Subankamo 300471606

Lab 4 - Configuring a Packet Filtering Firewall [Total 4 Marks]

Due Date – 21/05/2023

- Follow the instructions in this document,
- Answer the questions in the order they appear in this document and in the labs included in this document (See each lab's document for more instructions)
- Submit the file (.doc, .docx, .pdf) using the ecs submission system (i.e. Lab4).

  https://apps.ecs.vuw.ac.nz/submit/CYBR371

Instructions

Part A - Netlab [2 Marks Total]

Login into netlab at netlab.ecs.vuw.ac.nz, complete the following labs and answer the questions accordingly:

- **Configuring a Network-Based Firewall**
- **Testing Firewall Rules with Firewalking**

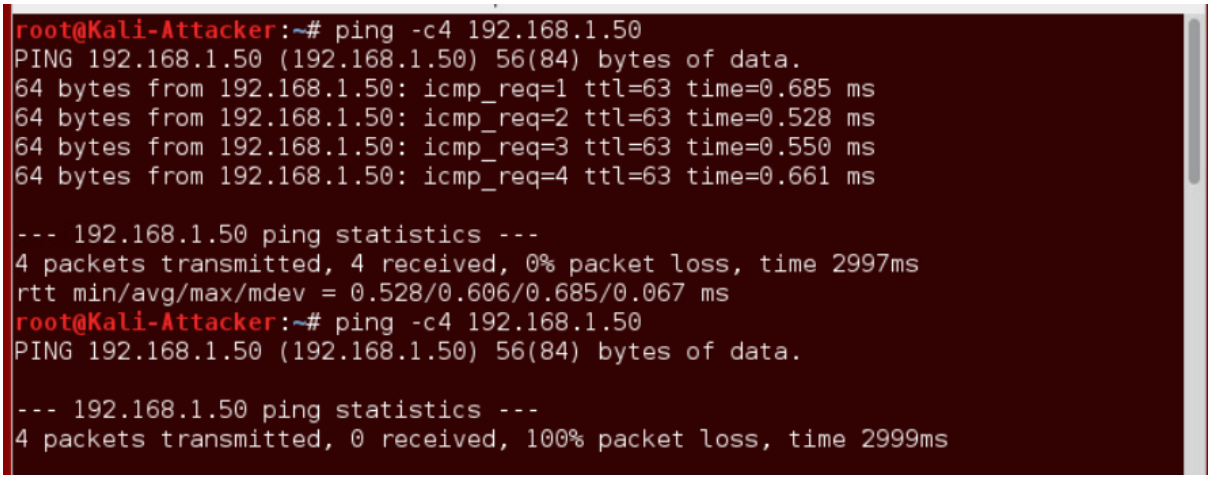
Questions (2 Marks Total):

1. **Question 1 - What is Firewalking (the technique and/or the tool) and how can it be used in an attack process against a potential target? [Max 250 words] [1 Mark]**

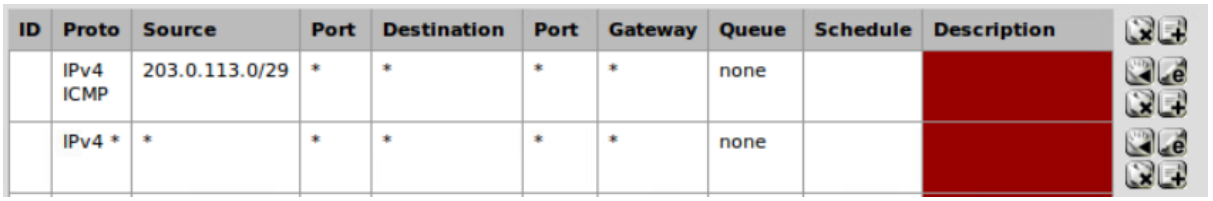
Firewalking is a technique used in network security to determine the status of network ports on a target system. It involves sending packets with varying TTL values to observe the response from the target. This information is valuable to attackers as it helps create a network map, identify services, and evaluate network security measures. Once open ports are identified, attackers can launch specific attacks. For example, if SSH port 22 is open, they can exploit SSH vulnerabilities or attempt brute-force attacks. Open web server ports can be assessed for web application vulnerabilities like SQL injection or XSS. Similarly, open RDP ports can be exploited using known RDP vulnerabilities or brute-force attacks on RDP credentials. Firewalking enables attackers to tailor their strategies and maximize the chances of success by focusing on the target's weaknesses. By understanding the target's network infrastructure and security measures, attackers can effectively plan further stages of an attack.

