likert scale types, yes/no responses or categorical, ordinal items, gender, ranks of personnel, etc.

The study conducted a survey of 23 questions covering a range of security issues that are of importance and of concern to IT and security administrators in small and medium size businesses [SMBs]. The questions were designed and conducted to obtain a snapshot of the state of security issues in SMBs and to confirm issues that have been raised in other security studies.

### **4. DATA ANALYSIS**

This study analyzes the response from participants regarding what they deem as security threats as it relates to these attacks: session hijacking, IP address spoofing attack, waterhole, web application attack, malware attacks, Java attacks, zero-day attacks, denial of service/distributed denial of service (DDos),



and advanced persistent threats (APTs), shell shock etc. A total of 11 hypotheses are analyzed. Paired

samples statistics and correlation techniques are used for data analysis.

#### 4.1 FINDINGS/RESULTS

**Table 1: Frequencies**

		Male	Female		Exec	IT
V3	Co_Nwk_Scr	87%	86%	Secure +	87%	85%
V4	Org_Nwk_Sec_Sys_Effective	85%	87%	Agree +	83%	87%
V5	More_Co_Invstmt_IDS	78%	72%	Agree +	76%	76%
V6	Freq_Phishing	80%	90%	Moderately +	88%	87%
V28	Java_Attks_Most_Freq	21%	15%	Moderately +	20%	18%
V29	WaterHole_Attks_Most_Freq	26%	21%	Moderately +	25%	20%
V30	OS_Attks_Most_Freq	81%	81%	Moderately +	82%	80%
V31	ZeroDay_Attks_Most_Freq	79%	75%	Moderately +	79%	77%

A sampling of the frequency output for all the variables indicates that the concern for cybersecurity threats is uniform across all variables. However, there does not seem to be that much concern for Java attacks and Waterhole attacks because the percentage of respondents that consider such threats to be very intrusive is less

than 30%. This resulted in performing a t-test to compare these 2 sets of variables.

Hypothesis 1: There is a significant difference between the concern for Java Attacks compared to concern for other types of Attacks.

**Table 2: Paired Comparison of Java Attacks Vs Other Types of Attacks**

Paired Samples Statistics						
		Mean	N	Std. Deviation	Std. Error Mean	
Pair 1	V28: How often have Java Attack Exploits hit your Organization?	3.19	244	.412	.026	
	V32_Aggregate	4.0519	244	.53462	.03423	
Paired Samples Correlations						
				N	Correlation	Sig.
Pair 1	V28: How often have Java Attack Exploits hit your Organization? & V32_Aggregate			244	.064	.317
Paired Samples Test						
		Paired Differences				
		Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference	

					Lower
Pair 1	V28: How often have Java Attack Exploits hit your Organization? - V32_Aggregate	-.86339	.65386	.04186	-.94584
<b>Paired Samples Test</b>					
		Paired Differences			
		95% Confidence Interval of the Difference			Sig.
		Upper		t	df
Pair 1	V28: How often have Java Attack Exploits hit your Organization? - V32_Aggregate	-.78093		- 20.626	243 .000

The sample mean of the two sets of variables (3.19 and 4.05), and the correlation coefficient of 0.064 support the results of the t-test with a value of -29.526. At the 1% significance level, this confirms the hypothesis that there is a significant

difference between the concern for Java attacks compared to concern for other types of Attacks. Hypothesis 2: There is a significant difference between the concern for WaterHole Attacks compared to concern for other types of Attacks.

<b>Table 3: Paired Comparison of WaterHole Attacks Vs Other Types of Attacks Paired Samples Statistics</b>					
		Mean	N	Std. Deviation	Std. Error Mean
Pair 1	V29: How often have WaterHole Attack Exploits hit your Organization?	3.14	244	.683	.044
	V32_Aggregate	4.0519	244	.53462	.03423
<b>Paired Samples Correlations</b>					
			N	Correlation	Sig.
Pair 1	V29: How often have WaterHole Attack Exploits hit your Organization? & V32_Aggregate		244	.065	.314
<b>Paired Samples Test</b>					
		Paired Differences			
				Std. Error Mean	95% Confidence Interval of the Difference
		Mean	Std. Deviation		Lower

Pair 1	V29: How often have WaterHole Attack Exploits hit your Organization? - V32_Aggregate	-.91257	.83947	.05374	-1.01843
--------	--	---------	--------	--------	----------

Paired Samples Test						
		Paired Differences		t	df	Sig. (2-tailed)
		95% Confidence Interval of the Difference				
		Upper	Lower			
Pair 1	V29: How often have WaterHole Attack Exploits hit your Organization? - V32_Aggregate		-.80671	-16.981	243	.000

The sample mean of the two sets of variables (3.14 and 4.05), and the correlation coefficient of 0.065 support the results of the t-test with a value of -16.981. At the 1% significance level, this confirms the hypothesis that there is a significant difference between the concern for Waterhole Attacks compared to concern for other types of Attacks.

