Research And Implementation of Computer Network User Behavior Forensics System based on System a Log

Wen Zhe Lu
Computer Network Information Center, Chinese Academy of Sciences
Beijing 100190, People's Republic of China
University of Chinese Academy of Sciences, Beijing 100190
People's Republic of China.

ABSTRACT: *In recent years, many computer forensics theoretical methods have been developed to provide efficient means to counter computer crimes. Computer evidence must be accurate and thorough. A design of architecture of a forensics system is given in this design, and the log data is the key Computer evidence to analyse. One of the key issues this paper tries to resolve is the log-data integrity. In the system, CES algorithm is used to protect and verify the integrity of log data. Another key issue is how to analyse the computer evidence accurately. A timestamp-based multi-characters log analysis method is also discussed in this paper. This method is to realize and tag the relationship of time-line sequence which is a reasonable way to identify the user's behaviour. The result comes out that the forensics technique will be more integrated and thorough.*

Keywords: Computer Security, Computer Forensic, System Log

Received: 24 May 2017, Revised 20 June 2017, Accepted 15 July 2017

1. Introduction

After more than 20 years of development, China's Internet development and popularity level has been the forefront of developing countries. According to the "37th China Internet Development Survey Report", released by China Internet Information Center (CNNIC), shows that as of the end of December 2015, the number of Chinese Internet users reached 688 million, the Internet penetration rate of 50.3%, and the total number of sites has risen to 423 million.

At the same time, computer crime has also caused significant losses to the public, and the key to combating computer crime is to find full, reliable, and convincing electronic evidences; that is why the computer and legal interdisciplinary - computer forensics (computer forensics) has been getting more and more attention. For several consecutive years, it become FIRST (forum of incident response and security teams) safety conference in recent years, where computer forensics technology has been widely used. Forensic experts can conduct in the laboratory analysis from the crime scene, use logs of the computer and the network device, trying to figure out who, when, where and how the illegal activity were carried out.

Computer forensics only applies computer research and analysis techniques to identifying potential legal evidence. May seek evidence in a wide range of computer offenses or misuse, including but not limited to theft of trade secrets, theft or

destruction of intellectual property and fraud. Computer experts can use a series of methods to discover data that resides in a computer system, or to recover deleted, encrypted, or corrupted file information. Any or all of this information may help to discover actual litigation.

On the other hand, the protection of evidence is essential. The knowledgeable computer forensics professionals will ensure that the subject computer system is carefully processed to ensure that:

• No possible evidence is damaged, destroyed, or otherwise compromised by the procedures used to investigate the computer.

• No possible computer virus is introduced to a subject computer during the analysis process.

• Extracted and possibly relevant evidence is properly handled and protected from later mechanical or electromagnetic damage.

• A continuing chain of custody is established and maintained.

• Business operations are affected for a limited amount of time, if at all.

• Any client-attorney information that is inadvertently acquired during a forensic exploration is ethically and legally respected and not divulged.

2. Literature Review

In the study of user behavior analysis technology, in 2000, Wood was the first one to propose an integrated approach to designing a model. Based on the accurate analysis of internal users with malicious attacks, he puts forward the three necessary conditions for internal attacks: internal attacks must have attack motives, identify targets and initial attacks. The downside is that it does not give the standard for measuring and quantifying internal threats, but only taking into account the user's intentional or malicious behavior, without taking into account the occasional misuse of users. In 2002, Schultz proposed a framework that could predict internal threats. The structure introduces the concept of Indicators, used to describe the attack category, and behavioural characteristics. But it has a certain degree of subjectivity, and the unpredictable attack on the lack of flexible and effective detection capabilities. Subsequently, Magklara proposed an internal threat prediction model. The model implements the quantification and qualitative approach to assessing metrics and captures internal threats in advance, including malicious events and incidental events. However, this internal threat prediction model is too abstract, mostly staying in the theoretical stage, and many implementation details are not taken into account.