2. If pfsense a stateful or stateless firewall? Demonstrate the statefullless or the statelessness of pfsense firewall using the lab "Configuring a Network-Based Firewall". You may include a screenshot. [1 Mark]

pfSense is primarily a stateful firewall. It is designed to track the state of network connections and allows only authorized traffic based on the connection state. However, it also has the capability to operate in a stateless mode if needed.
In a stateful firewall, pfSense keeps track of the state of network connections by monitoring the traffic passing through it. It maintains a state table that records information about each connection, such as source IP, destination IP, ports, and connection status. This state information is used to allow inbound traffic for established connections and to permit outgoing traffic in response to those connections.



>first ping prior to setting firewall rule to block packets from 203.0.113.0/29
>second ping after setting firewall rule to block packets from 203.0.113.0/29



ID	Proto	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description
	IPv4 ICMP	203.0.113.0/29	*	*	*	*	none		
	IPv4 *	*	*	*	*	*	none		

Subankamo 300471606

Part B - Packet Filtering Firewalls [2 Marks Total]

Login into netlab at netlab.ecs.vuw.ac.nz, complete the following lab and answer the questions accordingly:

- **Configuring a Packet Filtering Firewall**

Questions (2 Marks Total)

- **Question 1 - We have the following rule in our iptables on the server (i.e. 192.168.1.1). The client (192.168.1.78) however fails to initiate an SSH connection. Explain why this is the case. [0.5 Mark]**

    - #sudo iptables -A INPUT -p tcp -s 192.168.1.78 -d 192.168.1.1 --dport 22 -i <name-of-your-interface, probably “enp0s3”> --j ACCEPT
    - #sudo iptables -A OUTPUT -p tcp -s 192.168.1.78 --sport 22 -d 192.168.1.1 --dport 22 ACCEPT
    - sudo iptables -P INPUT DENY
    - sudo iptables -P OUTPUT DENY

In the given rules, there are two rules allowing SSH traffic from the client (192.168.1.78) to the server (192.168.1.1) on port 22. However, these rules are appended to the INPUT and OUTPUT chains after the default DENY policy.
The issue with the given iptables rules is that the default policies for both INPUT and OUTPUT chains are set to DENY, which means all incoming and outgoing traffic is blocked unless explicitly allowed.
To fix change sudo iptables -P INPUT DENY and sudo iptables -P OUTPUT DENY to sudo iptables -P INPUT ACCEPT and sudo iptables -P OUTPUT ACCEPT So all incoming and outgoing traffic is allowed by default.

- **Question 2 - Our server is running a Telnet service listening on port 23. We would like to stop any new Telnet connections from a client with the IP address of 10.0.2.5. Is the following a good rule to block the Client? Explain. [0.5 Mark]**

    - #sudo iptables -A INPUT -s 10.0.2.5 -p tcp --sport 23 -j DROP

The given rule sudo iptables -A INPUT -s 10.0.2.5 -p tcp --sport 23 -j DROP is not an effective rule to block new Telnet connections from the client with the IP address 10.0.2.5. The rule tries to block incoming traffic from the client by specifying the source IP address as 10.0.2.5 and the source port as 23. However, the source port specified in the rule (--sport 23) is the port on the client side, not the server side.
Telnet connections use destination port 23 on the server side, not source port 23. Therefore, the rule as it is written will not block Telnet connections from the client IP address.
To effectively block new Telnet connections from the client IP address 10.0.2.5, you should modify the rule to specify the destination port as 23 (--dport 23), not the source port.

- **Question 3 - Write a rule to drop all the new and ESTABLISHED outgoing SSH service requests received on your primary interface (eth0) which has a source MAC address of 30:65:EC:22:14:D1 [0.5 Mark]**

sudo iptables -A OUTPUT -o eth0 -m mac --mac-source 30:65:EC:22:14:D1 -p tcp --dport 22 --match state --state NEW,ESTABLISHED -j DROP

- **Question 4 - Write a rule to drop any incoming packets with “INVALID” state. [0.5 Mark]**

sudo iptables -A INPUT -m state --state INVALID -j DROP