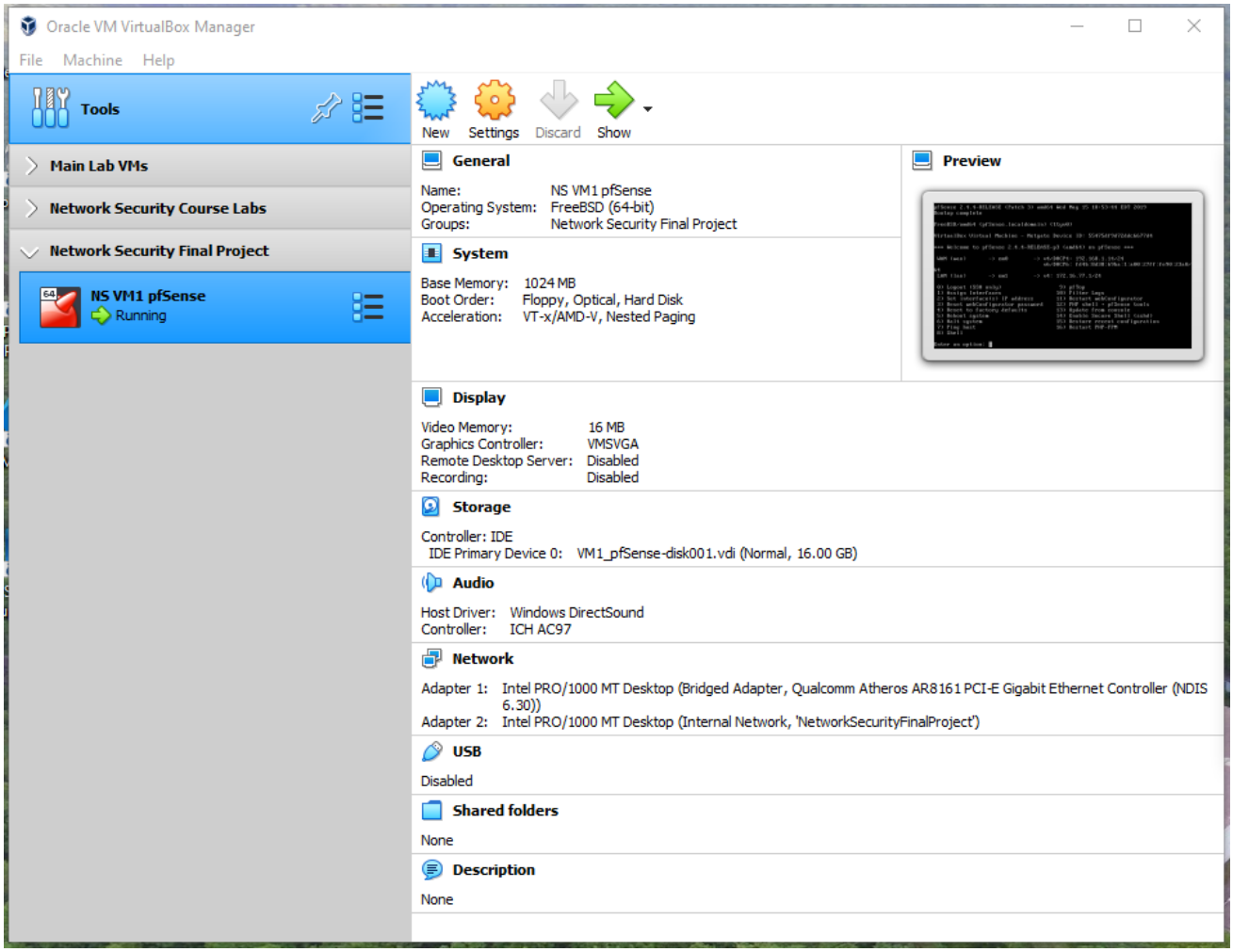
Christopher Dix – NJIT-CS-03 – January 2021
NS-10-L1: Secure Your Network

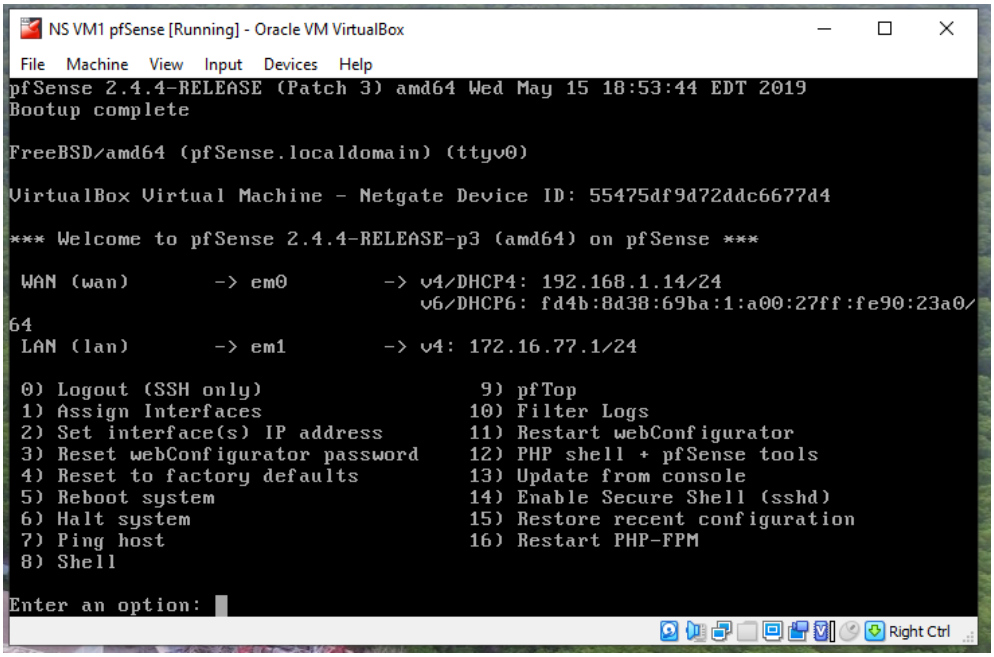
Project Background: Play the role of a Security Analyst to solve various challenges with a corporate environment's network security. (For example, users are still able to access some resources they should not be able to, and some inbound traffic that should be blocked is still allowed. Secure access for remote employees is also not set up.)

Virtual Environment Scenario and Setup:

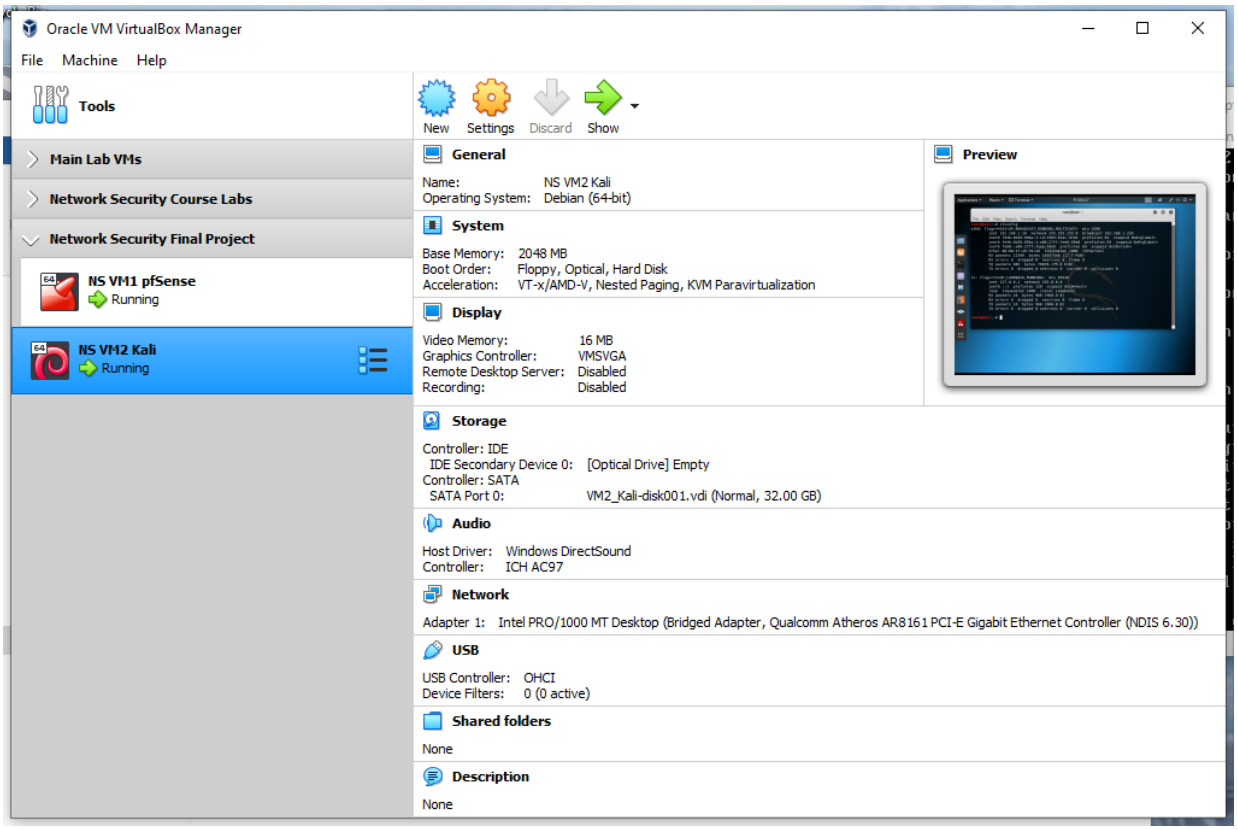
- **Host System:** The physical host (running the hypervisor) and its network connection will be used to mimic the Internet for this project. It will act as an external agent to access certain resources within the virtualized corporate network.

