COLLEGE OF TECHNOLOGY AND ENGINEERING

MAHARANA PRATAP UNIVERSITY OF AGRICULTURE & TECHNOLOGY

UDAIPUR (RAJ.)

A

SEMINAR REPORT

ON

INTRUSION DETECTION USING HONEYPOTS

submitted in partial fulfillment for the award of the Degree of Bachelor of
Technology in Department of Computer Science & Engineering
(Session 2020-2021)

Submitted By:
Asmita Vaishnav
B.Tech. Final Year
Computer Science & Engineering

ACKNOWLEDGEMENT

I would like to place on record my deep sense of gratitude to Dr. Naveen Choudhary, Head of Department of Computer Science and Engineering for his generous support. Further, I would like to thank Mrs. Kalpana Jain, Assistant Professor, Department of Computer Science and Engineering and Ms. Diksha Goyal, Assistant Professor (TEQIP-III) for their overwhelming support throughout the duration of the seminar. Their motivation always gave me the required inputs and momentum to continue with my work, without which the seminar work would not have taken its current shape. Also, their valuable suggestion and numerous discussions have always inspired new ways of thinking.

Asmita Vaishnav

DECLARATION

I hereby declare that the seminar titled “**HONEYPOTS**” has been presented by me and is not reproduced as it is from any other source. It has been submitted in partial fulfillment of the requirement for award of Bachelor of Technology in Computer Science and Engineering, College of Technology and Engineering, a constituent of Maharana Pratap University of Agriculture and Technology, Udaipur.

Date: 10-06-2021

Place: Udaipur

Asmita Vaishnav

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATION

1. **NID -** Network Intrusion Detection
2. **IDS -** Intrusion Detection System
3. **SIEM** - Security Information and Event Management System
4. **DMZ** - Demilitarized Zone
5. **NIDS -** Network Intrusion Detection System
6. **HIDS** - Host Intrusion Detection System
7. **NIC** - Network Interface Card
8. **VM** - Virtual Machine
9. **TCP** - Transfer Control Protocol
10. **IP -** Internet Protocol
11. **GNU** - General Public License
12. **LAN -** Local Area Network
13. **UDP** - User Datagram Protocol
14. **ICMP** - Internet Control Message Protocol
15. **FTP** - File Transfer Protocol
16. **SSH** - Secure Shell
17. **Nmap** - Network Mapper
18. **HTTP** - Hypertext Transfer Protocol
19. **OS** - Operating System
20. **IGMP** - Internet Group Management Protocol
21. **OSI** - Open Systems Interconnection Model
22. **GUI** - Graphical User Interface
23. **SIM -** Security Information Management
24. **SOC -** Security Operations Canter

ABSTRACT

Security of network becomes primary importance of every organization to protect it from any kind of cyber attack. For this, the analysis of traffic that travels in the network is an imperative task. Traffic analysis on network is done by installing various network intrusion detection (NID) devices. One of a very vital device for detecting network intrusions is a Honeypot. Honeypots interact with the attacker and collect the data which can be analyzed to retrieve information regarding the attacks and attacker in the network. Honeypots are security resources that are targeted by the attacker and generate data logs of the attacks. Honeypots provide relevant data in a small quantity so that security researchers could easily understand and the analysis of the data becomes feasible. This report, covers the major aspects of how honeypots are simulated to interact with attackers and the data collected by using honeypot is analyzed using various tools.

CHAPTER 1

INTRODUCTION

It is said that a good defense is a good offense. Past few years, computer security scholars and community took this idea into consideration and developed a concept of honeypot. Traditionally, the idea was more focused on the defensive side and they developed the powerful technologies and tools like Firewall and Intrusion Detection System (IDS) to defend the network from intruders. Today, they are more concerned in studying the types of attacks; the various tools used for attacking, the new kinds of virus and other security threats so that they can defend their system more securely. The idea behind the honeypot is to create a virtual or in some scenario a real system, put the system visible to the attackers so that they can compromised and probe. The system will keep track of the activities and later the logged information is analyzed to make sure the production services and network are secured with new threats.

Lance Spitzner defines honeypot technology as – "A honeypot is security resource whose value lies in being probed, attacked, or compromised."

1.1 WHAT IS AN INTRUSION DETECTION SYSTEM?

Intrusion detection can be defined as the ability to monitor and react to computer misuse. An **intrusion detection system** (**IDS**) is a device or software application that monitors a network or systems for malicious activity or policy violations. Any intrusion activity or violation is typically reported either to an administrator or collected centrally using a security information and event management (SIEM) system. A SIEM system combines outputs from multiple sources and uses alarm filtering techniques to distinguish malicious activity from false alarms.

IDS types range in scope from single computers to large networks. The most common classifications are:

