ISSN: 0975-833X

Available online at http://www.journalcra.com

International Journal of Current Research
Vol. 12, Issue, 02, pp.10299-10302, February, 2020

DOI: https://doi.org/10.24941/ijcr.38153.02.2020

INTERNATIONAL JOURNAL OF CURRENT RESEARCH

RESEARCH ARTICLE

AN INTERACTIVE HONEYPOT USING PACKET SNIFFER&SSH CLIENT (LOW LEVEL INTERACTION)

***Onuoha Chibuike Martins, Nnadi Leonard C. Amadi E.C. and Charles Ikerionwu**

Department of Information Technology, School of Information and Communication Technology, Federal University of Technology Owerri

ARTICLE INFO

Article History:
Received 24th November, 2019
Received in revised form
10th December, 2019
Accepted 09th January, 2020
Published online 28th February, 2020

Key Words:

Security, Honeypot,
Network, Compromise, Attacker, Port.

ABSTRACT

Information Security, as the word says, has always meant securing assets and providing controls and procedures to resist damage or potentialimpact on the system(s) under consideration. This definition as isunderstood in security circles has various potential inferences and istypically understood in the defensive sense. Protect the network, protect the server, protect the logs, the list never ends. Honeypot is a security mechanism whose value lies in unauthorized or illicit use of that resource.It is a resource which is intended to be attacked and compromised to gain more information about the attacker and his attack techniques. They are a highly flexible tool that comes in many shapes and Sizes. However, this paper examines the different types of honeypot and their levels of interaction with the attacker. This paper attempts to examine a honeypot environment using Tcpdump packet sniffer through SSH client.

Citation: ***Onuoha Chibuike Martins, Nnadi Leonard C. Amadi E.C. and Charles Ikerionwu.*** **2020.** "An Interactive Honeypot using packet sniffer & Ssh Client (Low Level Interaction)", *International Journal of Current Research*, 12, (02), 10299-10302.

INTRODUCTION

Organizations around the world not only in Nigeria are facing challenges in trying to combat network security. The connected electronic information network has become an integral part of our daily lives. All types of organizations, such as medical, financial and education institution, use this network to operate effectively. They utilize the network by either storing, processing, collecting or sharing vast amounts of digital information. As more digital information is gathered and shared, the protection of this information is becoming even more vital to our national security and economic stability. Cybersecurity is the ongoing effort to protect these networked systems and all of the data from unauthorized use or harm. On a personal level, you need to safeguard your data and your computing devices. At the corporate level, it is everyone's responsibility to protect the organization's reputation, data and customers. Threat in today's environment is increasing with a high magnitude and effect in spite of having different security mechanisms in place. Just by having antivirus software it is very difficult to say that the systems are free from virus threat. Same as sitting behind a firewall doesn't means that the network is out of reach of malicious activities and intents.

****Corresponding author:*** **Onuoha Chibuike Martins,**
Department of Information Technology, School of Information and Communication Technology, Federal University of Technology Owerri.

This is all because every new virus or new attack finds some different way to penetrate the security infrastructure, which often goes undetected by the security technologies in place. The answer to thisissue is to have some technology, which is designed to get compromised and ismeant to welcome the attackers. Because this is the excellent way to learn new developments in the attacker's community and their motive of penetrating into any security perimeter, it also has lots of other values and benefits discussed in the following sections. According to Lance Spitzner (active member of Project Honeynet 3), "A honeypot is an information system resource whose value lies in unauthorized or illicit use of that resource" 4. The main goal is to distract the attacker from real systems and to gain valuable information about them with the tools and exploits they use. All the traffic from and to a honeypot is suspicious and unauthorized because of the fact that no production services are provided by this resource. All data collected by a honeypot is therefore interesting and it never generates big number of logs. It can further be used for prevention of attacks and can debar attackers from other systems by occupying their resources for long duration. The information gathering for an attack depends on the level of tracking enabled on the honeypot. Common tracking level includes technologies and methods like firewall, system logs, sniffers, IDS tools, integrity checkers and few others.

