ISSN 2395-1621

# Survey on Behaviour Analysis of Client Side Cyber Attack

Miss. Bhavar Priyanka B., Miss. Shaikh Aliya M., Miss. Lende Priyanka D., Miss. Ghule Tanupriya B., Prof. S.C.Deshmukh

priyankabhavar123@gmail.com
shaikhaliya660@gmail.com
lendepriyanka13@gmail.com
ghuletanupriya@gmail.com
sandesh.deshmukh@avcoe.org

Department Of Information Technology,
Amrutvahini College of Engineering, Sangamner.

## ABSTRACT

**Day by day, more and more people are using internet all over the world. It is becoming a part of everyone's life. People are checking their e-mails, surfing over internet, purchasing goods, playing online games, paying bills on the internet etc. However, while performing all these things, how many people know about security? Do they know the risk of being attacked, infecting by malicious software? Even some of the malicious software are spreading over network to create more threats by users. The behavior analysis of client side cyber-attack play a significant role in computer security. In network surroundings find the activities that have an effect on Confidentiality, Integrity and accessibility on network knowledge. Currently, most computer systems use user IDs and passwords because the login patterns to verify users. However, many of us share their login patterns with co-workers and request these co-workers to help co-tasks, thereby creating the pattern united of the weakest points of computer security. Insider attackers, the valid users of a system who attack the system internally, are hard to find since most intrusion detection systems and firewalls establish and isolate malicious behaviors launched from the external world of the system solely. In file integrity concept if any user deletes the file, modify file or insert file into specific directory then by using our system we can detect it. If any file delete, modify or insert into specific folder then that file backup will save in folder which is specified by client. Then file integrity log send to server. Server sends the integrity of that file to the clients email id. So client will easily know which file is modified, client can recover that modified file from specified backup folder. The aim of the system is analysing, understanding, watching and tracking hacker's behaviors in order to create more secure systems.**

**KEYWORDS: Honeypot, hacking, security, IIDPS**

ARTICLE INFO

**Article History**

Received: 22nd June 2020

Received in revised form : 22nd June 2020

Accepted: 24th June 2020

**Published online : 24th June 2020**

## I. INTRODUCTION

In this system we work on network intrusion detection system and to guard the network with the advent of snooping agents and honeypot on the network in order that any intrusion took place in network may be detected and as a result may be prevented. The concept behind the use of snooping retailers and honeypot is to provide network control in term of tracking. Usually in wireless networks, attacks are main cause of malfunctioning and are difficult to monitor. In file integrity concept if any user delete the file, modify file or insert file into specific directory then by using our system we can detect it. If any file delete or modify of insert into specific folder then that file will save in folder which is specified by client. Then file integrity log send to server. Server sends the integrity of that file to the clients email id. So client will easily know which file is modified, client can recover that modified file from specified backup folder. In proposed system detecting the intrusion through many thing like integrity, checking currently running processes, by key log, etc. These all activities are performed by user. The first activity is file integrity. We are detecting intrusion through file integrity. If unauthorized person try to insert pen drive or access file then system send message to user and shut down as the intrusion is detected.

## II. RELATED WORK

[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.

Towards the design and implementation of a security architecture that prevents both classical and malware phishing attacks, this paper aims at making the first steps. For isolating applications of different trust level, and a trusted wallet for storing credentials and authenticating sensitive services, our approach is based on the ideas of compartmentalization. The solution requires no special care from users once the wallet has been setup in an initial step, for identifying the right Web sites while the disclosure of credentials is strictly controlled. Moreover, a prototype of the basic platform exists and briefly describe its implementation.

