Assignment 2 Report

Group 10

Tommaso De Nicola - 2006686
Lorenzo Colombini - 1973692
Mattia Romano - 1982886
Riccardo Capobianco - 1884636

Initial Brainstorming

We decided to create more VPN servers to efficiently divide our addresses into distint subnets to identify the user permissions through their IP Addresses.

The network assigned to the Road Warriors VPN was 100.100.253.0/24, which has been divided into the following subnets:

- 100.100.253.0/26 for the standard users
- 100.100.253.64/26 for the power user
- 100.100.253.128/26 for the administrators.

We also decided to include in the openvpn configuration the IPv6 addresses, which had been configured before on the ACME Network. The resulting subnets are:

- 2001:470:b5b8:a00::/64 for the Standard Users
- 2001:470:b5b8:a01::/64 for the Power Users
- 2001:470:b5b8:a02::/64 for the Administrators

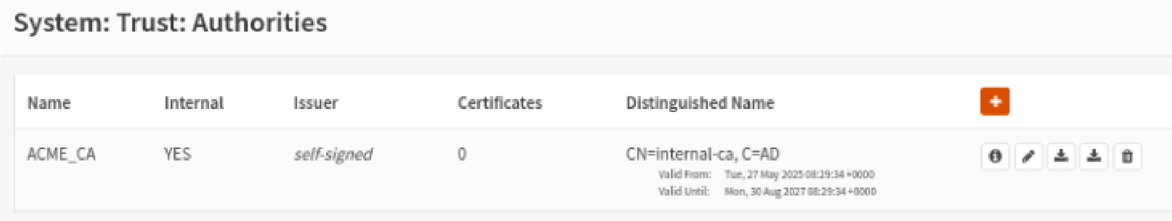
For the IPsec tunnel, we opted for a route-based implementation to meet the requirement that any packet running between the two routers should go through an IPSec tunnel. This setup involves creating an IPSec tunnel and configuring static routes on both routers to ensure that all the traffic flows through the secure tunnel.

Implementation of Road-warriors VPN

To implement the VPN we followed these steps in the Main Firewall:

1. Creation of Certification Authority and Certificates

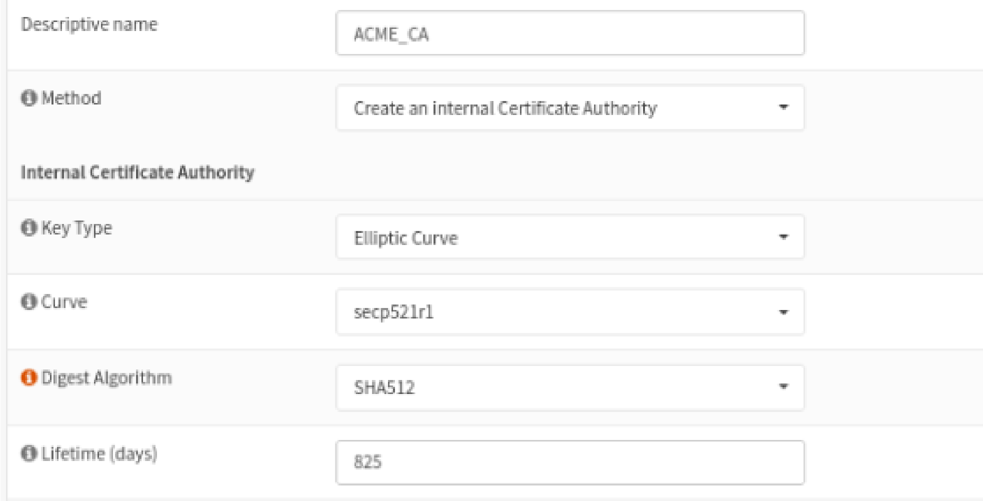
   At first, we created the CA, in *System > Trust > Authorities*:



System: Trust: Authorities

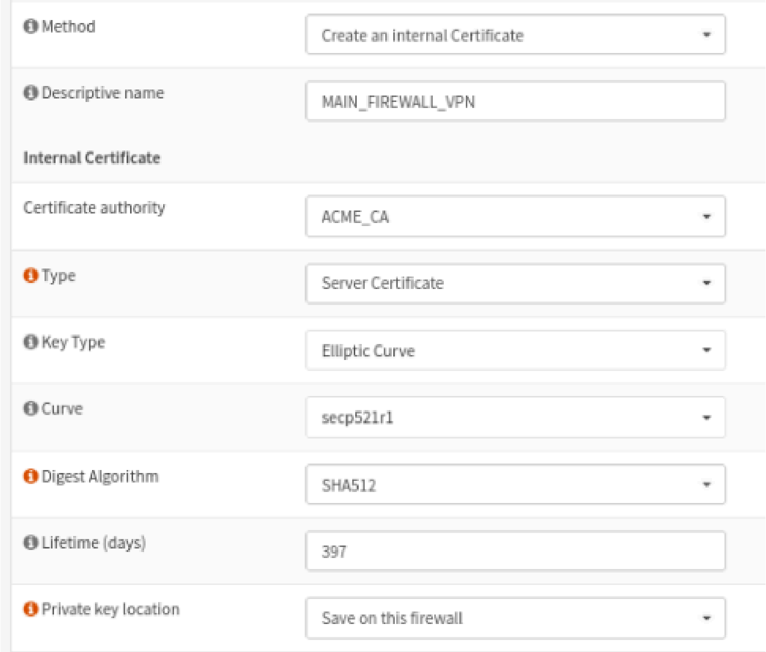
Name	Internal	Issuer	Certificates	Distinguished Name	+
ACME_CA	YES	*self-signed*	0	CN=internal-ca, C=AD Valid From: Tue, 27 May 2025 08:29:34 +0000 Valid Until: Mon, 30 Aug 2027 08:29:34 +0000	

Which has been created this way:

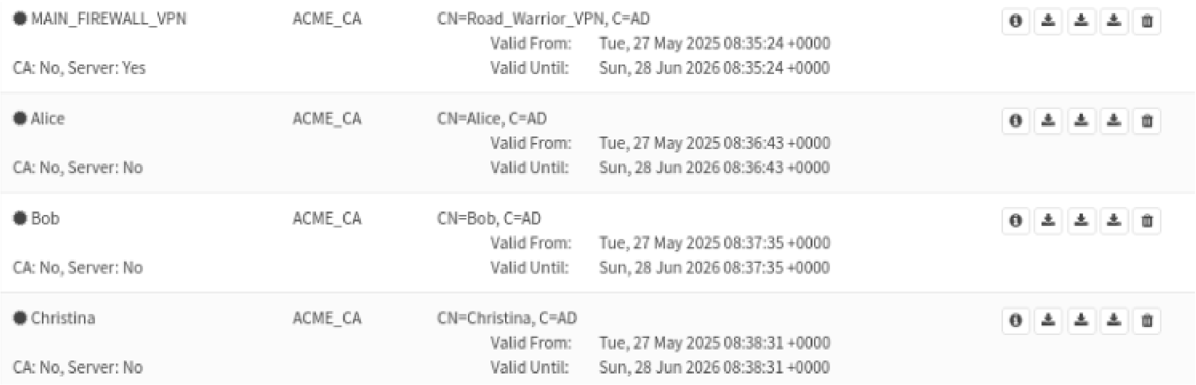


Descriptive name	ACME_CA
Method	Create an internal Certificate Authority
Internal Certificate Authority	
Key Type	Elliptic Curve
Curve	secp521r1
Digest Algorithm	SHA512
Lifetime (days)	825