Recently, computer forensics has garnered much attention over the past ten to fifteen years despite being a relatively nascent scientific field. This attention is due to the sheer amount of data being generated by modern computer systems, which has become an essential source of digital evidence (Geiger, 2005). Scientifically valid and lawful forensic investigation of this digital evidence seeks to uncover and discern its meaning where the evidence must be both reliable, accurate and complete (Harris, 2006).

There has not been, however, as much attention, especially in the form of academic research, towards what can be deemed as "anti-digital forensics", "anti-forensics", or "counter-forensics" (Baggili et al., 2012). Although, one could argue that some research such as cryptography may be regarded as anti-forensic, but has not been labelled as such in the literature, thus skewing our perception of the amount of anti-forensic research being conducted. Nonetheless, anti-forensics generally means: attempts to compromise the availability or usefulness of evidence during the forensics process.

3. Realization Model of Forensic Analysis of Computer Network User Behavior Based on System Log

3.1 System Architecture

This paper presents a system which based on the system log of computer network user behavior forensic analysis system model, the system is an agent based on the distributed forensics analysis system, and the main idea of system design is to focus on the LAN server system log to a dedicated evidence collection and forensic analysis on Forensic Server. The theoretical model of the system is described. As shown in the following figure, the system consists of two parts: the log collection agent and the forensic server. The system adopts the client-server structure. The log collection agent is client-side and is deployed on each protected

server in the local area network. It is responsible for verifying, collecting and sending the system log information on the target machine. The configuration file defines the collection agent various parameters such as acquisition object, acquisition time interval, server address and port, PKI certificate and key file name and path. The forensic server is the server side of the system, which is responsible for receiving and saving the log data sent by the Agent side. The original log data is also protected and authenticated. The forensic server is also responsible for the pre-processing and management of the log data. The forensic analysis is the most important function of the forensic server, and it is necessary to analyse the log data in the database according to the instructions of the forensic personnel. Finally, the forensic server is responsible for analysing the formation and submission of the results. According to the evidence analysis, the log records associated with the computer crime are extracted from the original log file, resulting in written evidence of compliance with legal requirements. The information transmission between the client and the server adopts a secure transmission mode, such as the SSL protocol, which enables the authentication between the client and the server to authenticate each other. The encrypted data and the digital signature are used to ensure the log information integrity.

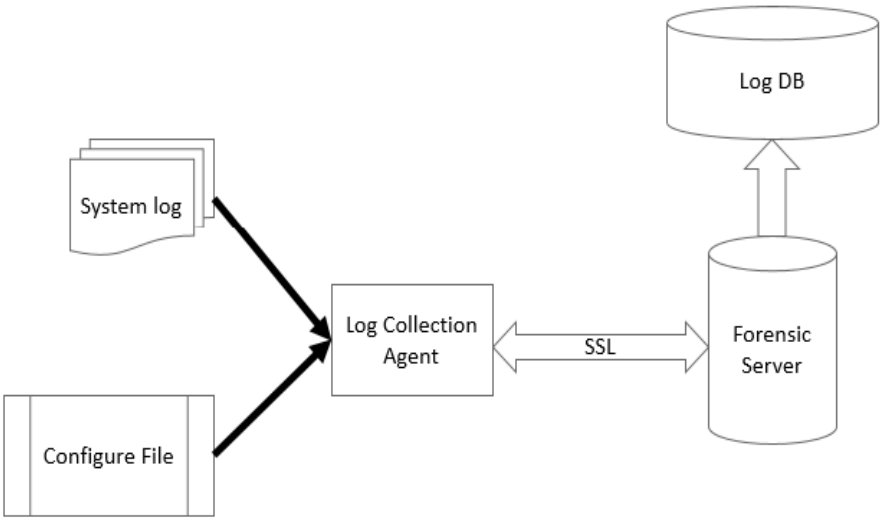

Figure 1. System Architecture

3.2 Key Methods

3.2.1 Collection Frequency

Operating system and the applications have a wide range of logs, not all of them are related to security events, and some log content are duplicated and redundant. Therefore, in order to improve the efficiency, it is better to select the security-related system logs in advance,. The following security audit events should be focused:

- **Auditing login events:** Every successful or failed intrusion is initiated from the login, collecting the information about successful or failed local logins, remote logins, and network shares in the computer system.

