

ISCS 3523-004

Lab 01: Intrusion Detection Tools

Christian Barba; fgb587

September 11, 2022

## **Table of Contents**

<b>Introduction</b>	<b>page 3</b>
<b>Setting up the Virtual Machine Lab</b>	<b>page 3</b>
<b>Verifications of a Working Virtual Environment</b>	<b>page 3</b>
<b>Six Applications used for Intrusion Detection on Kali Linux</b>	<b>page 4</b>
<b>How this Lab Works and why it is Useful</b>	<b>page 5</b>
<b>Appendix</b>	<b>page 7</b>
<b>Works Cited</b>	<b>page 10</b>

## Introduction

The purpose of this lab is to establish a virtual environment that will allow the user to simulate real world attacks in a risk-free zone. Many cybersecurity experts set sand boxes to dissect malicious code, test security features, and analyze attacks to obtain a deeper understanding on how to make their host computers a safer space. Without being able to view attacks head on, the ability to detect attacks and learn how to prevent them would become nearly impossible.

## Setting up the Virtual Machine Lab

Setting up the virtual machines that are used for the rest of this course was not an easy process, and I had to troubleshoot multiple problems that I ran into. During this lab, I had to install a PfSense VM, a Kali Linux VM, two Ubuntu server VMs (IPS and SIEM), a Metasploitable VM, and finally a Windows 10 VM. The process of installing all these ISOs and OVAs took me around three days in total as a lot of the mistakes I was making were minor. The problems that was most drastic was not being able to install Suricata on IPS, but due to the availability of an OVA, I chose to download that instead. Figure 4.1 shows all of the virtual machines that are in my VirtualBox.

## Verifications of a Working Environment

Due to the network topology that I had setup in the lab, I can ping Metasploitable 2 from my Kali machine as well as my PfSense, and I can browse the internet on Kali. The verification that I can ping Metasploitable 2 from my Kali machine is shown in figure 1.1, and figure 1.2 shows that I am able to ping my Kali machine from Metasploitable 2. My IPS machine can connect to the internet through the Host-only networking that was set up in the network configuration. I can use the commands *curl -I https://google.com* and *nslookup google.com* to get proper feedback as shown in figure 2.1. The *curl -I* returns the HTTP response headers from google.com, and the *nslookup* shows the NDS records for google.com. Through the Splunk website, I can inspect the activity and alerts that are detected on my SIEM VM. Figures 3.1 and 3.2 show that the website is functional and is reporting the activity that is

occurring on the VM. There have been no alerts because none of the commands I have used have been of concern.

When opening a Kali terminal and using the command *ip addr*, it displays 2 adapters shown in figure 5.1. These are shown as lo and eth0, and lo is the loopback interface, which is how the system talks to itself, while eth0 shows the IP address for the machine and how everyone else will recognize it if someone decides to ping it. This is not the exact same case when I type in the *ipconfig* and *ipconfig /all* commands in the Windows 10 command prompt. In Windows, the *ipconfig* command shows the IPv4 address, Subnet Mask, and the Default Gateway, but when you add the */all* extension, more information appears such as the DHCP server information. The physical address is also visible under this extension.

When using Wireshark on both Kali and Windows there was a noticeable difference. Figure 5.2 shows how a Windows version of Wireshark is presented. Wireshark on this OS gives out more information like the TCP packets that are regularly active during the system without any ping enabled from Kali. Windows also is able to show the TCP ports that were used when Windows ran into problems with some packets that came through the network. The TCP retransmissions are located throughout the report on the Windows machine, but in figure 5.3, Kali only shows the ping requests and the replies that occurred during this exchange.

