Volume 3, Issue 11, November 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Analysis on Different Algorithms of Dynamic Honeypots

Tejashree K. Shinde, D.A.Nikam

Assistant Professor, Department of Computer Science & Engineering,
Dr.J.J.Magdum College of Engineering, Jaysingpur
Shivaji University, Kolhapur ,India

Abstract— There are a lot of critical issues in the network security concern. One of these issues is deployment of dynamic honeypot. To overcome this matter, plug and play honeypot methods are being promoted by some researchers. Therefore, the main aim of this paper is to study on the usability and security features of the plug and play concept in order to have easy deployment of dynamic honeypot on a network environment. It also offers type of integration of existing network tools to deploy dynamic honeypot based on its placement within network topology.

Keywords —Dynamic honeypots, honeynet, plug and play honeypot, SIP, VoIP

I. INTRODUCTION

Today's it is obvious that the number of vulnerable threat of network assets increase significantly. The major examples of such threats are worms which can be found in the Internet. In spite of decades of research and experience, still incapable to make secure computer systems and also they immobilized manufacture ways to measure a computer systems and networks level of security. This is certainly true that intrusion to a network or system can never be eliminated but security helps reduce risk to an organization and its information-related resources. However, three particular concepts surrounded the IT security includes Detection, Prevention and Reaction. In one hand, you can't find end-to-end equipments solution to cover all or even two of concepts. In the other hand, secure digital resources in an organization by ruling detect and prevent vulnerabilities before exploited. Looking at network intrusion detection systems (NIDS) before starting with honeypots, could be useful to motivate the use of honeypots. NIDS by providing the amount of functional information causes an increasing number of protocols that employ encryption to protect network traffic from eavesdroppers and decreasing in the face of ever more complicated evasion techniques. Effectiveness of NIDSs is weak because of high false positive rates. Using honeypots assists to decrease the number of the false positives. In network security an Intrusion Prevention System (IPS) is the method of that monitors network and/or system activities for malicious or unwilling behavior. Moreover, IPS is accountable attempting to stop discovered possible occurrences in real time.

In the other hand intrusion prevention systems are the devices of performing intrusion detection .A novel strategy in the modernized defensive network security is Deception. In fact, this issue is relatively in intrusion detection field. The idea which follows these systems is to provide systems or services that deceive the intruder. Therefore, these systems help in learning the techniques that invaders use and they can also be viewed as a decoy to amuse hackers from the real systems and services. Thus, Honeypots can be categorized as deception systems. The purpose of honeypot in security community refers to a computer system that is used as a trap to catch the attention of attackers to attack this computer system. If so, gathering useful information from attacker, firstly, allows us to present controlled information to our adversaries. Secondly, a honeypot allows us to collect information concerning our attackers. Finally, a honeypot can serves as a disincentive against future attacks .This information could be pattern of attack, unknown security holes, tools and techniques they used, and even the keystrokes of attacker. Similarly, this information can be used to prevent, block, or turn off the attack. In additional, based on this information enforcing and strengthen existing intrusion detection tools or network firewalls will be possible. It is considerable that honeypots should not be viewed as a solution to network security; they should be seen as an aid to it. Regardless to detection and prevention the values of a honeypot is directly comparative to the amount and type of information we can successfully obtain from it. Sideways from information gathering, a honeypot has the capabilities of distracting adversaries from more valuable machines on a network, and can provide early warning signs concerning a new type of attack or exploitation trends, and allows in depth examination of adversaries during or after exploitation of a host.

II. DYNAMIC HONEYPOTS

Honeypots like all network security technology needs configuration. Traditionally whether low, medium or high interaction of honeypots, they normally configure manually. The current static honeypot faces the same requirements. Even if configuration and deploying was not sufficient, we need someone to maintain the honeypot and it can be more challengeable. Due to solve this problem, the idea of the Dynamic Honeypots comes out. Possibly it seems a plug and play solution. We can simply plug it into the network and the honeypot does all the work for us [8]. Dynamics honeypots are all in demands including how many honeypots need to deploy, how to deploy them, and what they should look like

will automatically determines so that they can blend in with the organization environment. Even better, the deployed honeypots can change and adapt to the organization environment. For instance, when we add Linux to the network, we rapidly have Linux honeypots or if we remove a Novell server from organization network, the
Novell honeypots automatically disappear. When we change the router in our network, honeypot routers will change as well. The final goal is an application and a solution where we just simply plug into our network, then it will learn the environment, deploys the honeypots and capture the damages done by the attacker.