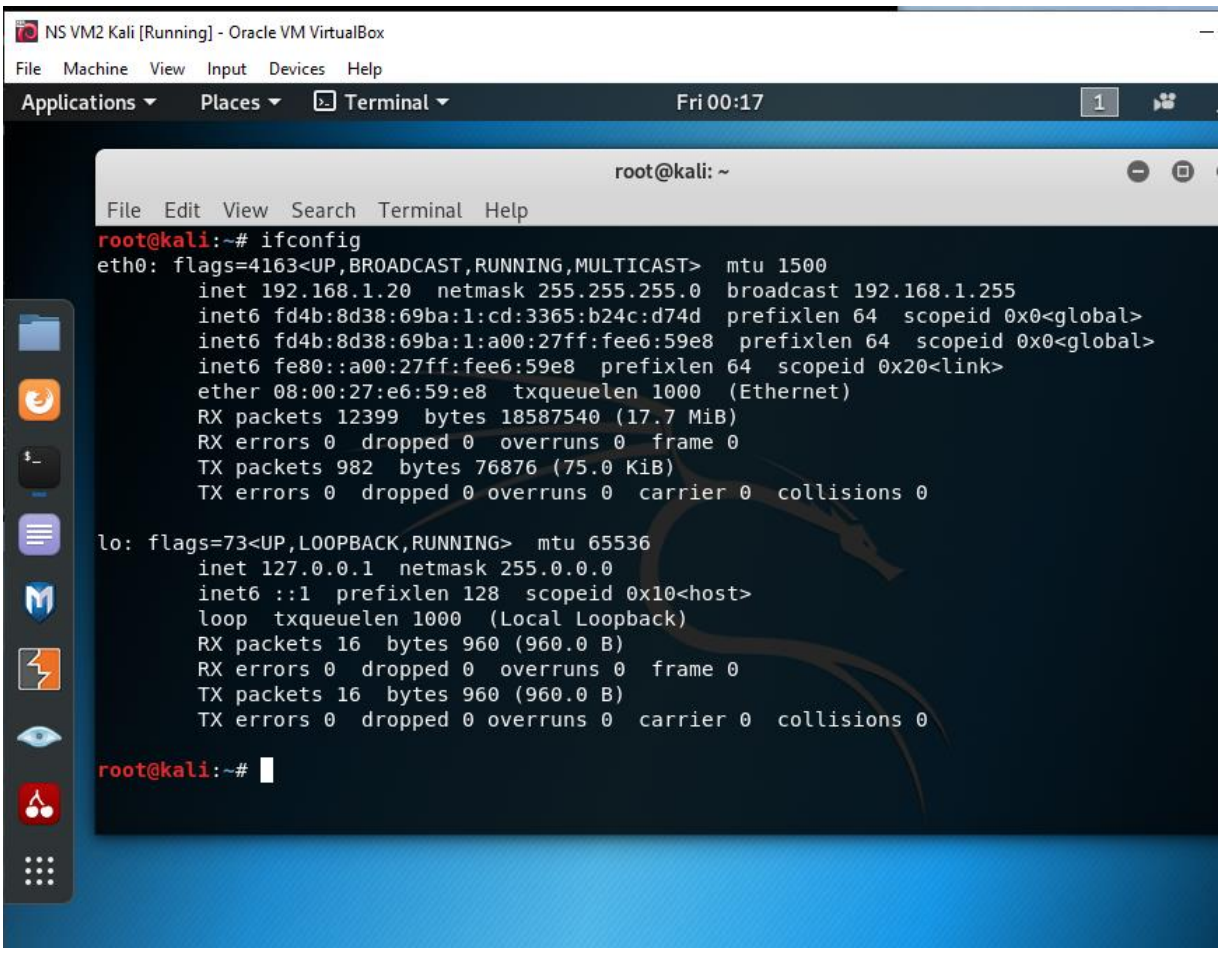
- **VM #1:** This virtual machine is a pfSense firewall. It separates the Internet from the internal scenario and works with NAT between the networks. The firewall is configured to get, on the WAN interface, an IP address from the same subnet as the physical host (DHCP client, bridged adapter in VirtualBox). For the LAN interface, pfSense acts as the DHCP server and the default gateway for the internal environment. pfSense also runs Suricata and OpenVPN for some tasks.



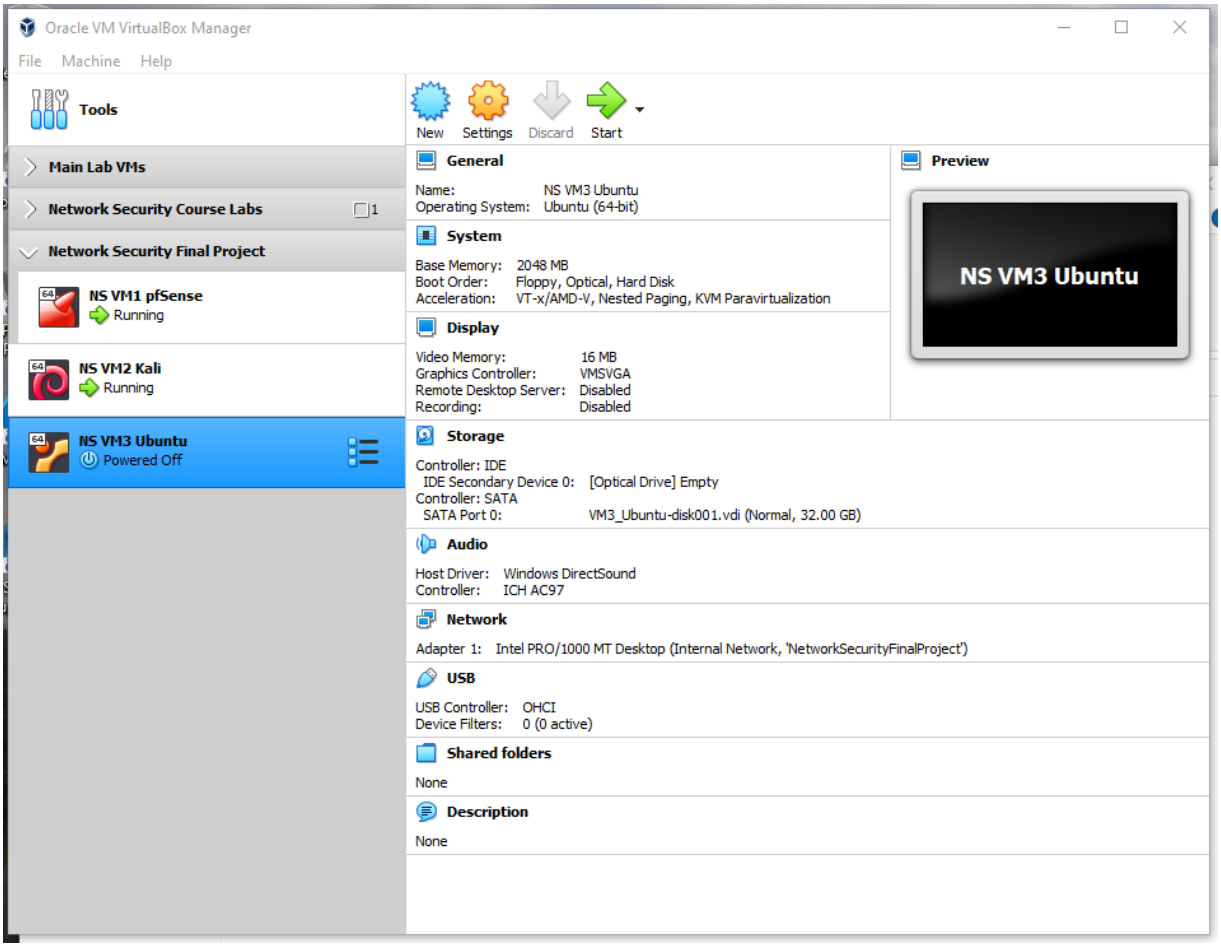


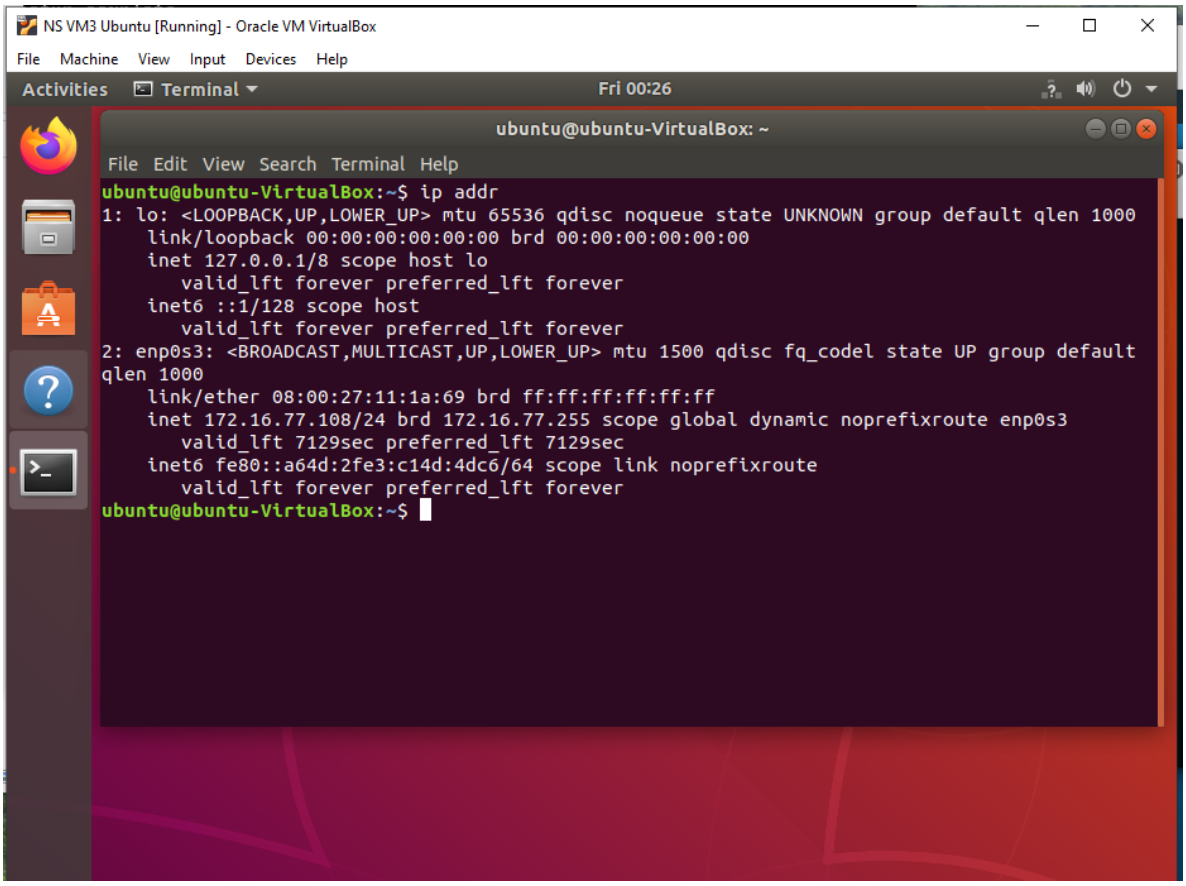
- **VM #2:** This virtual machine runs Kali Linux and will be used later in the project. It runs in bridged mode, meaning that it has an IP address on the same subnet as the physical host.