Honeypot and its classification: Honeypots are decoy systems or servers deployed alongside production systems within your network. When deployed as enticing targets for

attackers, honeypots can add security monitoring opportunities for blue teams and misdirect the adversary from their true target. A honeypot is a computer or computer system intended to mimic likely targets of cyberattacks. According to Antipolis, Pouget, & Dacier (2003), a Honeynet is actually a network made up of real systems designed to be hacked. It can be used to detect attacks or deflect them from a legitimate target. It can also be used to gain information about how cybercriminals operate. Like mice to cheese-baited mousetraps, cybercriminals are attracted to honeypots — not because they're honeypots. The bad guys think the honeypot is a legitimate target, something worthy of their time. That's because the bait includes applications and data that simulate a real computer system.

Honeypots come in a variety of complexities depending on the needs of your organization and can be a significant line of defense when it comes to flagging attacks early. For a honeypot to work, the system should appear to be legitimate. It should run processes a production system is expected to run, and contain seemingly important dummy files. The honeypot can be any system that has been set up with proper sniffing and logging capabilities. It's also a good idea to place a honeypot behind your corporate firewall—not only does it provide important logging and alerting capabilities, but you can block outgoing traffic so that a compromised honeypot cannot be used to pivot toward other internal assets. Sachan, & Panchagavi (2016) honeypots are designed to mimic the actual systems that the intruder wants to break into but limiting the intruder from accessing the entire network. In terms of objectives, there are two types of honeypots: research and production honeypots. Research honeypots gather information about attacks and are used specifically for studying malicious behavior out in the wild. Looking at both your environment and the wider world, they gather information about attacker trends, malware strains, and vulnerabilities that are actively being targeted by adversaries. This can inform your preventative defenses, patch prioritization, and future investments. Production honeypots, on the other hand, are focused on identifying active compromise on your internal network and tricking the attacker. Information gathering is still a priority, as honeypots give you additional monitoring opportunities and fill in common detection gaps around identifying network scans and lateral movement. Production honeypots sit with the rest of your production servers and run services that would typically run in your environment. Research honeypots tend to be more complex and store more types of data than production honeypots.

Advantages of research honeypots: In network security, there are many available solutions in the market to help defend yourself and your organization's reputation in the cyberspace, both IT personnel's and server administrators can check the internet to find solution that suites their needs. Aside monitoring traffic to better understand the angle attackers are coming from, here are some of the advantages of deploying a honeypot: According to Sharma (2013), Honeypot can capture new tools for detecting attacks too. It gives more ideas and deepness of the subject proving that it is possible to discover different point of views and apply them for our security solutions. Through the deployment of honeypots, log files are periodically checked to see the nature of the attacker's malicious behavior. Also through honeypots, new pattern of attacks can be discovered and from this pattern, new security measures (Patches) can be created by looking at the pattern.

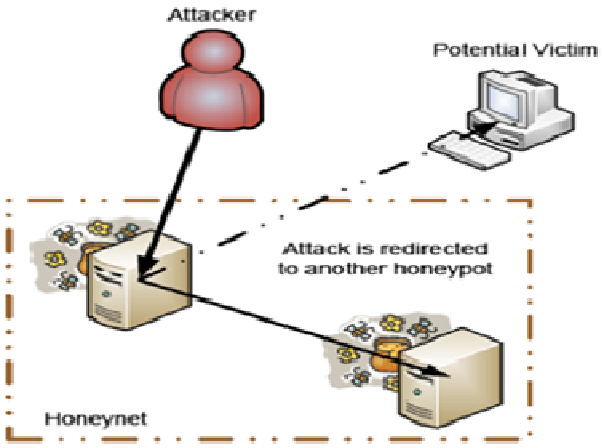

Fig a: A Honeypot Environment

Honeypot are cheap and easy to deploy and monitor.According to Spitzner(2002), Honeypots can give you the precise information you need in a quick and easy-to-understand format. This information can then be used for statistical modeling, trend analysis, detecting attacks, or even researching attackers.

