ACME_19_a2_report

1 Team Info

Team Name: Guardians of the Gateway

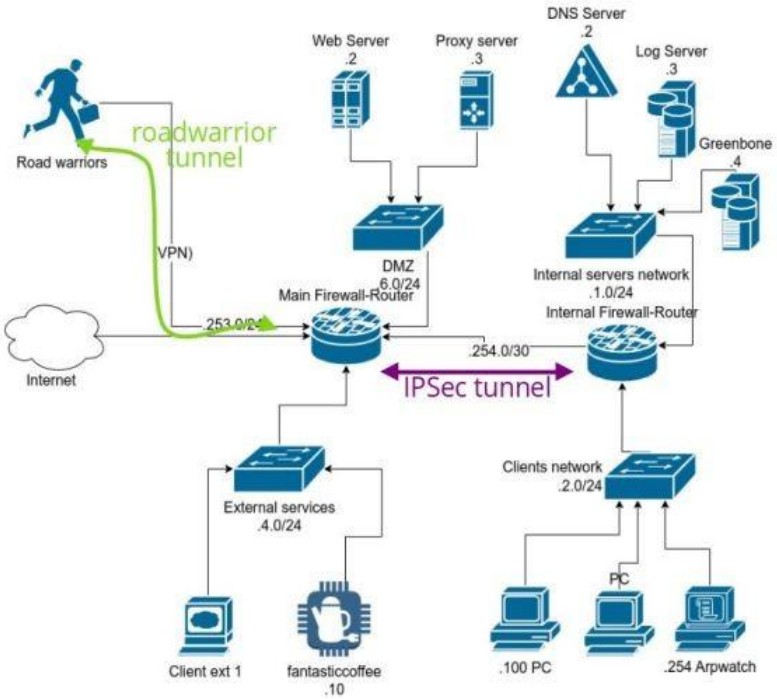
Team Number: 19 Team

Members:

- Simone Ciferri
- Matteo Concutelli
- Giorgio Pesce

2 Initial Approach

Our approach begins with the creation of a Certificate Authority. This CA will be used to issue digital certificates to both the client and the server.
Following the establishment of the CA, we will configure the OpenVPN server on OPNsense, our chosen firewall and routing platform.
The server will be set up to handle various user access levels, ensuring that sensitive areas of the network are protected while still providing necessary access to different user groups. Users will be categorized into two groups: operators and employees.
Firewall rules will be meticulously configured to allow the appropriate traffic to flow between the VPN clients and the internal network, ensuring seamless and secure connectivity.



3 Road Warrior Tunnel

3.1 Creating CA and Server Certificate

The first step in the configuration process involves creating a Certificate Authority (CA) on OPNsense. This CA is crucial as it will be responsible for issuing the certificates that secure the VPN connections. For the key type, RSA was chosen with key lenght of 2048 bits. The digest algorithm selected is SHA512, which offers enhanced security. Following the creation of the CA, the next task was to create a server certificate. This certificate is essential for the clients to verify the server's identity when they attempt to connect.

3.2 Adding Users

Alice was added to the operators group, which grants her comprehensive access to the entire ACME network. On the other hand, Bob and Charles were placed in the employees group, which is more restricted.
We configured the VPN such that Alice has permission to access all network segments within ACME. This includes all subnets, allowing her to connect to any part of the network without restrictions. Conversely, Bob and Charles have limited access: they are explicitly denied entry to the subnet that hosts the internal servers.
We then proceeded to create client certificates to log into the OpenVPN server.

alice	alice sebastiani	Operators
bob	bob marley	Employee
charles	charles bukowski	Employee
root	System Administrator	admins

Additionally, to enhance security, we enabled password authentication for all users. This means that, along with their certificates, users must also provide a valid password to establish a VPN connection.

alice:mitostemmacerulea
bob:passifloracerulea charles:tacsoniacerulea

3.3 VPN server configuration

With the foundational elements in place, the next step was to configure the OpenVPN server on the OPNsense firewall. We configured it to support remote access via SSL/TLS, which is suitable for our needs. We chose UDP as the protocol for its speed advantages and set the device to tun, which establishes a routed IP tunnel. The interface was set to WAN, allowing connections from the internet, and the port was set to 1194, a commonly used port for OpenVPN. TLS authentication was enabled, and, for encryption, we opted for AES-256-GCM, known for its strong security features, and SHA512 was chosen as the auth digest algorithm to ensure data integrity.
Next, we defined the network settings for the VPN tunnel. The IPv4 tunnel network was set to 192.168.10.0/24, which assigns IP addresses to VPN clients.
We limited the maximum number of clients to 10 to mitigate potential Denial of Service (DoS) attacks Since we added authentication and checked the option "Strict User/CN Matching", if we try to log to the OpenVPN server with our credentials but with the .openvpn file of another user, it won't work:

Date	Severity	Process	Line
2024-06-04T13:33:47	Warning	openvpn	Username does not match certificate common name (bob != charles), access denied.
2024-06-04T13:33:35	Warning	openvpn	Username does not match certificate common name (bob != charles), access denied.
2024-06-04T13:33:17	Warning	openvpn	Username does not match certificate common name (bob != charles), access denied.