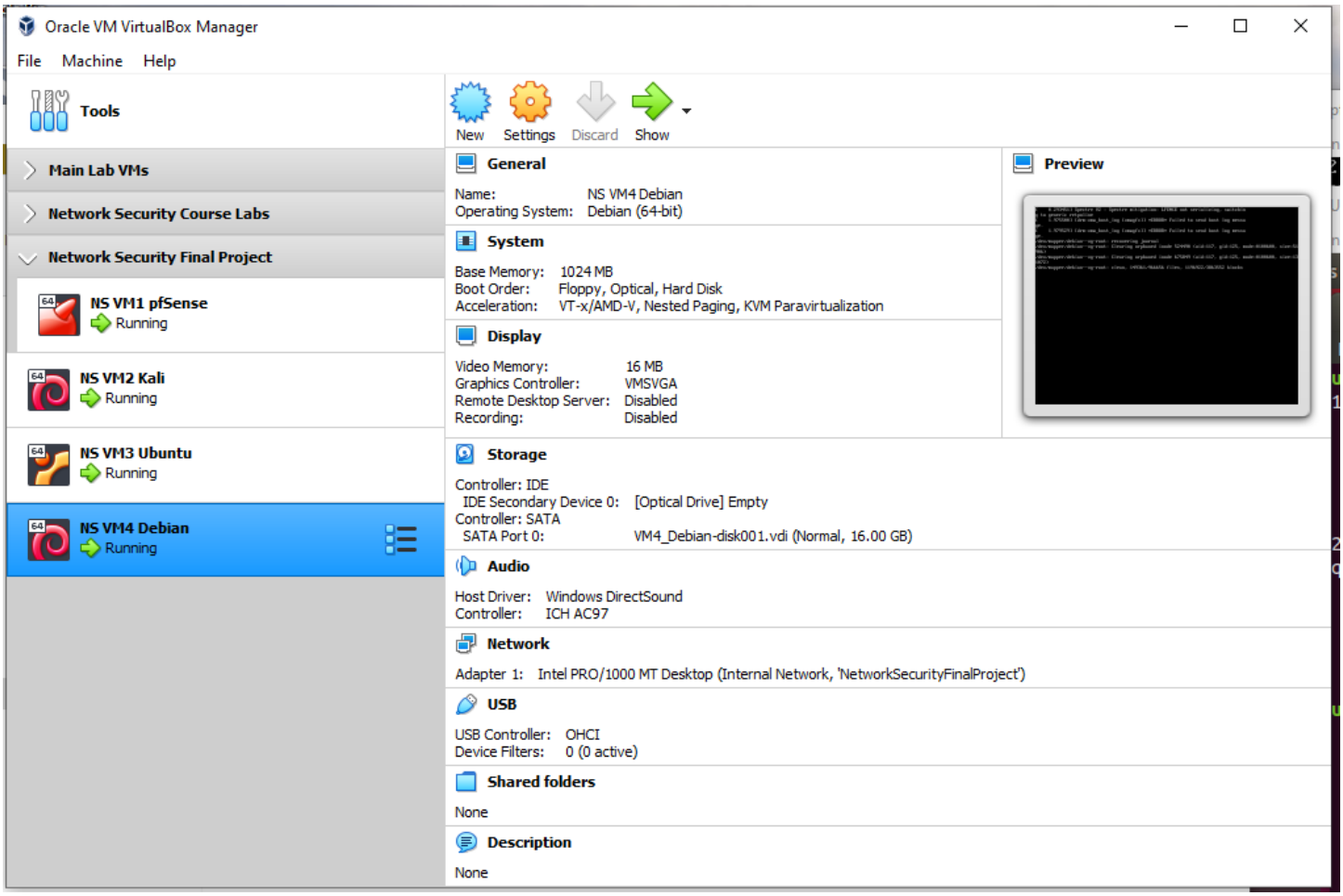


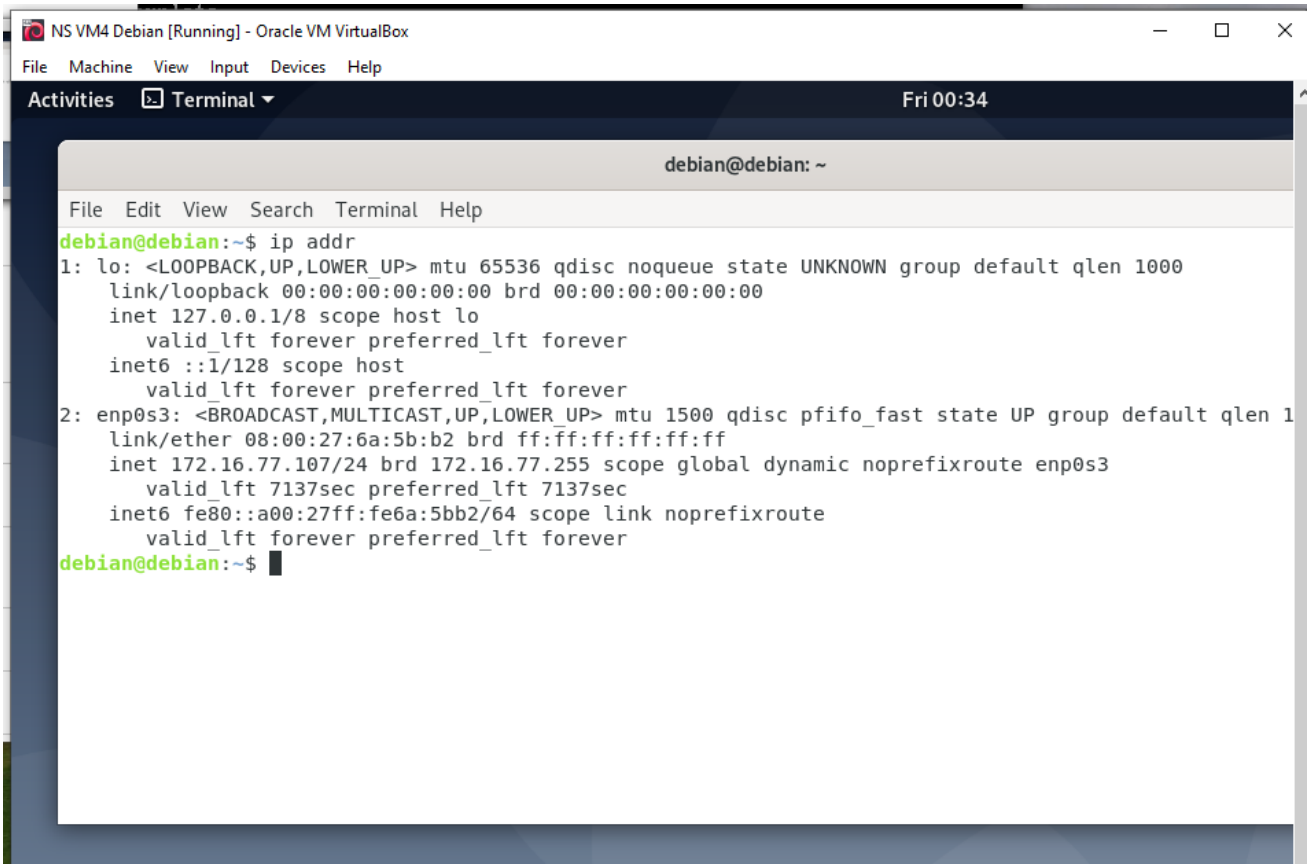
- **VM #3:** This virtual machine runs Ubuntu and represents servers in the corporate network. It hosts various services, including a web server that will simulate an internal web service for external access.



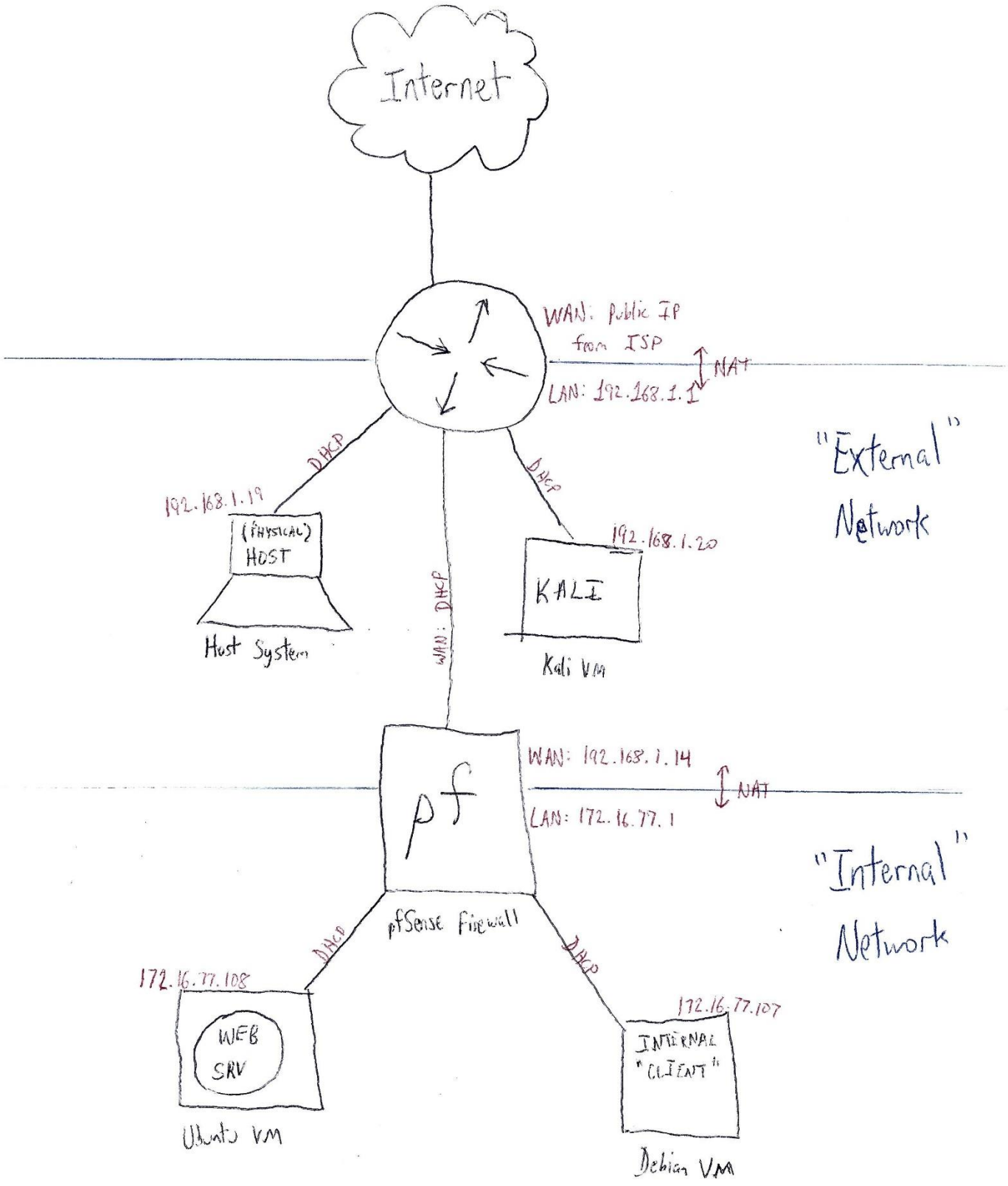


- **VM #4:** This virtual machine runs Debian and represents the internal network clients. It will be used for remote access tests, among other tasks, and is a DHCP client in the internal network.





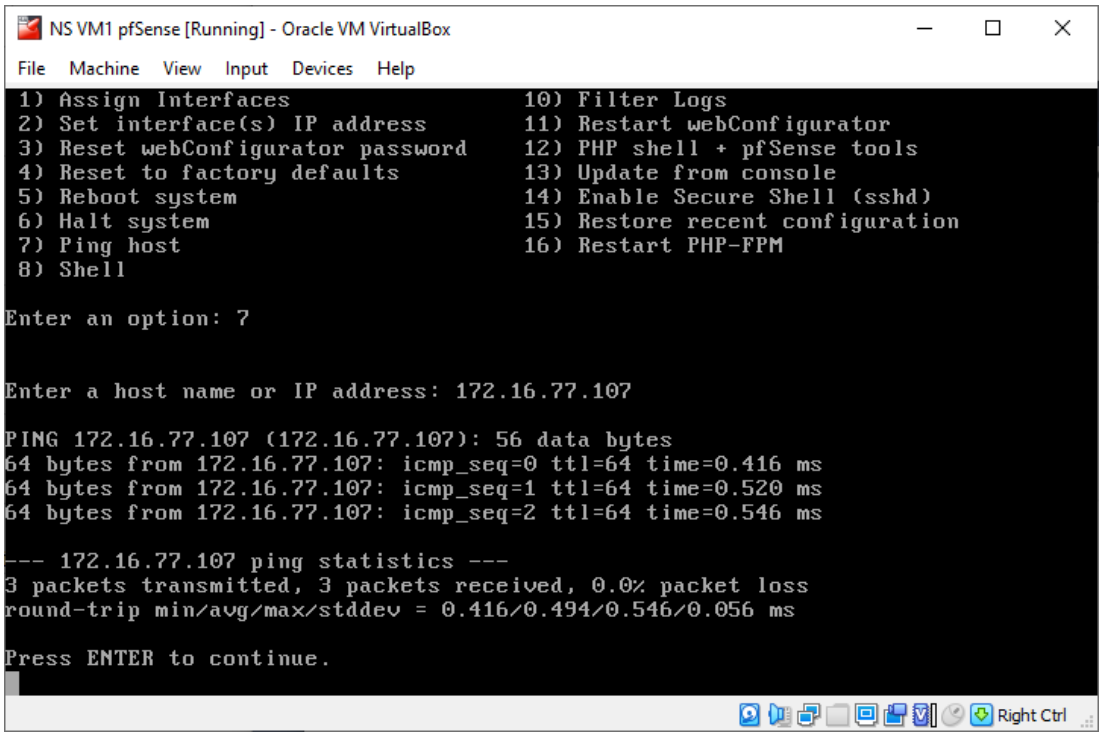
Network Diagram:



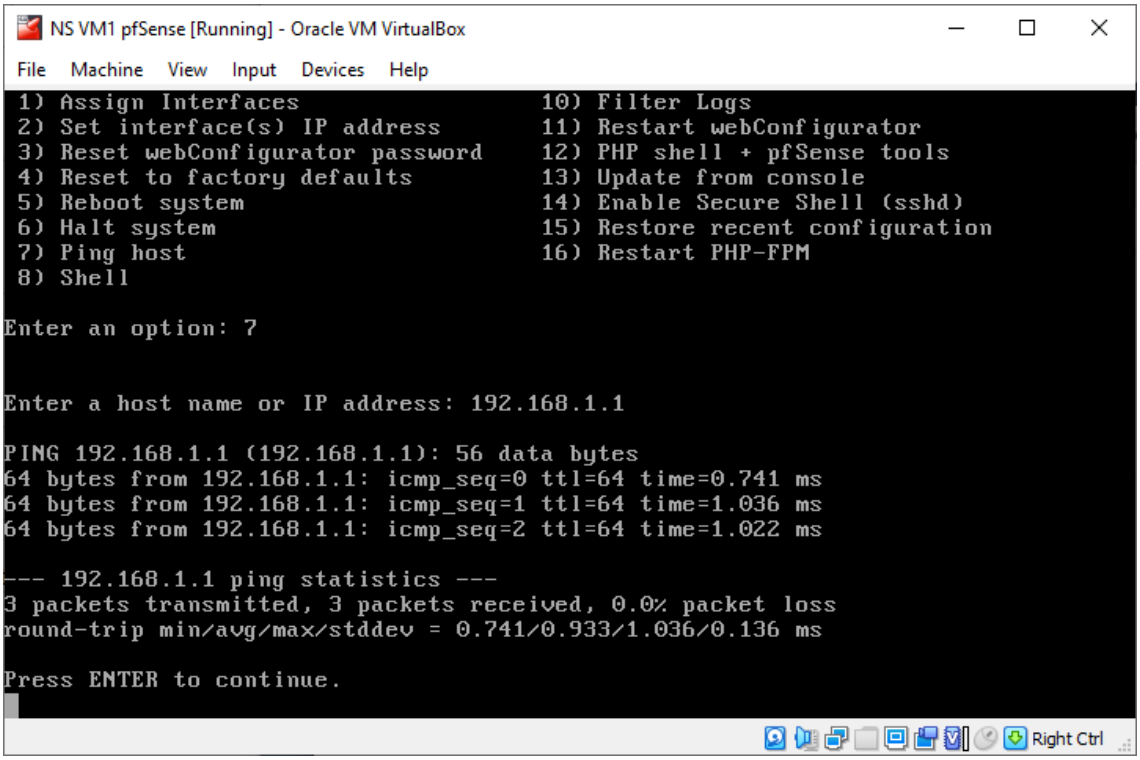
Scenario Validation:

1. IP addresses were verified to be correctly acquired/configured in the correct subnets, as shown in the above screenshots.

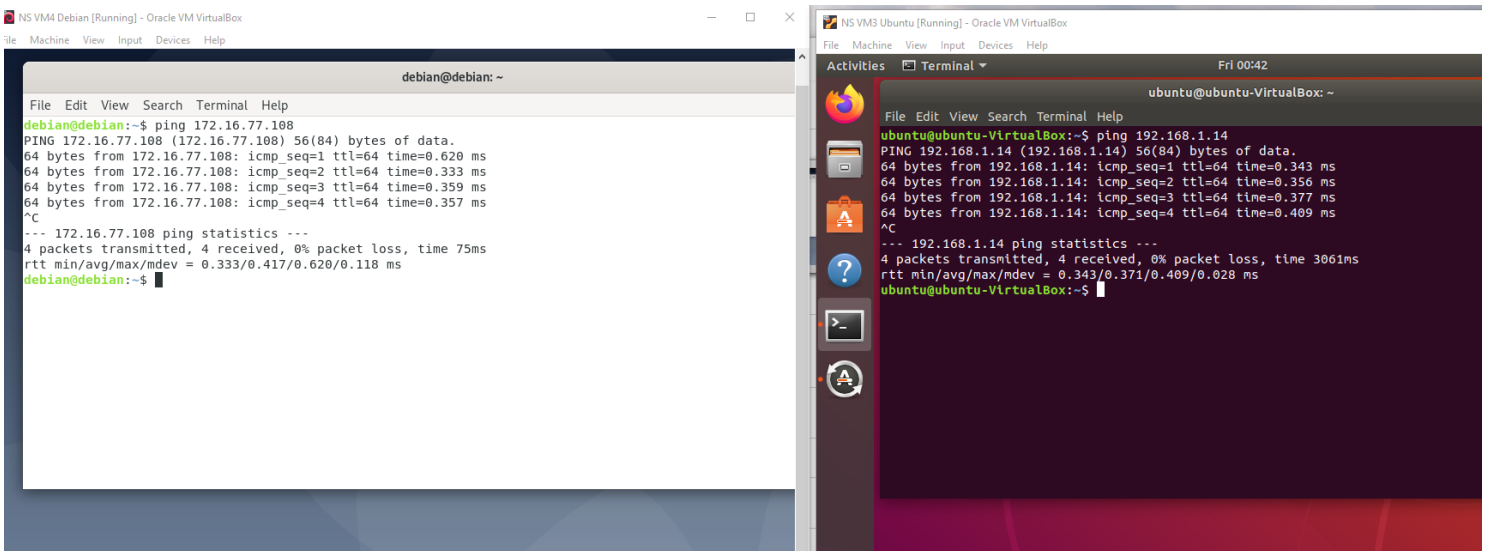
2. The pfSense machine was able to successfully ping another host on the internal subnet.



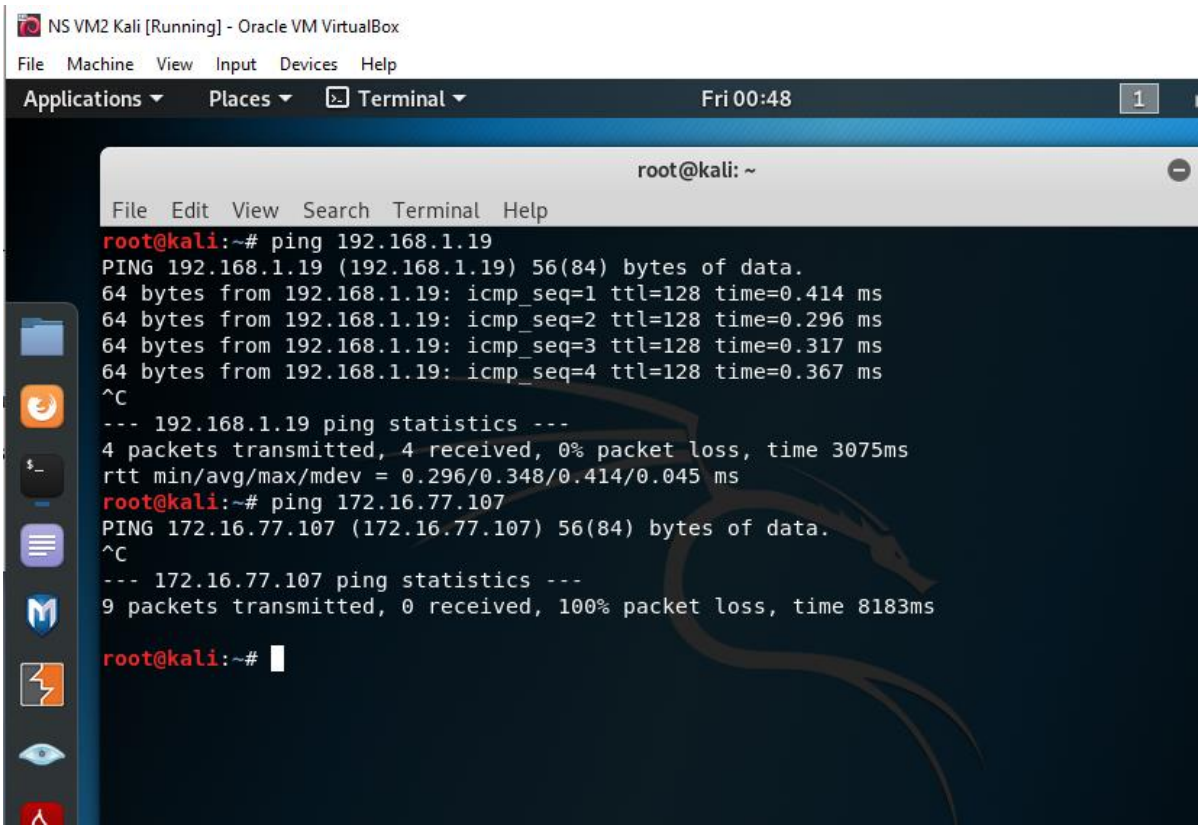
3. The pfSense machine was able to successfully ping another host on the external subnet.



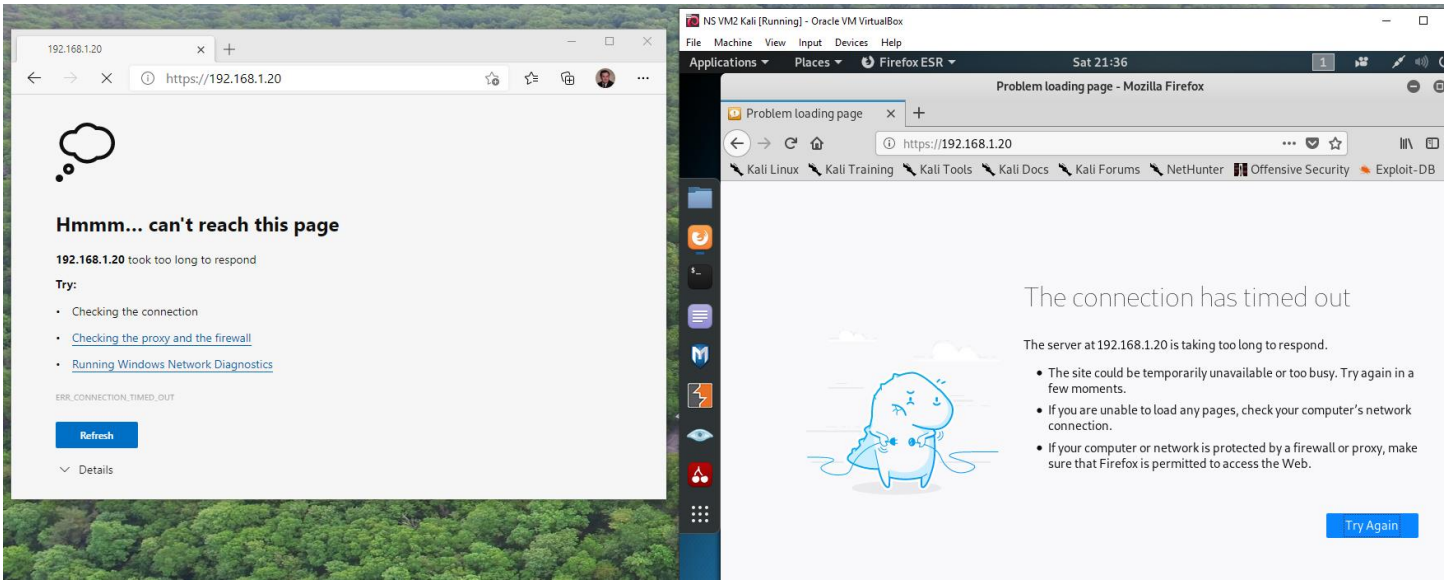
4. Any VM in the internal subnet can ping any other VM.