Disadvantages of research honeypots: Honeypots, when administered poorly can put the network and the entire security devices at risk. Because it can potentially be detected by the attacker. According to Peter&Schiller (2011), only monitor interactions made directly with the honeypot - the honeypot cannot detect attacks against other systems. Kambow & Passi (2014) opined that one huge drawback generally faced by honeypots is that they are worthless if no one attacks them. Obviously, they can accomplish wonderful things but if the attacker doesn't send any packet to honeypots then it would blissfully unaware of any unauthorized activity.

Research honeypots: monitoring the attacker: According toKrawetz (2019), a simple honeypot can listen to one port, multiple ports, or every port. However, you don't want it to listen on a port that is running an active service. Otherwise, it will just be collecting everything -- scanners, attackers, and legitimate users. In the course of our technical paper, we will be using the following to deploy a non-interactive honeypot and carefully monitor their IP's and port numbers:

- Packet Sniffer (tcpdump)
- Ssh client
- Ubuntu server core 14.0.4
- Windows SCP
- Internet connection

Packet Sniffer: Packet sniffers or protocol analyzers are tools that are used to capture and analyze log packets in other to diagnose network related issues. According to Andy (2019), packet sniffers work by intercepting and logging network traffic that they can 'see' via the wired or wireless network interface that the packet sniffing software has access to on its host computer. Examples of packet sniffers are wireshack, lifewire, tcpdump, snorts, tshark, bro, etc. Secure Shell client: SSH client is a software program which uses the secure shell protocol to connect to a remote computer. Ubuntu server core: This is a Linux open source, lightweight command line

interface server that we will be using as our operating system to integrate our honeypot server.

Configuring our honeypot server: In our honeypot server, we will be using SSH daemon as our running service. First, we will change our ssh default port from 23 to a non-standard port so that common attackers or scanners will not easily find it. To do that we open our etc/ssh/sshd_config and go to the line number have port 23 and change to 41244. We use sudonano /etc/ssh/sshd_config to open up the text editor in other to change the port to non-standard ports. Next, we try to make the service available and allow all inbound and outbound connections by dropping all other network traffic exiting through eth0 except for traffic on port 41244. First, we connect as root user and execute the below command:

iptables -A INPUT -I eth0 –p tcp –sport 41244 –m state –state ESTABLISHED –j ACCEPT
iptables -A INPUT -o eth0 –p tcp –dport 41244 –m state –state NEW.ESTABLISHED –j ACCEPT

Configuring the Tcpdump: According to Neal (2019), while tcpdump can capture entire packets, a really basic honeypot just needs to record IP addresses, port numbers, and network protocols (typically TCP, UDP, or ICMP). We will use the below basic tcpdump command for doing this is:

sudotcpdump -tttt -q -l -i eth0 -n -s0
"-tttt" tells tcpdump to preface each line of packet output with the current date and time in a human readable form. E.g., "2019-05-17 14:39:54.576416" (that's down to the fraction of a second).

"-q" tells it to be quiet (minimize the output).

"-l" (lowercase 'L') directs it to use line-buffering for output. This is great for streaming results into a file or another application.

"-i eth0" specifies the network interface for sniffing the traffic. If you leave it off, then it will listen on every interface. But if you have one network interface devoted to the honeypot (e.g., eth0), then use "-i" to tell tcpdump where to listen. (The interface "eth0" may not be right for you. If you don't know your interface's name, use sudotcpdump -D to list every available interface.)

"-n" prevents hostname and protocol lookups. Querying DNS and doing name resolution lookups can slow things down. Also, if DNS queries attempt to use the same interface where tcpdump listens, then it could end up capturing its own network requests, causing a feedback loop and infinite network traffic. If you want to look up hostnames on a honeypot, then do it after the fact and not during data collection. "-s0" determines how big of a packet buffer it should use.

