Assignment 2

**Federation University Australia at IIBIT (Sydney)**

**Sydney Campus**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ITECH** | **3** | **2** | **1** | **5** |

**COURSE NUMBER :**

**STUDENT’S GIVEN NAMES: KANCHAN SHRESTHA**

**STUDENT NUMBER FED UNI: 30344274**

**COURSE NAME: INFORMATION SECURITY**

**PROGRAM OF STUDY: Bachelor of Information Technology**

**TUTOR: ADNAN SAEED**

**TITLE OF ASSIGNMENT:** Securing system using IPTable firewall

**DUE DATE: week 8 DATE SUBMITTED: 7 May**

**DECLARATION:**

I have kept a copy of this assignment/work so that I can produce it if the original is lost or damaged. I hereby certify that no part of this assignment/work has been copied from any other student’s present or previous, published or unpublished, professional or amateur work or from any other source except where due acknowledgement is made in the assignment. I further certify that no part of this assignment/work has been written/produced for me by any other person except where such collaboration has been authorized by the unit/subject lecturer/tutor concerned.

KANCHAN SHRESTHA

----------------------------------------------------------------

Signature of Student

**Note**: It is necessary to sign the above declaration. A lecturer/tutor or an examiner reserves the right not to mark this assignment/work if the declaration has not been duly signed.

Table of Contents

1. [QUESTION 3](#_Toc39775404)

[1.1 Reject all incoming and outgoing ping packets. 3](#_Toc39775405)

[1.2. Reject all incoming telnet packets and allow all outgoing telnet packets. 4](#_Toc39775406)

[1.3. Reject all traffic coming to MySQL server. 5](#_Toc39775407)

[1.4. Block incoming packets to the IP address of your virtual machine. 6](#_Toc39775408)

[1.5. Allow packets inbound to port 80 (inbound) and reject packets going out (outbound) through port 80 7](#_Toc39775409)

1. [EXPERIMENT SETUP IN KALI 8](#_Toc39775410)
2. [FIREWALL 10](#_Toc39775411)
3. [IPTABLES 11](#_Toc39775412)

[4.1 Tables 11](#_Toc39775413)

[4.2 Chains 11](#_Toc39775414)

[4.3 Rules 12](#_Toc39775415)

[4.4 Using Iptables command 12](#_Toc39775416)

1. [Advantages of firewalls with Iptables 13](#_Toc39775417)
2. [Disadvantages of firewalls with Iptables 13](#_Toc39775418)
3. [Ways to overcome the disadvantages 14](#_Toc39775419)

# QUESTION

## 1.1 Reject all incoming and outgoing ping packets.

A screenshot of a cell phone

Description automatically generated

Here I am using left computer as a server and I have rejected all incoming and outgoing ping packets.

PROOF:

A screenshot of text

Description automatically generated

## 1.2. Reject all incoming telnet packets and allow all outgoing telnet packets.

A screenshot of a computer

Description automatically generated

PROOF:

A screenshot of text

Description automatically generated

## 1.3. Reject all traffic coming to MySQL server.

Screenshot

A screen shot of a computer

Description automatically generated

PROOF:

A screenshot of a cell phone

Description automatically generated

## 1.4. Block incoming packets to the IP address of your virtual machine.

Screenshot

A screenshot of text

Description automatically generated

PROOF:  
A screenshot of a cell phone

Description automatically generated

## 1.5. Allow packets inbound to port 80 (inbound) and reject packets going out (outbound) through port 80

Screenshot:

A black and silver text on a screen

Description automatically generated

PROOF:

A screen shot of a computer

Description automatically generated

# EXPERIMENT SETUP IN KALI

I have used Kali Linux disc image provided from my tutor. In this version of kali Linux Telnet, Apache web server and mysql are pre-installed. I am using Oracle virtual box to import the Kali linux. It is a free version and easy to use. For this assignment. I made a virtual Kali Linux creating a virtual hard disk of 50 GB and memory of 4 GB. Basically I will be working on terminal so I don’t need extra amount of resources for kali. I made another virtual kali environment by cloning the previous OS. So there are two OS now, one as a server and other as a client or other computer. They are two computers having different network address such as IP address and distinguished identity. The details about how I have tested in parts 1 -5 are listed below:

1. For rejecting incoming and outgoing ping packets I have put two active kali machine side by side and first I checked the IP address of both the computer using ifconfig command then ping the other computer, its pinging and working like a normal. After I applied the command for the above task, it could not ping the ip address and from the other machine also did not succeed with the message showing destination port unreachable.

