Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 5.258

IJCSMC, Vol. 5, Issue. 3, March 2016, pg.114 – 119

An Internal Intrusion Detection and Protection System Using Data Mining and ACO Techniques

[1]D. Dhanavandhini
CSE, IFET College of Engineering, Villupuram
[2]Mrs. S.Umadevi
CSE, Senior Assistant Professor, IFET College of Engineering, Villupuram

ABSTRACT: The most computer systems use user IDs and passwords as the login patterns to authenticate users. However, many people share their login patterns with coworkers and request these coworkers to assist co-tasks, thereby making the pattern as one of the weakest points of computer security. Insider attackers, the valid users of a system who attack the system internally, are hard to detect since most intrusion detection systems and firewalls identify and isolate malicious behaviors launched from the outside world of the system only. In addition, some studies claimed that analyzing system calls (SCs) generated by commands can identify these commands, with which to accurately detect attacks, and attack patterns are the features of an attack. In this paper, presents an intelligent learning approach using Ant Colony Optimization (ACO) based distributed intrusion detection system to detect intrusions in the distributed network. The algorithm improves the efficiency of intrusion detection, reduces false positives of intrusion detection. The results obtained as, the value of rates obtained from and increased efficiency of ACO is increased up to (97% approx.)

I. INTRODUCTION

In the past decades, computer systems have been widely employed to provide users with easier and more convenient lives. However, when people exploit powerful capabilities and processing power of computer systems, security has been one of the serious problems in the computer domain since attackers very usually try to penetrate computer systems and behave maliciously, e.g., stealing critical data of a company, making the systems out of work

or even destroying the systems. Generally, among all well-known attacks such as pharming attack, distributed denial-of-service (DDoS), eavesdropping attack, and spear-phishing attack, insider attack is one of the most difficult ones to be detected because firewalls and intrusion detection systems (IDSs) usually defend against outside attacks. To authenticate users, currently, most systems check user ID and password as a login pattern. However, attackers may install Trojans to pilfer victims' login patterns or issue a large scale of trials with the assistance of a dictionary to acquire users passwords. When successful, they may then log in to the system, access users' private files, or modify or destroy system settings. Fortunately, most current host-based security systems and network-based IDSs can discover a known intrusion in a real-time manner. However, it is very difficult to identify who the attacker is because attack packets are often issued with forged IPs or attackers may enter a system with valid login patterns. Although OS-level system calls (SCs) are much more helpful in detecting attackers and identifying users, processing a large volume of SCs, mining malicious behaviors from them, and identifying possible attackers for an intrusion are still engineering challenges.

Therefore, in this paper, we propose a security system, named Internal Intrusion Detection and Protection System (IIDPS), which detects malicious behaviors launched toward a system at SC level. The IIDPS uses data mining and forensic profiling techniques to mine system call patterns (SC-patterns) defined as the longest system call sequence (SC-sequence) that has repeatedly appeared several times in a user's log file for the user. The user's forensic features, defined as an SC-pattern frequently appearing in a user's submitted SC-sequences but rarely being used by other users, are retrieved from the user's computer usage history.

The contributions of this paper are: 1) identify a user's forensic features by analyzing the corresponding SCs to enhance the accuracy of attack detection; 2) able to port the IIDPS to a parallel system to further shorten its detection response time; and 3) effectively resist insider attack.

II. RELATED WORK

Supervisory Control and Data Acquisition (SCADA) systems [1], which are widely used in monitoring and controlling critical infrastructure sectors, are highly vulnerable to cyber attacks. Current security solutions can protect SCADA systems from known cyber assaults, but most solutions require human intervention. This paper applies autonomic computing technology to monitor SCADA system performance, and proactively estimate upcoming attacks for a given system model of a physical infrastructure. We also present the feasibility of intrusion detection systems for known and unknown attack detection. A dynamic intrusion response
system is designed to evaluate recommended responses, and appropriate responses are executed to influence attack impacts. To identify zero-day attacks (attacks that exploit previously unknown vulnerabilities) the signature database must be upgraded frequently. Disadvantage of this paper is Responses are not sufficient to mitigate attack impacts if the intrusion detection raises a false alarm.