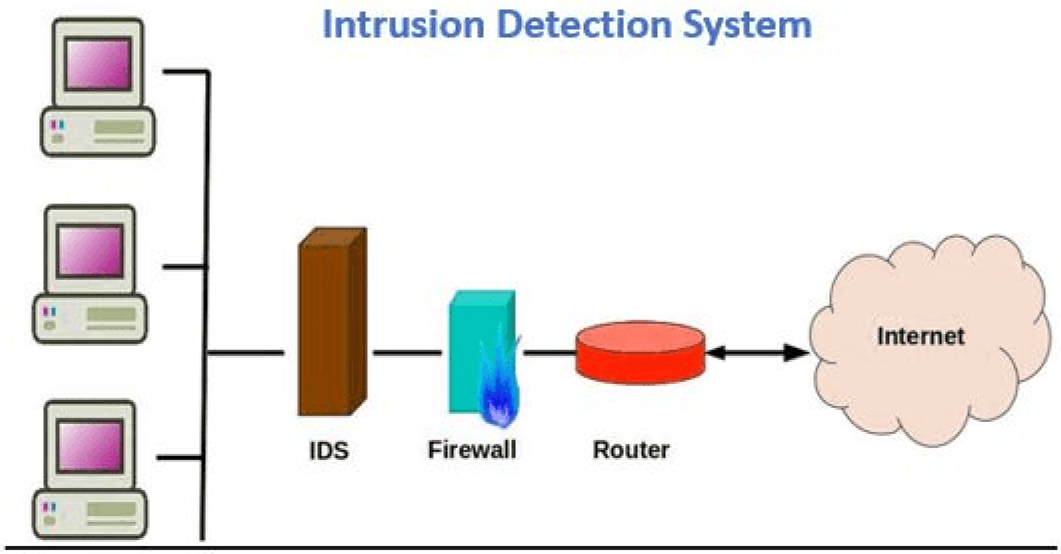


Fig 1.1 Intrusion detection system

- **Network intrusion detection systems (NIDS) -** Network intrusion detection systems (NIDS) are set up at a planned point within the network to examine traffic from all devices on the network. It performs an observation of passing traffic on the entire subnet and matches the traffic that is passed on the subnets to the collection of known attacks. Once an attack is identified or abnormal behavior is observed, the alert can be sent to the administrator. An example of a NIDS is installing it on the subnet where firewalls are located in order to see if someone is trying crack the firewall.

- **Host-based intrusion detection systems (HIDS) -** Host intrusion detection systems (HIDS) run on independent hosts or devices on the network. A HIDS monitors the incoming and outgoing packets from the device only and will alert the administrator if suspicious or malicious activity is detected. It takes a snapshot of existing system files and compares it with the previous snapshot. If the analytical system files were edited or deleted, an alert is sent to the administrator to investigate. An example of HIDS usage can be seen on mission critical machines, which are not expected to change their layout.

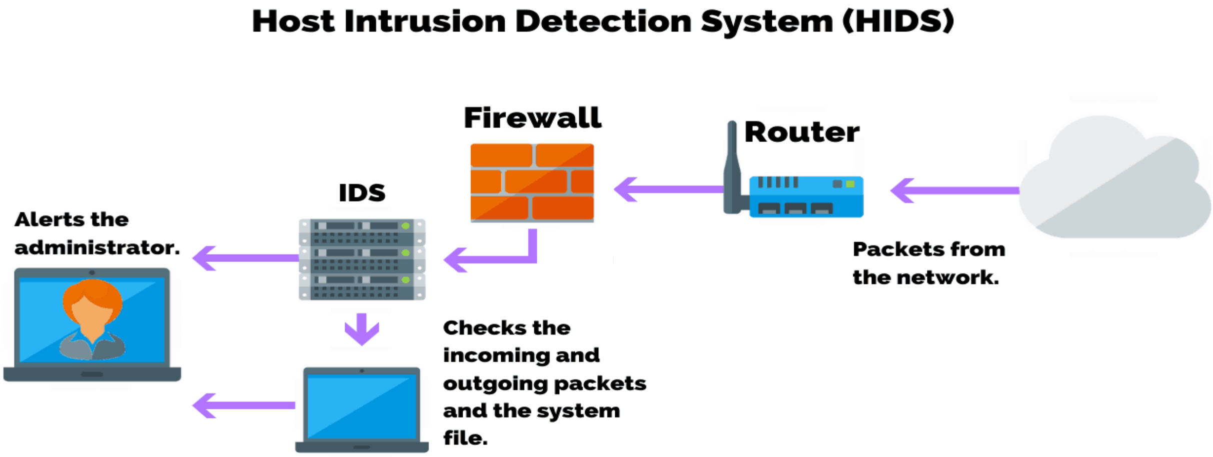

Host Intrusion Detection System (HIDS)
Firewall
Router
IDS
Alerts the administrator.
Packets from the network.
Checks the incoming and outgoing packets and the system file.

Fig 1.2 Host Intrusion Detection System

1.2 WHAT IS A FIREWALL?

A firewall is a network security device, either hardware or software-based, which monitors all incoming and outgoing traffic and based on a defined set of security rules it accepts, rejects or drops that specific traffic.

- **Accept :** allow the traffic
- **Reject :** block the traffic but reply with an “unreachable error
- **Drop :** block the traffic with no reply

A firewall establishes a barrier between secured internal networks and outside untrusted network, such as the Internet.

Firewalls are generally of two types:

- **Host- based Firewalls:** Host-based firewall is installed on each network node which controls each incoming and outgoing packet. It is a software application or suite of applications, comes as a part of the operating system. Host-based firewalls are needed because network firewalls cannot provide protection inside a trusted network. Host firewall protects each host from attacks and unauthorized access.
- **Network-based Firewalls:** Network firewall function on network level. In other words, these firewalls filter all incoming and outgoing traffic across the network. It protects the

internal network by filtering the traffic using rules defined on the firewall. A Network firewall might have two or more network interface cards (NICs). A network-based firewall is usually a dedicated system with proprietary software installed.

Comparison of IDS with Firewalls:

IDS and firewall both are related to the network security but an IDS differs from a firewall as a firewall looks outwardly for intrusions in order to stop them from happening. Firewalls restrict access between networks to prevent intrusion and if an attack is from inside the network it don't signal. An IDS describes a suspected intrusion once it has happened and then signals an alarm.

So Honeypots are basically a product of Intrusion Detection devices; they are decoy servers or systems, and they gather information about attackers to the network or server.