Commands used

|  |  |
| --- | --- |
| Sudo iptables -A INPUT -p icmp –icmp-type 8 -s 0/0 -j REJECT | Reject incoming ping packets |
| Sudo iptables -A OUTPUT -p icmp –icmp-type 8 -s 0/0 -j REJECT | Reject outgoing ping packets |
| Ping 192.168.x.x | Test ping request |

1. For rejecting all incoming telnet packets and accepting outgoing telnet packets. In this part, I have set up the environment similar to question 1. I used command to start the Telnet server and checked the status of my network. Telnets use port number 23 so that I used the command to block the outgoing packets in port 23 and then to see the result I try to request the Telnet server to reply from another Linux showing remote connection refuse and then I used similar command to allow the outgoing Telnet packets.

Commands Used

|  |  |
| --- | --- |
| Sudo /etc/init.d/xinetd restart | Restart the telnet server |
| Iptables -A INPUT -p tcp -m tcp –dport 23 -j REJECT | REJECT all incoming telnet packets |
| Iptables -A OUTPUT -p tcp -m tcp –dport 23 -j ACCEPT | ACCEPT all outgoing telnet packets |
| Iptables -L -n -v | to show the status to reject packets |
| telnet 192.168.x.x | Test telnet connection |

1. For rejecting all traffic coming to mysql server first started the apache server and mysql server then to reject all coming traffic i implemented the similar command like in question2 but to the port number 3306 in which mysql listens. Then to test the connectivity from the other computer using the username and password to get permission to access to the mysql server then the connection couldnot be performed because the following command was executed successfully.

Commands Used

|  |  |
| --- | --- |
| Sudo /etc/init.d/mysql start | Start mysql |
| Sudo /etc/init.d/apache2 start | Start apache server |
| Iptables -A INPUT -p tcp -m tcp –dport 3306 -j REJECT | Reject all traffic coming to MySQL server |
| Mysql -u username -p password -h IP address | Test connectivity to mysql server |

1. To block the incoming packets to ip address of virtual machine, I got the mac address of the network interface and then used the iptables rule. -mac-source is to specify the mac address

Commands used:

|  |  |
| --- | --- |
| Iptables -I INPUT -m mac –mac-source 08 :00 :27 :0Ee :ba :f1 -j REJECT | Block all incoming packets |
| Iptables -L | Show the status of the command |

1. To allow packets inbound to port 80 by To view the result iptables-L and it shows the implementation of the above rule. INPUT is used to specify the incoming packets wheras -dport 80 to specify the port number and then ACCEPT is to allow the packets, similarly OUTPUT specifies the outgoing packets and REJECT is to reject the packets

Commands used

|  |  |
| --- | --- |
| Sudo /etc/init.d/apache2 start | Start apache server |
| Iptables -A INPUT -p tcp –dport 80 -j ACCEPT | Allow packets inbound to port 80 |
| Iptables -A OUTPUT -p tcp –dport 80 -j REJECT | reject packets going out (outbound) through port 80 |
| Iptables -L | Show the status of the above commands |

# FIREWALL

A firewall is a network security device that works as a filter between computer network and the internet by allowing or blocking the networked traffic. Firewall performs task like Defending Resources, Validating access, managing and controlling network traffic , recording and reporting on events. Firewall is designed based on the concept that all traffic should pass through firewall by physically blocking all access to the local network and only the authorised traffic that is defined by the security policy is allowed to pass through firewall. (J Frahim, 2015) There are several methods the firewall uses to filter the information and specific to how it works at different layers of a network. The techniques that are used by firewall to control access are:

|  |  |
| --- | --- |
| Service control | Determine the type of internet service Inbound and Outbound |
| Direction control | Determine in which direction to allow a service |
| User control | Control access to a service by authorized users |
| Behaviour control | Control how particular services are used. |

**Firewall Visualisation**A close up of a map

Description automatically generated

Image source: <https://www.geeksforgeeks.org/types-of-firewall-and-possible-attacks/>

# IPTABLES

Iptable simply is a tool for managing firewall rules on a Linux based operating system. It provide interface to work with packet filtering framework on Linux kernel called Netfilter Iptables monitor traffic coming to user and to server using tables that contains set of rules called chains. Chains are the set of rules that will filter incoming and outcoming data packets. The three main terms that we need to understand while using Iptables are Tables, Chain and Rules. They are explained as follows: (Lucas, 2017)

## 4.1 Tables

There are three main tables on Iptables they are

|  |  |
| --- | --- |
| Table Name | Action |
| Filter (Default) | Packet Filtering |
| NAT (Network Added Translation) | Modify packets source & Destination Address |
| Mangle | Modify IP headers |

## 4.2 Chains

Iptables uses the policy chains to allow or block traffic. If there are no rules then it resorts to default action. Only the root user are privileged to operate and execute the iptables. System admin defines the tables containing rules how to treat the packets. The admin can create a desired amount of chains. There are a predefined chains that have a policy. The predefined chains with the policy are given below:

|  |  |
| --- | --- |
| Prerouting | The chains where packets are kept before the routing decisions are made |
| Input | Packet going to be locally delivered and are controlled by “local-delivery” |
| Forward | All packets that have been routed that were not locally delivered are included in this chain |
| Output | Packets sent from the machine itself falls under this chain |
| Postrouting | After the routing decision have been made, packet enter this chain exactly before passing to hardware. |

