ATLANTIS PRESS

Advances in Computer Science Research, (ACSR), volume 52

2016 International Conference on Computer Engineering and Information Systems (CEIS-16)

Network Forensic Analysis via Vulnerability Evidence Reasoning

Cheng-Yue Chang, Jing-Sha He

School of Software Engineering & Beijing Engineering Research Center for IoT Software and Systems

Beijing University of Technology

Beijing 100124, China

E-mail: cychang@emails.bjut.edu.cn, jhe@bjut.edu.cn

***Abstract*-In this article, we propose a novel method that uses vulnerability evidence reasoning in network forensics analysis. Central to our method is the evidence graph model to support evidence presentation and reasoning. Based on the evidence graph, we propose a network forensics method that built the evidence graph on the basis of the network system vulnerabilities and environmental information. At the same time, the proposed method can realize the reconstruction of attack scenarios with high efficiency and with the capability of identifying multi-staged at-tacks through evidence reasoning. Results of the experiment that we conducted would show that the proposed method is complete and credible with certain reasoning ability, which can be a powerful tool for rapid and effective network forensic analysis.**

Keywords-network forensics; evidence graph; event vector; vulnerability evidence reasoning

I. INTRODUCTION

With the rapid development of computer technology and network technology, the Internet is leading the world toward a revolution of information technology. The network has gradually penetrated into the society in various fields, and has begun to dramatically change the way in which people work, study and live. Network technology now connects the whole world, making the human society closely connected, to create wealth for the community so that each individual in the world can use the Internet technology to access and share all sorts of information anytime and anywhere. However, at the same time of facilitating people and creating wealth, network security issues have inadvertently emerged. Due to the imperfectness of law and morality as well as the lack of technology development in the technology and experience, a lot of security issues could be ignored, thus creating network loopholes. As the result, criminals may take advantage of such loopholes as system vulnerabilities to obtain benefits illegally via the Internet. Along with the continuous development of computer networking technologies, offenses that use the network as a tool or medium are increasing over time. To some extent, the development of network technologies with a lag on corresponding forensic means has quickly become a big problem for network security. According to a report on the security of the Internet in China released in 2015 [1], in the third quarter of 2015 alone, 240 million web vulnerabilities were identified for 360 websites, 454 thousand websites were attacked through these web vulnerabilities, a bandwidth of 577.1 Gb/s were consumed by identified DDoS attacks, and 43.84 billion HTTP Flood attacks were intercepted. During the same period, the Sky platform received a total number of 967 "white hat" and 9,355 valid vulnerabilities with 102 effective records everyday on the average. Among them, there were 538structural vulnerabilities and 8,817 event vulnerabilities, and 63.2% vulnerabilities are considered as high-risk ones.

Statistics from the report showed that with the development of network technologies, as the number of Internet users increases, the number of network security incidents increases too. Disclosure of account information, computer worms and attacks resulting from arbitrary execution of malicious code have posed serious threat to the Internet. The main reason is that of the computer system security vulnerabilities so that hackers would exploit such vulnerabilities to realize the attacks, causing losses to the target systems and users.

Cyber delict involves not only technical but also social issues so that governments are considering ways of maintaining the rights of legitimate user's by the law and imposing tough sanctions on cybercriminals. On the technical side, the need for network forensics technologies has become an urgent issue [2]. Just like computer forensics [3], the purpose of network forensics is to obtain or identify potential and legal electronic evidence. However, the interconnection of computers makes network forensics more complex and different than host-oriented forensics. Network forensics needs to extract and analyze dynamic data in the network to provide valid electronic evidence about the attacker's orientation and to reconstruct attack scenarios [4] to be for court ruling.

In this article, we present a novel method that applies vulnerability evidence reasoning to network forensics analysis. Central to our method is an evidence graph model to facilitate evidence presentation and reasoning. Based on the evidence graph, we propose a network forensics method on the basis of the network system vulnerabilities and environmental information. The proposed method can realize attack scenario reconstruction with high efficiency using the ability of identifying multi-staged attacks through evidence reasoning. We will perform some experiment to compare our method to Wang's network forensics analysis method [5] by using MIT Lincoln Laboratory LLDOS 1.0 and LLDOS 2.0.2 datasets to demonstrate the validity and the accuracy of our method. The experiment would show that the evidence graph as well as the method is effective with certain

Copyright © 2016, the Authors. Published by Atlantis Press.

This is an open access article under the CC BY-NC license (http://creativecommons.org/licenses/by-nc/4.0/).

reasoning ability, which can be used as a powerful tool for rapid and effective network forensics analysis.