Concern for security in an organization is predicated on the actual incidence of certain types of cyber-attacks. If the respondent experiences a high incidence of a particular type of attack, there is the likelihood that they will be more concerned

about that type of attack. A regression analysis of this dependency is run on a number of variables, and the results are shown below:

Hypothesis 3: The concern about network security threats is strongly dependent on several factors – including the incidence of IP address spoofing attack.

When a regression analysis is run on *concern about network security* as a function of other types of attacks, the result shows that the most significant factor that predicts such concern is the frequency of IP address spoofing attacks.

**Table 4: Regression of Concern about Network Security on incidence of Attacks.**

Coefficients <sup>a</sup>						
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	.175	.235		.745	.457
	V16: How often has your Organization been hit by IP address spoofing Attack	.732	.051	.719	14.239	.000

Hypothesis 4. The concern that *employees pose the greatest network security threats* is strongly dependent on How often has your Organization

been hit by Web Application Attacks, How often has your Organization been hit by IP address spoofing Attack.

**Table 5: Regression of Concern about Employee Threats on Frequency of Attacks.**

Coefficients <sup>a</sup>						
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	.218	.470		.463	.644
	V9: How often has your Organization been hit by Web Application Attacks [buffer overflows, XML,SQL injections]?	.183	.089	.146	2.062	.040
	V11: How often has your Organization been hit by Advanced persistent threats [APTs] targeted Attacks - RATs	.328	.167	.240	1.964	.051
	V16: How often has your Organization been hit by IP address spoofing Attack	1.013	.103	.782	9.845	.000

Hypothesis 5. The concern that *Foreign Nation-States pose the greatest network security threats* is strongly dependent on how often your Organization has been hit by IP address spoofing Attack, and how often have Operating System Attack Exploits hit your Organization.

**Table 6: Regression of Concern about Foreign Nation States on Frequency of Attacks.**

Coefficients <sup>a</sup>						
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	.534	.234		2.279	.024
	V16: How often has your Organization been hit by IP address spoofing Attack	.824	.051	.806	16.067	.000

	V30: How often have Operating System Attack Exploits hit your Organization?	.122	.059	.117	2.061	.040
--	---	------	------	------	-------	------

Hypothesis 6. The concern that *3<sup>rd</sup> party Contractors/Vendors pose the greatest network security threats* is strongly dependent on How often your Organization has been hit by Mobile

device malware, How often Sandboxes - environment with limited functionality are used to test untrusted code, and How often Operating System Attack Exploits hit your Organization.

**Table 7: Regression of Concern about 3<sup>rd</sup> Party Contractors on Frequency of Attacks.**

Coefficients <sup>a</sup>						
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	-.027	.174		-.153	.878
	V13: How often has your Organization been hit by Mobile device malware [smartphones, tablets] Attacks	.419	.064	.405	6.568	.000
	V27: How often are Sandboxes - environment with limited functionality used to test untrusted code?	.408	.039	.410	10.394	.000
	V30: How often have Operating System Attack Exploits hit your Organization?	.258	.044	.246	5.842	.000

Hypothesis 7. The frequency of Anti-virus, Anti-Malware techniques to counter Advanced Persistent Threats is strongly dependent on How

often your Organization has been hit by Web Application Attacks.

**Table 8: Regression of Frequency of Mitigation Techniques on Frequency of Attacks.**

Coefficients <sup>a</sup>						
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	4.416	.599		7.367	.000
	V9: How often has your Organization been hit by Web Application Attacks [buffer overflows, XML, SQL injections]?	.230	.113	.190	2.037	.043

Hypothesis 8. The frequency of Network Technologies to counter Advanced Persistent Threats is strongly dependent on How often your Organization has been hit by Zero-Day Attacks, How often your Organization has been hit by Advanced persistent threats [APTs] targeted

Attacks; How often your Organization has been hit by Mobile device malware [smartphones, tablets] Attacks; How often Sandboxes - environment with limited functionality are used to test untrusted code; and how often Operating System Attack Exploits hit your Organization.