3.4 Addition of firewall rules

To ensure that traffic flows correctly between VPN clients and the internal network, we added the necessary firewall rules. On the WAN interface, we created a rule to allow incoming traffic on the VPN port, specifically UDP traffic on port 1194. This rule ensures that OpenVPN traffic is permitted through the firewall. Additionally, on the OpenVPN interface, we added a rule to allow traffic from VPN clients to the ACME network.

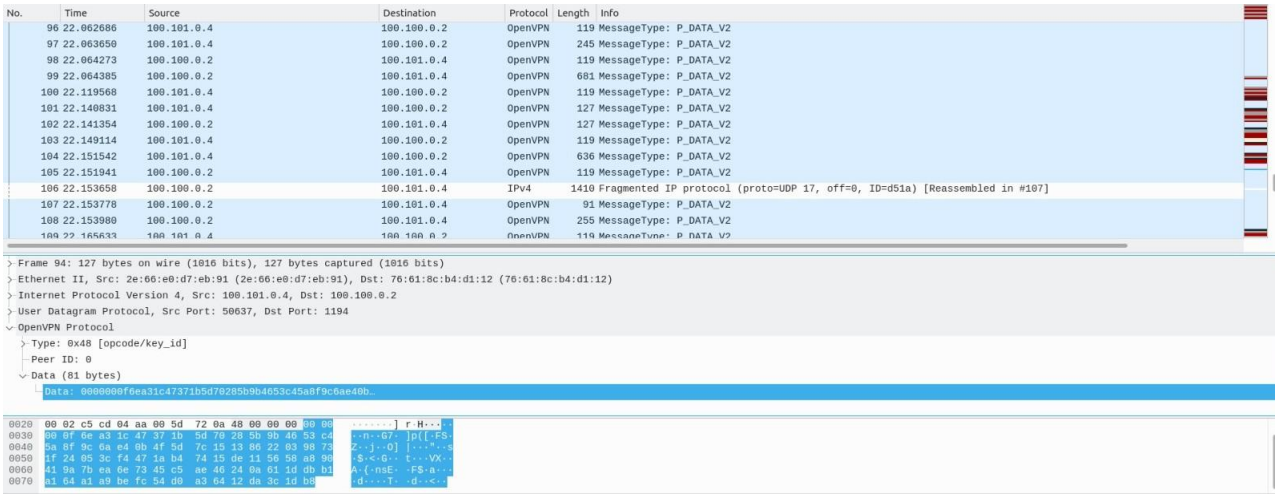
Firewall	Interface	Direction	Source	Destination	Protocol	Port	Action
Main	WAN	in	any	WAN address	UDP - IPv4	1194	Pass
Main	OpenVPN	in	OpenVPN net	any	any - IPv4	any	Pass

Table 1: Allow OpenVPN to ACME

3.5 Testing phase

For testing, we first verify that the OpenVPN interface is actually working, by sending some pings from "bob" to the web server:

Next, we check that the encryption process is actually working by capturing some packets:



3.6 Prevent Restriction Bypass

After the configuration, we noticed that users belonging to the "employees" group, even if they don't have route to the SERVERS net, can manually add the route using the *ip route add* command. By doing so, they're able to access the SERVERS net. To avoid this, we decided to assign static IP address to every user:

- alice: 192.168.10.2
- bob: 192.168.10.3

- charles: 192.168.10.4

Now we can add explicit firewalls rules to block employees from accessing the SERVERS net:

Firewall	Interface	Direction	Source	Destination	Protocol	Port	Action
Main	OpenVPN	in	bob	SERVERS net	any - IPv4	any	Block
Main	OpenVPN	in	charles	SERVERS net	any - IPv4	any	Block

Table 2: Block employees from accessing SERVERS net

Now, if bob tries to manually add a route to the SERVERS net and perform, for example, the *host* command, it won't succed:

```
┌──(kali㉿kali)-[~/Desktop/PND/assigment2_tests]
└─$ sudo ip route add 100.100.1.0/24 via 192.168.10.1 dev tun0
[sudo] password for kali:

┌──(kali㉿kali)-[~/Desktop/PND/assigment2_tests]
└─$ host webserver 100.100.1.2
^C
```

4 IPSec Tunnel

4.1 Configuration

First, we identified the interfaces that required configuration for the IPsec tunnel. These were the INTERNAL interface on the Main firewall and the EXTERNAL interface on the Internal firewall.

Next, we addressed the necessary firewall rules. For the IPsec tunnel to function correctly, specific protocols and ports needed to be allowed. These included the ESP protocol, UDP port 500 (ISAKMP), and UDP port 4500 (NAT-T). However, due to the existing "allow everything" rule between the two firewalls that we added during the ACME firewall configuration of the first assignment, no additional changes were required.