### **Six Applications used for Intrusion Detection on Kali Linux**

Kali Linux is host to many cyber security applications that can be used to detect intrusions on a given system. Some of these applications include: “lbd”, “Wireshark”, “nmap”, “netmasks”, “wafw00f”, and “Suricata”. According to Packt, lbd is a tool used to detect load balancing on a given DNS or HTTP, and this would indicate if a router were getting throttled from a DDoS attack for example. Load balancing is important in a company environment because the network has to be function to a high standard to supply access to multiple users with proper internet access. If network traffic appears to be building up there will be an easier way to detect why this is happening due to lbd. Wireshark is used to see the traffic

coming into a network by displaying all the packets being sent and received. If there is to be believed that a surplus of network traffic, perhaps detected by the Idd, then Wireshark allows us to find the source of this traffic and understand why it is happening. The tool nmap is used to display the network topology on a given network, and this can show who is on the network, what OS they are running, and manage the host machine, as well as other services (Javapoint). A way hackers try to steal information is by exploiting a company's internet connections by logging onto their servers. Thanks to nmap, there is a way to detect who is on the server, and if there appears to be someone running a non-native OS then it will be easy to detect and shutdown. Netmasks can be used to find a subnet mask to identify hosts that may be connected to a network (Kali Linux, 2022). This appears to go hand in hand with nmap at detecting who is on a server. Wafw00f detects if the website a person is accessing is protected by a firewall by sending certain attacks to validate the website is official or not because a safe website should have an active firewall (mohdshariq). Dealing with co-workers may entail that they might encounter suspicious websites that appear to be friendly, but with wafw00f, it is now simpler to detect if these websites are actually valid. This application could be used to defend against SQL injections. Lastly, a tool that can be used for IPS is Suricata because Suricata is used to detect threats on a network by using deep packet inspection. All these tools are useful for intrusion detection because the main purpose of them tools is to detect any malicious traffic that may be flooding into the network a person is operating on. Whether it be monitoring the packets coming into the network or displaying the map of machines connected in a single topography, these tools can detect if something is wrong or out of place.

### **How this lab works and why it is useful**

Installing multiple machines onto one host computer is one of the most important tools that a penetration tester, malware analyst, forensic analyst, et cetera, can use to obtain research. The topology of this lab is the most important part because we are able to load destructive material into the machines without being afraid that it will leach onto our host computers. This is due to PfSense acting as the router that connects all of the machines together. PfSense is able to use network management protocol

DHCP to establish a range of IP addresses and assign certain machines these IP addresses in order to keep everything on a specific range. This is especially useful when working with multiple virtual machines on a host computer as mapping out the network topology becomes organized and precise as to not mix up specific machines. Throughout the lab, we have taken multiple precautions like setting up firewall rules and squid proxy servers to ensure nothing can escape a Kali or a Metasploitable 2 machine through the network and reach the host computer. Both Kali and Metasploitable 2 are functioning on the intnet network setting, which will allow them to connect to the internet through the PfSense router we have established early on. On this virtual router we have disabled the IPv4 protocol for Metasploitable 2 to deny any packets from dispersing outwards as well as blocking out ports 80 and 21 to deny any HTTP and FTP requests to go out. However, through Suricata acting as a three-way handshake on our IPS machine, we are able to send packets from Kali to Metasploitable 2 and detect whether or not these packets violate the rules that Suricata has defined with the af\_packet. Another feature we have disabled on these machines is the ability to drag and drop files from the VMs to the host computer. Since we will be dealing with malicious code, it is important that there is no possibility of a slip up that may cause the viruses to jump from the machines onto our host computer.

The purpose of setting up this virtual ecosystem is to have a place to load in malicious code as well as see how a network may be reacting to certain events that may occur by an attacker. Being able to simulate attacks that have been used before can allow an analyst to gain knowledge and get familiar with the tools that can help with attacks that may be coming in the foreseeable future. For penetration testing, we could use certain tools and applications that will allow us to test the defenses of our environment, and with this information, we can fortify our real-world systems to stop certain attacks from happening again. For forensics, being able to load a hard drive into a virtual machine would be the ideal situation for the analyst. This is due to not knowing what is on a hard drive that you may want to analyze because it could be full of malicious code. If this is the case, then the virtual machine will be able to take all of the damage and not the host computer. These situations appear a lot of the times throughout

a cyber security experts' career. Having this virtual environment allows these security experts to have a sandbox to test anything that they may seem fit.