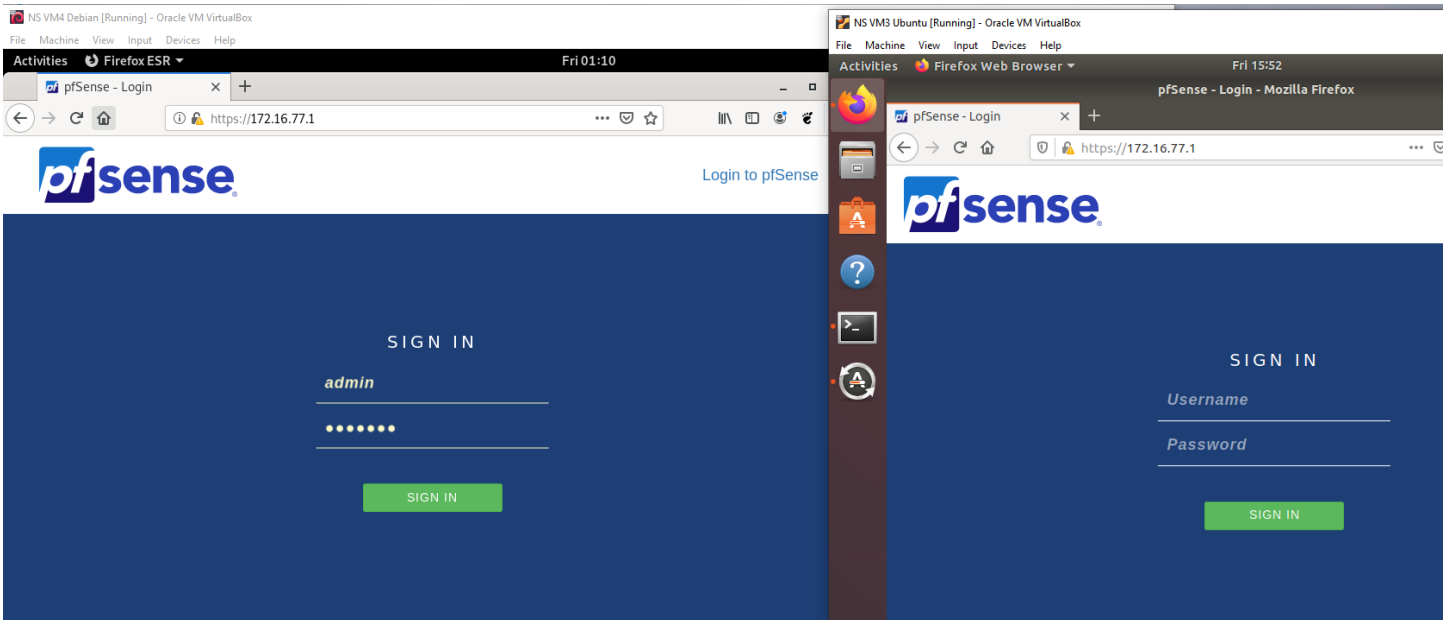
5. The Kali VM can ping the physical host but not the other VMs.



6. The pfSense management webpage cannot be accessed from the Kali VM and from the physical host (on the external network).



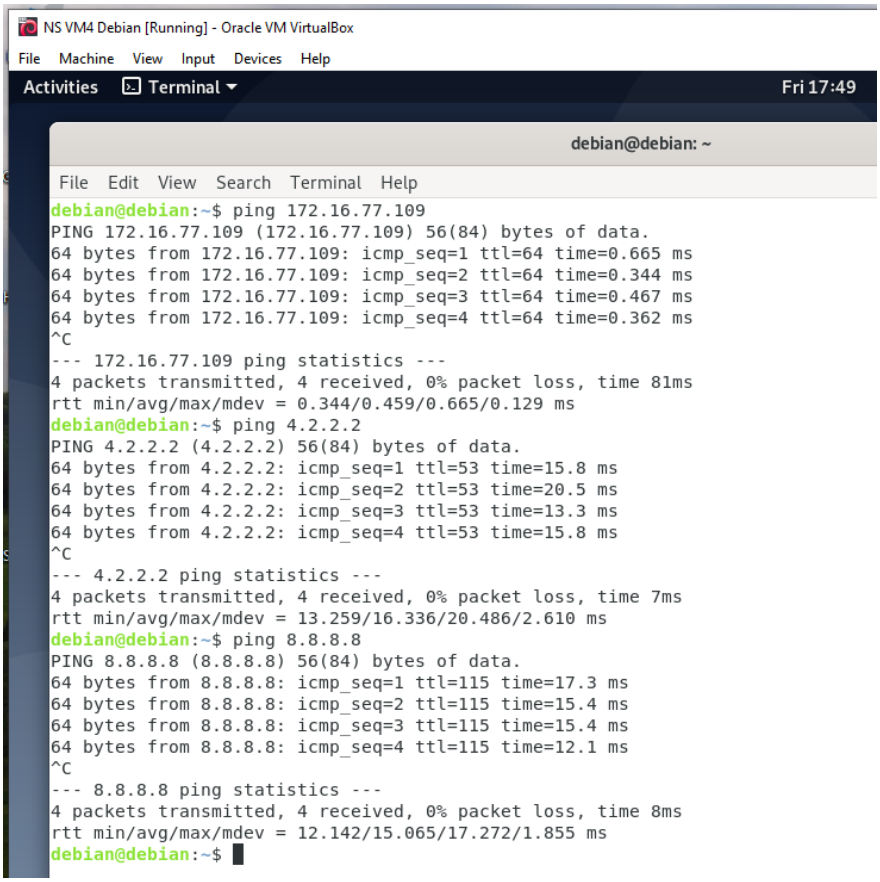
7. The pfSense management webpage can be opened from VMs on the internal subnet.



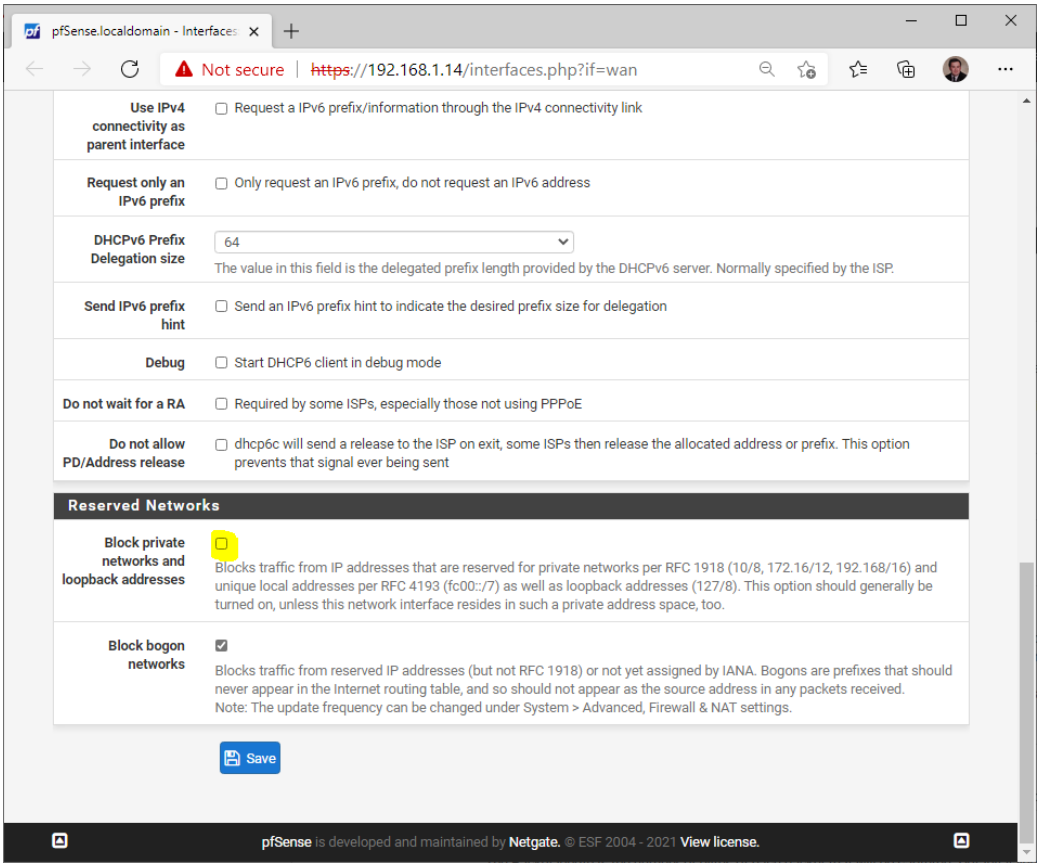
Project Task 1: Blocking Unwanted Traffic

- **Scenario:** Employees in the corporate / "internal" network can access online resources which they should not use (such as online games). Configure the firewall to block this unwanted traffic. For this scenario, the ICMP protocol will be used to represent the unwanted traffic.

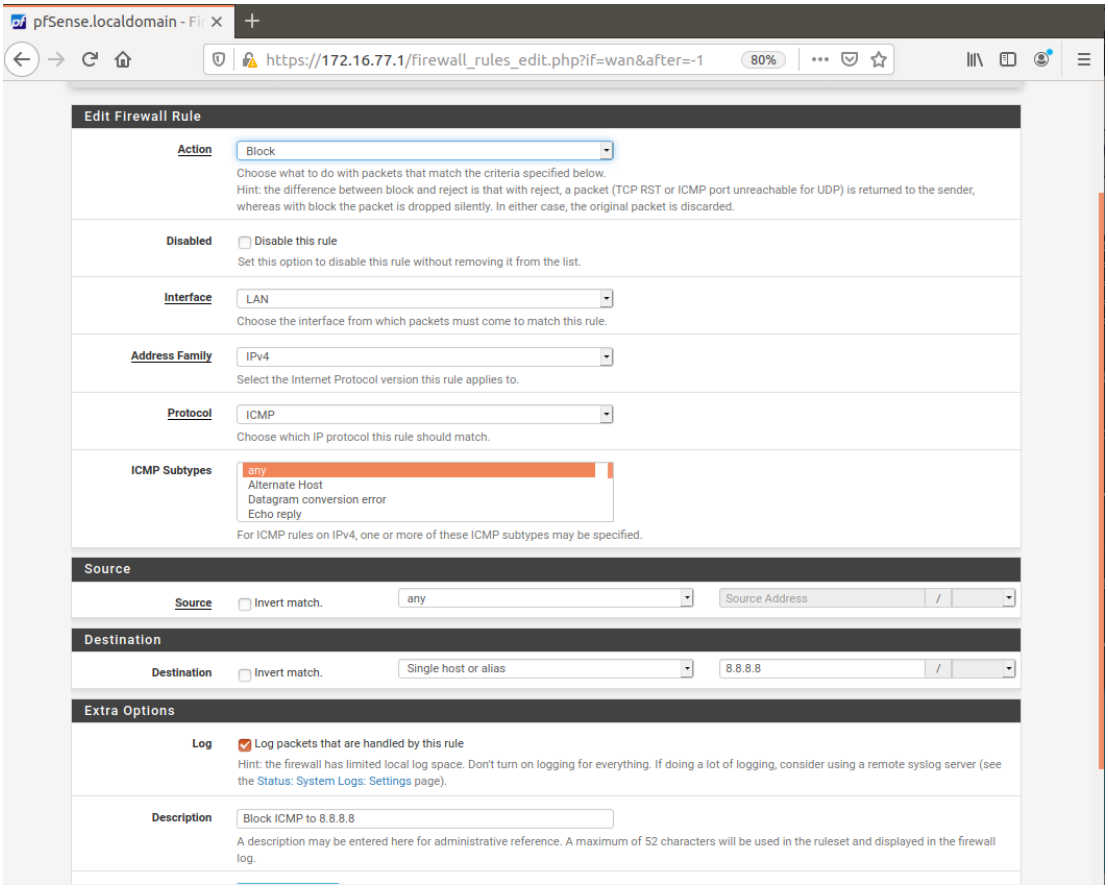
1. From the Debian VM, it was verified that ICMP traffic is allowed to other internal VMs and to external addresses on the Internet (4.2.2.2 and 8.8.8.8 were tested). (Note: Systems were rebooted, so, from this point forward, the Ubuntu VM is now 172.16.77.109 and the Debian VM is now 172.16.77.110).