- **Auditing account management:** Including account creation, deletion, renaming, locking and activation, monitoring the user-rights, for example, the user password.

- **Auditing object access:** Mainly include the directory and file access on the drive, the printer, the registry access, etc., involving files, directories and other objects to open, close, delete, change the object content and attributes, software installation and uninstallation.

- **Auditing privilege use:** Whether the user-rights conform to the user rights defined in Group Policy or local policy.

- **Auditing process tracking:** The creation and the end of the process, and the ACL of the process.

• **Auditing system events:** System events are generated when the user or process changed, such as system start-up and shutdown, system time adjustment, etc.

• **Auditing policy event:** Used to determine whether each event of a change user privilege assignment policy, audit policy, or trust policy is audited.

3.2.2 Algorithm Process

There are three important factors to be considered, velocity of syslog generation, importance of the computer system, and the risk of the computer system. These factors will be considered as same weight of influence, and define $\boldsymbol{F}_{velocity}$, $\boldsymbol{F}_{importance}$ and $\boldsymbol{F}_{breach}$ as the three factors. The values could be 0.1-1.0, and the increment is 0.1, which means the average speed and severity of the log are increasing. Define *t* as the time interval.

The formula for single log file is:

$$t = 3 + 30 * (1 - \frac{F_{velocity} + F_{importance} + F_{breach}}{3})$$

t is discrete, and it could be one of the ten values, 3 minutes, 6 minutes or 30 minutes. When the three influencing factors take the maximum value of 1.0, ***t*** takes the minimum, T_min = 3 minutes. When the three factors take the minimum 0.1, t is the maximum, T_inc = 30 minutes, the time increment is T_inc = 3 minutes. The minimum time interval and the maximum time interval and the time interval increment value are based on the system log and experimental experience. The formula is also an experienced one. The value of the three influencing factors can be based on the past system log statistics and network environment to decide. When the network environment changed, or the network security situation changed, the time interval parameters in the configuration files could be changed at any time.

To adjust the time interval, define ***V*** as the average speed of the log generation, and ***Sum*** is the total time of the system log in the one cycle, and ***t*** is the time interval, ***V=Sum/t***, ***V1*** and ***V2*** represent the average generation speed of the previous cycle and the current cycle. ***Told*** and ***Tnew***, respectively, on behalf of the current collection interval and the next interval, time adjustment algorithm is as follows:

Time Intervel Adjust Algorithm

if $\left(\frac{V_2 - V_1}{V_1} > 10\%\right)$ *then*
    $\boldsymbol{T_{new}} = \boldsymbol{T_{old}} - \boldsymbol{T_{inc}}$;
else if $\left(\frac{V_2 - V_1}{V_1} < -10\%\right)$ *then*
    $\boldsymbol{T_{new}} = \boldsymbol{T_{old}} - \boldsymbol{T_{inc}}$;
else
    $\boldsymbol{T_{new}} = \boldsymbol{T_{old}}$
if ($T\_min < \boldsymbol{T_{new}} < T\_max$) *then*
    return $\boldsymbol{T_{new}}$
else
    return $\boldsymbol{T_{old}}$

3.2.3 A Timestamp-based and Multi-characters Matching Log Correlation Analysis

Hacker's invasion behavior is not independent, but measured by a series of actions. These actions belong to the different stages of the attack series. The early stage for the later part of the preparation, the later state is the result of pre-behavior. In another word, there is a certain correlation between the invading events issued by the same intruder, the success of the previous invasion phase is the starting point and the necessary condition of the latter invasion stage, that is, the success of an attack is a series of attacks stage of success. And this series of intrusion must be in the log system to leave a series of related log records, these records may belong to a different network equipment or a different log file. To sum up, in the process of forensic analysis of the system log not only in a log file to identify and intrusion-related entries, but also as much as possible to identify the invasion of the incident to

reflect all the log records, and based on the time chain. These log records form a complete sequence of security actions to reconstruct the intrusion event, and the results of the evidence are more proof.

