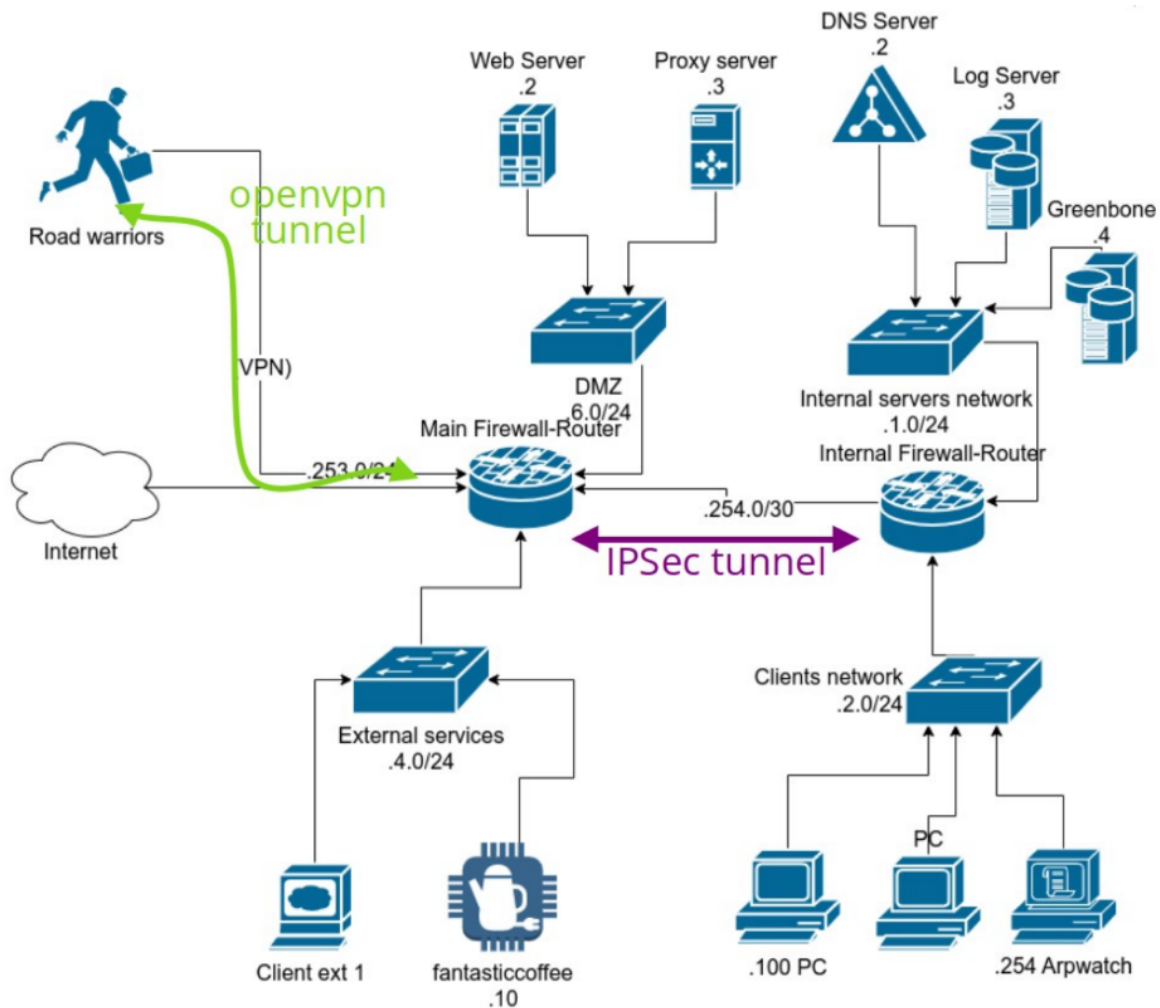


Assignment 1 Group

Student names and numbers:

- [REDACTED]
- [REDACTED]
- [REDACTED]
- [REDACTED]

The network infrastructure in which we will perform the assignments requested is the following:



INITIAL BRAINSTORMING

The assignment required us to establish two things into the network:

-A VPN at the main firewall router, for the road warriors to connect from the internet.

-An IPsec tunnel between the main firewall router and the internal firewall router.