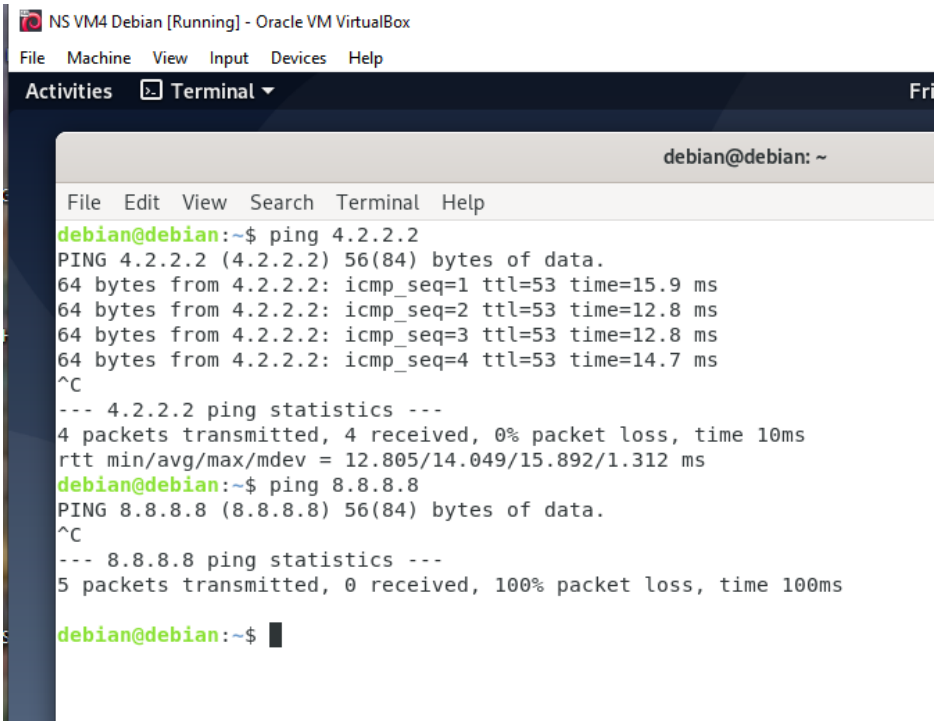
2. The WAN interface of pfSense was configured to allow traffic for private networks and loopback addresses.



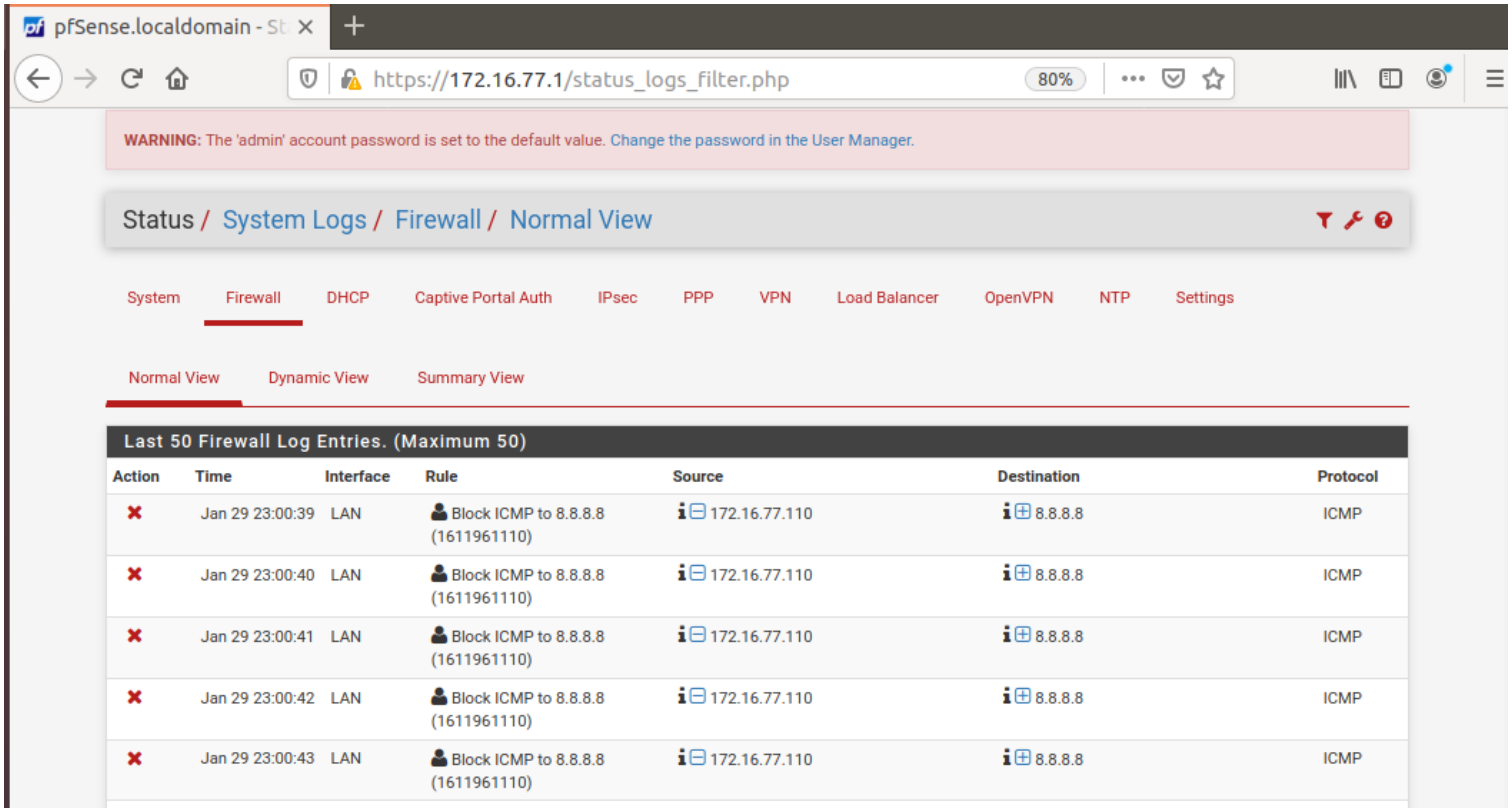
3. A firewall rule was created within the pfSense LAN interface to block ICMP traffic for a specific destination, 8.8.8.8, and logging was enabled for packets that match the rule.



4. By pinging both WAN addresses again, it was verified that the firewall rule successfully blocks ICMP traffic to 8.8.8.8 only.



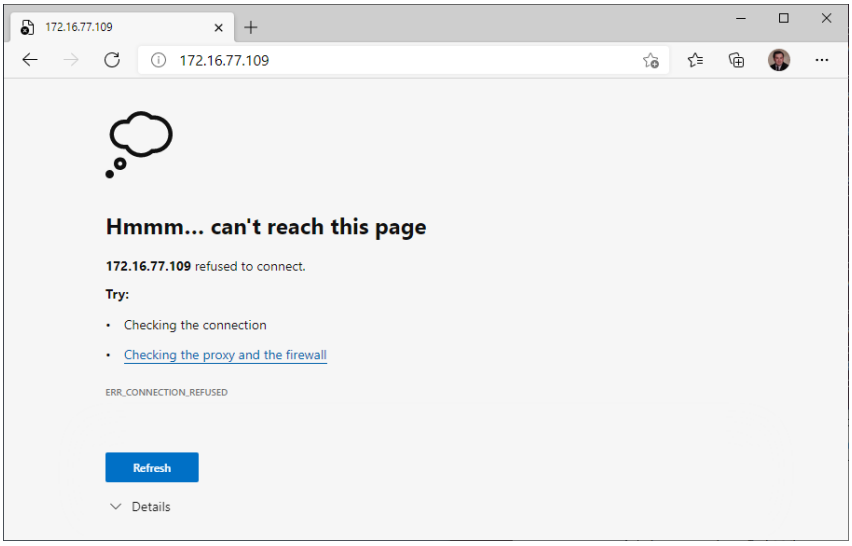
- **Evidence:** Since logging was enabled for the firewall rule, the pfSense firewall logs show that the rule was successfully configured and is blocking ICMP traffic to 8.8.8.8.



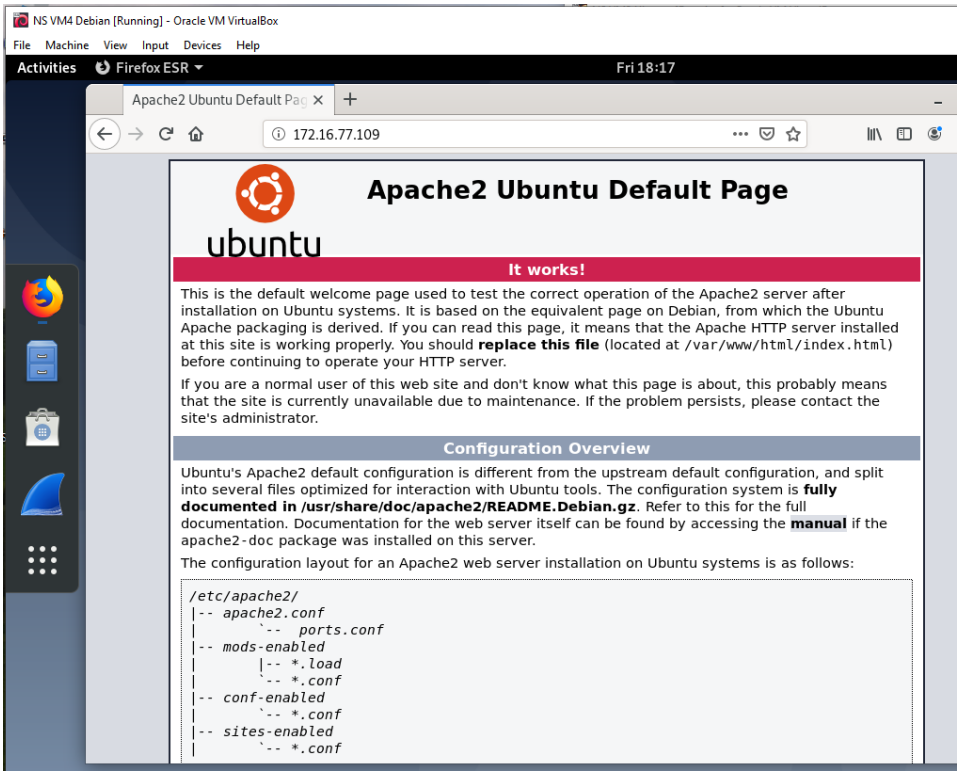
Project Task 2: Quick Solution for Remote Access

- **Scenario:** During the COVID-19 outbreak, some services needed to be performed remotely. An employee requires access to systems running on the web server at the HQ office and the ability to manage the web server that runs on the Ubuntu VM via SSH. The Warehouse Manager requested VPN access for the employees, but, while the firewall is not licensed and configured to work as needed, he wants the employees to have temporary access via other means. In this task, the web server and SSH service is to be made available for connection from remote networks by means other than a VPN.