The timestamp is an indispensable attribute item for system log. And some log records also provide the timestamp of action or the end of the intrusion time. And the time-stamps of the system log records associated with the intrusion event must have a sequence relationship. Timestamps are important attributes that are of interest in associative analysis.

The relevancy of the intrusion event has reflection in the log record that some of the attributes are the same or similar. For example, when a user login a computer system, a folder is created, then there are login log and file in the access log, and the usernames of the two records are exactly the same. Therefore, the signatures of the events can be associated by different log records, and the different events can be associated.

According to Yijun's method, it is assumed that all attribute sets of an attack phase Y of an intrusion event are denoted as Y (attributes). Where attributes is the attribute of the attack event, such as timestamp, IP address, port, user name, ext. $\triangle t$ is the pre-defined time interval, h1 and h2 are two attack events, and q is the attribute set of the attack event, and can be based on the security event begintime, endtime is the start time and the end time of the action respectively, the result, condition respectively The result and condition of the action, which is supplemented by the log type when the log data is pre-processed:

If $\forall q \in Y(h1)$

$Then\ q \in Y(h2), and(h1.endtime < h2.begintime) \wedge (h2.begintime - h1.endtime < \triangle t) \wedge (h1.result = h2.condition)$

That means that h2 is the follow-up intrusion event of h1, the two events are associated.

$\triangle t$ is the time interval. It could be adjusted according to the network condition.

If $\forall q \in Y(h1)$

Then $q \in Y(h2)$,and($/ h1.begintime - h2.begintime / < \triangle t) \wedge (h1.result = h2.result)$

That means that h1 and h2 are the same event.

4. Protect the Integrity of System Log

Even the log files are transferred to the forensic server, it is necessary to protect the original log data from being intentionally or unintentionally destroyed or modified by the non-forensic personnel. The main methods of data integrity and protection include information digest technology and digital signature technology. The CES signature algorithm is a suitable solution to sign and verify the log data.

4.1 Introduction of CES

In 2001, CES (Content Extraction Signature) algorithm was proposed by Ron Steinfel, Laurence Bull and Professor Yuliang Zheng in Seoul, at the ICISC conference. CES algorithm is designed to sign the documents with multiple sub-documents. The CES algorithm can provide signature protection for any subset of N data items in a document. It can derive the signature of any data from the signature of the document, the efficiency is higher than the N standard signature cost.

4.2 CES Algorithm

CES has four verification modes: CommitVector, HashTree, RSAProd (based on RSA signature), and MERSAProd (based on RSA signature). Only the CommitVector is used in this system. The definition of signals are following:

***M*:** The information will be signed.
***X*:** The extracted information.
***S*, *V*, *K*:** The standard signature algorithm, verification algorithm and key-generation algorithm.

sk, *pk*: private key and public key
SK, *PK*: private key and public key based on CES.

(1) key-generation algorithm

Algorithm GK(k)

$(sk, pk) \leftarrow K(k)$
$Sp \leftarrow I(k)$
$PK \leftarrow \langle pk \| cp \rangle$
$SK \leftarrow \langle sk \| cp \rangle$
$Return(SK, PK)$

(2) Signature algorithm

Algorithm Sig (*sk*, *M*, *CEAS*)

$C_i \leftarrow Com_{cp}(M[i], r_i)$ for $i \in [n]$
$C \leftarrow Conc_{i=1}^{n} C_i$
$\delta_c \leftarrow S_{sk}(CEAS \| c)$
$\delta_{Full} \leftarrow \langle CEAS \| \delta_c \| Conc_{i=1}^{n} r_i \rangle$
$Return(\delta_{Full})$