# Appendix

## 1.1- Pinging and Curling Metasploitable 2 from Kali Linux:

```
cbarba@kali: ~  
File Actions Edit View Help  
cbarba@kali-[~]  
$ ping -c 4 172.16.2.3 66 curl 172.16.2.3  
PING 172.16.2.3 (172.16.2.3) 56(84) bytes of data.  
64 bytes from 172.16.2.3: icmp_seq=1 ttl=64 time=0.889 ms  
64 bytes from 172.16.2.3: icmp_seq=2 ttl=64 time=0.542 ms  
64 bytes from 172.16.2.3: icmp_seq=3 ttl=64 time=0.519 ms  
64 bytes from 172.16.2.3: icmp_seq=4 ttl=64 time=0.493 ms  
  
--- 172.16.2.3 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3038ms  
rtt min/avg/max/mdev = 0.493/0.610/0.889/0.161 ms  
<html><head><title>Metasploitable2 - Linux</title></head><body>  
<pre>  
  
metasploitable2  
  
</pre>  
  
Warning: Never expose this VM to an untrusted network!  
Contact: msfdev[at]metasploit.com
```

## 1.2- Pinging Kali Linux from Metasploitable 2:

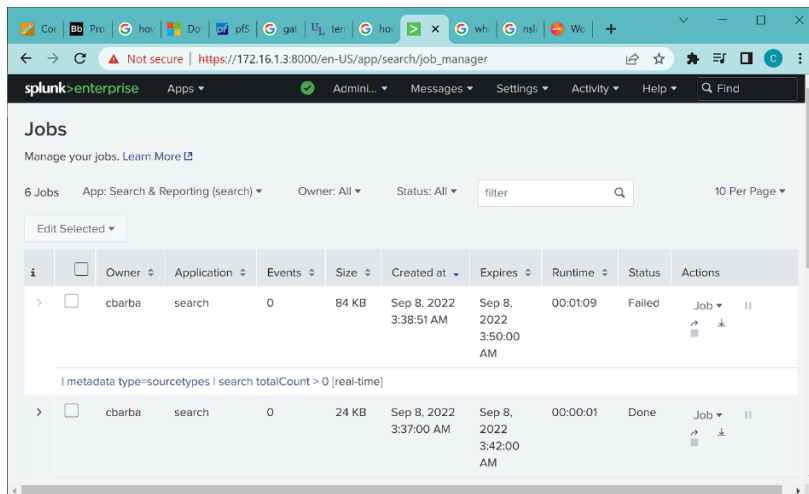
```
Metasploitable 2 (Snapshot 1) [Running] - Oracle VM Virt...  
File Machine View Input Devices Help  
Last login: Tue Sep 6 16:16:33 EDT 2022 on tty1  
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/*copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To access official Ubuntu documentation, please visit:  
http://help.ubuntu.com/  
No mail.  
msfadmin@metasploitable:~$ ping 172.16.2.2  
PING 172.16.2.2 (172.16.2.2) 56(84) bytes of data.  
64 bytes from 172.16.2.2: icmp_seq=1 ttl=64 time=5.94 ms  
64 bytes from 172.16.2.2: icmp_seq=2 ttl=64 time=0.782 ms  
64 bytes from 172.16.2.2: icmp_seq=3 ttl=64 time=0.591 ms  
64 bytes from 172.16.2.2: icmp_seq=4 ttl=64 time=0.457 ms  
64 bytes from 172.16.2.2: icmp_seq=5 ttl=64 time=0.559 ms  
  
--- 172.16.2.2 ping statistics ---  
5 packets transmitted, 5 received, 0% packet loss, time 4004ms  
rtt min/avg/max/mdev = 0.457/1.666/5.944/2.141 ms  
msfadmin@metasploitable:~$
```