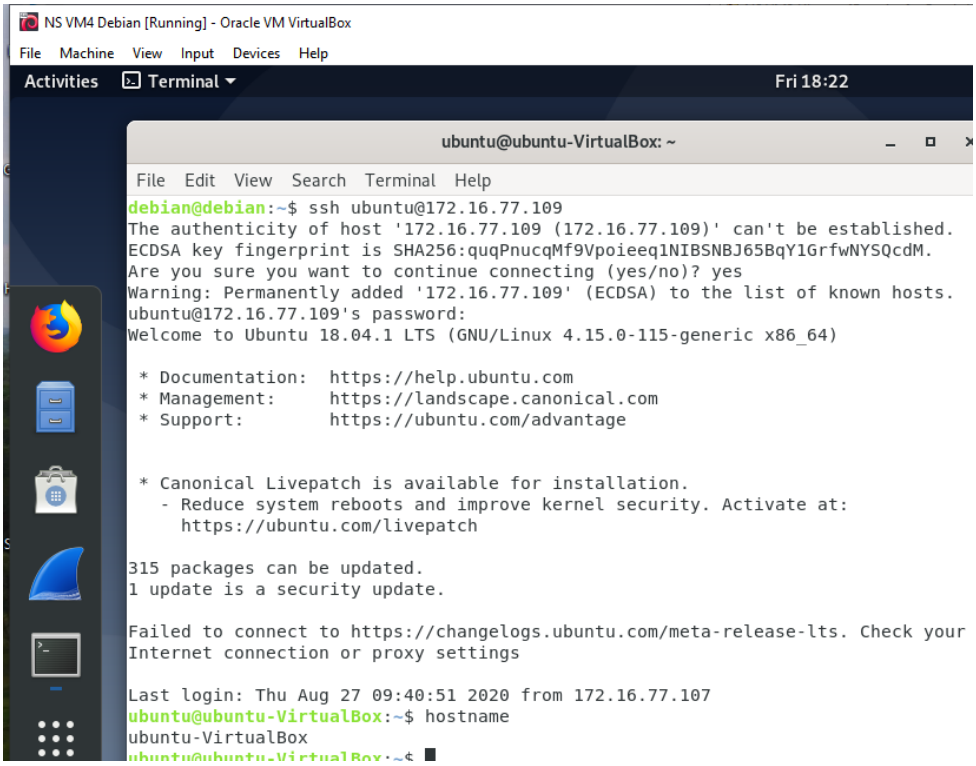
1. The web server cannot be accessed from the host machine. This is because the Ubuntu VM (which hosts the web server) is on a different subnet that cannot be reached from the external network. (The internal machines are able to access the external network through NAT, but not vice-versa.)



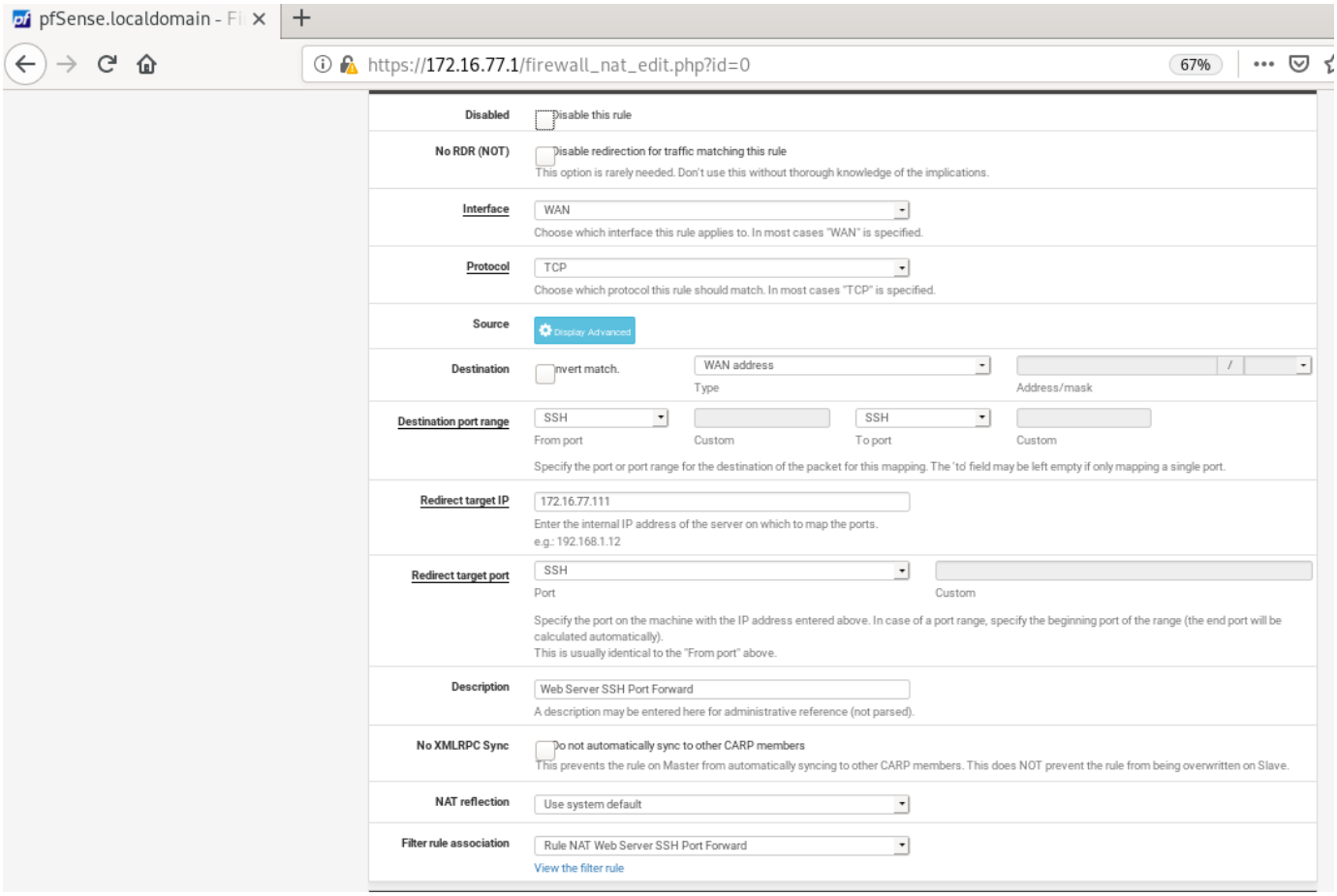
2. The web server can be successfully accessed from the Debian VM (on the same internal network), verifying its functionality.



3. I can successfully access the Ubuntu VM's SSH service from the Debian VM, verifying its functionality.



4. Two NAT port forwarding rules were created in pfSense to translate the external requests to the web server. Firewall rules were automatically created on the WAN port to authorize this traffic. (Note: Systems were rebooted, so, from this point forward, the Ubuntu VM's IP address is now 172.16.77.111.)



Port Forward | 1:1 | Outbound | NPt

Rules

	Interface	Protocol	Source Address	Source Ports	Dest. Address	Dest. Ports	NAT IP	NAT Ports	Description	Actions
☐	WAN	TCP	*	*	WAN address	22 (SSH)	172.16.77.111	22 (SSH)	Web Server SSH Port Forward	
☐	WAN	TCP	*	*	WAN address	80 (HTTP)	172.16.77.111	80 (HTTP)	Web Server HTTP Port Forward	

Firewall / Rules / WAN

The firewall rule configuration has been changed.
The changes must be applied for them to take effect.

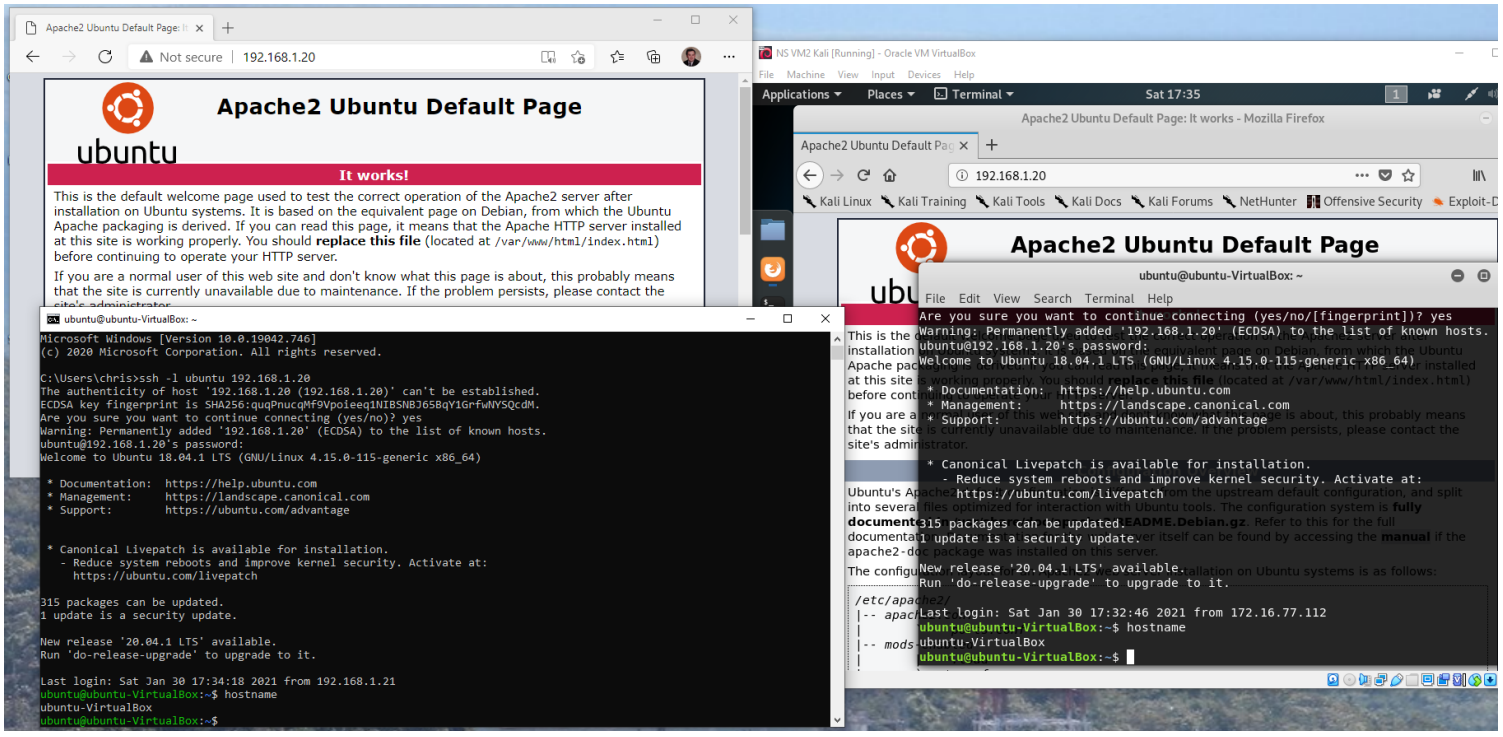
Apply Changes

Floating | WAN | LAN

Rules (Drag to Change Order)

	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
☐	0/16 KiB	IPv4 TCP	*	*	172.16.77.111	80 (HTTP)	*	none		NAT Web Server HTTP Port Forward	
☐	0/0 B	IPv4 TCP	*	*	172.16.77.111	22 (SSH)	*	none		NAT Web Server SSH Port Forward	

- **Evidence:** Multiple systems on the external network (the physical host and the Kali VM) were able to successfully access both the web server and SSH service via the pfSense WAN address. (Note: Systems were rebooted, so, from this point forward, the pfSense WAN IP address is now 192.168.1.20.)