II. RELATED WORK

The development of computer forensics technologies has accompanied the ever increasing incidents of computer crimes. In the United States, Computer-derived evidence has been accepted by the court since 1969 [6]. Network forensics was first proposed by Ranum in the 1990s [7]. In 2001, a workshop on digital forensic research was held in New York in which a formal definition was given [8], i.e., the use of scientifically proven techniques to collect, fuse, identify, examine, correlate, analyze, and document digital evidence from multiple, actively processing and transmitting digital sources for the purpose of uncovering facts related to the planned intent, or measured success of unauthorized activities meant to disrupt, corrupt, and/or compromise system components as well as providing information to assist in response to or recovery from these activities.

Kaushik, through scanning ports, found that attackers can create potential loopholes [9] and then use these vulnerabilities to attack hosts. After these vulnerabilities are discovered, through monitoring the ports, attackers can be captured. However, this method consumes too much energy because the forensic system has to execute in a listening state.

In 2001, Amman proposed a method based on the "map" of network vulnerability analysis [10], opening the door for showing evidence with evidence graph. For vulnerability assessment based on graph theory, Dinah et al. proposed a novel vulnerability assessment framework [11] through the measure of connections to transform network vulnerability assessment into an optimization problem in graph theory so as to ensure the safety of network processing and to keep the effort within a certain level by reducing the number of edges and vertices in the graph. By connecting nodes in the network, optimization can be performed by removing the most vulnerable nodes to reduce the risk of the network [12], [13].

Liu suggested to reconstruct scenarios by relationships between attacks [14]. Wang introduced the notion of Target-Oriented Effective Event Sequence (TOEES) to semantically reconstruct stealthy attack scenarios, which less dependent on ad-hoc expert knowledge [15], [16]. Rasmi et al. proposed a method that uses attack graphs for network intrusion detection to identify the intention of attackers [17], [17], which can provide help for the detection of network intrusion. Network forensics technology combines digital forensics with network intrusion detection attack graph to analyze the attacker's route to reconstruct evidence graphs. Tian proposed a real-time network intrusion forensics method called NetForensic and introduced vulnerabilities into the network intrusion forensics field. Thus, we use evidence graph to show network evidence in this paper, which was introduced nearly a decade ago and has gradually been improved over the years.

Existing network forensics has the following shortcomings:

- Evidence from a single source may lead to incomplete or invalid evidence.
- The lack of data preprocessing would result in low efficiency in the case of a huge amount of data.
- Thus, the network forensics method we propose in this paper will have the following advantages:
- Our method relies on multiple data sources to collect evidence data, which would help to produce more evidences.
- Our method combines network vulnerability and link analysis to make analysis results more authoritative.

III. THE PROPOSED METHOD

A. *Heterogeneous Data Sources*

The evidence in network forensics should have multiple sources from different channels. In our study, the sources of cyber-crime should include, but not limited to, the following:

- Network packets;
- Operating system log files;
- Anti-virus software log files;
- IIS server log files, FTP server log files, etc.;
- Firewall log files.

Heterogeneous data are collected from the above sources, among which network packets are the primary resource. Although these heterogeneous data are different in structure, they still have some fields in common, such as the source and the target of the visit, the operations and the timestamp of the visit event. So, we can extract the common fields from these data, convert them into a unified format and present the result as an event vector.

B. *Event Vector*

Event vector (EV) is defined as : EV = < ID, Source, Target, Operation, Timestamp, Source>, which can be expressed as:

$$\text{Source} \xrightarrow[\text{Timestamp}]{\text{Operation}} \text{Target}$$

Source represents the main enforcer, Target represents the object that is the bearer of the event, Operation is an action that is executed by the Source and would influence the Target, Timestamp indicates the time of the Operation. These collected heterogeneous data are stored in the form of the event vector, which could help to build the evidence graph efficiently.

Based on the definition of EV, network packets can be defined as <ID, IP1, IP2, Visit, Timestamp, Net-packet>, which can be expressed as:

$$\text{IP1} \xrightarrow[\text{Timestamp}]{\text{Visit}} \text{IP2}$$

Event vectors are used to unify data format and build evidence graph, which will facilitate the analysis of evidence graphs to provide evidence to investigation officers.

C. Evidence Graph

The method of building evidence graph is mainly about relying on the time series to link the event vectors. Evidence graph gives the most intuitive representation between the evidences. Therefore, how to efficiently generate a complete evidence graph is the most important step in our method.