(3) Extract algorithm

Algorithm Extract (M, δ_{Full}, X)

$\delta_{Full} \leftarrow \langle CEAS \| \delta_c \| Conc_{i=1}^{n} r_i \rangle$
$C_i \leftarrow Com_{cp}(M[i], r_i)$ for $i \in [n] - X$
$\delta_{ext} \leftarrow \langle CEAS \| \delta_c \| Conc_{i \in X} r_i \| Conc_{i \in [n]-X} c_i \rangle$
$Return(\delta_{ext})$

(4) Verify algorithm

Algorithm Verify (M', δ_{ext}, pk)

$\delta_{ext} = \langle CEAS \| \delta_c \| Conc_{i \in Cl(M')} r_i \| Conc_{i \in [n]-Cl(M')} r_i \rangle$
$c_i \leftarrow Com_{cp}(M'[i], r_i)$ for $i \in Cl(M')$
$c \leftarrow Conc_{i-1}^{n} C_i$
$d \leftarrow V_{pk}(CEAS \| c, \delta_c)$
Output "Accept" iff d = "Accept" and $CL(M') \in CEAS$

4.3 Realization of the Algorithm
Based on the OpenSSL, the system realize the CES algorithm. Firstly, generate the key-pairs by OpenSSL command line, then realize the CES functions by API of OpenSSL.

Commit Function:

```
char * Commit(char * msg, char *rand)
{
        EVP_MD_CTX mdctx;
        Const EVP_MD *md;
        OpenSSL_add_all_digests();
        Md = EVP_get_digestbyname("md5");
        EVP_DigestUpdate(&mdctx, msg, strlen(msg));
        EVP_DigestUpdate(&mdctx, rand, strlen(rand));
        EVP_DigestFinal(&mdctx, md_value, &md_len);
}
```

Signature Function:

```
Sign(char * Msg, char * priv_key)
{
        EVP_MD_CTX md_ctx;
        EVP_PKEY *pkey;
        BIO *b;
        b=BIO_new(BIO_s_file());
        BIO_read_filename(b, priv_key)
        pkey=PEM_read_bio_PrivateKey(b, NULL, NULL, NULL);
        EVP_SingInit(&md_ctx, EVP_sha1());
        EVP_SignUpdate(&md_ctx, data, strlen((char*)data));
        EVP_SignFinal(&md_ctx, sig_buf, &sig_len, pkey);
        ……
}
```

Extracting Function:

```
void Extract(char *Msg, char * Xmsg, char * RandName, char *Xcommit, char
*Xrand)
{
        ……
        fMsg = fopen(Msg, "r");
        fXmsg = fopen(Xmsg, "r");
        fRand = fopen(RandName, "r");
        fXcommit = fopen(Xcommit, "w");
        fXrand = fopen(Xrand, "w");
        ……
```

```
        fread(fXmsg, msg', sizeof(msg'));
        fread(rRand, rand',sizeof(rand');
        strcpy(commit, Commit(msg', rand'));
        fwrite(fXCommit, commit, sizeof(commit));
        fwrite(fXrand, rand, strlen(rand));
        ……
    }
    Verification Function:
    Verify(char *Xmsg, char*cert)

{
    EVP_MD_CTX MD_CTX;
    EVP_PKEY * pkey;
    X509    *x509;
    ……
    BIO *c;
    c=BIO_new(BIO_s_file());
    BIO_read_filename(c, cert);
    X509=PEM_read_bio_X509(c, NULL, NULL, NULL);
    pkey=X509_get_pubkey(x509);
    EVP_VerifyInit(&md_ctx, EVP_sha1());
    EVP_VerifyUpdate(&md_ctx, data, strlen((char *)data));
    EVP_VerifyFinal(&md_ctx, sig_buf, sig_len, pkey);
    ……
}
```

5. Conclusions

A considerable limitation of this work was that the system only use the log data. Actually, additional information on the network could be involved in the forensic scope. For example, the connection of the network, the status of the routers could be associated with the system log to identify the security issues.