Also, we needed to distinguish between the roles of operators and employees for the road warriors. What to do was not difficult to understand, as we knew the concept of both VPN and IPsec. For how to do it, all was about understanding how to implement them using Proxmox and OPNSense.

So, we decided the essential steps to do:

-First of all, learn how to use these two environments, to be able to understand what we were about to do.

- Then, proceed with understanding how VPN and IPSec were manageable using the environment's interfaces.
- And finally implementing what was required by the assignment, and testing it.

VPN

Access OPNSense on an interface of the main firewall (typing the ip on the url) and select VPN -> OpenVPN -> Servers to create a new VPN server. The server in this case will run on the main router since we want to secure the communications between the road warriors and the external edge of the ACME network (the main router itself). Once the packets enter the main router, they will be inside the ACME infrastructure so we will not need the VPN anymore. Therefore the endpoint devices of the VPN network we are creating will be the road warriors as clients and the main router as server.

The steps to create a VPN Server are:

- Configure it as Local User Access Server.
- Provide a certification authority and since we didn't have one we proceeded to create it (we called it ACME_ca).
- To create a new certificate for our vpn, the majority of parameters are the same as the default CA, the only thing that we changed is the name, which we called ACME_RoadWarriors_server.
- Setup the actual server with the following configurations:
 - interface: WAN since the road warrior are outside the ACME network so all their traffic will have to go through the WAN interface
 - Protocol: UDP that is the default one for OpenVPN and since we had no need of a TCP connection we chose that
 - Port: 1194 that is the openVPN port
 - IPv4 Tunnel Network: 100.100.253.0/24 that is the virtual subnet that the vpn will create
 - IPv4 Local Network: 100.100.1.0/24, 100.100.2.0/24, 100.100.4.0/24, 100.100.6.0/24 that is the list of all the ACME subnets that will be reachable from the new vpn subnet
- Check the rule to permit traffic from client connected over the internet to OpenVPN and the the rule to allow traffic to pass across VPN tunnel (we will see these rules shortly)










Now the VPN server is ready.

The assignment requires the creation of three users: "Alice", "Bob" and "Charles" and the steps are the following:

Navigate to System -> Access -> Users.

- To create a new user it is necessary to insert the name and password (we chose the form of #Name# as username and #Name123# as password for brevity) and then to check the box that makes Opnsense create the certificate for this new user. The certificate is generated using the certification authority created in the previous step (ACME_ca) and it is necessary to make each user connect to the Main Firewall Router Gateway using a VPN connection.
- The internal certificate that we use is created at the moment with the default parameters.

Each user has a role and to define the roles and the two groups that are needed by the assignment: “Operator” and “Employee” we accessed System -> Access -> Groups , defined the name of the group then assigned “Alice” to the “Operator” group and “Bob” and “Charles” to the “Employee” group.

Group name	Member Count	Description	
 admins	1	System Administrators	
 Employee	2	Cannot access Internal servers network	 
 Operator	1	Can access all the networks of the company	 

The vpn server assigns the IP addresses of his hosts dynamically choosing the IPs available in the vpn subnet pool. This makes the task of limiting their access inside the ACME network more difficult. Therefore we decided to assign to each client a specific ip address converting the dynamic method to a static one: the address 100.100.253.1 is the one of the server so it is already taken; the IPs from 100.100.253.2 to 100.100.253.14 will be assigned to the operators and the remaining part of the subnet will be assigned to employee hosts. To do so we went in the opnsense section VPN > OpenVPN > Client Specific Overrides and we added three rules (one for each road warrior existing) by specifying the server (Remote Clients vpn on UDP port 1194), the name of the road warrior (respectively Alice, Bob and Charles), the IPv4 tunnel network (100.100.253.0/24) and in the Advanced part we added the command:







```
ifconfig-push IP_ADDRESS 255.255.255.0
```

and assigned to Alice 100.100.253.5 (since she’s an operator), to Bob 100.100.253.15 and to Charles 100.100.253.16 (since they are employees).