CHAPTER 2

HONEYPOTS

2.1 WHAT IS A HONEYPOT?

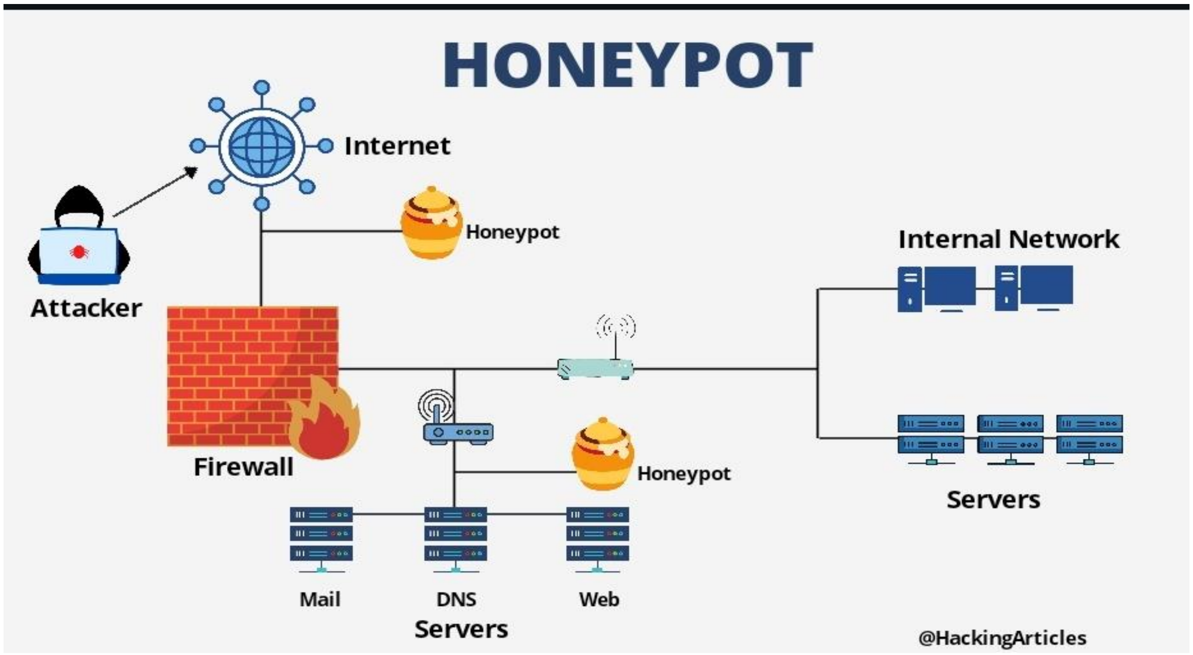


Fig 2.1 Honeypot network

- Honeypots are decoy systems used in the network to lure the attackers so that attacker gets trapped in the decoy system and their activities are captured for further analysis. A honeypot is a resource whose value is being in attacked and compromised. This means, that a honeypot is expected to get probed, attacked and potentially exploited.

- Honeypot do not fix anything. They provide us additional, valuable information. A honeypot is a resource, which pretends to be real target. The main goals are the distraction of an attacker and the gain of the information about the attack and the attacker.

- Global communication is getting more important every day. At the same time, computer crimes are increasing. Counter measures are developed to detect or prevent attacks - most of these measures are based on known facts, known attack patterns. As in the military, it is important to know, who your enemy is, what kind of strategy he uses, what tools he utilizes and what he is aiming for. Gathering this kind of information is not easy but important. By knowing attack strategies, countermeasures can be improved and vulnerabilities can be fixed.

- A honeypot is primarily an instrument for the information gathering and learning. Its primary purpose is not to be ambush for the Blackhat community to catch them in action and to press charges against them. The lies on silent collection of as much information as possible about their attack patterns, used programs, purpose of attack and Blackhat community itself. All this information is used to learn more about the Blackhat proceedings and motives as well as their technical knowledge and abilities.

- Honeypots are not the perfect solution for solving or preventing computer crimes. Honeypots are hard to maintain and they need the good knowledge about the operating systems and network security. In the right hands honeypot is effective tool for the information gathering. In the wrong, inexperienced hands, a honeypot can become another infiltrated machine and an instrument for the black hat community.

2.2 WORKING OF HONEYPOT

- A honeypot works by being an intentionally vulnerable hole in security. Typically these devices will take the form of a virtual machine (VM) that has been deliberately weakened and placed in an accessible area of the network. These VMs will often have critical security updates missing, along with open ports and unnecessary services enabled for a hacker to exploit. Additionally, a honeypot device will usually have administrator accounts with weak passwords or no password at all, making it easy for an attacker to escalate their privileges without difficulty.
- All of these security weaknesses will cause an attacker to think that they've found an easy target to infiltrate, when in reality their time is being wasted as the administrator monitors their activity and shuts down access to the rest of the network. The end result

is an attacker being caught in a trap with nothing to show for it in terms of valuable data or systems access. By the time the hacker has realized what’s going on, the administrator has gathered sufficient information to further reinforce the network or report the activity to authorities.

- Honeypots are often placed in a demilitarized zone (DMZ) on the network. That approach keeps it isolated from the main production network, while still being a part of it. In the DMZ, a honeypot can be monitored from a distance while attackers access it, minimizing the risk of the main network being breached.

- Honeypots may also be put outside the external firewall, facing the internet, to detect attempts to enter the internal network. The exact placement of the honeypot varies depending on how elaborate it is, the traffic it aims to attract and how close it is to sensitive resources inside the corporate network. No matter the placement, it will always have some degree of isolation from the production environment.

- All in all the honey pot helps a company prepare for attacks and respond to those attacks by learning from information gathered.

A few key points to be remembered about the operation of honeypots:

- They look and behave as if they are real.
- They don’t disclose existence at any time.
- They are partially disabled so hackers cannot take it over.
- They contain firewalls that protect outbound traffic in the case one is taken over.
- They live in a network that is untouched by normal traffic.
- They sound silent alarms when traffic goes to or from it.
- They begin logging all intruder activity the moment it first senses intrusion.

2.3 CLASSIFICATION OF HONEYPOTS

Honeypots are classified based on their deployment and the involvement of the intruder.

1. Based on their deployment

- **Research Honeypots:**

  They run to gather information about the motives and tactics of the attacker targeting different networks. These honeypots do not add direct value to a specific organization; instead, they are used to research the threats that organizations face and to learn how to better protect against those threats. Research honeypots are complex to deploy and maintain, capture extensive information, and are used primarily by research, military, or government organizations.