Then we proceeded with the certificates in *System > Trust > Certificates*, one for the server and one for each user. The parameters for the certificates are the following:



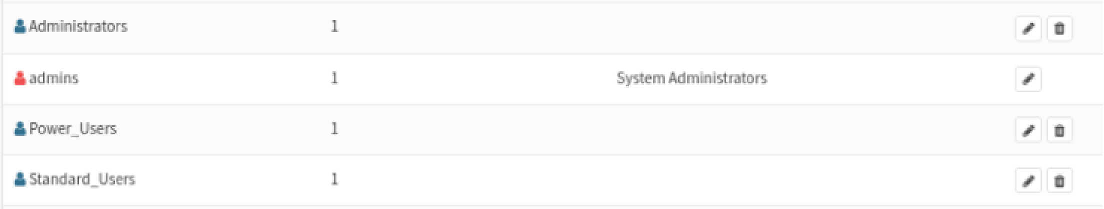
Method	Create an internal Certificate
Descriptive name	MAIN_FIREWALL_VPN
Internal Certificate	
Certificate authority	ACME_CA
Type	Server Certificate
Key Type	Elliptic Curve
Curve	secp521r1
Digest Algorithm	SHA512
Lifetime (days)	397
Private key location	Save on this firewall



MAIN_FIREWALL_VPN CA: No, Server: Yes	ACME_CA	CN=Road_Warrior_VPN, C=AD Valid From: Tue, 27 May 2025 08:35:24 +0000 Valid Until: Sun, 28 Jun 2026 08:35:24 +0000
Alice CA: No, Server: No	ACME_CA	CN=Alice, C=AD Valid From: Tue, 27 May 2025 08:36:43 +0000 Valid Until: Sun, 28 Jun 2026 08:36:43 +0000
Bob CA: No, Server: No	ACME_CA	CN=Bob, C=AD Valid From: Tue, 27 May 2025 08:37:35 +0000 Valid Until: Sun, 28 Jun 2026 08:37:35 +0000
Christina CA: No, Server: No	ACME_CA	CN=Christina, C=AD Valid From: Tue, 27 May 2025 08:38:31 +0000 Valid Until: Sun, 28 Jun 2026 08:38:31 +0000

2. Creation of Users and Groups

After setting up the certificates, we proceeded by creating the groups for the users, in *System > Access > Groups* (admin is default):

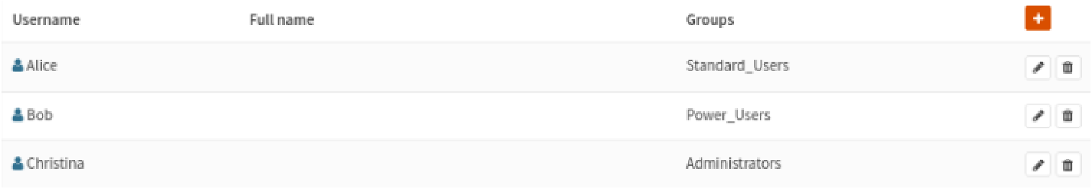


Administrators	1	
admins	1	System Administrators
Power_Users	1	
Standard_Users	1	

Then we created the 3 users through the *System > Access > Users* menu, using the following passwords, generated in a secure way :

- Alice: ***i3c5:6?Q2H,w***
- Bob: ***c#<2232YfawZ***
- Christine: ***-e>\N3P9r!5£***

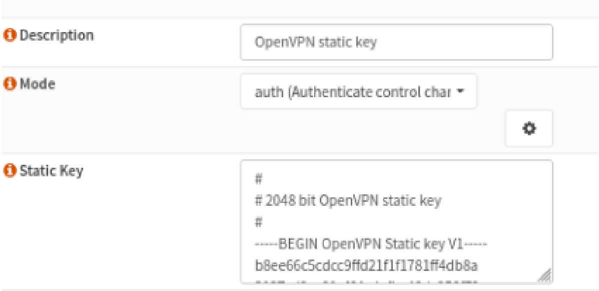
Here is the Users interface:



Username	Full name	Groups
Alice		Standard_Users
Bob		Power_Users
Christina		Administrators

3. Creation of the VPN Servers

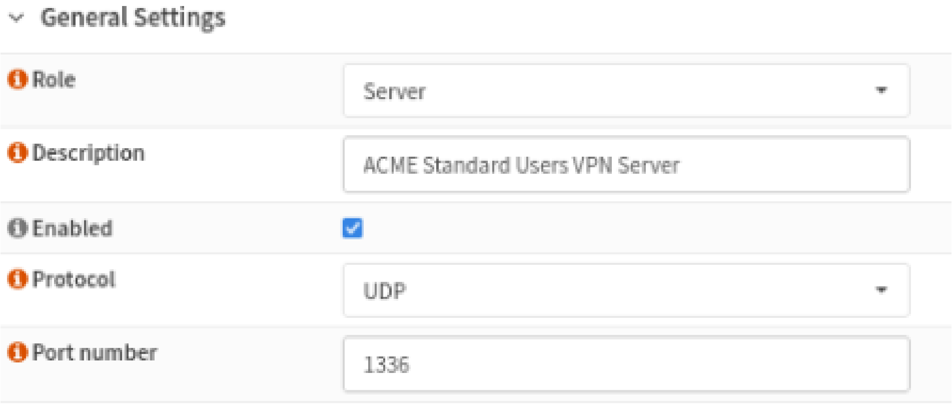
We then created the Static key to authenticate the VPN server through *VPN > OpenVPN > Instances > Static Keys*, as shown in the image:



Description	OpenVPN static key
Mode	auth (Authenticate control char
Static Key	# # 2048 bit OpenVPN static key # -----BEGIN OpenVPN Static key V1----- b8ee66c5cdcc9ffd21f1f1781ff4db8a

After obtaining the key, we needed to create the servers for every type of user, as described in the initial brainstorming.