**Table 9: Regression of Frequency of Network Technologies to Counter Threats, on Frequency of Attacks.**

Coefficients <sup>a</sup>						
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	.637	.199		3.207	.002
	V8: How often has your Organization been hit by Zero-Day Attacks?	-.728	.353	-.738	-2.063	.040
	V11: How often has your Organization been hit by Advanced persistent threats [APTs] targeted Attacks - RATs	.294	.071	.306	4.170	.000



V13: How often has your Organization been hit by Mobile device malware [smartphones, tablets] Attacks	.207	.073	.223	2.838	.005
V27: How often are Sandboxes - environment with limited functionality used to test untrusted code?	.307	.045	.345	6.853	.000
V30: How often have Operating System Attack Exploits hit your Organization?	.448	.050	.479	8.908	.000

Hypothesis 9. The frequency of Network Segregation to counter Advanced Persistent Threats is strongly dependent on How often your Organization has been hit by SSL-encrypted threats - BOT distribution; How often your Organization has been hit by Mobile device

malware [smartphones, tablets] Attacks, How often Sandboxes - environment with limited functionality are used to test untrusted code; and how often Operating System Attack Exploits hit your Organization.

**Table 10: Regression of Frequency of Network Segregation on Frequency of Attacks**

Coefficients <sup>a</sup>						
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	1.036	.378		2.742	.007
	V12: How often has your Organization been hit by SSL-encrypted threats - BOT distribution	.186	.087	.182	2.141	.033
	V13: How often has your Organization been hit by Mobile device malware [smartphones, tablets] Attacks	.378	.138	.353	2.730	.007
	V27: How often are Sandboxes - environment	.182	.085	.177	2.136	.034

	with limited functionality used to test untrusted code?					
	V30: How often have Operating System Attack Exploits hit your Organization?	.272	.096	.251	2.842	.005

Hypothesis 10. The frequency of *Rogue Device Scanning* to counter Advanced Persistent Threats is strongly dependent on how often your Organization has been hit by Phishing / spear—

Phishing Attacks; how often are Sandboxes - environment with limited functionality used to test untrusted code; and how often Operating System Attack Exploits hit your Organization.

**Table 11: Regression of Frequency of Rogue Device Scanning to Counter Threats, on Frequency of Attacks.**

Coefficients <sup>a</sup>						
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	-.095	.119		-.799	.425
	V6: How often has your Organization been hit by Phishing / spear—phishing Attacks?	-.045	.018	-.049	-2.462	.015
	V27: How often are Sandboxes - environment with limited functionality used to test untrusted code?	.137	.027	.137	5.119	.000
	V30: How often have Operating System Attack Exploits hit your Organization?	.916	.030	.875	30.538	.000

Hypothesis 11. The frequency of *Log Monitoring/Event Correlation* to counter Advanced Persistent Threats is strongly dependent on how often your Organization has been hit by Malware [virus, worms, Trojans]

Attacks; how often your Organization has been hit by Zero-Day Attacks; how often your Organization has been hit by SSL-encrypted threats - BOT distribution; how often Sandboxes - environment with limited functionality are used

to test untrusted code; and, how often Operating System Attack Exploits hit your Organization.

**Table 12: Regression of Frequency of Log Monitoring/Event Correlation to Counter Threats, on Frequency of Attacks.**

Coefficients <sup>a</sup>						
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	1.432	.360		3.980	.000
	V7: How often has your Organization been hit by Malware [virus, worms, Trojans] Attacks?	-.171	.076	-.149	-2.251	.025
	V8: How often has your Organization been hit by Zero-Day_Attacks?	-1.252	.639	-1.022	-1.961	.051
	V12: How often has your Organization been hit by SSL-encrypted threats - BOT distribution	.177	.083	.161	2.140	.033
	V16: How often has your Organization been hit by IP address spoofing Attack	-.104	.079	-.092	-1.327	.186
	V27: How often are Sandboxes - environment with limited functionality used to test untrusted code?	.388	.081	.351	4.781	.000
	V30: How often have Operating System Attack Exploits hit your Organization?	.434	.091	.374	4.767	.000

## CLUSTER ANALYSIS

A Cluster Analysis of the data indicates that the respondents can be segmented into 3 basic clusters.