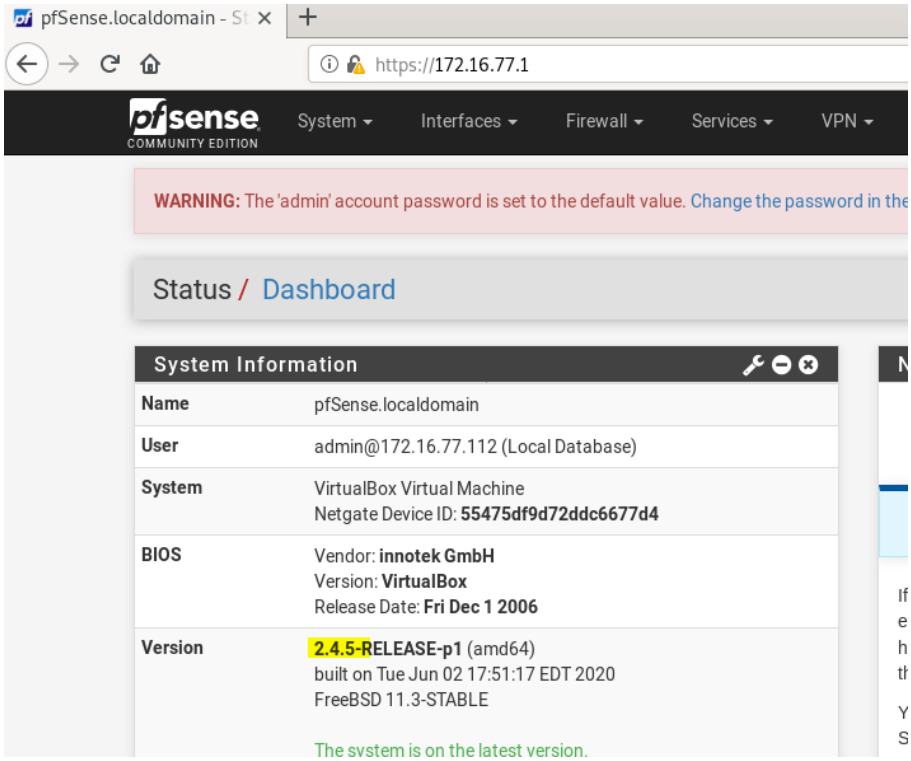


Project Task 3: The All-Seeing Eye

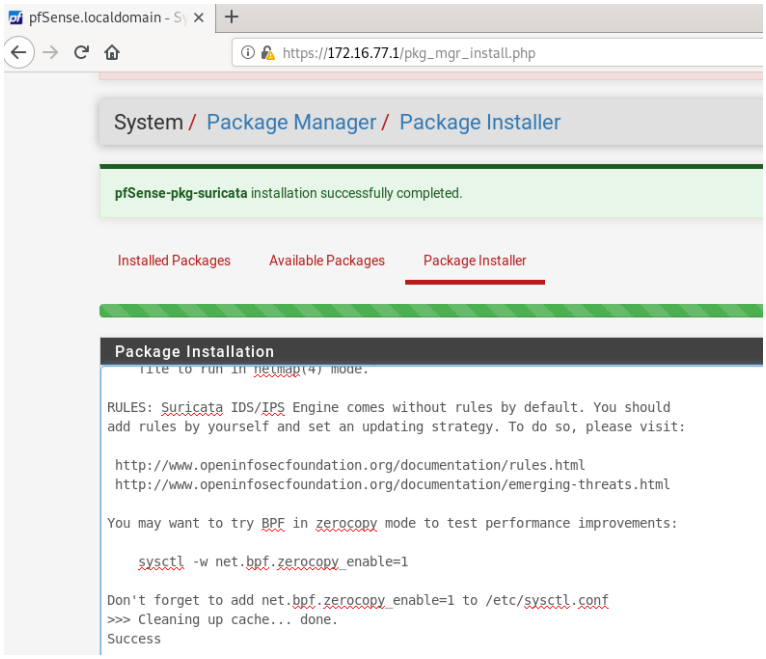
- **Scenario:** The CISO of the company decided to implement a detection and prevention system against potential known network attacks. It was requested to set up a mechanism capable of detecting DoS and brute-force attacks, and to verify that they function properly.

1. The Suricata IDS can be installed on top of pfSense to meet these requirements.

2. pfSense was updated to the latest version, 2.4.5, in order to support the most recent release of Suricata.

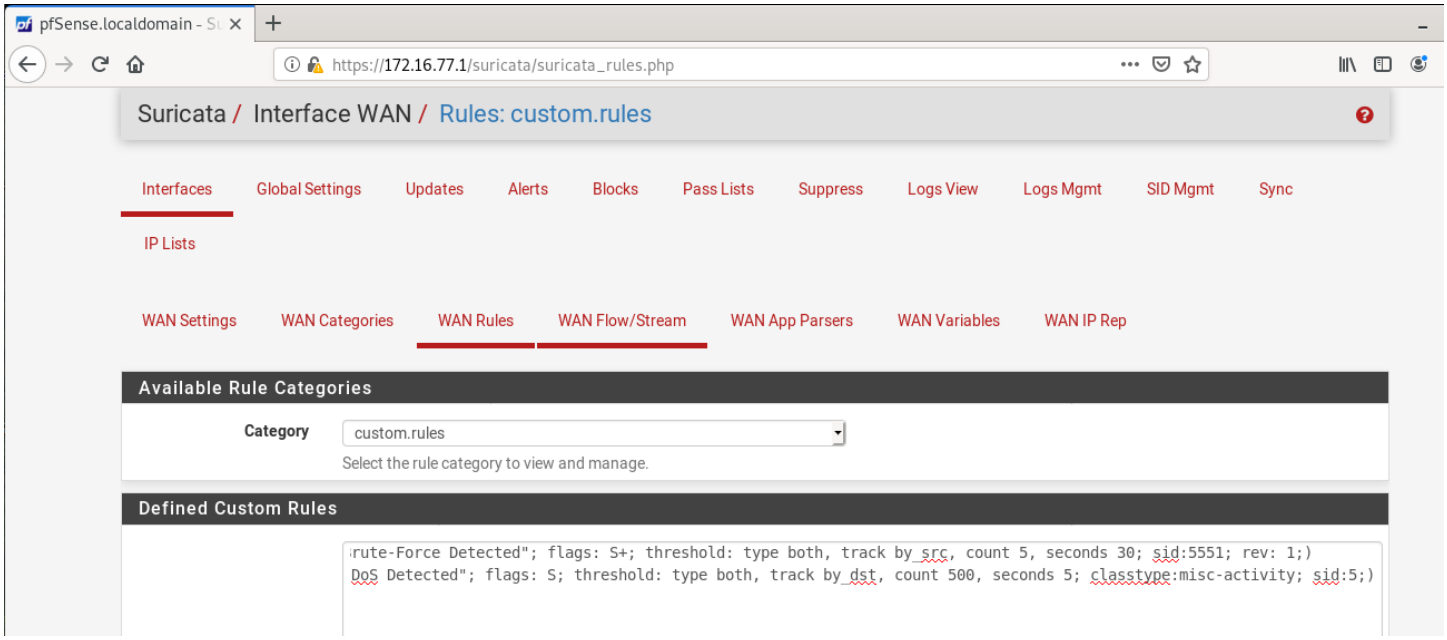


3. The Suricata package was installed onto the pfSense system.

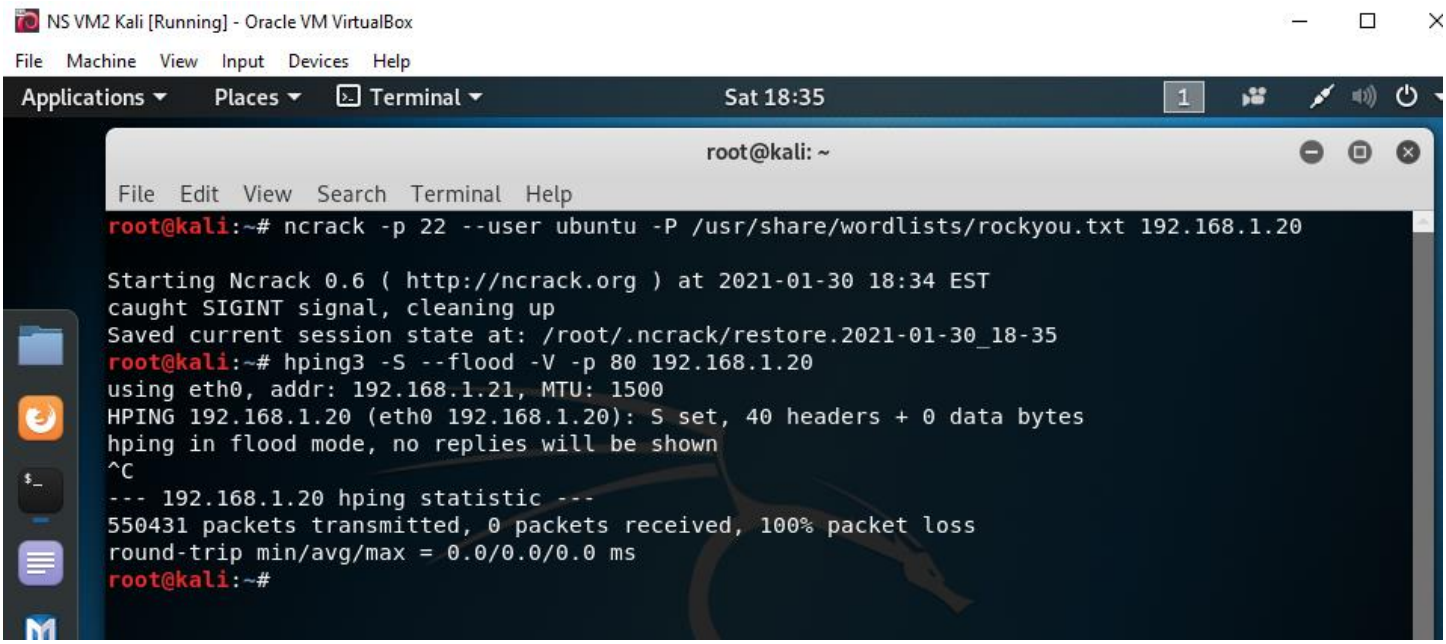


4. Suricata was configured in pfSense to monitor the WAN interface, with custom rules to detect abnormal traffic. Rules applied were:

```
alert tcp any any -> any 22 (msg:"SSH Brute-Force Detected"; flags: S+; threshold: type both, track by_src, count 5, seconds 30; sid:5551; rev: 1;)
alert tcp any any -> any 80 (msg:"HTTP DoS Detected"; flags: S; threshold: type both, track by_dst, count 500, seconds 5; classtype:misc-activity; sid:5;)
```



5. An SSH brute-force attack and a HTTP DoS attack were executed, each for 30 seconds, from the Kali Linux machine.



6. **Evidence:** The Suricata alert entries within pfSense verify that everything works as expected and that the attacks were detected.