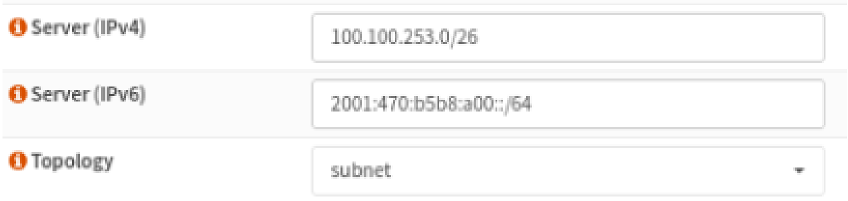
We started from the Standard Users instance, as follows:



General Settings

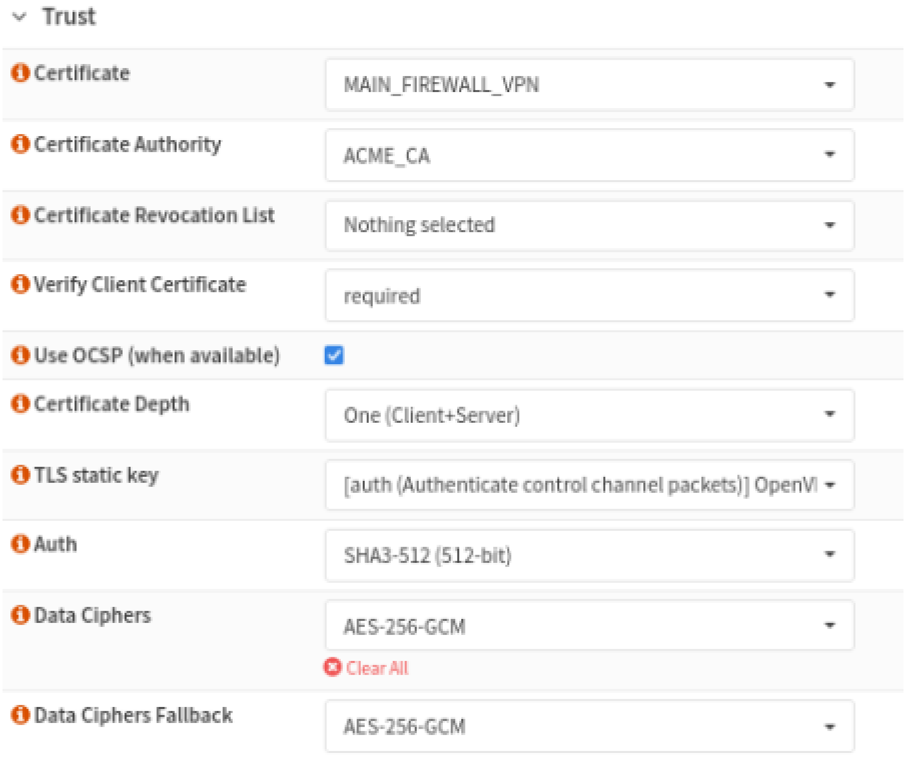
Role	Server
Description	ACME Standard Users VPN Server
Enabled	☑
Protocol	UDP
Port number	1336

We used port 1336 for the standard users, incrementing to 1337 and 1338 for the other user groups. We had to use different ports because the VPN servers will run on the same interface. The IPv4 and IPv6 of Standard users will be into the following pools:



Server (IPv4)	100.100.253.0/26
Server (IPv6)	2001:470:b5b8:a00::/64
Topology	subnet

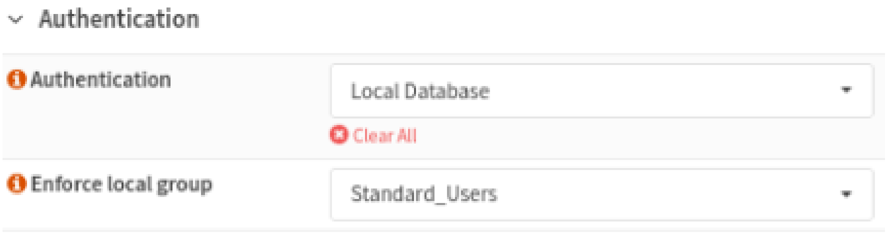
In Trust settings we changed the ciphers and auth to enhance security:



Trust

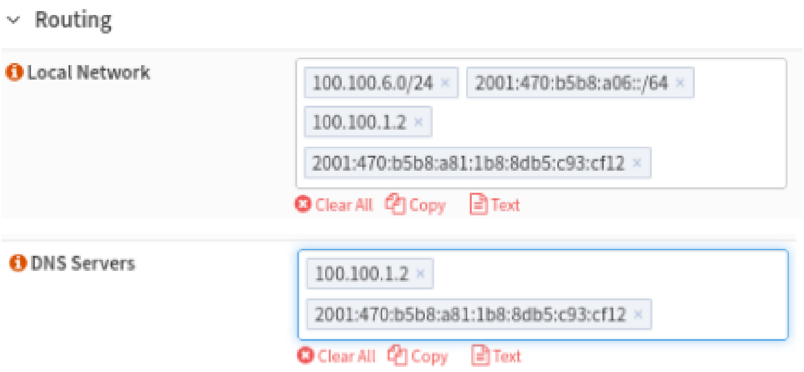
Certificate	MAIN_FIREWALL_VPN
Certificate Authority	ACME_CA
Certificate Revocation List	Nothing selected
Verify Client Certificate	required
Use OCSP (when available)	☑
Certificate Depth	One (Client+Server)
TLS static key	[auth (Authenticate control channel packets)] OpenVI
Auth	SHA3-512 (512-bit)
Data Ciphers	AES-256-GCM Clear All
Data Ciphers Fallback	AES-256-GCM

Finally, we set the certificate to use to authenticate and the networks through which the users can navigate. We allowed them to use the DNS server in order to resolve hostnames of the machines:



Authentication

Authentication	Local Database Clear All
Enforce local group	Standard_Users

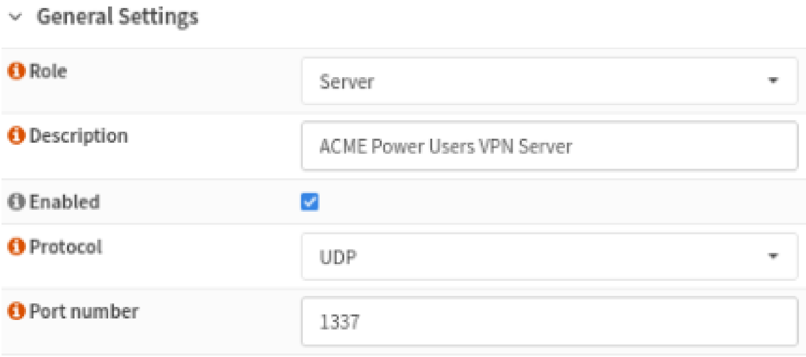


Routing

Local Network
100.100.6.0/24 ×
2001:470:b5b8:a06::/64 ×
100.100.1.2 ×
2001:470:b5b8:a81:1b8:8db5:c93:cf12 ×

DNS Servers
100.100.1.2 ×
2001:470:b5b8:a81:1b8:8db5:c93:cf12 ×

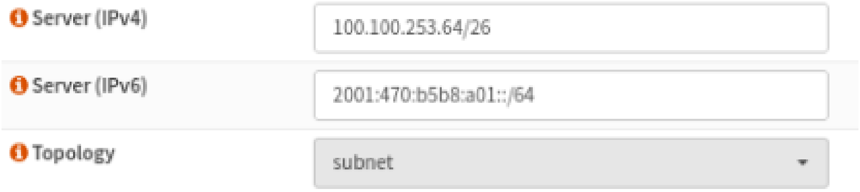
For Power Users, the configuration is very similar. We changed the description of the servers and the port, which has been increased:



General Settings

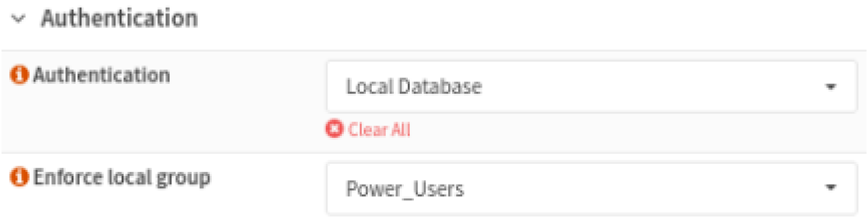
Role	Server
Description	ACME Power Users VPN Server
Enabled	☑
Protocol	UDP
Port number	1337

Then we changed the address pool:



Server (IPv4)	100.100.253.64/26
Server (IPv6)	2001:470:b5b8:a01::/64
Topology	subnet

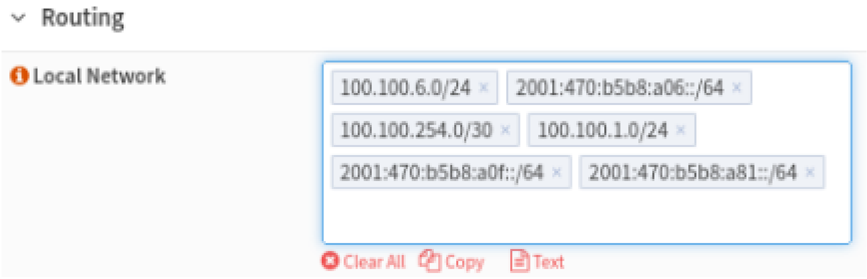
The Trust menu is the same; we just inserted the right certificate into the authentication settings:



Authentication

Authentication	Local Database
Enforce local group	Power_Users

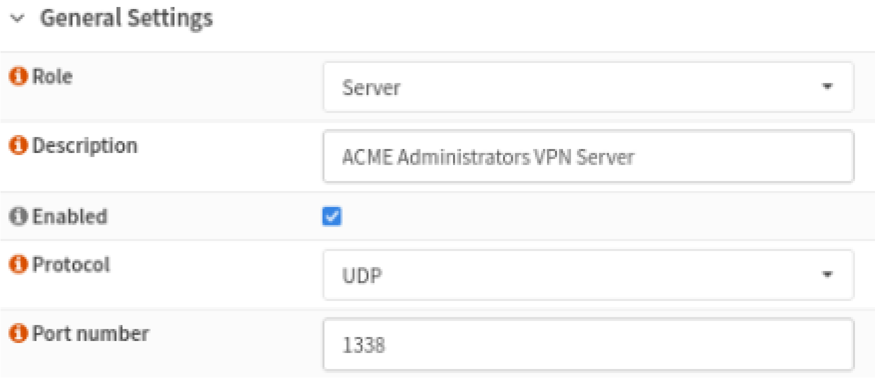
The Routing part has been adjusted too, while the DNS is the same.



Routing

Local Network
100.100.6.0/24 ×
2001:470:b5b8:a06::/64 ×
100.100.254.0/30 ×
100.100.1.0/24 ×
2001:470:b5b8:a0f::/64 ×
2001:470:b5b8:a81::/64 ×

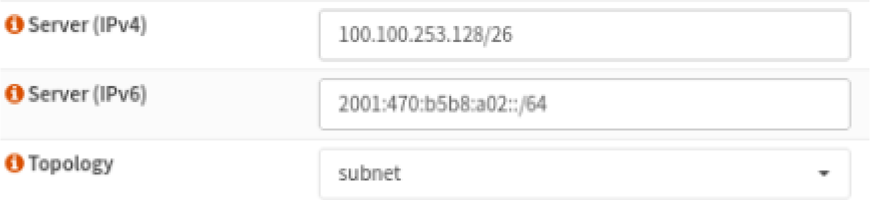
Finally, for the Administrators, we created the VPN server on port 1338:



General Settings

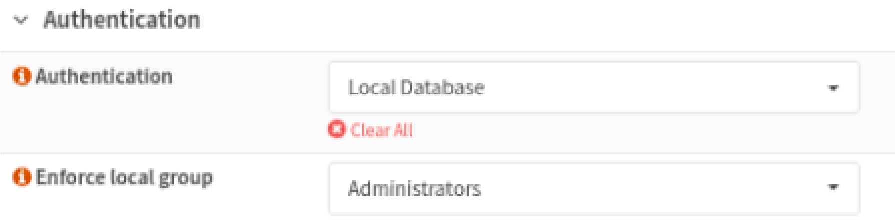
Role	Server
Description	ACME Administrators VPN Server
Enabled	☑
Protocol	UDP
Port number	1338

We assigned the following address pool:



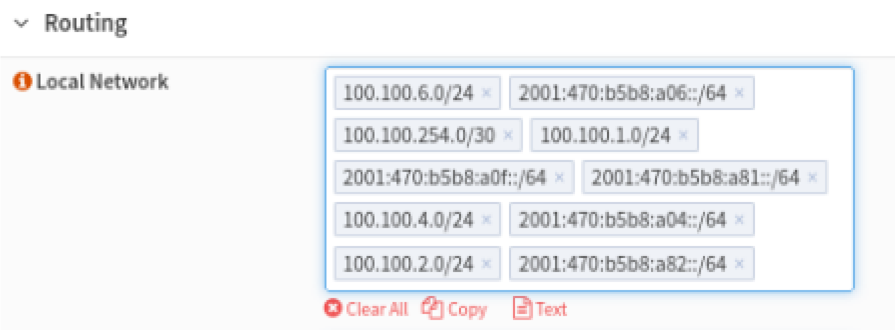
Server (IPv4)	100.100.253.128/26
Server (IPv6)	2001:470:b5b8:a02::/64
Topology	subnet

The Trust configurations are the same, but we changed the certificate used and the routing settings, keeping the DNS settings unchanged



Authentication

Authentication	Local Database
Enforce local group	Administrators

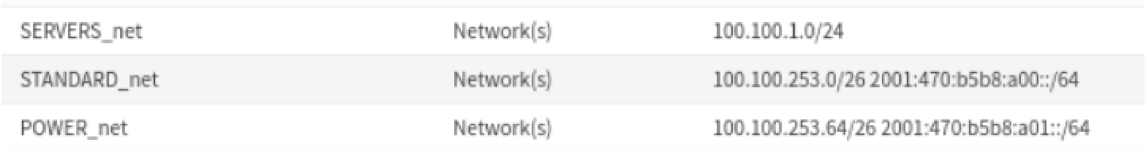


Routing

Local Network	100.100.6.0/24 2001:470:b5b8:a06::/64 100.100.254.0/30 100.100.1.0/24 2001:470:b5b8:a0f::/64 2001:470:b5b8:a81::/64 100.100.4.0/24 2001:470:b5b8:a04::/64 100.100.2.0/24 2001:470:b5b8:a82::/64