- **Production Honeypots:**

  These are easy to use, capture only limited information, and are used primarily by corporations. Production honeypots are placed inside the production network with other production servers by an organization to improve their overall state of security. Normally, production honeypots are low-interaction honeypots, which are easier to deploy. They give less information about the attacks or attackers than research honeypots.

2. Based on involvement

- **Low Interaction Honeypots**

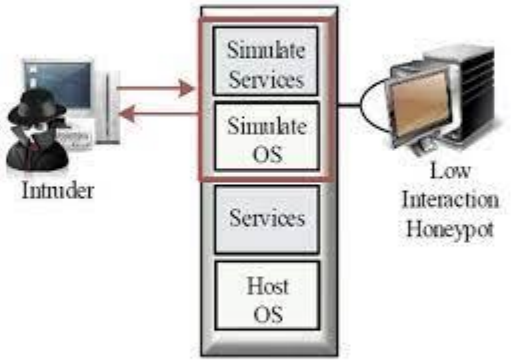


Fig 2.2 Low interaction honeypot

Low interaction honeypots gives very little insight and control to the hacker about the network. It simulates only the services that are frequently requested by the attackers. The main operating system is not involved in the low interaction systems and therefore it is less risky. They require very fewer resources and are easy to deploy. The only disadvantage of these honeypots lies in the fact that experienced hackers can easily identify these honeypots and can avoid it.

- **Medium Interaction Honeypots:**

  Medium interaction honeypots allow more activities to the hacker as compared to the low interaction honeypots. They can expect certain activities and are designed to give certain responses beyond what a low-interaction honeypot would give. Developing a mid-involvement honeypot is complex and time consuming. Special care has to be taken for security check as all developed fake daemons need to be as secure as possible.

- **High Interaction Honeypots:**

  A high interaction honeypot offers a large no. of services and activities to the hacker, therefore, wasting the time of the hackers and trying to get complete information about the hackers. These honeypots involve the real-time operating system and therefore are comparatively risky if a hacker identifies the honeypot. High interaction honeypots are also very costly and are complex to implement. But it provides us with extensively large information about hackers.

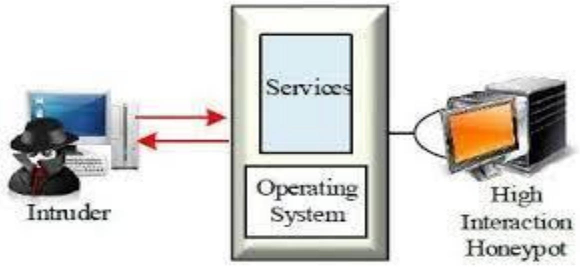


Fig 2.3 High interaction honeypot

2.4 Other Types of Honeypots:

- **Malware Honeypots** - These types of honeypots detect malware based on known replication techniques and propagation vectors.

- **Database Honeypots** - Since attacks on databases like SQL injections are fairly common, you can use database honeypots to distract an attacker from your legitimate database servers by setting up decoy databases.

- **Client Honeypots** - These honeypots typically act as servers, listening in for incoming connections. Client honeypots actively engage with malicious servers that attack clients. They pose as a client to monitor and record any modifications.

- **Email Honeypots** - Email honeypots are a list of email addresses used by email service providers to detect spammers. Typically, accounts inactive over a long period of time are used for this purpose.

- **Spider Honeypots** - These honeypots are used to trap web-crawlers by creating fake web pages and links only reachable by crawlers.

2.5 Honeynet: A Network of Honeypots

A honeynet is a decoy network that contains one or more honeypots. It looks like a real network and contains multiple systems but is hosted on one or only a few servers, each representing one environment. For example, a Windows honeypot machine, a Mac honeypot machine and a Linux honeypot machine. A "honeywall" monitors the traffic going in and out of the network and directs it to the honeypot instances. You can inject vulnerabilities into a honeynet to make it easy for an attacker to access the trap. Any system on the honeynet may serve as a point of entry for attackers. The honeynet gathers intelligence on the attackers and diverts them from the real network. The advantage of a honeynet over a simple honeypot is that it feels more like a real network, and has a larger catchment area. This makes honeynet a better solution for large, complex networks – it presents attackers with an alternative corporate network which can represent an attractive alternative to the real one.