An additional limitation of this work was that the result of statistical analysis of the log data is more reliance on forensic personnel. The data mining, artificial intelligence or other technologies could be considered to be applied to the computer forensics. With the help of data mining and intelligent analysis, the computer forensic work could be more efficient and accurate.

References

[1] CNNIC. (2016). 37th China Internet Development Survey Report, Jan 2016.

[2] Wang, Ling., Qian, Hualin. (2003). Computer Forensics and Its Future Trend. *Journal of Software.* 14 (9).

[3] Gary Palme. (2001). A Road Map for Digital Forensic Research. Tcchntcal Report. DTRT0010-01ÿDFRWSÿNovember 2001. 15-20

[4] Luoma, V. M. (2006). Computer forensics and electronic discovery: The new management challenge. Computers & Security, 25. p.91-96.

[5] Reis, M.A., Geus, P. L. (2002). Standardization of computer forensic procedures and protocols. *In:* Proceedings of the FIRST2002

14th Annual Computer Security Incident Handling Conference. Waikoloa, USA, June, 2002

[6] Wundram, M., Freiling, FC., Moch, C. (2013). Anti-forensics: the next step in digital forensics tool testing. *In*: 2013 Seventh International Conference on IT Security Incident Management and IT forensics, p. 83e97. IEEE.

[7] Thuen, C. (2007).Understanding counter-forensics to ensure a successful investigation.

[8] Stamm MC, Lin WS, Liu K. Forensics vs. anti-forensics: a decision and game theoretic framework. In: Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on. IEEE; 2012. p. 1749e52.

[8] Dahbur, K., Mohammad, B. (2012). Toward understanding the challenges and countermeasures in computer anti-forensics. Cloud Compute Adv Des Implement Technol 2012:176.

[9] Fairbanks, KD., Lee CP., Xia YH. (2007). Owen III HL. Timekeeper: a metadata archiving method for honeypot forensics. In: Information Assurance and Security Workshop, 2007. IAW'07. IEEE SMC. IEEE. p 114e8.

[10] Garfinkel, S. (2007). Anti-forensics: techniques, detection and countermeasures. *In*: 2nd International Conference on i-Warfare and Security. p. 77.

[11] Geiger, M. (2005). Evaluating commercial counter-forensic tools. DFRWS. 2005.

[12] Grugq. (2002). The art of defiling: defeating forensic analysis. Blackhat briefings; 2005.14th Annual Computer Security Incident Handling Conference.Waikoloa, USA, June.

[13] Harris, R. (2006). Arriving at an anti-forensics consensus: examining how to define and control the anti-forensics problem. Digit Investig, 3: 14th Annual Computer Security Incident Handling Conference. Waikoloa, USA, June, 2002

[14] kessler, G.C. (2006). Anti Forensics and the digital investigator. *In*: Australian Digital Forensics Problem. Digit Investig 3; 44e9.

[15] Pajek, P., Pimenidis, E.(2009). Computer anti-forensics methods and their impact on computer forensic investigation. Glob Secur Saf Sustain. 145e55 [Springer].

[16] Peron, CS., Legary, M. Digital anti-forensics: emerging trends in data transformation techniques. 2005.

[17] Rekhis, S., Boudriga, N. A system for formal digital forensic investigation aware of anti-forensic attacks. *Inf Forensics Secur IEEE Trans* 2012;7:635e50.

[18] Rogers, M. (2006). Anti-forensics: the coming wave in digital forensics. Retrieved September, 7.

[19] Shirani, B. (2007). Anti-forensics. High Technology Crime Investigation Association. 2002.

[20] Smith, A. (2007). Describing and categorizing disk-avoiding anti-forensics tools. *J Digit.Forensic Pract*, 1:309e13.

[21] Sremack, J C., Antonov, AV. (2007). Taxonomy of anti-computer forensics threats. IMF 2007:103e12.