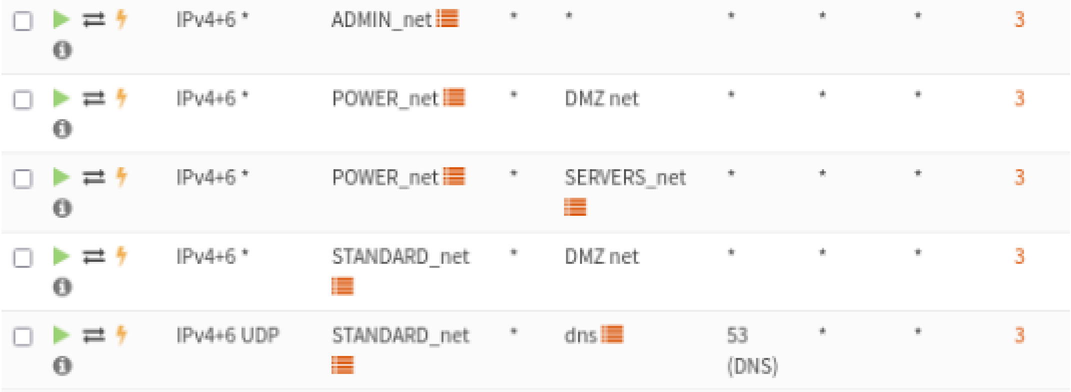
4. Creation of Aliases and Firewall Rules

At first we created the 3 aliases for the 3 types of users



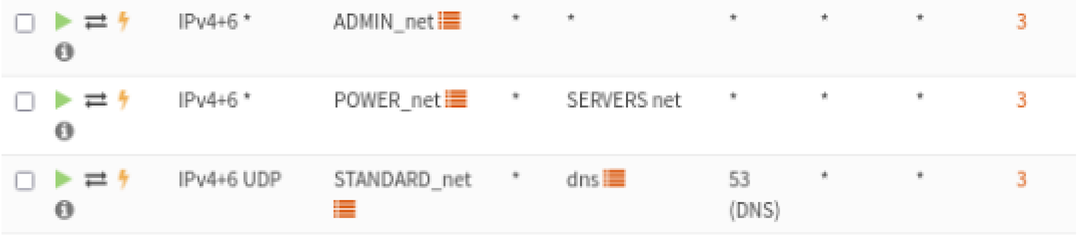
SERVERS_net	Network(s)	100.100.1.0/24
STANDARD_net	Network(s)	100.100.253.0/26 2001:470:b5b8:a00::/64
POWER_net	Network(s)	100.100.253.64/26 2001:470:b5b8:a01::/64

Then we added the following floating rules on the *main firewall*:



☐	▶ ⇄ ⚡ ℹ	IPv4+6 *	ADMIN_net ☰	*	*	*	*	*	3
☐	▶ ⇄ ⚡ ℹ	IPv4+6 *	POWER_net ☰	*	DMZ net	*	*	*	3
☐	▶ ⇄ ⚡ ℹ	IPv4+6 *	POWER_net ☰	*	SERVERS_net ☰	*	*	*	3
☐	▶ ⇄ ⚡ ℹ	IPv4+6 *	STANDARD_net ☰	*	DMZ net	*	*	*	3
☐	▶ ⇄ ⚡ ℹ	IPv4+6 UDP	STANDARD_net ☰	*	dns ☰	53 (DNS)	*	*	3

And the following ones on the *internal firewall*:



☐	▶ ⇄ ⚡ ℹ	IPv4+6 *	ADMIN_net ☰	*	*	*	*	*	3
☐	▶ ⇄ ⚡ ℹ	IPv4+6 *	POWER_net ☰	*	SERVERS net	*	*	*	3
☐	▶ ⇄ ⚡ ℹ	IPv4+6 UDP	STANDARD_net ☰	*	dns ☰	53 (DNS)	*	*	3

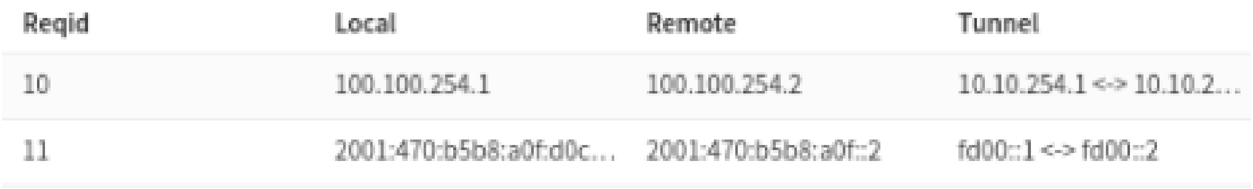
With these rules, we allow standard users just to use the DNS server, blocking any other connections outside the DMZ network. For power users we allowed DMZ and Servers networks, as written in the assignment, while for the administrators we allowed full access.

Implementation of IPSec Tunnel

IPsec setup in OPNsense can be divided into two phases:

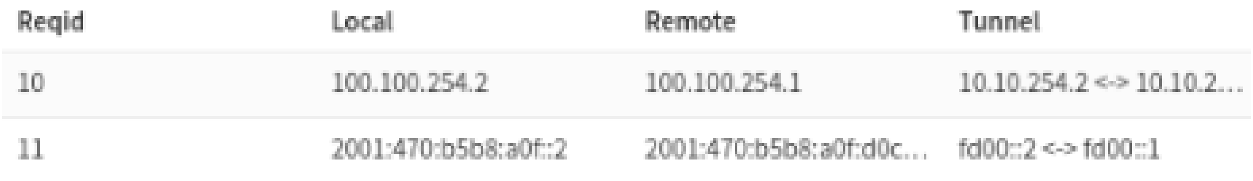
1. authentication and creation of a secure channel
2. encryption and encapsulation of packets into ESP frames.

We started with the configuration in *VPN > IPsec > Virtual Tunnel Interfaces*, adding one interface for IPv4 and one for IPv6. Here we specify the tunnel addresses which are essential as they fill in the source and destination IP fields in the ESP header, which encapsulates the encrypted IP packet.
In the Main Firewall:



Reqid	Local	Remote	Tunnel
10	100.100.254.1	100.100.254.2	10.10.254.1 <-> 10.10.2...
11	2001:470:b5b8:a0f:d0c...	2001:470:b5b8:a0f::2	fd00::1 <-> fd00::2

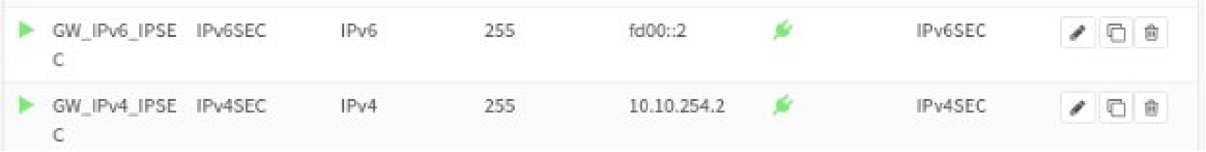
In the Internal Firewall:



Reqid	Local	Remote	Tunnel
10	100.100.254.2	100.100.254.1	10.10.254.2 <-> 10.10.2...
11	2001:470:b5b8:a0f::2	2001:470:b5b8:a0f:d0c...	fd00::2 <-> fd00::1