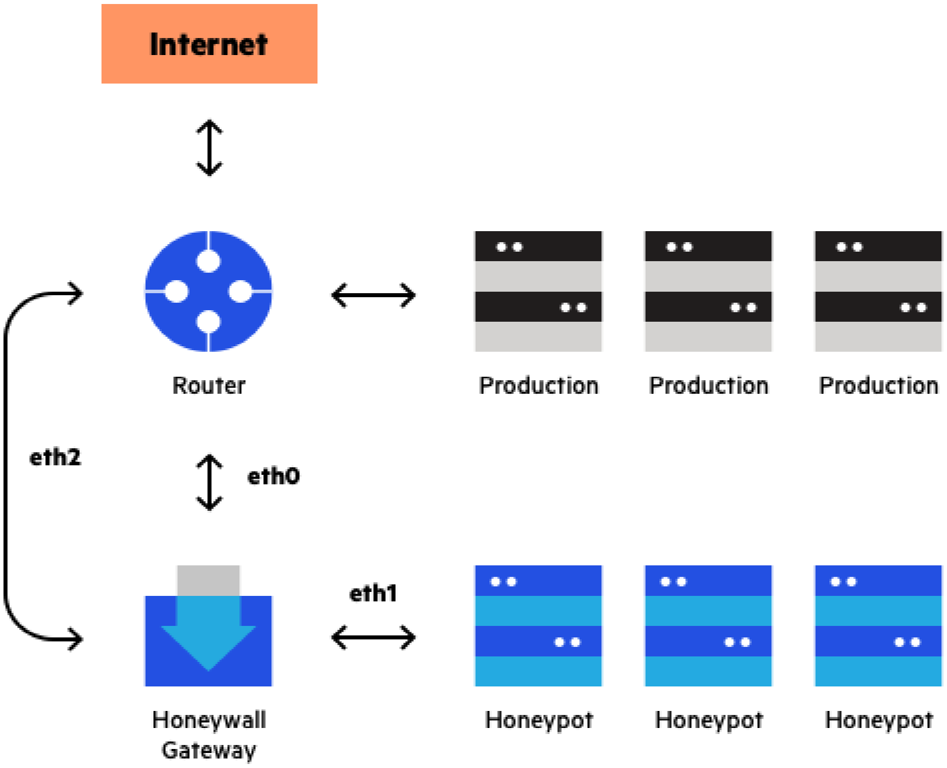


Fig 2.4 Honeynet

CHAPTER 3

IMPLEMENTATION

A honeypot system is simulated in different environments both in Windows and Linux environment using appropriate honeypot tool. The honeypot system is connected to the network to attract data. From the data packets that are collected on the network, data is analyzed. Data collected by using proposed simulated honeypot is compared with the existing honeypot techniques. We will discuss the comparison based on TCP. TCP is a protocol standard that defines how to maintain and establish network conversations via which application programs exchange data. It also defines how computer send packets of data to each other.

TOOLS USED:

3.1 HONEYD

Honeyd is an open source program by Niels Provos that enables a client to set up and run numerous virtual hosts on a network. The hosts can be designed to run random services, and their personality can be adjusted with the goal that they give off an impression of being running certain operating systems. Honeyd enhances cyber security by providing various mechanisms for threat detection and assessment. When a virtual IP in system is attacked, then Honeyd interact with attackers and play the role of the victim.

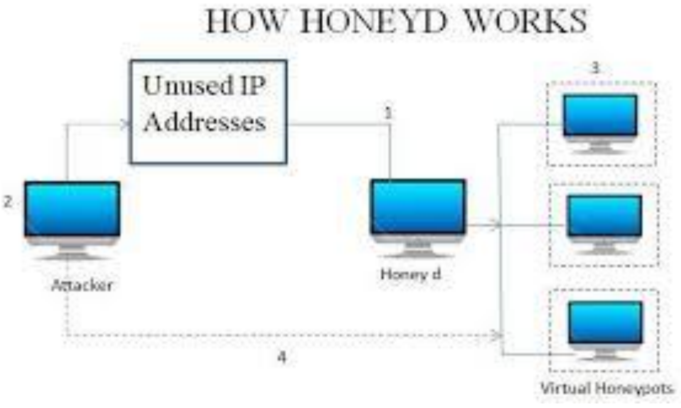


Fig 3.1 Honeyd working

Honeyd is an open source programming tool released under GNU General Public License.

Features:

- Honeyd is a low-interaction honeypot structure that permits a solitary host machine to assert unused IP addresses on Local Area Network (LAN) and reproduce a self-assertive virtual system of honeypots.

- It reproduces the network stack for the simulated systems so that they respond to the three major IP protocols: Transmission Control Protocol (TCP), User Datagram Protocol (UDP) and Internet Control Message Protocol (ICMP).

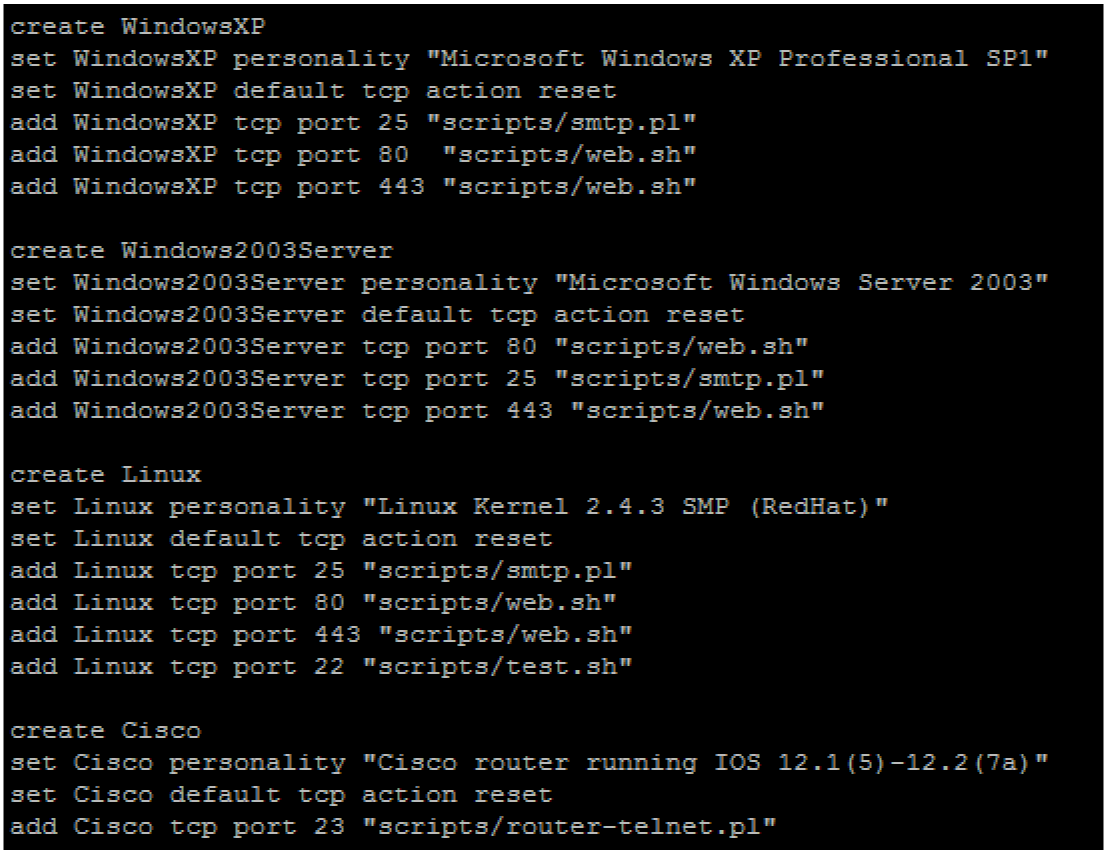
- By design, it reacts to network packet whose destination IP belong to one of the virtual honeypot network. Honeyd also support tunneling of network that allows simulation of address spaces that are topologically dispersed and load sharing.