Output: IP address of my honeypot server is 10.0.2.15. From the output log, we can discover how intruders flood the honeypot server. We see the intruders ip address, port number they are scanning in my honeypot server and also the protocol. We can decide to further analyze where intruders are coming from by using their IP addresses to carry out a WHOIS lookup on https://www.whoishostingthis.com/ to see the intruders information, whois record, hosting provider company and also the country from which attack is coming from.

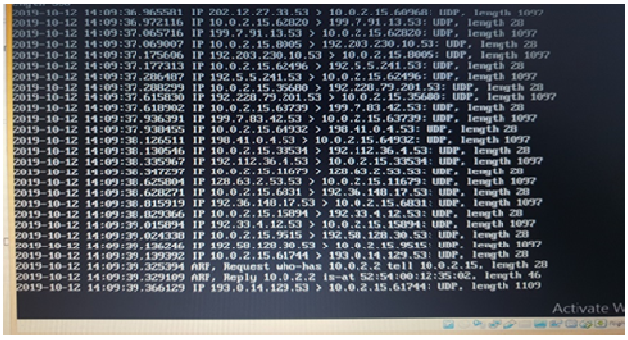

Fig b. An Output log in 5mins

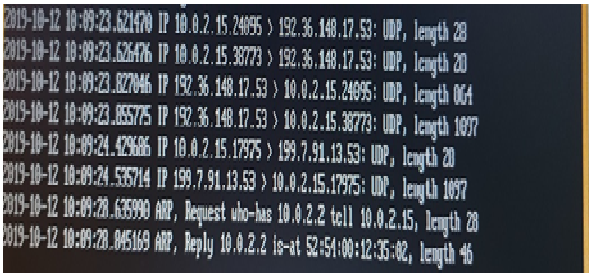

Fig c. An Output log after 2hrs

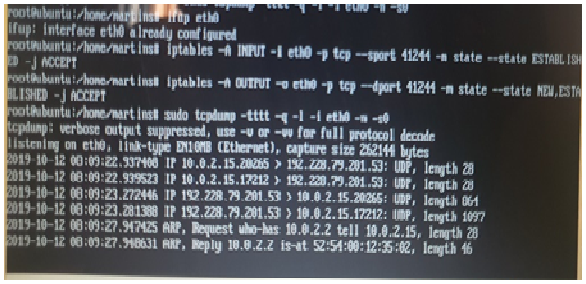

Fig d. An Output log after 4hrs

Similarly we can decide to check the ports they are scanning and also make a port lookup using https://www.speedguide.net/ports.php to know if these ports are always vulnerable, a target for attackers and if these ports are for trojan horses. If all theseinformation are collected effectively, then we can always know what to implement in our firewall rules. Also gives a little insight on the next patch that will be introduced and deploy. All a network security administrator should do is to always implement patches to stay up to date.

Review of related works: According to Samu (2016), He opined that in trying to detect real time attacks, a honeypot could be implemented using puppet enterprise agent to gather hacker's activities and monitor their operations. Crowie is an open source debian honeypot technology which is intended to mimic a server. To appear legitimate to attackers, it has imitations of files and folders one would expect on a Linux machine such as /etc and /bin (Liu, Mahar, Virdi& Zhou, 2016). According to Spitzner (2002), he setup a honeypot to track hackers using an application called Specter. Specter must be manually installed on any system that you want to be a honeypot. This can be a time-intensive process. However, Specter has a major advantage over BOF in that it can be remotely managed. Specter comes with a second application called Specter Remote, which allows an administrator to remotely manage all the Specter honeypots.