An evidence graph (EG) is a directed graph which has two tuples EG = <V, E> in which V is the set of vertices that should include all the objects in the event vectors and E is the set of edges E = {<S, T, Timestamp | S, T∈V && Path (S, T)>} that describe the relationships between the vertices of a finite set. Path (S, T) indicates a one-way path from vertex S to vertex T.

Based on the definition of the evidence graph, we propose the algorithmic procedure in Figure 1 for building an evidence graph:

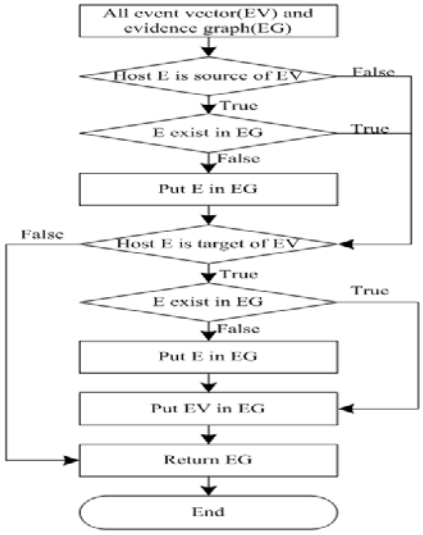

Figure 1. The algorithm for building an evidence graph.

D. Vulnerability Evidence Reasoning

Our method applies evidence reasoning to network forensics based on evidence graph. Our algorithm would conduct evidence reasoning based on the evidence graph that has been constructed in which it takes into consideration of the vulnerabilities that exist in the nodes and in the connections between nodes. The analysis results will show the details of network forensics evidence as well as the attack scenarios. This algorithm is able to identify the key nodes in the evidence graph and locate key steps of attacks.

Through analyzing a number of attacks in the past that exploit network vulnerabilities, we find adversaries to attack host or network usually use multiple hosts and vulnerabilities to enter the network or the host. Because of this, the relevance of vulnerabilities have become increasingly important in network security.

National Vulnerability Database (NVD) has widely used in various areas. Based on the regular rules of vulnerabilities, we can establish the relationships between vulnerabilities in NVD. Common Vulnerabilities & Exposures (CVE) is just like a data dictionary and can unify the names of information security vulnerabilities or weaknesses. Common Vulnerability Scoring System (CVSS) is a free and open industry standard which can be used to assess and mark security vulnerabilities.

The process of scanning vulnerabilities includes four steps: (1) scan the target network; (2) determine the existence of vulnerabilities through comparing with NVD records; (3) query CVE to determine the names of vulnerabilities; (4) calculate vulnerability scores through the CVSS vulnerability scoring system. Through this process, we can obtain the vulnerability scores for the unique identification codes of network hosts and store them in a network vulnerability database. Then, the database can be used to provide information about vulnerabilities of each host in the network.

VERA (Vulnerability Evidence Reasoning Algorithm) that we propose in this paper works on top of the evidence graph and the network vulnerability database. Using this algorithm, we can infer attacked nodes as well as attacking routes. In VERA, each network node has two important values: the degree of importance of the node (called Noteworth) and that of the links (Linkworth). Nodeworth refers to the value of vulnerabilities that a host has while Linkworth is the sum of the Nodeworth of all the nodes that have out-going links to the node. If a network host is vulnerable, it is easy to be attacked. The initial value of Nodeworth is the average vulnerabilities value of a node.

Following is the steps of the VERA algorithm to be applied to network forensics based on evidence graph:

Step 1: Construct the adjacency matrix H for the evidence graph assuming the set of node N= {n_1, n_2,……,n_n}. For any node n_i, $n_i \in N$, y_i is the Nodeworth of the node, z_i is the Linkworth of the node, A_i is the set of nodes that have out-going links to the node, and B_i is the set of nodes to which this node has out-going links.

Step 2: Initialize Nodeworth vector y and Linkworth vector z.

$$y^0 = (Vul_1, Vul_2, \ldots, Vul_n)^T \quad (1)$$

$$z^0 = (1/\sqrt{n}, 1/\sqrt{n}, \ldots, 1/\sqrt{n})^T \quad (2)$$

Then, (Vul_1, Vul_2, …, Vul_n) is the average vulnerability score for a host node.

Step 3: At the kth iteration, the Nodeworth of node n_i is y_i^k which is equal to the sum of z_i, i.e., $y=H^TZ$.

Step 4: After a new Nodeworth vector y is obtained in the previous step, the Linkworth of the host is the sum of y_i. i.e., Z=Hy.