Honeyd tool is used to simulate the honeypot on the network and to interact with the attacker. The platform used for Honeyd is VM Ware on Ubuntu 16.04.

Honeyd is simulated for various ports. It is also simulated for various servers like FTP server, SSH server etc. When a remote system is trying to connect to the ftp server, it asks for user name and password to connect to ftp server. Similarly remote host is also captured when trying to connect to SSH server. Attacker assume that he is connected with real server but in reality, he is captured by the honeypot It shows the activities of attacker like source IP, source port, destination IP and destination port. When the nmap scans the IP given to the virtual honeypot, it shows the port open.

3.1.1 Honeyd Installation and simulation

A virtual machine is deployed on esxi vm. Esxi vm is type1 bare metal (without running an OS) hypervisor that means it has direct access to the hardware it is running on and the virtual machines running directly on the top of the hypervisor. Honeyd is installed on that virtual machine which is deployed on the esxi vm. After installation Honeyd is simulated for various ports like TCP, UDP and ICMP. It is also simulated for various services like FTP, HTTP and SSH.



```
create WindowsXP
set WindowsXP personality "Microsoft Windows XP Professional SP1"
set WindowsXP default tcp action reset
add WindowsXP tcp port 25 "scripts/smtp.pl"
add WindowsXP tcp port 80  "scripts/web.sh"
add WindowsXP tcp port 443 "scripts/web.sh"

create Windows2003Server
set Windows2003Server personality "Microsoft Windows Server 2003"
set Windows2003Server default tcp action reset
add Windows2003Server tcp port 80 "scripts/web.sh"
add Windows2003Server tcp port 25 "scripts/smtp.pl"
add Windows2003Server tcp port 443 "scripts/web.sh"

create Linux
set Linux personality "Linux Kernel 2.4.3 SMP (RedHat)"
set Linux default tcp action reset
add Linux tcp port 25 "scripts/smtp.pl"
add Linux tcp port 80 "scripts/web.sh"
add Linux tcp port 443 "scripts/web.sh"
add Linux tcp port 22 "scripts/test.sh"

create Cisco
set Cisco personality "Cisco router running IOS 12.1(5)-12.2(7a)"
set Cisco default tcp action reset
add Cisco tcp port 23 "scripts/router-telnet.pl"
```

Fig 3.2 Honeyd Configuration for threats

Description of simulation parameters is given below:

- **TCP:** It is a protocol standard that defines how to maintain and establish network conversations via which application programs exchange data. It also defines how computer send packets of data to each other.

- **UDP:** It is part of the internet protocol suite and is used by programs running on different computers on a network. It is used to send short messages but it is an unreliable connectionless protocol.

- **ICMP:** It is an extension to the internet protocol defined in RFC 792. It supports packets that contain error, control and informational messages. It is used by network devices to generate messages that contain error when there are problems delivering IP packets.

- **IGMP:** It is a protocol that allows a host to advertise its multicast group membership to neighboring switches and routers. It is used by the TCP/IP protocol suite to achieve dynamic multicasting.

- **FTP:** It is a protocol used to transfer files between client and server on computer network.

- **SSH:** It is a protocol used for operating network services securely over an unsecured network.

For simulation a configuration file is created in the honeypot. In this configuration files various ports and services are simulated. When scanning of these services is done by n-map, it shows positive results. Honeyd is simulated for almost 30 days.

3.2 KFSENSOR

KFSensor is a window-based honeypot system that is designed to detect and attract hackers and worms by simulating various services of susceptible systems. KFSensor is preconfigured to scrutinize all TCP and UDP ports, along with ICMP. It is also configured with the emulation of common services. KFSensor perform monitoring right after the installation and can be easily modified to add extra services later on. KFSensor is basically designed for use in a window based corporate environment.

KFSensor is easy to install and configure. It takes just five minutes to set up and become operational. No special hardware is required and its efficient design enables it to run even on low specification Windows machines. Its straight forward Windows interface controls all functionality.

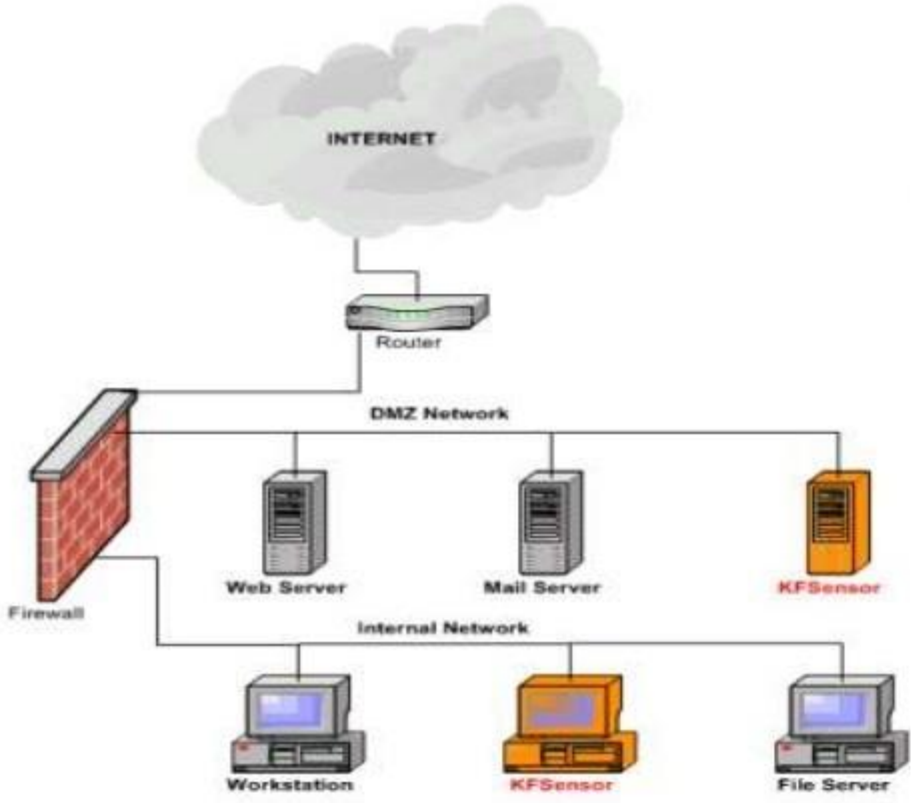


Fig 3.3 KFSensor Network

KFSensor works by simulating systems services at the highest level of the OSI Network Model - the application layer. This enables it to make full use of Windows security mechanisms and networks libraries, reducing the risk of detection and compromise by not introducing additional drivers and custom IP stacks.