## 2.1- IPS being able to curl and Nslookup google.com:

```
IPS [Running] - Oracle VM VirtualBox  
File Machine View Input Devices Help  
applicable law.  
  
Last login: Tue Sep 6 19:32:56 UTC 2022 on tty1  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
cbarba@suricata:~$ curl -I https://www.google.com  
HTTP/2 200  
content-type: text/html; charset=ISO-8859-1  
3p: CP="This is not a P3P policy! See g.co/p3phelp for more info."  
Date: Thu, 08 Sep 2022 03:03:50 GMT  
Server: gws  
x-xss-protection: 0  
x-frame-options: SAMEORIGIN  
Expires: Thu, 08 Sep 2022 03:03:50 GMT  
Cache-Control: private  
Set-Cookie: IP_JAR=2022-09-08-03; expires=Sat, 08-Oct-2022 03:03:50 GMT; path=/; domain=.google.com;  
Secure  
Set-Cookie: AEC=AakniG017LsI_4umm-vxbMez145C8523XuXS5J41e-iyxKUj0F1t00aUuCI; expires=Tue, 07-Mar-202  
3 03:03:50 GMT; path=/; domain=.google.com; Secure; HttpOnly; SameSite=lax  
Set-Cookie: NID=511=oB05MbM0vd73_oPA3fVVA5F0MEKJR-vuA_HSgv5Mt00Fah_aNoSUUK5Z0KwSkZpQ9QzYl_DC-VznUXz  
5YrCbF_EowgcMnQbNBB6mhSziDK2fy5J9aYkRRywoiul_g2NTY_0yH8pX0n1_5E0os_yaBTgaIQ039uU0KUK3mQE2p0; expires  
=Fri, 10-Mar-2023 03:03:50 GMT; path=/; domain=.google.com; HttpOnly  
alt-svc: h3=":443"; ma=2592000,h3-29=":443"; ma=2592000,h3-Q050=":443"; ma=2592000,h3-Q046=":443"; m  
a=2592000,h3-Q043=":443"; ma=2592000,quic=":443"; ma=2592000; v="46,43"  
  
cbarba@suricata:~$ nslookup google.com  
Server: 127.0.0.53  
Address: 127.0.0.53#53  
  
Non-authoritative answer:  
Name: google.com  
Address: 142.251.40.78  
Name: google.com  
Address: 2607:f8b0:4000:819::200e  
  
cbarba@suricata:~$
```



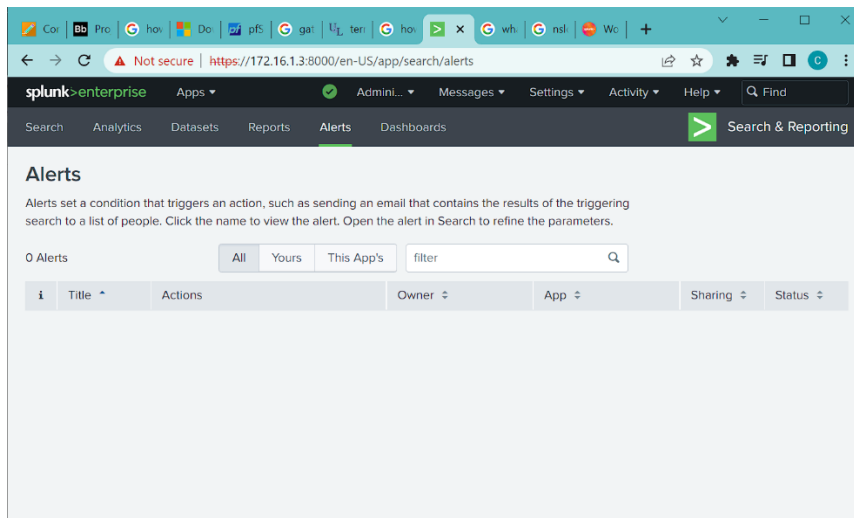
### 3.1- Splunk activity logs (Jobs page):



The screenshot shows the Splunk Jobs page in a web browser. The page title is "Jobs" with a subtitle "Manage your jobs. Learn More". Below the title, there are filters for "App: Search & Reporting (search)", "Owner: All", and "Status: All". A search bar with the text "filter" is present. The table below lists search jobs. The first job is "cbarba" with a size of 84 KB, created on Sep 8, 2022 at 3:38:51 AM, and a status of "Failed". The second job is also "cbarba" with a size of 24 KB, created on Sep 8, 2022 at 3:37:00 AM, and a status of "Done".

i	Owner	Application	Events	Size	Created at	Expires	Runtime	Status	Actions
>	cbarba	search	0	84 KB	Sep 8, 2022 3:38:51 AM	Sep 8, 2022 3:50:00 AM	00:01:09	Failed	Job ▾
I metadata type=sourcetypes   search totalCount > 0 [real-time]									
>	cbarba	search	0	24 KB	Sep 8, 2022 3:37:00 AM	Sep 8, 2022 3:42:00 AM	00:00:01	Done	Job ▾