III. DYNAMIC HONEYPOTS REVIEWED ALGORITHMS

A. Algorithm 1

The article [9] has shown an independent honeypot capable of adapting in a dynamic and constantly changing network environment .In fact, they response to a proposed idea by honeynet organization. Thus, the dynamic honeypot appeal to integrate passive or active probing and virtual honeypots. Their propose design overview was the component which listed below:

1) An active probing tools to learn network such as Nmap [10].
2) A passive fingerprinting tools to learn network ,such as Passive Operation Finger Printing (Pof) [11].
3) A low interaction honeypot.
4) Physical honeypot to receive redirect traffic, such asSebek.
5) A Database consist of hosts description and Log files.

Their implementation was combination of low and high interaction honeypot. They used Nmap and passive fingerprinting (P0f) to learn about network and they also use snort which is the network intrusion detection system that performs real-time traffic analysis and detects a variety of attacks and probes, such as buffer overflows, stealth port scans, and OS fingerprinting attempts. Honeyd used as a low interaction honeypots and Sebek as a high interaction honeypots. A most critical advantage of using this method to approach of dynamic honeypot trick in its capability to capture malicious attacks both on the small scale i.e. from
host perspective (physical honeypot logs) and the large scale i.e. from the network perspective (virtual honeypot logs).

B. Algorithm 2

Researchers in the article [12], clarify a method to an automatically and dynamically configure and construct honeypots in consequences of the network scanning results. In addition to it, by collecting information the administrator will be capable to protect network in superior way .Their purpose of this research is to produce one product which system administrator or researcher can swiftly construct a honeypot without the need of professional knowledge of the honeypot configuration mechanism. Moreover, the honeynet administrator is not obligatory to identify the details of the network topology or installed systems in the network to be mimicked. Furthermore, simplify configuration and reconfigured to reproduce the current network.

The specification to approach this phenomenon is considered as follows. For manager tools and identified network Honeyd [14] in Perl and Linux OS environment was used as part of this project which .Then, Nmap for operation system detection and to determine which TCP/UDP ports which are open. To construct the configuration file for honeypot or honeynet, active scanning quickly provides sufficient data while it does consume some bandwidth. The Nmap results are analyzed real-time by the honeyd Configuration Manager to establish the resulting honeynet configuration, which is stored as a honeyd configuration file. Honeyd use the IP assignment method to create the file which administrator use to determine the network configuration of the honeypots. These IP assignments consent to the administrator to produce replicas of the scanned machines. The four options for IP address assignment are as follows:

First, configure the honeypots to use the same IP addresses as the real systems. Second, adjust the network component of the IP address, while preserving the hostcomponent. Third, use a selected IP address range in which to place the ensuing honeypots. Forth, mingle the honeypot systems into the construction network where possible, although this requires that the assigned IP addresses on agiven production network can be identified. In addition, to initiate a configuration which emulates the production hosts and open ports, a small MySQL database was used to accumulate the details about the obtainable scripts, including which services they emulate, and for what operating systems they are suitable. Thereby, when an open port is found by the Honeyd Configuration Manager is possible to quickly query the database for an appropriate script.

C. Algorithm 3

Honeyd@WEB [13] is low-interaction, production, dynamic and manageable virtual honeypots via a web interface. Base on the placement of honeypot, may be set up in front of a firewall, in the Demilitarized Zone (DMZ), or behind a firewall. However, this research recommended that the best place to set up a honeypot is behind a firewall. Consequently it is competent to detect internal attackers. In addition, the earlier the honeypot is to actual servers (which are probable behind a firewall), the more expected it is to attract intruders. Practically, the low-interaction honeypot (honeyd) was integrated with P0f to develop a dynamic and manageable honeypot. Honeyd@WEB using a web-based approach to sets and deploys the honeypots. It runs P0f, Arpd (Address Resolution Protocol Daemon) and honeyd through the web interface. The aim of using P0F is listening to traffic of the production network and determine the hosts ‟ operating systems by passive fingerprinting technique''. Regarding to information which are gathering by P0f such as the number of active hosts and their operation system it will suggest the deployment of honeyd.

Using of ARP is to map Internet Protocol address (IP address) to a physical machine address that is identified in the local network. An Arp daemon which is known as Arpd listens on a particular interface and answers ARP requests for some preferred IP addresses. Here, Arpd is used to direct network traffic towards the honeypot and to get honeyd to respond to all unused IP addresses on the network.

To deploying, editing or deleting administrator can then use the web interface honeypots. For logging and analysis purposes, Snort and ACID are used to detect attacks .But we should notice the amount of captured data is limited to probes and connection attempts since low interaction honeypots are used.