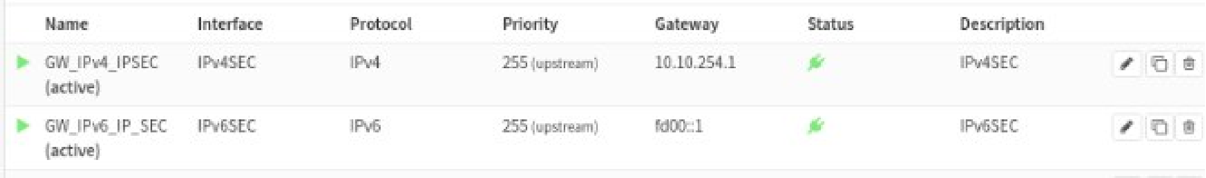
We continued the configuration on System > Gateways > Configuration

In the Main Firewall:



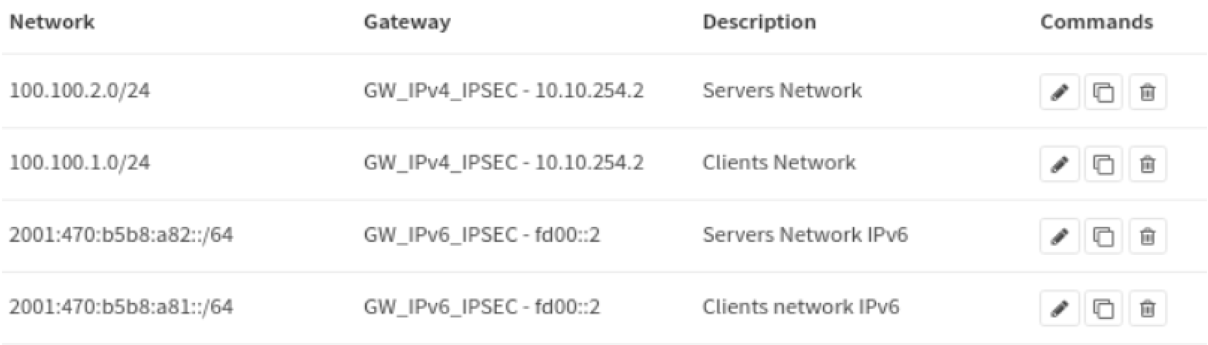
▶	GW_IPv6_IPSEC	IPv6SEC	IPv6	255	fd00::2	🔌	IPv6SEC	✎ ⧉ 🗑
▶	GW_IPv4_IPSEC	IPv4SEC	IPv4	255	10.10.254.2	🔌	IPv4SEC	✎ ⧉ 🗑

In the Internal Firewall



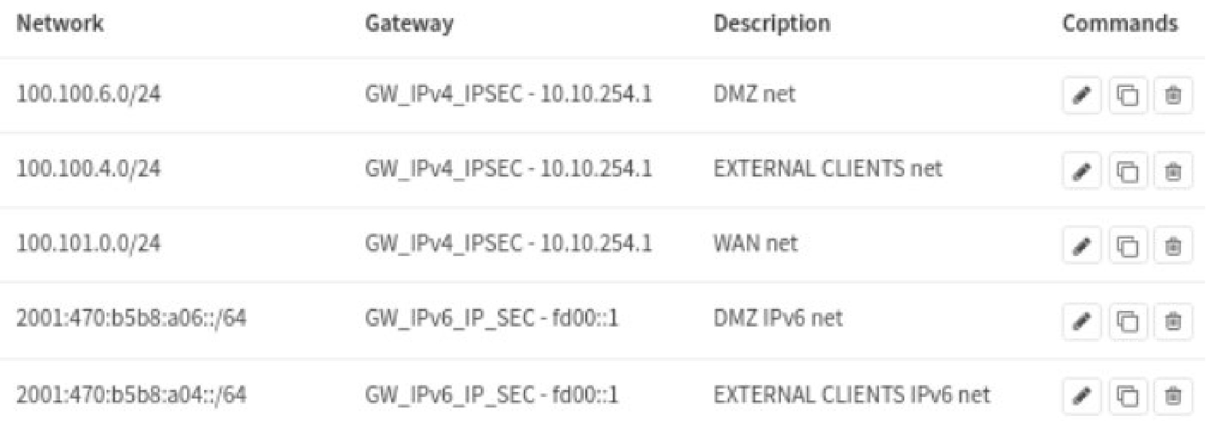
	Name	Interface	Protocol	Priority	Gateway	Status	Description	
▶	GW_IPv4_IPSEC (active)	IPv4SEC	IPv4	255 (upstream)	10.10.254.1	🔌	IPv4SEC	✎ ⧉ 🗑
▶	GW_IPv6_IP_SEC (active)	IPv6SEC	IPv6	255 (upstream)	fd00::1	🔌	IPv6SEC	✎ ⧉ 🗑

The “Upstream Gateway” option is enabled only in the internal firewall because the only way packets have to go from the internal to the internet is to pass through the Main Firewall.
In order to use the gateways, we inserted new static routes in System > Routes > Configuration. In the Main Firewall, we have:



Network	Gateway	Description	Commands
100.100.2.0/24	GW_IPv4_IPSEC - 10.10.254.2	Servers Network	✎ ⧉ 🗑
100.100.1.0/24	GW_IPv4_IPSEC - 10.10.254.2	Clients Network	✎ ⧉ 🗑
2001:470:b5b8:a82::/64	GW_IPv6_IPSEC - fd00::2	Servers Network IPv6	✎ ⧉ 🗑
2001:470:b5b8:a81::/64	GW_IPv6_IPSEC - fd00::2	Clients network IPv6	✎ ⧉ 🗑

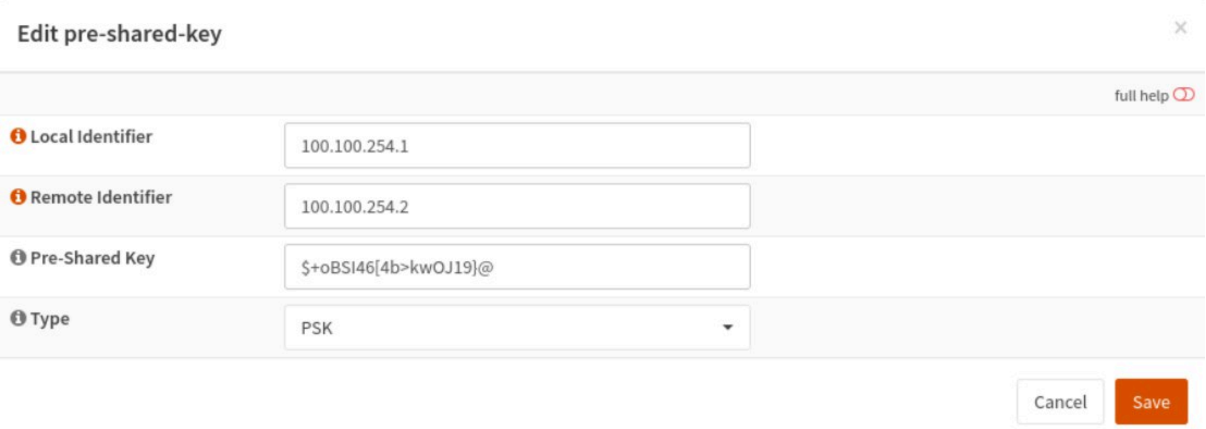
We must do this because we need to redirect all the traffic from the Internal interface of the main firewall to the External interface of the internal firewall to the IPsec interface. We just need to specify the interface and the IP address of the other IPsec endpoint. This way, if the packet is destined to one of the internal networks, it will go through the IPsec tunnel