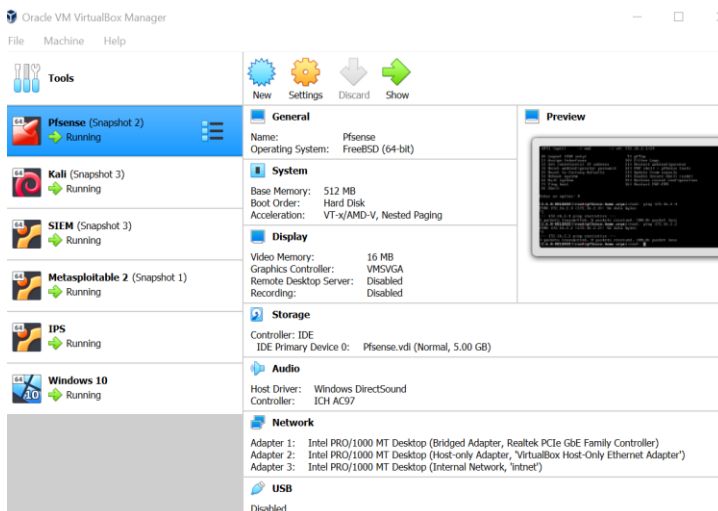
### 3.2- Splunk alert page:



The screenshot shows the Splunk Alerts page in a web browser. The page title is "Alerts" with a subtitle "Alerts set a condition that triggers an action, such as sending an email that contains the results of the triggering search to a list of people. Click the name to view the alert. Open the alert in Search to refine the parameters." Below the title, there are filters for "All", "Yours", and "This App's". A search bar with the text "filter" is present. The table below lists alerts. The first alert is "Pfense" with a size of 512 MB, created on Sep 8, 2022 at 3:38:51 AM, and a status of "Running".

i	Title	Actions	Owner	App	Sharing	Status
>	Pfense					Running

### 4.1- List of all VMs working on the host computer:



The screenshot shows the Oracle VM VirtualBox Manager interface. On the left, there is a list of VMs: "Pfense (Snapshot 2)", "Kali (Snapshot 3)", "SIEM (Snapshot 3)", "Metasploitable 2 (Snapshot 1)", "IPS", and "Windows 10". The main panel shows the settings for the selected VM, "Pfense". The settings are organized into sections: General, System, Display, Storage, Audio, Network, and USB. The General section shows the Name as "Pfense" and the Operating System as "FreeBSD (64-bit)". The System section shows the Base Memory as 512 MB and the Boot Order as Hard Disk. The Display section shows the Video Memory as 16 MB and the Graphics Controller as VM5VGA. The Storage section shows the Controller as IDE and the IDE Primary Device as Pfense.vdi (Normal, 5.00 GB). The Audio section shows the Host Driver as Windows DirectSound and the Controller as ICH AC97. The Network section shows the Adapter 1 as Intel PRO/1000 MT Desktop (Bridged Adapter, Realtek PCIe GbE Family Controller). The USB section shows the Controller as Disabled.

VM Name	Snapshot	Status
Pfense	Snapshot 2	Running
Kali	Snapshot 3	Running
SIEM	Snapshot 3	Running
Metasploitable 2	Snapshot 1	Running
IPS		Running
Windows 10		Running

## 5.1- Kali Linux IP address configurations.

```
cbarba@kali: ~  
File Actions Edit View Help  
cbarba@kali: ~  
$ ip addr  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000  
    link/ether 08:00:27:1f:f2:38 brd ff:ff:ff:ff:ff:ff  
    inet 172.16.2.2/24 brd 172.16.2.255 scope global dynamic noprefixroute eth0  
        valid_lft 6164sec preferred_lft 6164sec  
    inet6 fe80::a00:27ff:fe1f:f238/64 scope link noprefixroute  
        valid_lft forever preferred_lft forever
```

## 5.2- Windows Wireshark results:

Windows 10 (Snapshot 1) [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Ethernet