A common application of virtual machines (VM) [2] is to use and then throw away, basically treating a VM like a completely isolated and disposable entity. The disadvantage of this approach is that if there is no malicious activity, the user has to re-do all of the work in her actual workspace since there is no easy way to commit (i.e., merge) only the benign updates within the VM back to the host environment. In this work, we develop a VM

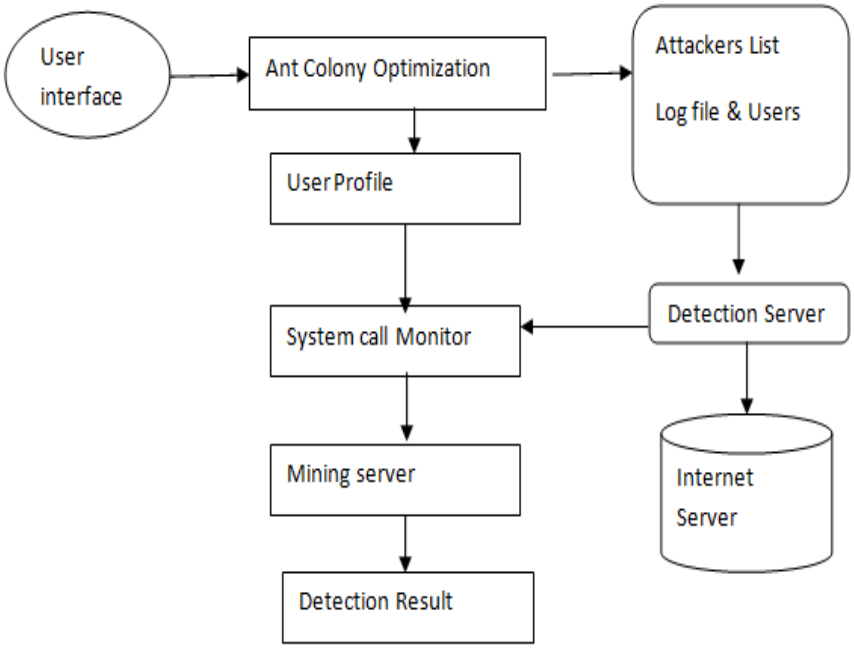
commitment system called Secom to automatically eliminate malicious state changes when merging the contents of an OS-level VM to the host. Secom consists of three steps: grouping state changes into clusters, distinguishing between benign and malicious clusters, and committing benign clusters. Secom has three novel features. First, instead of relying on a huge volume of log data, it leverages OS-level information flow and malware behavior information to recognize malicious changes. Advantage of the Secom prototype has a smaller number of false negatives and thus can more thoroughly clean up malware side effects. In addition, the number of false positives of the Secom prototype is also lower than that achieved by the on-line behavior-based approach of the commercial tools. Disadvantage is no secure commitment mechanism to save the benign changes within the VM back to the host environment.

A distributed denial of service attacks [3] are the most serious factor among network security risks in cloud computing environment. This study proposes a method of integration between HTTP GET flooding among DDOS attacks and Map Reduce processing for a fast attack detection in cloud computing environment. This method is possible to ensure the availability of the target system for accurate and reliable detection based on HTTP GET flooding. Advantage is the processing time for performance evaluation compares a pattern detection of attack features with the Snort detection. The proposed method is better than Snort detection method in experiment results because processing time of proposed method is shorter with increasing congestion. But attacks are difficult to distinguish between normal traffic and DDoS.

Peer-to-peer streaming [4] has witnessed a great success thanks to the possibility of aggregating resources from all participants. Nevertheless, performance of the entire system may be highly degraded due to the presence of malicious peers that share bogus data on purpose. In this paper we propose to use a statistical inference technique, namely Belief Propagation, to estimate the probability of peers being malicious. The detection algorithm is run by a set of trusted monitor nodes that receives notification messages (checks) from peers whenever they obtain a chunk of data; these checks contain the list of the chunk up loaders and a flag to mark the chunk as polluted or clean. Peers are able to detect if the received chunk is polluted or not but, since multi-party download is employed, they are not capable to identify the source(s) of bogus blocks.

Advantages are the accuracy, robustness, and complexity of our technique by running a real peer-to-peer application on Planet Lab. The proposed approach is very accurate and robust against malicious nodes misbehaving (different pollution intensity, presence of fake checks, churning, and total un-cooperation from malicious nodes), increasing number and colluding behavior of malicious nodes. Disadvantage is they can lie when sending checks to the monitor node. They can churn by alternating between connection and disconnection periods.

III. PROPOSED SYSTEM



The data set contains different types of intrusions present in networking environment. Mainly TCP/IP data combined with several attacks. It analyzes what attackers have done such as spreading computer viruses, malwares, and malicious codes and conducting DDoS attacks. The SC monitor and filter, as a loadable module embedded in the kernel of the system being considered, collects those SCs submitted to the kernel and stores these SCs in the format of _uid, pid, SC_ in the protected system where uid, pid, and SC respectively represent the user ID, the process ID, and the SC c submitted by the underlying user, i.e., c ∈ SCs. It also stores the user inputs in the user's log file, which is a file keeping the SCs submitted by the user following their submitted sequence. To find out what SCs are typical ones generated by a shell command, the statistic model of term frequency-inverse document frequency (TF-IDF) is used to analyze the importance of intercepted SCs collected in a user log file. The mining server analyzes the log data with data mining techniques to identify the user's computer usage habits as his/her behavior patterns, which are then recorded in the user's user profile. The two process are 1.Mining User and Attacker Habits: 2.Creating User Profiles and Attacker Profiles. An attack pattern (or a signature), which may be an attacker-specific pattern or a pattern commonly used by attackers, can be identified in the same method. The detection server compares users' behavior patterns with those SC-patterns collected in the attacker profile, called attack patterns, and those in user profiles to respectively detect malicious behaviors and identify who the attacker is in real time. When an intrusion is discovered, the detection server notifies the SC monitor and filter to isolate the user from the protected system.