D. Algorithm 4

[14] Proposed a methodology in Linux base host by having a single physical network interface card. The goal of this research was to establish a virtual honeynet on a VMware Server running Honeywall CDROM [15] then, using data capture mechanisms reports help administrator for risk evaluation. According to the results of this study administrator is capable to enhance the overall security of our network resources. This project was based on free and open source tools soLinux base operation system have been used the host OS (OS virtual machine). Likewise, the particular honeypot that integrated with Linux was Honeywall. Honeywall CDROM is a bootable CD that setups onto a hard drive and comes with all the tools and functionality for you to implement data capture, control and analysis. Moreover, VMware Server was used as the virtualization solution for this projectdue to free, reliable and an enormous proportion of support. Furthermore, Sebek High interaction honeypot was sets as a printer driver name. During the implementation author identify one problem regarding the loop in the topology and the honeypot LAN segment was causing due to configuration in eth0 and eth1 interfaces which were as a VMware bridge and eth2 as a Vmware host-only interface, but they overcome this problem by reference the Pakistanian solution research [16]. They suggest Vmnet0 on the etho as a VMware bridge interface in the direction of router. VMnet 1 on the eth1 is a VMware host only interface leading to the internal LAN segment where honeypot is there and VMnet2 on the eth2 as a VMware bridge interface which is firewalled and accessible for remote management purpose. In additional, the public IP assign to the management interface within the honeynet subnet but it has a limited access to it from ROO configuration. Finally, this virtual honeynet was online for estimate 60 days from the attack results were documented as attacked ports and services, attacker IP's and country of origin. However, the method which this researcher offered required a great amount of memory for virtualized environments and can be used as a performance benchmark. Similarly, using tools and technology such as Linux, Vmware and Snort demand a high degree of skill and customization. Not only installing, maintenance needs understanding of the system with deep knowledge regarding
the network topology, nevertheless requires several software to approach this method.

F. Algorithm 5

The Dynamic Honeynet System presented in this paper[17] is based on the Dionaea-Honeypot-Framework [19] which can be used as Low or Medium Interaction Honeypot [20]. Dionaea provides different simulated services such as SMB, HTTP, FTP or SIP. The test setup requires the SIP function only. As the Dionaea SIP component does not provide all SIP methods (01/2011) like REGISTER, this implementation is not sufficient to detect e.g., a Sipvicious attack. Due to the fact that there are unsupported SIP functions and performance issues, [17]developed a new SIP component which is completely integrated in the Dionaea Honeypot framework and has low hardware requirements. The new implementation supports all necessary interfaces and simulates all SIP methods. Therefore, this component reacts as a standard SIP server and is an interesting-looking SIP device for attackers. [18] present initial SIP-based Honeypot System. They show that it is not feasible to identify an attacker over the whole attack chain (attack stages one to four) because attackers typically change their IP addresses before starting a stage four attack. This issue is solved by the Dynamic Honeynet System, which is presented in Section IV. [21] present SIP Trace Recorder (STR), which allows passive attack monitoring in SIP-based networks. Due to the fact honeypots are configured in real-time, it is necessary to detect an attack and to send a notification message to the attacked honeypot. To solve this issue, [17] developed the Sensor component which does active monitoring based on signatures and sends alarm messages. Furthermore, [18] use a static Honeypot configuration with predefined SIP accounts. The new implementation provides dynamic functions based on the attacker's behaviour and controlled by the Sensor component which is call Enable Extension Function (EEF).

IV. **RESULT AND ANALYZIE**

Studies indicate integration of networking tools and applications is a fundamental issue to approach easy deployment of honeypot. Table I illustrates the summary of
methods regarding the researcher's algorithms.

TABLE I : COMPARITIVE TABLE ON DYNAMIC HONEYPOTS

Research Title	**Method**	**Comment**
A dynamic honeypot design for intrusion detection	Combination of low and high interaction honeypots (Honeyd,Nmap,P0f,Sebek).	Complicated ,Buffer overflow,Using different software,Linux OS,Difficult to configuration.

Dynamic honeypot construction	Low interaction honeypot and Active finger printing (Nmap).	Consume a lot of bandwidth,Linux OS.
Honeyd@Web	Low interaction honeypot and Passive finger printing (honeyd,P0f,ARPD,Snort).	Limit of dynamic property,Inside firewall , Linux OS,Needs user for maintenance.
Experiences with a generation III virtual honeynet	Honeywall with High interaction honeypot (Sebek),Vmware,Snort,TCp dump,H flow P0f.	Linux OS,Utilizing Large amount of memory, Demand high degree of knowledge for using and customizable oftools,Needs a lot of maintenance
Improved Detection and Correlation of Multi-Stage VoIP Attack Patterns by using a Dynamic Honeynet System	Low or Medium Interaction Honeypot. (Dionaea)	to release the Dynamic Honeynet System under an open source license.

V. CONCLUSIONS

The value of honeypot is obvious in the security. Despite; deploying a honeypot was the most controversial concerns in this era. To deal with this problem researcher propose solution for easy deployment of dynamic honeypot. In this paper we consider five different solutions which is suggested to deal with installing and configuration problem .Then we compare them and account their advantages based on the easy deployment aims.