A machine running KFSensor can be treated as just another server on the network, without the need to make complex changes to routers and firewalls. KFSensor provides immediate benefits in revealing the nature and quantity of attacks on a network. By consolidating all the network traffic of an attack into a single alert KFSensor makes it easy to explain a security threat to non-specialist staff. The information KFSensor generates can be used to refine firewall rules and produce new signatures for network intrusion detection systems.

KFSensor is an extremely cost-effective way of enhancing network security infrastructure. Every byte of an attack is recorded in KFSensor's logs. Events can be assigned different colour coded severities, making it easy to spot anything unusual or serious. Custom reports can be defined and the log can be filtered to show just those from a certain port, protocol or source IP address.

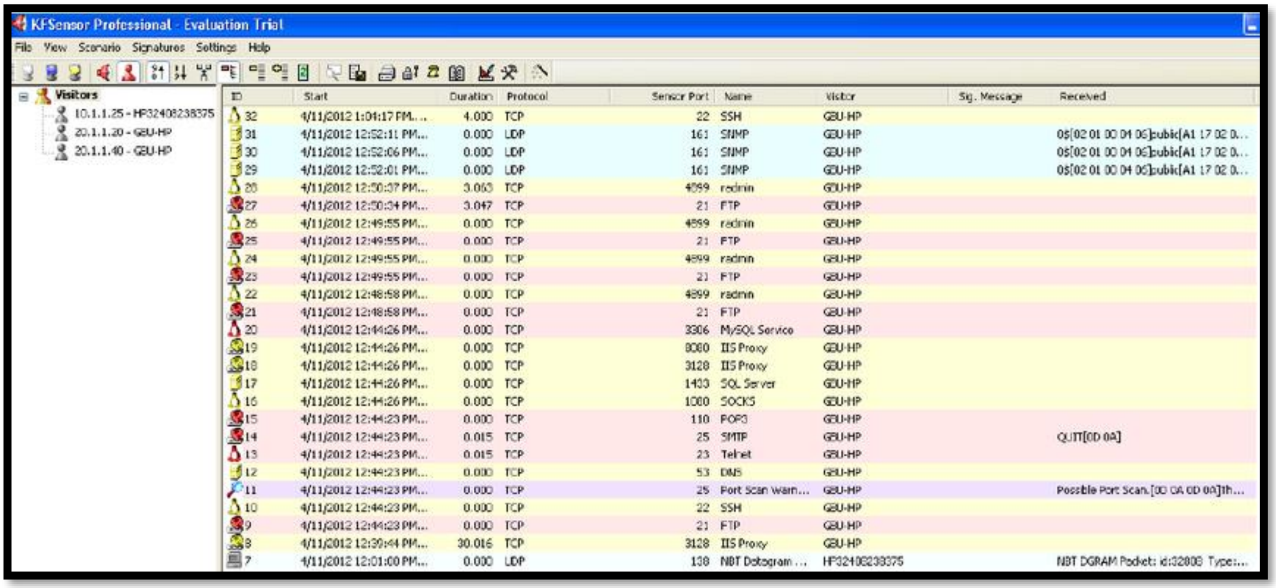
Features:

- It contains many unique and innovative features such as remote management, emulation of windows networking protocols and Snort compatible signature engine.

- KFSensor provides a cost-effective way of improving a network security of an organization with its GUI based management console, low maintenance and extensive documentation.
- KFSensor allow an easily integration with a common SIM/SOC & Syslog system & databases.

KFSensor honeypot is used on Windows platform for data analysis. The data collected from the KFSensor we analyze the most attacked port in system and also the type of Trojan captured by it.

3.2.1 KFSensor Installation and simulation


KFSensor Professional - Evaluation Trial
File View Scenario Signatures Settings Help