The ACO identifies who the underlying user is by computing the similarity scores between the user's current inputs, i.e., SCs, and the behavior patterns stored in different users' user profiles.

IV. SYSTEM IMPLEMENTATION

Intrusion means someone penetrate the security of the system without permission. Intrusion Detection System (IDS) can detect the illegal activities performed by the Intruders and can report to the higher authorities. IDS is a set of methods and techniques to detect the illegal activities in System level and Network level. IDS can be broadly classified into two, Host Based Intrusion Detection Systems and Network Based Intrusion Detection Systems.

V. CONCULSION AND FUTURE WORK

In Intrusion Detection Data Mining refers to the process of extracting hidden, previously unknown and useful information from large databases. It is a convenient way of extracting patterns and focuses on issues relating to their feasibility, utility, efficiency and scalability. Thus data mining techniques help to detect patterns in the data set and use these patterns to detect future intrusions in similar data.

REFERENCES

[1] S. Gajek, A. Sadeghi, C. Stuble, and M. Winandy, "Compartmented security for browsers—Or how to thwart a phisher with trusted computing," in Proc. IEEE Int. Conf. Avail., Rel. Security, Vienna, Austria, Apr. 2007,pp. 120–127.

[2] C. Yue and H. Wang, "BogusBiter: A transparent protection against phishing attacks," ACM Trans. Int. Technol., vol. 10, no. 2, pp. 1–31, May 2010.

[3] Q. Chen, S. Abdelwahed, and A. Erradi, "A model-based approach to self-protection in computing system," in Proc. ACM Cloud Autonomic Comput. Conf., Miami, FL, USA, 2013, pp. 1–10.

[4] F. Y. Leu, M. C. Li, J. C. Lin, and C. T. Yang, "Detection workload in a dynamic grid-based intrusion detection environment," J. Parallel Distrib. Comput., vol. 68, no. 4, pp. 427–442, Apr. 2008.

[5] H. Lu, B. Zhao, X. Wang, and J. Su, "DiffSig: Resource differentiation based malware behavioral concise signature generation," Inf. Commun. Technol., vol. 7804, pp. 271–284, 2013.

[6] Z. Shan, X. Wang, T. Chiueh, and X. Meng, "Safe side effects commitment for OS-level virtualization," in Proc. ACM Int. Conf. Autonomic Comput., Karlsruhe, Germany, 2011, pp. 111–120.

[7] M. K. Rogers and K. Seigfried, "The future of computer forensics: A needs analysis survey," Comput. Security, vol. 23, no. 1, pp.12–16, Feb. 2004.

[8] J. Choi, C. Choi, B. Ko, D. Choi, and P. Kim, "Detecting web based DDoS attack using MapReduce operations in cloud computing environment," J. Internet Serv. Inf. Security, vol. 3, no. 3/4, pp. 28–37, Nov. 2013.

[9] Q. Wang, L. Vu, K. Nahrstedt, and H. Khurana, "MIS: Malicious nodes identification scheme in network-coding-based peer-to-peer streaming," in Proc. IEEE INFOCOM, San Diego, CA, USA, 2010, pp. 1–5.

[10] Z. A. Baig, "Pattern recognition for detecting distributed node exhaustion attacks in wireless sensor networks," Comput. Commun., vol. 34, no. 3, pp. 468–484, Mar. 2011.

[11] H. S. Kang and S. R. Kim, "A new logging-based IP traceback approach using data mining techniques," J. Internet Serv. Inf. Security, vol. 3, no. 3/4, pp. 72–80, Nov. 2013.

[12] K. A. Garcia, R. Monroy, L. A. Trejo, and C. Mex-Perera, "Analyzing log files for postmortem intrusion detection," IEEE Trans. Syst., Man, Cybern., Part C: Appl. Rev., vol. 42, no. 6, pp. 1690–1704, Nov. 2012.

[13] M. A. Qadeer, M. Zahid, A. Iqbal, and M. R. Siddiqui, "Network traffic analysis and intrusion detection using packet sniffer," in Proc. Int. Conf. Commun. Softw. Netw., Singapore, 2010, pp. 313–317.

[14] S. O'Shaughnessy and G. Gray, "Development and evaluation of a dataset generator tool for generating synthetic log files containing computer attack signatures," Int. J. Ambient Comput. Intell., vol. 3, no. 2, pp. 64–76, Apr. 2011.