Network	Gateway	Description	Commands
100.100.6.0/24	GW_IPv4_IPSEC - 10.10.254.1	DMZ net	✎ ⧉ 🗑
100.100.4.0/24	GW_IPv4_IPSEC - 10.10.254.1	EXTERNAL CLIENTS net	✎ ⧉ 🗑
100.101.0.0/24	GW_IPv4_IPSEC - 10.10.254.1	WAN net	✎ ⧉ 🗑
2001:470:b5b8:a06::/64	GW_IPv6_IP_SEC - fd00::1	DMZ IPv6 net	✎ ⧉ 🗑
2001:470:b5b8:a04::/64	GW_IPv6_IP_SEC - fd00::1	EXTERNAL CLIENTS IPv6 net	✎ ⧉ 🗑

This way, if the packet is destined to an external network, it will go through the IPsec tunnel.

The last step to use the IPsec VPN is to set up the connection and the authentication method. We started from VPN > IPsec > Pre-Shared Keys:



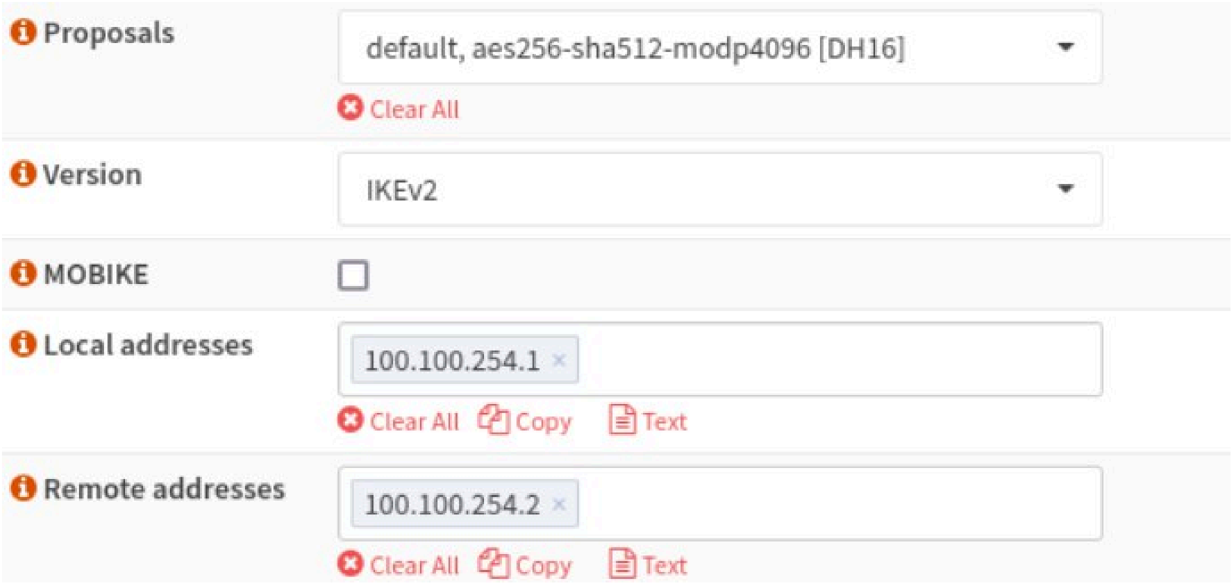
Edit pre-shared-key

Local Identifier	100.100.254.1
Remote Identifier	100.100.254.2
Pre-Shared Key	$+oBSI46[4b>kwOJ19}@
Type	PSK

Both firewalls have the Mutual Pre-Shared Key (PSK): *$+oBSI46[4b>kwOJ19}@*

Both are also using IP addresses as identifiers and AES-256-GCM with 128 bit ICV + SHA512 + Diffie Hellman Key Group 16 (4096) bits as encryption algorithms.

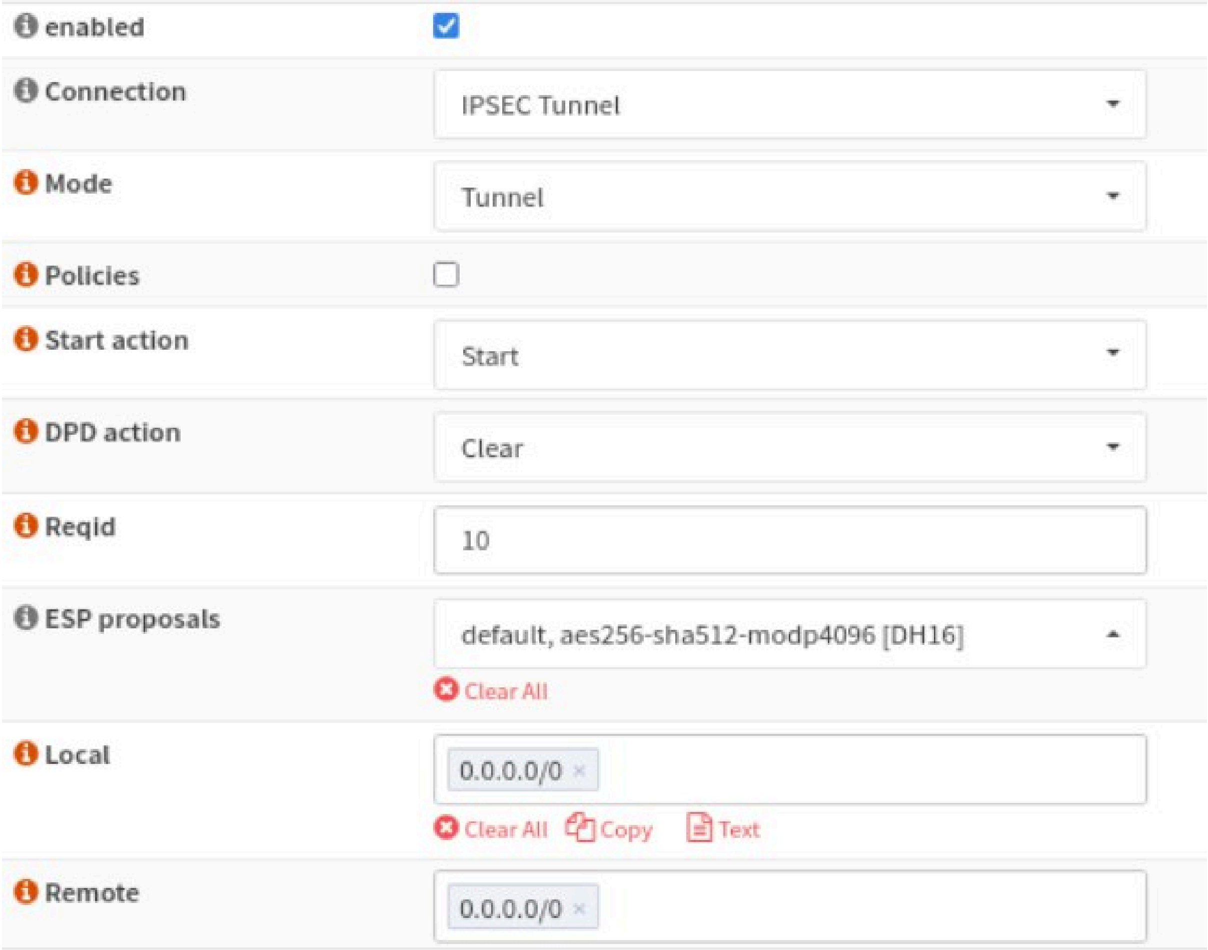
We finally setup the connection in VPN > IPsec > Connections by simply specifying the network addresses, the version IKEv2, the Phase 1 proposals and the PSK to use. On the Main we have:



Proposals	default, aes256-sha512-modp4096 [DH16]
Version	IKEv2
MOBIKE	□
Local addresses	100.100.254.1
Remote addresses	100.100.254.2

The Internal Firewall has specular addresses.

Then we selected the created PSK for Local and Remote authentication. Lastly, we created a children with a security policy that matches all the traffic, so everything that enters the tunnel gets encrypted.



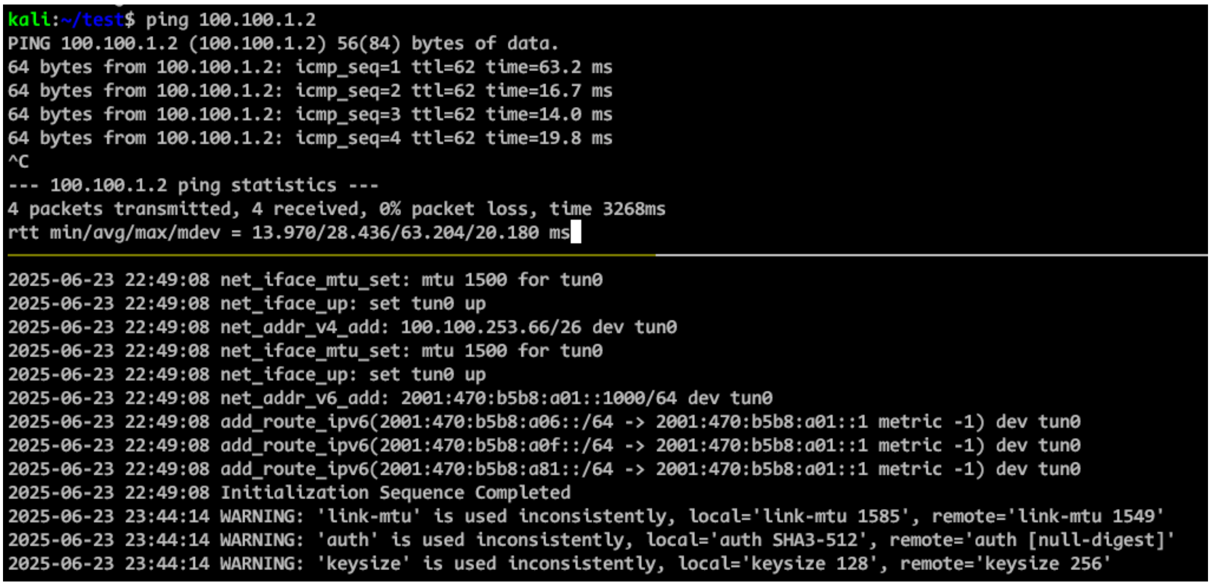
enabled	☑
Connection	IPSEC Tunnel
Mode	Tunnel
Policies	□
Start action	Start
DPD action	Clear
Reqid	10
ESP proposals	default, aes256-sha512-modp4096 [DH16]
Local	0.0.0.0/0
Remote	0.0.0.0/0

Once the tunnel is established and the routers are authenticated, they can start encrypting the traffic encapsulating every packet in an ESP frame.

To start the tunnel, we just go to VPN > IPsec > Connections, enable the IPSec Tunnel, and we finally apply.

Testing of Road-warriors VPN & IPSec Tunnel

Using the Bob's profile (Power User) we can check the connettivity between the user and the DNS to ensure that both the openVPN and IPSeC works



```
kali:~/test$ ping 100.100.1.2
PING 100.100.1.2 (100.100.1.2) 56(84) bytes of data.
64 bytes from 100.100.1.2: icmp_seq=1 ttl=62 time=63.2 ms
64 bytes from 100.100.1.2: icmp_seq=2 ttl=62 time=16.7 ms
64 bytes from 100.100.1.2: icmp_seq=3 ttl=62 time=14.0 ms
64 bytes from 100.100.1.2: icmp_seq=4 ttl=62 time=19.8 ms
^C
--- 100.100.1.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3268ms
rtt min/avg/max/mdev = 13.970/28.436/63.204/20.180 ms
```

```
2025-06-23 22:49:08 net_iface_mtu_set: mtu 1500 for tun0
2025-06-23 22:49:08 net_iface_up: set tun0 up
2025-06-23 22:49:08 net_addr_v4_add: 100.100.253.66/26 dev tun0
2025-06-23 22:49:08 net_iface_mtu_set: mtu 1500 for tun0
2025-06-23 22:49:08 net_iface_up: set tun0 up
2025-06-23 22:49:08 net_addr_v6_add: 2001:470:b5b8:a01::1000/64 dev tun0
2025-06-23 22:49:08 add_route_ipv6(2001:470:b5b8:a06::/64 -> 2001:470:b5b8:a01::1 metric -1) dev tun0
2025-06-23 22:49:08 add_route_ipv6(2001:470:b5b8:a0f::/64 -> 2001:470:b5b8:a01::1 metric -1) dev tun0
2025-06-23 22:49:08 add_route_ipv6(2001:470:b5b8:a81::/64 -> 2001:470:b5b8:a01::1 metric -1) dev tun0
2025-06-23 22:49:08 Initialization Sequence Completed
2025-06-23 23:44:14 WARNING: 'link-mtu' is used inconsistently, local='link-mtu 1585', remote='link-mtu 1549'
2025-06-23 23:44:14 WARNING: 'auth' is used inconsistently, local='auth SHA3-512', remote='auth [null-digest]'
2025-06-23 23:44:14 WARNING: 'keysize' is used inconsistently, local='keysize 128', remote='keysize 256'
```

Final Remarks

We just added a floating rule on both firewalls to allow all the traffic through the IPsec interfaces. More specific decisions are taken before forwarding / after receiving the packet to / from the interface. We repeated all the tests (of the first two assignments) after configuring the VPNs and everything worked as expected.