ACKNOWLEDGMENT

We would like to express our sincere gratitude to Prof. Mrs. D. A. Nikam for their supervision and guidance. We would like to express our appreciation to our parents and all the teachers and lecturers who help us to understand the importance of knowledge and show us the best way to gain it.

REFERENCES

[1] K. Scarfone and P. Mell, "Guide to Intrusion Detection and Prevention Systems (IDPS)," U. S. D. o. Commerce, Ed, ed. Gaithersburg: lInstitute of Standards and Technology, 2007.

[2] M. Yuqing, et al, "J-Honeypot: a Java-based network deception tool with monitoring and intrusion detection," in Information Technology: Coding and Computing, 2004. in Proc. ITCC 2004. International Conference on, 2004, pp. 804-808 Vol.1.

[3] G. M. Bednarski and J. Branson. (2004, 2010). Understanding Network Threats through HoneypotDeployment.[InformationWarfare:].Available:http://www.infinitel00p.com/library/honeypot.pdf

[4] K. STEDING-JESSEN, et al, "Using Low-Interaction Honeypots to Study the abuse of Open Proxies to Send Spam," ed: Brazilian Network Information Center - NIC.br Computer Emergency Response Team Brazil - CERT.br, São Paulo (SP), 2007, p. 9.

[5] G. Wicherski, "Collecting and Managing Accumulated Malware Automatically"presented at the IN-DEPTH SECURITY CONFERENCE EUROPE, Vienna, Austria 2007.

[6] A. A. Ashour, "Designing high interaction windows honeynets," M.S. 1437339, University of Louisville, United States - Kentucky, 2006.

[7] A. Lanoy and G. W. Romney, "A Virtual Honey Net as a Teaching Resource," in Information Technology Based Higher Education and Training, 2006. ITHET '06. 7th International Conference on, 2006, pp. 666-669.

[8] R. Budiarto, et al, "Honeypots: why we need a dynamics honeypots," in Information and Communication Technologies: From Theory to Applications, 2004. Proceedings. 2004 International Conference on, 2004, pp. 565-566.

[9] I. Kuwatly, et al, "A dynamic honeypot design for intrusion detection," in Pervasive Services, 2004. ICPS 2004. Proceedings. The IEEE/ACS International Conference on, 2004, pp. 95-104.

[10] Nmap, "Nmap Security Scanner 5.20 ed: Gordon Lyon, 1997, p. Used to discover hosts and services on a computer network.

[11] M. Zalewski, " Passive OS fingerprinting tool," 2000-2006, Ed, 2 ed: Michal Zalewski, 2000.

[12] C. Hecker, et al, "Dynamic Honeypot Construction," in Proc. of the 10th Colloquium for Information Systems Security Education, University of Maryland, University College ,Adelphi, MD, 2006.
[13] HONEYD,"Developments of the Honeyd Virtual Honeypot, "Honeyd1.5 c ed: Monkey.org, 2007.
[14] F. H. Abbasi and R. J. Harris, "Experiences with a Generation III virtual Honeynet," in Telecommunication Networks and Applications Conference (ATNAC), 2009 Australasian, 2009, pp. 1-6.
[15] Honeynetproject. (2005, 1st November). Know Your Enemy:Honeywall CDROM Roo 3rd Generation Technology.
[16] F. A. Shuja. (2006, 1st November). Virtual Honeynet: Deploying Honeywall using VMware.
[17] Dirk Hoffstadt, Niels Wolff, Stefan Monhof, Erwin Rathgeb, " Improved Detection and Correlation of Multi-Stage VoIP Attack Patterns by using a Dynamic Honeynet System" Communications (ICC), 2013 IEEE International Conference,
Computer Networking Technology Group ,University of Duisburg-Essen ,Essen, Germany
[18] Dirk Hoffstadt, Alexander Marold, and Erwin Rathgeb, "Analysis of SIP-Based Threats Using a VoIP Honeynet System," in Conference proceedings of the 11th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom-2012), Liverpool, UK, 2012.
[19] (2012, Aug.) dionaea. [Online]. http://dionaea.carnivore.it/
[20] Iyatiti Mokube and Michele Adams, "Honeypots: concepts, approaches, and challenges," in Proceedings of the 45th annual southeast regional conference, Winston-Salem, North Carolina, USA, 2007, pp. 321-326.
[21] Dirk Hoffstadt, Stefan Monhof, and Erwin P. Rathgeb, "SIP Trace Recorder: Monitor and Analysis Tool for threats in SIP-based networks," in TRaffic Analysis and Classification Workshop (IWCMC2012-TRAC), Limassol, Cyprus, Aug. 2012.