This allows us to give each role the necessary permissions by creating aliases inside the main firewall:

To do so we followed the path Firewall -> Aliases and created two aliases:

- The alias “Employee” with content “100.100.253.15,100.100.253.16” associated with “Bob” and “Charles”
- The alias “Operator” with content “100.100.253.5” associated with “Alice”

<input type="checkbox"/> Enabled	Name	Type	Description	Content	Loaded#	Last updated	Commands
<input checked="" type="checkbox"/>	Employees	Host(s)		100.100.253.15,100.100.253.16	2	2023-05-13T09:11:00.490174	  
<input checked="" type="checkbox"/>	Operators	Host(s)		100.100.253.5	1	2023-05-13T09:11:00.474740	  

So the next step is to restrict the access only to the RoadWarriors by defining firewalls rules with the aliases now available on the WAN interface of the Main Firewall Router. Let’s see all the rules that we defined in Firewall -> Rules -> WAN:

<input type="checkbox"/>	Protocol	Source	Port	Destination	Port	Gateway	Schedule	Description ?					
<input type="checkbox"/>								Automatically generated rules					
<input type="checkbox"/>	IPv4 *	100.101.0.0/24	*	*	*	*	*	to allow opnense connection from our machine					
<input type="checkbox"/>	IPv4+6 UDP	*	*	WAN address	1194 (OpenVPN)	*	*	OpenVPN Access for the vpn users from wizard allow					

The first rule is necessary for us to work on the ACME network from our pc without being blocked by the firewall, it's not a necessary rule and we disabled it in the testing phase. The second rule it's automatically created by the openvpn wizard to allow the openvpn traffic to enter the network.

The assignment requires us to block the access from the employees to the internal server network. To accomplish this result, we defined a specific deny rule in Firewall -> Rules -> OpenVPN. This rule blocks all the traffic originating from the employees, identified by their IP addresses associated at the defined alias, and directed to the subnet corresponding to the internal server network (100.100.1.0/24).

<input type="checkbox"/>	Protocol	Source	Port	Destination	Port	Gateway	Schedule	Description ?					
<input type="checkbox"/>	IPv4 *	Employees	*	100.100.1.0/24	*	*	*						
<input type="checkbox"/>	IPv4+6 *	*	*	*	*	*	*	OpenVPN Access for the vpn users from wizard					

To test the proper working of the VPN, we need to download the certificate of a user that allows us to connect to the VPN as that host. We downloaded the certificate of Alice by navigating the path VPN -> OpenVPN -> Client Export .To use it we ran the command

```
sudo openvpn Access_for_the_vpn_users_from__Alice.ovpn
```

then we insert the Username (Alice) and the Password (Alice123) and we are in. Running the command `ip a` we verified that we obtained a new address in the VPN on the tun0 interface. We tried to ping the Main Firewall Router from the tun0 interface and inspect the packets captured with tcpdump on its tun0 interface. We can see that all of the packets use the ICMP protocol as expected.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	100.100.253.5	100.100.253.1	ICMP	88	Echo (ping) request id=0xb893, seq=1/256, ttl=64 (reply in 2)
2	0.000166	100.100.253.1	100.100.253.5	ICMP	88	Echo (ping) reply id=0xb893, seq=1/256, ttl=64 (request in 1)
3	1.009080	100.100.253.5	100.100.253.1	ICMP	88	Echo (ping) request id=0xb893, seq=2/512, ttl=64 (reply in 4)
4	1.009122	100.100.253.1	100.100.253.5	ICMP	88	Echo (ping) reply id=0xb893, seq=2/512, ttl=64 (request in 3)
5	2.011111	100.100.253.5	100.100.253.1	ICMP	88	Echo (ping) request id=0xb893, seq=3/768, ttl=64 (reply in 6)
6	2.011128	100.100.253.1	100.100.253.5	ICMP	88	Echo (ping) reply id=0xb893, seq=3/768, ttl=64 (request in 5)
7	3.046253	100.100.253.5	100.100.253.1	ICMP	88	Echo (ping) request id=0xb893, seq=4/1024, ttl=64 (reply in 8)
8	3.046283	100.100.253.1	100.100.253.5	ICMP	88	Echo (ping) reply id=0xb893, seq=4/1024, ttl=64 (request in 7)
9	4.061184	100.100.253.5	100.100.253.1	ICMP	88	Echo (ping) request id=0xb893, seq=5/1280, ttl=64 (reply in 10)
10	4.061202	100.100.253.1	100.100.253.5	ICMP	88	Echo (ping) reply id=0xb893, seq=5/1280, ttl=64 (request in 9)

This is fine because we are seeing the packets directly inside the vpn interface so they are not encapsulated. While if we do the exact same thing but dumping the packet on the tap0 interface, we can see that the packets are correctly encapsulated and encrypted:

No.	Time	Source	Destination	Protocol	Length	Info
51	2.111101	100.100.0.2	100.101.0.2	OpenVPN	150	MessageType: P_DATA_V2
52	2.111210	100.101.0.2	100.100.0.2	OpenVPN	82	MessageType: P_DATA_V2
53	2.111297	100.101.0.2	100.100.0.2	OpenVPN	150	MessageType: P_DATA_V2
91	3.161165	100.100.0.2	100.101.0.2	OpenVPN	150	MessageType: P_DATA_V2

To test the access control request, we can simply connect as an employee (Bob) and try to ping some devices inside the internal server network (100.100.1.0/24) with no response while if we access as an operator (Alice) we are able to ping this part of the network.

IPsec tunnel

In order to set up an IPsec Tunnel between the Main Firewall Router and the Internal Firewall Router we had to work on both these systems. We started from the Main Firewall Router accessing VPN -> IPsec -> Tunnel Settings and took the following steps:

- Phase 1 :
 - Select the INTERNAL interface
 - Set the remote gateway on 100.100.254.2 which is the IP of the Internal Firewall Router public interface
 - Choose Mutual PSK as Authentication Method
 - Insert a Pre-Shared Key that we generated online
 - Choose AES as Encryption algorithm
- Phase 2 :
 - Set ESP protocol for the secure association
 - Tick the box "Enable IPsec"

We followed almost the same steps on the Internal Firewall Router:

- Phase 1 :
 - Select the EXTERNAL interface
 - Set the remote gateway on 100.100.254.1 which is the IP of the Main Firewall Router interface
 - Choose Mutual PSK as Authentication Method
 - Insert the same Pre-Shared Key
 - Choose AES as Encryption algorithm
- Phase 2 :
 - Set ESP protocol for the secure association
 - Tick the box "Enable IPsec"

To test it, we logged in the VPN as Alice, and started to ping some devices located in the Clients and Internal servers network; to capture the traffic we used the command tcpdump on the main firewall and on the internal firewall. It is shown in the pictures below that, as requested, all the traffic is encapsulated into encrypted ESP packets.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	100.100.254.2	100.100.254.1	ESP	186	ESP (SPI=0xcfb06407)
2	0.000391	100.100.254.1	100.100.254.2	ESP	186	ESP (SPI=0xc59f091d)
3	0.001386	100.100.254.1	100.100.254.2	ESP	154	ESP (SPI=0xc59f091d)
4	0.102928	100.100.254.2	100.100.254.1	ESP	186	ESP (SPI=0xcfb06407)
5	0.103366	100.100.254.1	100.100.254.2	ESP	186	ESP (SPI=0xc59f091d)
6	1.104444	100.100.254.2	100.100.254.1	ESP	186	ESP (SPI=0xcfb06407)
7	1.104807	100.100.254.1	100.100.254.2	ESP	186	ESP (SPI=0xc59f091d)
8	2.106203	100.100.254.2	100.100.254.1	ESP	186	ESP (SPI=0xcfb06407)
9	2.106587	100.100.254.1	100.100.254.2	ESP	186	ESP (SPI=0xc59f091d)
10	2.729356	100.100.254.1	100.100.254.2	ESP	154	ESP (SPI=0xc59f091d)
11	2.743066	100.100.254.1	100.100.254.2	ESP	154	ESP (SPI=0xc59f091d)
12	3.107957	100.100.254.2	100.100.254.1	ESP	186	ESP (SPI=0xcfb06407)
13	3.108586	100.100.254.1	100.100.254.2	ESP	186	ESP (SPI=0xc59f091d)
14	3.745592	100.100.254.1	100.100.254.2	ESP	154	ESP (SPI=0xc59f091d)
15	4.109863	100.100.254.2	100.100.254.1	ESP	186	ESP (SPI=0xcfb06407)
16	4.110228	100.100.254.1	100.100.254.2	ESP	186	ESP (SPI=0xc59f091d)
> Frame 1: 186 bytes on wire (1488 bits), 186 bytes captured (1488 bits)						0000 d2 cb c3 2b 36 7e 3a 98 dd 40 99 44 08 00 45 00 ...+6~:~ @·D··E·
> Ethernet II, Src: 3a:98:dd:40:99:44 (3a:98:dd:40:99:44), Dst: d2:cb:c3:2b:36:7e (d2:cb:c3:2b:36:7e)						0010 00 ac 67 87 00 00 40 32 4d cc 64 64 fe 02 64 64 ...g~:~@2 M·dd·dd
> Internet Protocol Version 4, Src: 100.100.254.2, Dst: 100.100.254.1						0020 fe 01 cf b0 64 07 00 00 18 59 b7 3e de eb 97 bb ...~d~:~·T~:~<[·
> Encapsulating Security Payload						0030 83 1d 78 07 e4 53 0e 42 b0 e6 96 c6 90 6d 54 52 ...x~:~S·B ~:~·~mTR
						0040 29 84 07 0f a4 53 dd 4c 8c b0 60 a2 3c b4 0f bf)~:~·S·L ~:~·~<~:~
						0050 17 7a 44 b6 f5 ef f3 29 e5 5e 93 67 22 85 71 fa ~zD~:~:~) ^·g~"·q·
						0060 b8 37 b0 d5 45 ae de 97 1b 04 17 45 d4 6f 04 81 ~7~·E~:~ ~:~·E~o~·