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

Many anti-phishing mechanisms currently focus on helping users verify whether a Web site is genuine. However, usability studies have demonstrated that prevention-based approaches alone fail to effectively suppress phishing attacks and protect Internet users from revealing their credentials to phishing sites. In this paper, instead of preventing human users from ``biting the bait,” a new approach to protect against phishing attacks with ``bogus bites.” We develop Bogus Biter, a unique client-side anti-phishing tool, which transparently feeds a relatively large number of bogus credentials into a suspected phishing site. Bogus Biter conceals a victim's real credential among bogus credentials, and moreover, it enables a legitimate Web site to identify stolen credentials in a timely manner. Leveraging the power of client-side automatic phishing detection techniques, BogusBiter is complementary to existing preventive anti-phishing approaches. The implemented BogusBiter as an extension to the Firefox 2 Web browser, and evaluated its efficacy through real experiments on both phishing and legitimate Web sites. The experimental results indicate that it is promising to use BogusBiter to transparently protect against phishing attacks.

[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.

Approach to estimate, detect and identify security attacks along with planning a sequence of actions to effectively protect the networked computing system this paper introduces a model-based autonomic security management (ASM). Sensors collect system and network parameters and send the data to the forecasters and the intrusion detection systems (IDSes) In the proposed approach. to recover the system based on the signature of attacks A multi-objective controller selects the optimal protection method. On several case studies including Denial of Service (DoS) attacks, SQL Injection attacks and memory exhaustion attacks, the proposed approach is demonstrated. Experiments show that from known and unknown attacks the ASM approach can successfully defend and recover the victim host while maintaining QoS with low overheads.

[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.

In this paper, to detect a massive amount of intrusion packets and to manage a dynamic environment a Grid-based platform, named the dynamic grid-based intrusion detection environment (DGIDE), which exploits Grid's abundant computing resources. A detector, a node that detects attacks, can dynamically join or leave the DGIDE. The system can obtain its key performance curves because A newly joined detector is tested, which are used to balance detection workload among detectors. The DGIDE backs up network packets. The DGIDE allocates another available detector to take over when, for some reason, a detector cannot continue its detection thus leaving an unfinished detection task. Therefore, the drawbacks of ordinary security systems as mentioned above can be avoided.

[5] 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.

In this work, to automatically eliminate malicious state changes when merging the contents of an OS-level VM to the host, develop a VM commitment system called Secom. 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, to recognize malicious changes it leverages OS-level information flow and malware behavior information. As a result, a smaller performance overhead the approach imposes. Second, Secom classifies objects into clusters and then identifies malicious objects on a cluster by cluster basis which different from existing intrusion detection and recovery systems that detect compromised OS objects one by one. Third, it simultaneously considers two malware behaviors that are of different types and the origin of the processes that exhibit these behaviors to reduce the false positive rate when identifying malicious clusters, rather than considers a single behavior alone as done by existing malware detection methods. On the Feather-weight Virtual Machine (FVM) system, a Windows-based OS-level virtualization system, Moreover, the Secom prototype has a smaller number of false negatives and thus can more thoroughly clean up malware side effects compared with the commercial anti-malware tools.

[6] Liu Dongxia, ``An Intrusion Detection System Based on Honeypot Technology" 2012 IEEE

This paper presents an intrusion detection module based on honeypot technology, which utilizes IP Traceback technique. By using the mobile agents, this module has the capability of distributed detection and response; the whole detection module can be extended with convenience and be configured dynamically and flexibly. By using honeypot technology, this module traces the intrusion source farthest.

[7] Vivek anand Rajbhar, ``INTRUSION DETECTION & PREVENTION USING HONEYPOT" Volume 9, No. 4, July – August 2018

Computers & information technology (IT) revolutionized the world & growing day by day. Computer networks enable us to communicate with remote computer network and access resources effectively & efficiently. But these networks are not secure it's prone to intrusion, threats and attacks. Now a days industries use Intrusion detection system (IDS) & Intrusion prevention system (IPS) to monitor the system or a network for attacks, intrusion or threats & prevent the system or network from such vulnerabilities. However IDS/IPS is very expensive & complex to be implemented on your IT systems

## III. PROPOSED SYSTEM

In this approach, log file is stored into two different forms as well as in two different places. Log file in plain text from is stored on target host and a copy of same log file is stored in another host called log manager. When an intruder tries to alter log file on target host, IDS running on the target host detects an intrusion and sends an alert message to the security administrator about the intrusion which in turn takes the required steps to mitigate it.