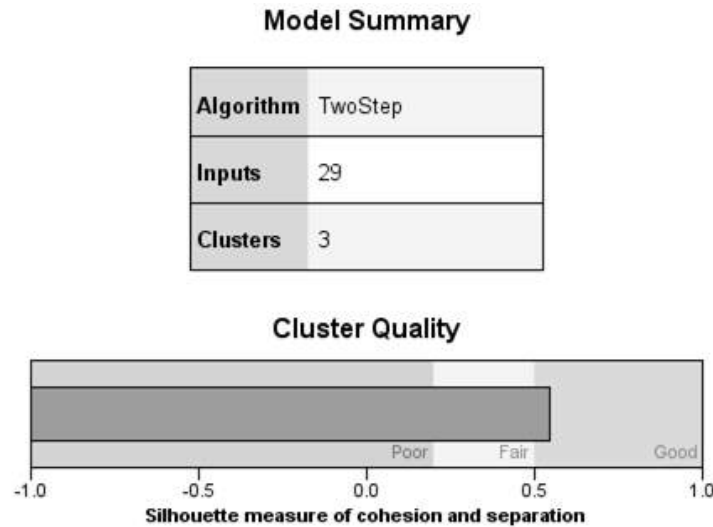
Cluster 1 respondents, which comprises 61% of the respondents, tend to have a slightly higher than moderate attitudes to cyber-attacks.

Cluster 2 respondents, which comprise 21% of the respondents, tend to have a slightly higher than moderate attitude to cyber-attacks; but in addition, have a few cases that are apparently not

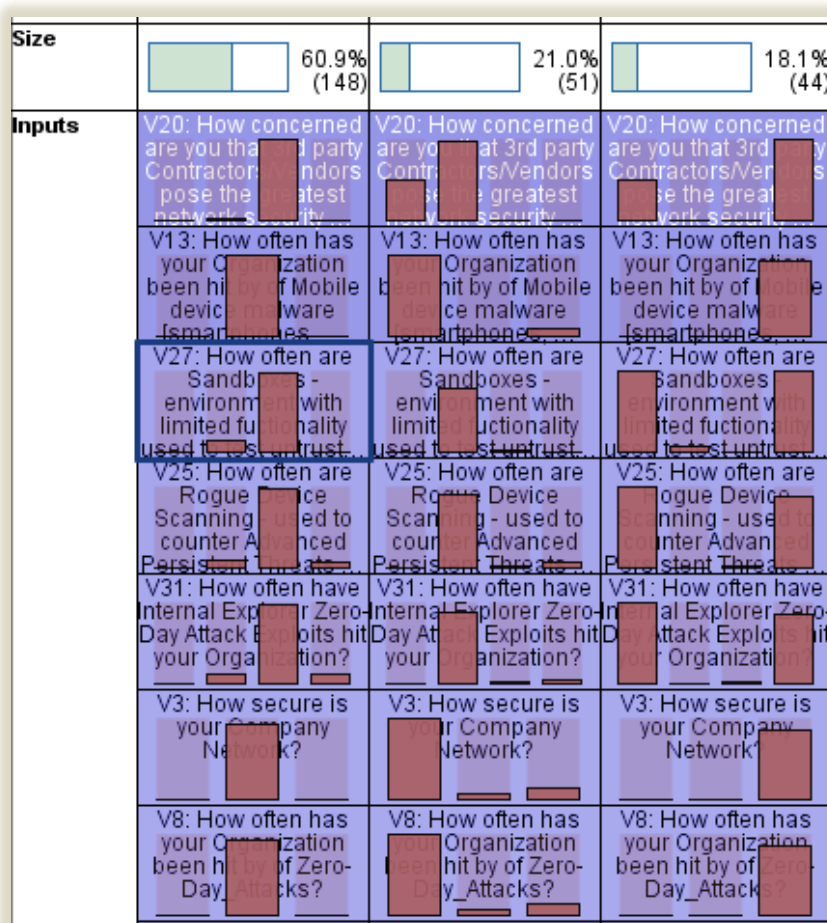
as concerned about cybersecurity as the other respondents.

Cluster 3 respondents, which comprise 18% of the respondents, tend to have extreme attitudes towards cybersecurity. Overall, the proportion of respondents who are not as concerned about cybersecurity threats as there are those that are concerned, is higher in this cluster than in clusters 1 & 2.

The Cluster quality is determined to be reliable, as shown in the chart below.