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-F>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.16.2.4	208.111.176.128	TCP	66	49804 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
2	0.343605	172.16.2.4	72.21.91.29	TCP	66	49802 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
3	0.593799	172.16.2.4	72.21.91.29	TCP	66	49803 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
4	0.702920	172.16.2.4	72.21.91.29	TCP	66	49800 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
5	1.000579	172.16.2.4	208.111.176.128	TCP	66	[TCP Retransmission] [TCP Port numbers reused] 49804 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
6	1.640660	172.16.2.4	184.50.50.164	TCP	66	49801 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
7	2.010410	172.16.2.2	172.16.2.4	ICMP	98	Echo (ping) request id=0x049f, seq=1/256, ttl=64 (reply in 10)
8	2.010568	PcsCompu_5d:6b:d8	Broadcast	ARP	42	Who has 172.16.2.2? Tell 172.16.2.4
9	2.011051	PcsCompu_1f:f2:38	PcsCompu_5d:6b:d8	ARP	60	172.16.2.2 is at 08:00:27:1f:f2:38
10	2.011071	172.16.2.4	172.16.2.2	ICMP	98	Echo (ping) reply id=0x049f, seq=1/256, ttl=128 (request in 7)
11	3.012158	172.16.2.2	172.16.2.4	ICMP	98	Echo (ping) request id=0x049f, seq=2/512, ttl=64 (reply in 12)
12	3.012287	172.16.2.4	172.16.2.2	ICMP	98	Echo (ping) reply id=0x049f, seq=2/512, ttl=128 (request in 11)
13	3.015291	172.16.2.4	208.111.176.128	TCP	66	[TCP Retransmission] [TCP Port numbers reused] 49804 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
14	3.093878	PcsCompu_5d:6b:d8	PcsCompu_f2:83:ea	ARP	42	Who has 172.16.2.1? Tell 172.16.2.4
15	3.094592	PcsCompu_f2:83:ea	PcsCompu_5d:6b:d8	ARP	60	172.16.2.1 is at 08:00:27:f2:83:ea
16	3.469508	172.16.2.4	208.111.176.128	TCP	66	49805 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
17	4.030855	172.16.2.2	172.16.2.4	ICMP	98	Echo (ping) request id=0x049f, seq=3/768, ttl=64 (reply in 18)
18	4.036174	172.16.2.4	172.16.2.2	ICMP	98	Echo (ping) reply id=0x049f, seq=3/768, ttl=128 (request in 17)
19	4.359502	172.16.2.4	72.21.91.29	TCP	66	[TCP Retransmission] [TCP Port numbers reused] 49802 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
20	4.484084	172.16.2.4	208.111.176.128	TCP	66	[TCP Retransmission] [TCP Port numbers reused] 49805 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
21	4.609199	172.16.2.4	72.21.91.29	TCP	66	[TCP Retransmission] [TCP Port numbers reused] 49803 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
22	5.059846	172.16.2.2	172.16.2.4	ICMP	98	Echo (ping) request id=0x049f, seq=4/1024, ttl=64 (reply in 23)
23	5.059989	172.16.2.4	172.16.2.2	ICMP	98	Echo (ping) reply id=0x049f, seq=4/1024, ttl=128 (request in 22)
24	5.226518	172.16.2.4	72.21.91.29	TCP	66	49806 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
25	5.265776	172.16.2.4	72.21.91.29	TCP	66	49799 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
26	6.003692	172.16.2.2	172.16.2.4	ICMP	98	Echo (ping) request id=0x049f, seq=5/1280, ttl=64 (reply in 27)
27	6.003783	172.16.2.4	172.16.2.2	ICMP	98	Echo (ping) reply id=0x049f, seq=5/1280, ttl=128 (request in 26)
28	6.234123	172.16.2.4	72.21.91.29	TCP	66	[TCP Retransmission] [TCP Port numbers reused] 49806 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
29	6.484306	172.16.2.4	208.111.176.128	TCP	66	[TCP Retransmission] [TCP Port numbers reused] 49805 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
30	7.015744	172.16.2.4	208.111.176.128	TCP	66	[TCP Retransmission] [TCP Port numbers reused] 49804 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
31	7.075984	PcsCompu_1f:f2:38	PcsCompu_5d:6b:d8	ARP	60	Who has 172.16.2.4? Tell 172.16.2.2
32	7.076001	PcsCompu_5d:6b:d8	PcsCompu_1f:f2:38	ARP	42	172.16.2.4 is at 08:00:27:5d:6b:d8
33	7.108073	172.16.2.2	172.16.2.4	ICMP	98	Echo (ping) request id=0x049f, seq=6/1536, ttl=64 (reply in 34)
34	7.108167	172.16.2.4	172.16.2.2	ICMP	98	Echo (ping) reply id=0x049f, seq=6/1536, ttl=128 (request in 33)
35	8.240225	172.16.2.4	72.21.91.29	TCP	66	[TCP Retransmission] [TCP Port numbers reused] 49806 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
36	8.687898	172.16.2.4	72.21.91.29	TCP	66	49807 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
37	8.715204	172.16.2.4	72.21.91.29	TCP	66	[TCP Retransmission] [TCP Port numbers reused] 49800 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
38	9.601483	PcsCompu_5d:6b:d8	PcsCompu_f2:83:ea	ARP	42	Who has 172.16.2.1? Tell 172.16.2.4