- Target Host:

Crucial data (i.e. log files) is stored in the Target Host. Continuous monitor of log file is prime requirement to preserve the integrity and confidentiality of the data stored in it. To achieve this, IDS is deployed on target host and it is a continuous process round the clock. Whenever an attacker tries to intrude the target host, IDS running on target host detects the intrusion; sends an alert message to security center as well as log server. Further, it invokes the digital forensic tool to capture the state of the system (RAM image and log file image). Newly captured log file image is compared with previous log file image to confirm the intrusion. Our Target Host is nothing but our Operating System as it is a Host based System. The intruder shall be able to access the system but if he tries to alter any of the system properties and manipulate the records then the IDS comes into picture.

- Server:

It stores the copy of the log file in an encrypted form. Encryption key maintained only by the log server and it kept secret. Periodically back up of the Target host log file is taken and it is stored on the log server. Upon receiving the log file as a backup, it encrypts the received log file and stores within it. Whenever the log server receives an alert message from target host, it decrypts the log file, computes the image of the decrypted log file using digital forensic tool and sends it to the target host to perform the comparison. The main job of the Log server is encryption and decryption of log files such that the intruder doesn't have access to them. If the intruder gets to know the location and condition of the log files then their safety comes under scrutiny. The most important part will be the key. The key that is used to encrypt and decry-pt the log files shall only be available with the owner and nobody else. It shall be provided at the time of delivering the software as a complete product.

- Security Center(Admin):

This is the system used by the security administrator to monitor the alerts generated by IDS. It receives alerts from Target Host. Once the target host has sent the alert to the Security center, the job of the Security center starts. The attack is hence detected and looked into at the Security center. The Security center is the most essential component of the IDS. Its job is to track the intrusion in such a way that as soon as he/she tries to access the system, an alert should be sent to the real owner. This shall be accompanied by the webcam image capturing activity in order to prove the offense in the court of law. If the intruder tries to access the files without the net connection, the system shall shut down by itself within 10 seconds, and if he has the net connection intact, then we shall also be able to inform the true owner about the intrusion with the help of an e-mail.

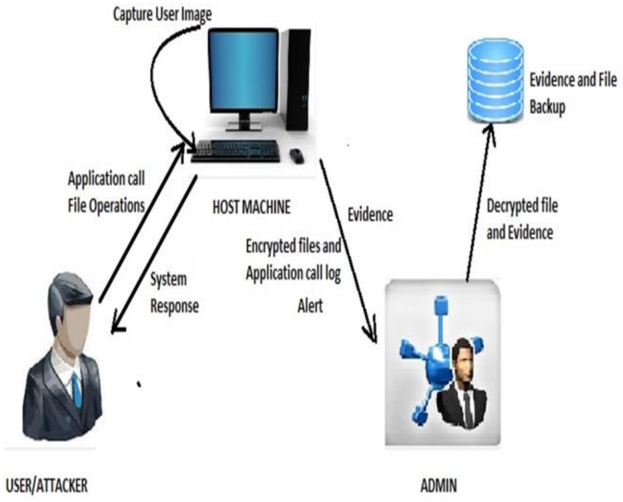

FIGURE 1: SYSTEM ARCHITECTURE

## IV. CONCLUSION AND FUTURE WORK

The time that a habitual SC-pattern appears in the user's log file is counted, the most commonly used SC-patterns are filtered out, and then a user's profile is established. By identifying a user's SC-patterns as his/her computer usage habits from the user's current input SCs, the IDS resists suspected attackers. The experimental results demonstrate that the average detection accuracy is higher than 94%. when the decisive rate threshold is 0.9, indicating that the IDS can assist system administrators to point out an insider or an attacker in a closed environment. The further study will be done by improving IDS's performance and investigating third-party shell commands.

## 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] Megha Mandlik, Trupti Akolkar ``Host Based Internal Intrusion Detection and Protection Techniques" Vol. 4, Issue 9, September 2016

[10] Jayesh Surana, Jagrati Sharma ``A Survey On Intrusion Detection System" 2017 IJEDR, Volume 5, Issue 2 ISSN: 2321-9939