**FIGURE 1: SUMMARY OF CLUSTER QUALITY CHART**



**5. CONCLUSION**

There is consistency in the attitude of respondents to the cyber threats. By focusing on specific factors, it is possible to predict their effects on the

attitude of the respondents to the threat of cyberattacks. By performing a Cluster Analysis, one can conclude that the respondents fall into three basic categories – the Moderate respondents, the respondents who are more than

moderately concerned about cyberattacks, and the respondents who view cyberattacks as an imminent threat to their organization.

## **6. IMPLICATIONS FOR PRACTITIONERS AND RESEARCHERS**

When signature-less malicious code malware is identified and analyzed, the information from a PCAP, Wire-shack or a Virus-Total engine should be fully leveraged. The goal should be to use the fingerprint or pattern of the malicious signature-less code to identify and remediate compromised network and prevent the infection from spreading to other parts of the network. Forensic records from a Virus-Total result, can then be run through automated offline mechanism to attest and dissect the malicious code.

## **7. SUMMARY**

Cyber-attacks today represent an immediate threat to the cyber world. Effective safeguards that can thwart signature-less sophisticated attacks need to be deployed that avoid enterprise systems increasing risk of devastating breaches that result in the compromise of confidential and classified information.

Implementing solutions that supplement traditional and generation firewalls, IPs, AV and gateways will stop advanced targeted attacks that use shellshock, advanced malware, zero-day, advanced persistent exploits [APTs], Ghost RAT etc. creating holes in the network.

Deploying a real-time, coordinated security posture capable of preventing today's signature-less attacks, will ensure enterprise sensitive assets are not compromised.

## 8. REFERENCES

1. Acohido, B.: Improving Detection, Prevention, and Response with Security Maturity Modeling. <https://www.sans.org/reading-room/Whitepapers/analyst/improving-detection-prevention-response-security-maturity-modeling-35985>. (2015, May),
2. Chambers, J., Stewart, J.N.: Why Cybersecurity Leadership Must Start At The Top. *Forbes*, <http://www.forbes.com/sites/frontline/2015/07/13/why-cybersecurity-leadership-must-start-at-the-top/#5af3a2234f3e> (2015, July, 13).
3. Department of Homeland Security (DHS). Continuous Mitigation and Diagnostics. <http://www.dhs.gov/cdm>. (2015, November 6).
4. FINRA Report on Cybersecurity Practices. [https://www.finra.org/sites/default/files/p602363%20Report%20on%20Cybersecurity%20Practices\\_0.pdf](https://www.finra.org/sites/default/files/p602363%20Report%20on%20Cybersecurity%20Practices_0.pdf) (2015, February).
5. Harris, K.D.: California Data Breach Report. <https://oag.ca.gov/breachreport2016#findings> (2016, February).
6. Vaughan N.S.: Securing the Internet: Let's Encrypt to Release First Security Certificate September 7, DNet, and August 24. <http://www.zdnet.com/article/securing-the-Internet-let-encrypt/> (2015).
7. Villeneuve N.: TeslsaaCrypt: Following the Money Trail and Learning the Human Costs of Ransomware, <https://www.fireeye.com/blog/threat-research/2015/05/teslsaa-crypt-followin.html>. (2015, May 15).
8. Berthiaume, D.: Amazon Shuts Digital Wallet, Chain Store Age, <http://www.chainstoreage.com/article/amazon-shuts-digital-wallet> (2015, January 21).
9. Carter, M.: Mobile Wallets Are Not Convenient Enough for Consumers, Payments Source. <http://www.paymentsource.com/news/paythink/mobile-wallets-are-not-convenient-enough-for-consumers-3022186-1.html> (2015, August 27).
10. Cyber Source.: 15<sup>th</sup> Annual Online Fraud Report: Online Payment Fraud Trends, Merchant Practices and Benchmarks. Visa-Cyber Source, San Francisco, CA (2015).
11. Lawrence, D.: Spy vs Spy: The US Government Designed and Funds the Best Defense Against its Own Surveillance, *Bloomberg Businessweek*, 42--47 (2014, January 27).
12. Perlroth, N.: I.R.S. Breach Demonstrates the Need to Guard Personal data, *The New York Times*, May 28, B2 (2015),
13. Smith, A.: Newly Discovered Sophisticated Malware Has Been Spying on Computers for Six Years, *Newsweek*, <http://europe.newsweek.com/new-sophisticated-malware-has-been-spying-computers-six-years-2886640> (2014, November 24).
14. Zetter, K.: Attackers Stole Certificate from Foxconn to Hack Kaspersky With Duqu 2.0, *Wired*, <http://www.wired.com/2015/06/foxconn-hack-kaspersky-duqu-2/> (2015, June 15).
15. Regalado, D.: Backdoor Ploutus Reloaded –Ploutus Leaves Mexico, Symantec (*blog*), [www.symantec.com/connect/blogs/backdoorploutus-reloaded-ploutus-leaves-mexico](http://www.symantec.com/connect/blogs/backdoorploutus-reloaded-ploutus-leaves-mexico) (2013, Oct. 25).
16. Moore, H.: Serial Offenders: Widespread Flaws in Serial Port Servers, *Security Street Rapid*, <http://community.rapid7.com/community/metasploit/blog/2013/04/23/serial-offenders-widespread-flaws-in-serial-port-servers> (2013, Apr. 23).
17. Watson, J.: Zombie Apocalypse' Training Drill Organized by Halo Corp. for Military, Police Set for Oct. 31 in San Diego, *Huffington Post*, [www.huffingtonpost.com/2012/10/29/zombie-apocalypse-training-military-halo-corp-2036996.htm](http://www.huffingtonpost.com/2012/10/29/zombie-apocalypse-training-military-halo-corp-2036996.htm) (2012, Oct. 27).
18. Radcliffe, J.: Hacking Medical Devices for Fun and Insulin: Breaking the Human SCADA System, *Blackhat Briefing & Training USA + 2011*, [www.blackhat.com/html/bh-us-11/bh-us-11-briefings.html](http://www.blackhat.com/html/bh-us-11/bh-us-11-briefings.html)
19. Smith, A.: Newly Discovered Sophisticated Malware Has Been Spying on Computers for Six Years, *Newsweek*, <http://europe.newsweek.com/new-sophisticated-malware-has-been-spying-computers-six-years-2886640> (2014, November 24).