**Filter** is one of the default table of iptables that contains three chains they are

INPUT – Control incoming packets to the server

FORWARD – Filter incoming packet to forward to outside world

OUTPUT- Filter packet going out from your server

## 4.3 Rules

Rules are the command that are used to make changes in the Network Traffic

When the packet matches a rule, it is given a values that are:  
ACCEPT – Allow packet to pass through

DROP – Won’t let packet to pass through

RETURN – Stop packet from traversing through a chain and tell it to go back to previous chain

## 4.4 Using Iptables command

Iptables is pre-installed in linux however we can install or update it using the command:

**Sudo apt-get install iptables**

After that, to check the status of iptables

**Sudo iptables -L -v**

-L is used to list all the rules and -v is used to show information in detailed format.

At first when using the iptables there are no any rules for packets filtering so we can define the rules.

To define new rule we need to type following command

***Sudo iptables -A***

-A is to append or defining new rules

-I (interface)> Network interface of the target traffic eg. eth0, lo

-p (protocol) > Network protocol where filtering process is done eg. tcp, udp

-s (source) > Address of the incoming traffic eg. hostname or IPaddress

-dport (destination port) > Destination port number of protocol eg. 443(https)

-j (target) > Target name ( ACCEPT, DROP, RETURN)

The order of the command to write is:

***Sudo iptables -A <chain> -I <interface> -p <protocol>***

Example: ***sudo iptables -A INPUT -I lo -j ACCEPT***

To filter packets based on a particular IPaddress or IPadress range -s is used to specify

Example: sudo iptables -A INPUT -s 192.168.1.9 -j ACCEPT

For a range of IPaddresses -m option and iprange module should be used and -src-range to specify the IPaddress range

Example: sudo iptables -A INPUT -m iprange –src-range 192.168.1.10-192.168.1.100 -j DROP

To drop all other traffic after specifying the target traffic using -dport

Example: sudo iptables -A INPUT -j DROP

To save the command in order to use it after we restart

Example: sudo /sbin/iptables-save

To disable the the iptables

Example: sudo iptables -F

Sudo /sbin/iptables-save

So, in this way we can use iptables as a firewall security tool to secure Linux server or Network by defining our own set of rules and desires.

# Advantages of firewalls with Iptables

1. Iptables allow access to the lowest level to the linux kernel firewall
2. It provides administrator with an interface to add, modify and remove packet rules
3. We can define our own custom rules or we can jump from one chain to another in firewall.
4. It is versatile tool and flexible using command line to control
5. The concept of the iptables firewall is easy and useful
6. There are a lot of documentation about the iptables firewall which is easily available
7. There are a default activities in the Iptables if the rules donot match (Rick, 2019)

# Disadvantages of firewalls with Iptables

1. Iptables while using at first have the setting to ‘allow’ everything and making changes to it is like installing new security software without having to do anything in it
2. It is easy to mess up things once we screw up the remotely setup firewall system
3. Unable to protect from the attack bypassing the firewall
4. Execution time is low for the high packet rates
5. It donot protect firewall against the malware infected from the outside source
6. Iptables uses linear processing which increase the cost as the number of packets increase
7. There is a flaw in the iptables-restore which donot function properly with the iptables-save command.
8. It is easy to exploit if any hacker is observing the data patterns. (Wick, 2020)

# Ways to overcome the disadvantages

1. Developers should give more updates on iptables by adding more features as there are number of users using iptables.
2. There should be a version of iptables with a predefined set of features that user donot have to configure.
3. Another layer of security should be added to the application as it is the standalone security feature for system
4. The rules and guidance for configuring the network properly should be considered carefully to be safe
5. There should be a restriction in installing a parallel internet connection or dial up connection to install things needed from the main computer to avoid firewall limitation. (Wiliamson, 2019)

Bibliography

J Frahim, O. S. (2015). *ALl in one Firewall.* california: Pearson Education.

Lucas, M. (2017). *Firewall policies and VPN COnfiguration.* Natural Publication.

Rick, L. w. (2019, july). *IPtable tutorial*. Retrieved from Hostinger: https://www.hostinger.com/tutorials/iptables-tutorial

Wick, C. (2020). *Managing Iptables and Improving Network Security*. Retrieved from Tech Republic: https://www.techrepublic.com/article/linux-101-configuring-and-managing-iptables-to-improve-network-security/

Wiliamson, J. (2019). *Computing Security*. Retrieved from Bright Hub: https://www.brighthub.com/computing/smb-security/articles/62024.aspx