Main-Firewall.pcap shown on Wireshark

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	100.100.254.2	100.100.254.1	ESP	186	ESP (SPI=0xcfb06407)
2	1.000044	100.100.254.1	100.100.254.2	ESP	186	ESP (SPI=0xc59f091d)
3	1.000101	100.100.254.1	100.100.254.2	ESP	154	ESP (SPI=0xc59f091d)
4	1.000726	100.100.254.2	100.100.254.1	ESP	186	ESP (SPI=0xcfb06407)
5	1.001661	100.100.254.1	100.100.254.2	ESP	186	ESP (SPI=0xc59f091d)
6	1.919890	100.100.254.1	100.100.254.2	ESP	154	ESP (SPI=0xc59f091d)
7	2.002001	100.100.254.2	100.100.254.1	ESP	186	ESP (SPI=0xcfb06407)
8	2.002972	100.100.254.1	100.100.254.2	ESP	186	ESP (SPI=0xc59f091d)
9	3.003599	100.100.254.2	100.100.254.1	ESP	186	ESP (SPI=0xcfb06407)
10	3.004497	100.100.254.1	100.100.254.2	ESP	186	ESP (SPI=0xc59f091d)
11	4.004756	100.100.254.2	100.100.254.1	ESP	186	ESP (SPI=0xcfb06407)
12	4.005696	100.100.254.1	100.100.254.2	ESP	186	ESP (SPI=0xc59f091d)
13	4.647691	100.100.254.1	100.100.254.2	ESP	154	ESP (SPI=0xc59f091d)
14	4.777661	100.100.254.1	100.100.254.2	ESP	154	ESP (SPI=0xc59f091d)
15	5.006465	100.100.254.2	100.100.254.1	ESP	186	ESP (SPI=0xcfb06407)
16	5.007439	100.100.254.1	100.100.254.2	ESP	186	ESP (SPI=0xc59f091d)
> Frame 1: 186 bytes on wire (1488 bits), 186 bytes captured (1488 bits)						0000 d2 cb c3 2b 36 7e 3a 98 dd 40 99 44 08 00 45 00 ...+6~:~ @·D··E·
> Ethernet II, Src: 3a:98:dd:40:99:44 (3a:98:dd:40:99:44), Dst: d2:cb:c3:2b:36:7e (d2:cb:c3:2b:36:7e)						0010 00 ac 95 45 00 00 40 32 20 0e 64 64 fe 02 64 64 ...E·~·@2 ~dd~dd
> Internet Protocol Version 4, Src: 100.100.254.2, Dst: 100.100.254.1						0020 fe 01 cf b0 64 07 00 00 17 54 e7 fb 1c 3c 5b e5 ...~d~:~·T~:~<[·
> Encapsulating Security Payload						0030 69 61 9b 91 01 cf 6a 8d e4 05 5d 3b 6a 59 ac 48 ia~:~·~j~:~·~];jY-H
						0040 84 59 99 76 02 86 94 e6 c1 78 07 10 db a4 20 3f ~Y·v~:~·~x~:~·~?
						0050 55 1d b5 d7 34 06 a9 f9 84 76 f9 6d df 0d c7 45 U~:~·4~:~·~·v~:~m~:~E

Internal-Firewall.pcap shown on Wireshark