➤ The main objectives of this journal with regard to security, privacy, digital forensics, hacking, and cyber warfare are as follows:

- Encouraging the study, improve the practice, and advance the knowledge;
- Providing the intended audiences with the latest advancements;
- Transferring the knowledge;
- Closing the gap between academia and the industry;
- Providing trusted source of knowledge;
- Encouraging talents and innovations;
- Supporting collaboration and communication;
- Encouraging the applied research.

The IJCSDF scope covers the following areas (but not limited to): cyber security, computer forensics, privacy, trust, hacking techniques, cyber warfare, cryptography, cybercrime, cyber-terrorism, cryptography, formal methods application in security and forensics, data piracy, database security and forensics, wired and wireless network security and investigation, mobile network security and forensics, incident handling, malware forensics and steganography.

➤ The IJCSDF is published four (4) times a year and accepts three types of papers as follows:

**Research papers:** that are presenting and discussing the latest, and the most profound research results in the scope of IJCSDF. Papers should describe new contributions in the scope of IJCSDF and support claims of novelty with citations to the relevant literature. Maximum word limit of 8000!

**Technical papers:** that are establishing meaningful forum between practitioners and researchers with useful solutions in various fields of digital security and forensics. It includes all kinds of practical applications, which covers principles, projects, missions, techniques, tools, methods, processes etc. Maximum word limit of 5000

**Review papers:** that are critically analyzing past and current research trends in the field. Maximum word limit of 12000!

**Book reviews:** providing critical review of a recently published book in the scope of IJCSDF. Maximum word limit of 1000!



## Volume 5, Issue 4

## CONTENTS

## ORIGINAL ARTICLES

<b>Method for Detecting a Malicious Domain by using only Well-known Information.....</b>	<b>166</b>
Author(s): Masahiro Kuyama, Yoshio Kakizaki, Ryoichi Sasaki	
<b>Digital Forensic Analysis of Ubuntu File System.....</b>	<b>175</b>
Author(s): Dinesh N. Patil, Bandu B. Meshram	
<b>A Preferential Analysis of Existing Password Managers from End-Users' View Point.....</b>	<b>187</b>
Author(s): S. Agholor, A. S. Sodiya, A. T. Akinwale, O. J. Adeniran and D. O. Aborisade	
<b>Intrusion Detection System with Spectrum Quantification Analysis.....</b>	<b>197</b>
Author(s): Yusuke Tsuge, Hidema Tanaka	
<b>“The Unwitting Danger Within - Detection, Investigation and Mitigation of a Compromised Network” .....</b>	<b>208</b>
Author(s): Emmanuel U Opara, Oredola A. Soluade	