## 5.3- Kali Wireshark results:

Kali (Snapshot 4) [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Ethernet

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-F>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.00000000	172.16.2.4	172.16.2.2	ICMP	74	Echo (ping) request id=0x0001, seq=1/256, ttl=128 (reply in 2)
2	0.000031059	172.16.2.2	172.16.2.4	ICMP	74	Echo (ping) reply id=0x0001, seq=1/256, ttl=64 (request in 1)
3	1.013371976	172.16.2.4	172.16.2.2	ICMP	74	Echo (ping) request id=0x0001, seq=2/512, ttl=128 (reply in 4)
4	1.013493406	172.16.2.2	172.16.2.4	ICMP	74	Echo (ping) reply id=0x0001, seq=2/512, ttl=64 (request in 3)
5	2.020109991	172.16.2.4	172.16.2.2	ICMP	74	Echo (ping) request id=0x0001, seq=3/768, ttl=128 (reply in 6)
6	2.029140761	172.16.2.2	172.16.2.4	ICMP	74	Echo (ping) reply id=0x0001, seq=3/768, ttl=64 (request in 5)
7	3.043882716	172.16.2.4	172.16.2.2	ICMP	74	Echo (ping) request id=0x0001, seq=4/1024, ttl=128 (reply in 8)
8	3.043902766	172.16.2.2	172.16.2.4	ICMP	74	Echo (ping) reply id=0x0001, seq=4/1024, ttl=64 (request in 7)
9	4.934358239	PcsCompu_5d:6b:d8	PcsCompu_1f:f2:38	ARP	60	Who has 172.16.2.2? Tell 172.16.2.4
10	4.934377109	PcsCompu_1f:f2:38	PcsCompu_5d:6b:d8	ARP	42	172.16.2.2 is at 08:00:27:1f:f2:38
11	5.209256156	PcsCompu_1f:f2:38	PcsCompu_5d:6b:d8	ARP	42	Who has 172.16.2.4? Tell 172.16.2.2
12	5.209628545	PcsCompu_5d:6b:d8	PcsCompu_1f:f2:38	ARP	60	172.16.2.4 is at 08:00:27:5d:6b:d8

## Works Cited

“HTTP and DNS Load Balancer Detection.” *Packt*,

<https://subscription.packtpub.com/book/networking-and-servers/9781783982165/5/ch05lv11sec50/http-and-dns-load-balancer-detection>.

mohdshariq. “Identification of Web Application Firewall Using WAFW00F in Kali Linux.”

*GeeksforGeeks*, 30 June 2021, <https://www.geeksforgeeks.org/identification-of-web-application-firewall-using-wafw00f-in-kali-linux/>.

“Netmask: Kali Linux Tools.” *Kali Linux*, 5 Aug. 2022, <https://www.kali.org/tools/netmask/>.

“Nmap Commands in Kali Linux - Javatpoint.” *Www.javatpoint.com*,

<https://www.javatpoint.com/nmap-commands-in-kali-linux>.