Step 5: Unitize the vectors y and z through the third and the fourth steps. The sum of the vectors is equal to 1.Then, execute the two steps until they converge.

Y contains the Nodeworth values of all the nodes which are sorted in an ascending order. The more vulnerable a node is, the higher the score that the node has, so the node need to draw more attention. Z contains the Linkworth values of all the nodes in descending order. If a link connects important nodes, it is easy to be attacked. According to the definition of Nodeworth and Linkworth, we can know that the highest Nodeworth represents the attacker and the highest Linkworth represents the victim. Based on network vulnerability, we take the attacker and the victim into the evidence graph and then rebuild the attack scene.

IV. EXPERIMENTS AND ANALYSIS

A. *Experiments*

Compared our method with the method proposed by Wang, using the same dataset LLDOS 1.0 with the size of 179MB, which from the MIT Lincoln Laboratory. LLDOS uses 14 hosts simulated external Internet environment to visit the services provider, 39 hosts within the network and 6 hosts consisting of DMZ zone, those covered Windows, Linux Red Hat 5.0, SunOS 4.1.4 and Solaris2.7 etc., including a variety of general-purpose operating system. This dataset has a strong representation.

Figure 2 shows the evidence graph of the initial construction of the event vector. As can be seen from the figure, it is convenient to find all of the nodes and edges for the evidences.

Figure 2. LLDOS 1.0 dataset original evidence graph.

Figure 3 shows the final evidence graph by running the VERA algorithm, which gives a clear and intuitive evidence of the attacker. First, the attacker determines the range of the IP addresses of the destination hosts and use host "202.77.162.213" to scan the entire range of the IP addresses,. Second, the attacker executes "Ping" with the Sadmind option to check whether the Sadmind service is available from which the attacker selects host "172.16.115.20" as source of attack. Third, buffer overflow attack is launched through Sadmind vulnerability in Solaris operation system. Fourth, the attacker acquires the root authority via Telnet and RPC to install the Mstream backdoor. Finally, the attacker uses "172.16.115.20" to manage "131.84.1.131" to launch a DDoS attack. This attack scenario is exactly the same as the attack detected by MIT Lincoln Laboratory.

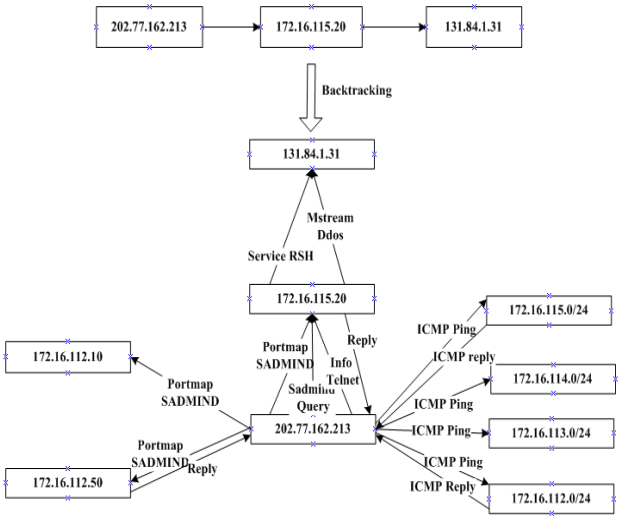

Figure 3. LLDOS 1.0 dataset evidence graph (Omission DDoS attack).

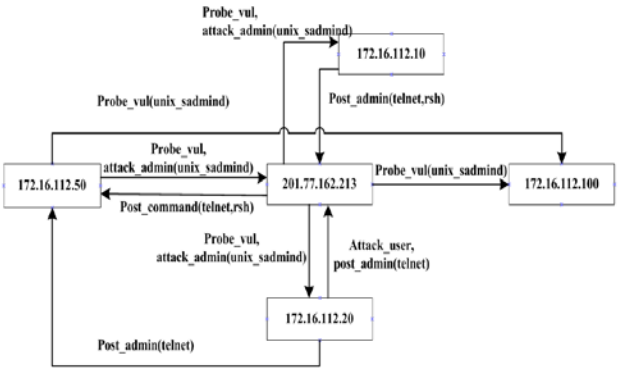

Figure 4. Wang's Simple Evidence Graph (Omission DDoS Attack).

B. *Analysis of results*

Figure 4 shows the evidence graph by Wang [[5]]. Comparison of Figure 3 and Figure 4 clearly indicates that our method can produce a more complete evidence graph. In addition, the evidence graph can show the process of attack in more detail. Although Figure 4 identifies "172.16.112.100" as a victim, it is actually not a victim in the official statement, indicating the deficiency of Wang's network forensics method.