Visitors
- 10.1.1.25 - HP32408238375
- 20.1.1.20 - GBU-HP
- 20.1.1.40 - GBU-HP

ID	Start	Duration	Protocol	Sensor Port	Name	Visitor	Sig. Message	Received
32	4/11/2012 1:04:17 PM...	4.000	TCP	22	SSH	GBU-HP		
31	4/11/2012 12:52:11 PM...	0.000	UDP	161	SNMP	GBU-HP		0$[02 01 00 04 06]public[A1 17 02 0...
30	4/11/2012 12:52:06 PM...	0.000	UDP	161	SNMP	GBU-HP		0$[02 01 00 04 06]public[A1 17 02 0...
29	4/11/2012 12:52:01 PM...	0.000	UDP	161	SNMP	GBU-HP		0$[02 01 00 04 06]public[A1 17 02 0...
28	4/11/2012 12:50:37 PM...	3.063	TCP	4899	radmin	GBU-HP		
27	4/11/2012 12:50:34 PM...	3.047	TCP	21	FTP	GBU-HP		
26	4/11/2012 12:49:55 PM...	0.000	TCP	4899	radmin	GBU-HP		
25	4/11/2012 12:49:55 PM...	0.000	TCP	21	FTP	GBU-HP		
24	4/11/2012 12:49:55 PM...	0.000	TCP	4899	radmin	GBU-HP		
23	4/11/2012 12:49:55 PM...	0.000	TCP	21	FTP	GBU-HP		
22	4/11/2012 12:48:58 PM...	0.000	TCP	4899	radmin	GBU-HP		
21	4/11/2012 12:48:58 PM...	0.000	TCP	21	FTP	GBU-HP		
20	4/11/2012 12:44:26 PM...	0.000	TCP	3306	MySQL Service	GBU-HP		
19	4/11/2012 12:44:26 PM...	0.000	TCP	8080	IIS Proxy	GBU-HP		
18	4/11/2012 12:44:26 PM...	0.000	TCP	3128	IIS Proxy	GBU-HP		
17	4/11/2012 12:44:26 PM...	0.000	TCP	1433	SQL Server	GBU-HP		
16	4/11/2012 12:44:26 PM...	0.000	TCP	1080	SOCKS	GBU-HP		
15	4/11/2012 12:44:23 PM...	0.000	TCP	110	POP3	GBU-HP		
14	4/11/2012 12:44:23 PM...	0.015	TCP	25	SMTP	GBU-HP		QUIT[0D 0A]
13	4/11/2012 12:44:23 PM...	0.015	TCP	23	Telnet	GBU-HP		
12	4/11/2012 12:44:23 PM...	0.000	TCP	53	DNS	GBU-HP		
11	4/11/2012 12:44:23 PM...	0.000	TCP	25	Port Scan Warn...	GBU-HP		Possible Port Scan.[0D 0A 0D 0A]th...
10	4/11/2012 12:44:23 PM...	0.000	TCP	22	SSH	GBU-HP		
9	4/11/2012 12:44:23 PM...	0.000	TCP	21	FTP	GBU-HP		
8	4/11/2012 12:30:44 PM...	30.016	TCP	3128	IIS Proxy	GBU-HP		
7	4/11/2012 12:01:00 PM...	0.000	UDP	138	NBT Datagram ...	HP32408238375		NBT DGRAM Packet: id:32808 Type:...

Fig 3.4 KFSensor Configuration

KFSensor is installed on Window system. It is easier to install. KFSensor simulate vulnerable system services and Trojans and act as a honeypot to attract and identify attackers and worms. KFSensor is simulated for one month.

3.3 DANGERS OF USING A HONEYPOT:

Running a honeypot is not something that should be underestimated- there are some dangers one must be aware of which basically are:

a) Unnoticed takeover of the honeypot by an attacker.

b) Lost control over the honey pot installation.

c) Damage done to third party.

3.4 BENEFITS OF HONEYPOTS:

- **Small Data sets -** Honeypots only collect attack or unauthorized activity, dramatically reducing the amount of data they collect. Organizations that may log thousands of alerts a day may only log a hundred alerts with honeypots. This makes the data honeypots collect much easier to manage and analyze.

- **Reduced False Positives -** Honeypots dramatically reduce false alerts, as they only capture unauthorized activity.

- **Catching False Negatives -** Honeypots can easily identify and capture new attacks never seen before.

- **Minimal Resources -** Honeypots require minimal resources, even on the largest of networks. This makes them an extremely cost effective solution.

- **Encryption -** Honeypots can capture encrypted attacks.

3.5 LIMITATIONS OF HONEYPOTS:

- **Single Data Point -** Honeypots all share one huge drawback; they are worthless if no one attacks them. They can accomplish wonderful things, but if the attacker does not sent any packets to the honeypot, the honeypot will be blissfully unaware of any unauthorized activity.

- **Risk -** Honeypots can introduce risk to your environment. Different honeypots have different levels of risk. Some introduce very little risk, while others give the attacker entire platforms from which to launch new attacks, Risk is variable, depending on how one builds and deploys the honeypot.

CONCLUSION

A honeypot is just a tool. How to use that tool is up to you. Honeypots, like all technologies have their drawbacks, the biggest being their limited field of view. Honeypots only capture activity directed against them and will miss attacks on other systems. For that reason, security experts do not recommend replacing existing security technologies with those systems. Instead, they see honeypots as a complementary technology to protect against network-and host-based intrusion.

Lastly about the implementation done on the two tools of honeypot- Honeyd and KFSensor concludes that the data on Linux honeypot is captured in both secured and unsecured network. Various suspicious IP are detected in Honeyd whereas KFSensor shows the IP of only the network in which KFSensor is deployed. Overall honeypots are an effective tool for detecting hackers and shutting them down before they can do any major damage to a system.

FUTURE WORK

The growing pace of internet and more and more devices getting connected to the internet has increased the threat to the network security. This increasing threat mandates the activity monitoring of the malicious activities and security bugs. With this growing internet usage, a better and more advanced Honeypots are required. In the field of network security, honeypots can be used as a guard to identify malicious activities occur on the network. In future a better protective network can be built by using the analysis of the current study of Honeyd and KFSensor, and also besides Honeyd can also be simulated for more services to test the network, for both, research and intrusion detection.

REFERENCES

1. Bhagat, Neeraj, and Bhavna Arora. "Intrusion detection using honeypots." *2018 Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC)*. IEEE, 2018.
2. **https://www.giac.org/paper/gsec/97/honey-pots-intrusion-detection/100494 as cited on 10th June 2021.**
3. **https://www.geeksforgeeks.org/introduction-of-firewall-in-computer-network/ as cited on 10th June 2021.**
4. **https://www.geeksforgeeks.org/intrusion-detection-system-ids/ as cited on 10th June 2021.**
5. **https://hide.me/en/blog/what-is-honeypot/ as cited on 10th June 2021.**
6. **https://www.imperva.com/learn/application-security/honeypot-honeynet/ as cited on 10th June 2021.**
7. **https://searchsecurity.techtarget.com/definition/honey-pot as cited on 10th June 2021.**
8. **http://www.123seminarsonly.com/CS/Honeypots.html as cited on 10th June 2021.**
9. **http://www.diva-portal.org/smash/get/diva2:327476/fulltext01 as cited on 10th June 2021.**