In a journal titled Honeypot: A trap for hackers, Paliwal (2017) was able to deploy a low interactive honeypot using Honeyd. Honeyd is a freely available framework for setting up virtual honeypots. With honeyd it is possible to setup honeypots with different personalities and services on one machine. Honeyd emulates the different operating system's IP stack and binds certain script to a desired port to emulate a specific service. Honeyd is able to fool network fingerprinting tools (which store the fingerprints of operating systems present in the network) to think they are dealing with a real operating system ranging from a Windows NT to an AIX box(Advanced Interactive eXecutive box) It is a proprietary operating system developed by IBM based on UNIX System

Conclusion

Honeypot is a useful mechanism to combat network bridges and challenges to services that is connected to the Internet. It is expected that network administrators should always be careful when dealing with attacks. It is worthy of note that attackers are constantly looking for targets to attack or blindly attacking.Through the concept of honeypot, new threats are discovered and analyze to know its purpose and to provide ways of mitigating them.

Recommendations and future works

Recommendations

High level interaction honeypot networks should be implemented by network security experts with sound knowledge of what they want and how result will be beneficial to them. We should experiment with honeypots in safe place close by to the production systems. It is the best solution for entrapment, because it can easily be configured. It is important to deploy a low interactive honeypot at least once or twice for those who run a web service. The findings can be of great help to the administrator in creating baseline for what to expect. Also it will serve as a warning model for future potential issues. Finally this practice may be used to provide useful tips in improving and implementing firewall rules. However, low interaction honeypots are emulating services of an operating system. Thus, as a hacker, you can come up with some conclusions by using this very basic information. As the services are emulated, low interaction honeypot cannot handle complicated services inside. Trying to break the system using this technique may work efficiently. The key point is to look for information through network; also we also know that low interaction honeypots are using the system's resources that they are on it. Attackers may quickly detect this honeypot server and use it as a tool to infiltrate attacks. In deploying low interactive honeypots, we may have variation as a result of different ISPs and also bandwidth specification. So in other to solve this issue we try to deploy honeypot on a droplet or cloud servers.

Future Works: Researchers who want to understand how an interactive honeypot works should deploy a full modern honeypot network on a droplet (virtual private server). A modern honeypot network includes both an intrusion detection applications and they include both snort and bro with others for effective capture and analysis. Modern Honeypot deploy scripts include several common honeypot technologies, including Snort, Cowrie, Dionaea, and glastopf, among others. A close comparison of these honeypot technologies could be carried out to know their strength and weaknesses. Furthermore, analysis of log files from a modern honeypot network should be done using supervised machine learning. Where output logs could be classified using logic based classifiers for easy predictions. Finally, work should be done on setting up an high interactive honeynet to further understand the patterns, operation and activities of attackers trying to gain access to our server.

REFERENCES

Antipolis, S., Pouget, F., & Dacier, M. 2003. *White Paper : " Honeypot, Honeynet, Honeytoken: Terminological issues 1 ."* 1–26.

Kambow, N., Passi, L. K. 2014. *Honeypots : The Need of Network Security*. *5*(5), 6098–6101.

Krawetz N. 2019. *Building a Basic Honeypot*: Retrived from https://www.https://www.hackerfactor.com/blog/index.php?/archives/841-Building-a-Basic-Honeypot.html

Liu L., Mahar K., Virdi C. & Zhou H, 2016. *White Paper: Using a Honeypot to Spy on Attackers.* Computer & Network Security

Paliwal, S. 2017. *Honeypot : A Trap for Attackers*. *6*(3), 842–845. https://doi.org/10.17148/IJARCCE.2017.63197

Peter, E., & Schiller, T. 2011. *A Practical Guide to Honeypots*. 1–19.

Sachan, A., & Panchagavi, R. 2016. *Honeypots : Sweet OR Sour spot in Network Security ?6*(3), 904–907.

Samu, F. 2016. *Design and Implementation of a Real-Time Honeypot System for the Detection and Prevention of Systems Attacks*.

Sharma, A. 2013. *HONEYPOTS IN NETWORK SECURITY*. *1*(5), 7–12

Spitner L. 2002. *Honeypots Tracking Hackers*: *The Lure of Honeypots.* Boston, MA: Addison Wesley

****