V. CONCLUSION

In this paper, we presented a novel network vulnerability evidence reasoning method. Central to the method is the evidence graph model that facilitates evidence presentation and reasoning. The method can be used to conduct network

forensics in which the evidence graph is built based on network system vulnerabilities and environmental information. At the same time, our method can realize reconstruction of the attack scenario with high efficiency and has the capability of analyzing multi-staged attacks through evidence reasoning. Experimental results show that the evidence graph constructed in our model is more complete and credible and thus has certain reasoning capability. Our future research will focus on determining the distribution of network vulnerabilities and on relating the network hosts.

ACKNOWLEDGMENT

The work presented in this paper has been supported by National High-Tech R&D Program (863 Program) (2015AA017204).

REFERENCES

[1] 360 Internet Security Center, In the Third Quarter of 2015 The Chinese Internet Security Report. 2015.

[2] A. R. Arasteh, M. Debbabi and A. Sakha, "Analyzing Multiple Logs for Forensic Evidence," Digital Investigation, vol. 4, June. 2007, pp. 82-91.

[3] L. Hu, W. B. Wang, and K. Zhao, "Computer Forensics Review," Journal of Jilin University: Information Science Edition, vol. 28, 2007, pp. 378-384.

[4] Z. H. Tian, X. Z. Yu, and H. L. Zhang, "A Real-time Network Intrusion Forensics Method Based on Evidence Reasoning Network," Chinese Journal of Computers, vol. 5, 2014, pp. 1184-1194.

[5] W. Wang, "A graph oriented approach for network forensic analysis," Ph.D. Dissertation, Iowa State University, 2010.

[6] P. Sommer, "Computer evidence: A forensic investigations handbook," Computer Fraud & Security, vol. 1, 1997, pp. 17-18.

[7] M. J. Ranum, "Network forensics and traffic monitoring," COMPUT SECUR J, vol. 13, 1997, pp. 35-39.

[8] G. Palmer, "A road map for digital forensic research," First Digital Forensic Research Workshop, 2001, pp. 27-30.

[9] A. K. Kaushik, E. S. Pilli, and R. C. Joshi, "Network Forensics System for Port Scanning Attack," Advance Computing Conference (IACC), 2010 IEEE 2nd International, 2010, pp. 310-315.

[10] P. Ammann, D. Wijesekera, and S. Kaushik, "Scalable, graph-based network vulnerability analysis," Proceedings of the 9th ACM Conference on Computer and Communications Security, 2001, pp. 217-224.

[11] T. N. Dinh, Y. Xuan, and M. T. Thai, "On New Approaches of Assessing Network Vulnerability: Hardness and Approximation," IEEE/ACM Transaction on Networking, April. 2012, pp. 609-619.

[12] Y. Shen and M. T. Thai, "Network Vulnerability Assessment under cascading failures," 2013 IEEE Global Communications Conference (GLOBECOM), Dec. 2013, pp. l526-1531.

[13] N. P. Nguyen, M. A. Alim, and Y. Shen, "Assessing network vulnerability in a community structure point of view," Advances in Social Networks Analysis and Mining (ASONAM), 2013 IEEE/ACM International Conference, Aug. 2013, pp. 231-235.

[14] C. Liu, A. Singhal, D. Wijesekera, "Relating Admissibility Standards for Digital Evidence to Attack Scenario Reconstruction," Journal of Digital Forensics Security & Law, vol. 9, June. 2014, pp. 181-196.

[15] W. Wang, and T. E. Daniels, "A Graph Based Approach toward Network Forensics Analysis," ACM Transactions on Information & System Security, vol. 12, october. 2008, pp. 427-438.

[16] W. Wang, and T. E Daniels, "Network Forensics Analysis with Evidence Graphs," Proc. of the 2005 Digital Forensic Research Workshop (DFRWS), August 2005, pp.17-19.

[17] M. Rasmi, and A. Jantan, "AIA: Attack Intention Analysis Algorithm Based on D-S Theory with Causal Technique for Network Forensics - A Case Study," International Journal of Digital Content Technology & its Applications, vol. 5, 2011, pp. 230-237.

[18] M. Rasmi, and A. Jantan, "Attack Intention Analysis Model for Network Forensics," Proc. of 2nd International Conference on Software Engineering and Computer Systems, ICSECS Press, 2011, pp.403-411