This configuration process included defining Phase 1 and Phase 2 settings for each endpoint. For Phase 1, we generated a pre-shared key and configured the necessary encryption and authentication parameters. In particular we chose aes256gcm16 as the encryption algorithm and SHA512 as the hash algorithm.
Phase 2 involved specifying the local and remote networks that needed to communicate through the tunnel. In particular, for the Main firewall:

Type	Local Subnet	Remote Subnet	Phase 2 Proposal	Description
ESP IPv4 tunnel	DMZ	100.100.2.0/24	aes256gcm16 + SHA512+ DH Group 14	CCP - Internal - DMZ to CLIENTS
ESP IPv4 tunnel	DMZ	100.100.1.0/24	aes256gcm16 + SHA512+ DH Group 14	CCP - Internal - DMZ to SERVERS
ESP IPv4 tunnel	EXTERNAL_CLIENTS	100.100.2.0/24	aes256gcm16 + SHA512+ DH Group 14	CCP - Internal - EXTERNAL SERVICES to CLIENTS
ESP IPv4 tunnel	EXTERNAL_CLIENTS	100.100.1.0/24	aes256gcm16 + SHA512+ DH Group 14	CCP - Internal - EXTERNAL SERVICES to SERVERS

And for the Internal firewall:

Type	Local Subnet	Remote Subnet	Phase 2 Proposal	Description
ESP IPv4 tunnel	CLIENTS	100.100.6.0/24	aes256gcm16 + SHA512+ DH Group 14	CCP - Main - CLIENTS to DMZ
ESP IPv4 tunnel	CLIENTS	100.100.4.0/24	aes256gcm16 + SHA512+ DH Group 14	CCP - Main - CLIENTS to EXTERNAL SERVICES
ESP IPv4 tunnel	SERVERS	100.100.6.0/24	aes256gcm16 + SHA512+ DH Group 14	CCP - Main - SERVERS to DMZ
ESP IPv4 tunnel	SERVERS	100.100.4.0/24	aes256gcm16 + SHA512+ DH Group 14	CCP - Main - SERVERS to EXTERNAL SERVICES

4.2 Addition of firewall rules

One critical aspect was the addition of firewall rules to allow IPsec traffic on the new IPsec interface.

Firewall	Interface	Direction	Source	Destination	Protocol	Port	Action
Main	IPsec	in	SERVERS net	DMZ net	any - IPv4	any	Pass
Main	IPsec	in	SERVERS net	EXT. SERVICES net	any - IPv4	any	Pass
Main	IPsec	in	CLIENTS net	DMZ net	any - IPv4	any	Pass
Main	IPsec	in	CLIENTS net	EXT. SERVICES net	any - IPv4	any	Pass

Table 3: Allow IPsec from Internal to External subnets

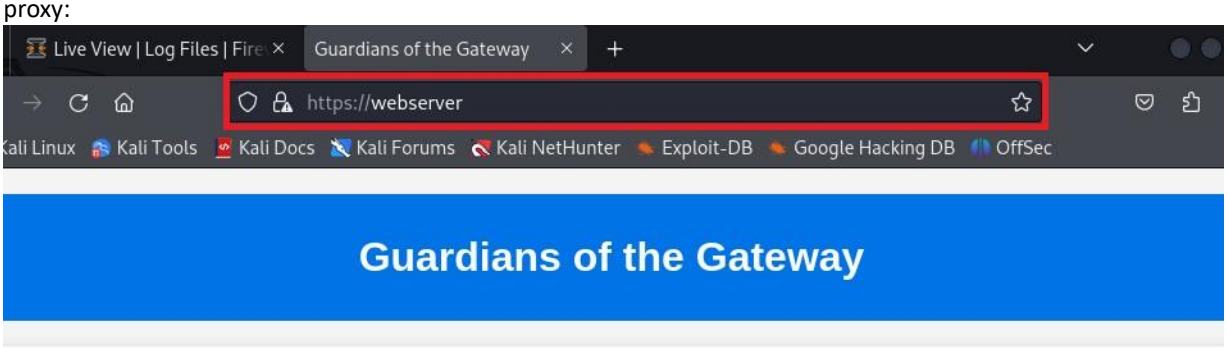
Firewall	Interface	Direction	Source	Destination	Protocol	Port	Action
Internal	IPsec	in	DMZ net	SERVERS net	any - IPv4	any	Pass
Internal	IPsec	in	DMZ net	CLIENTS net	any - IPv4	any	Pass
Internal	IPsec	in	EXT.SERVICES net	SERVERS net	any - IPv4	any	Pass
Internal	IPsec	in	EXT. SERVICES net	CLIENTS net	any - IPv4	any	Pass

Table 4: Allow IPsec from External to Internal subnets

These rules were specifically designed to ensure that traffic could flow from the appropriate network interfaces of the other network.

4.3 Testing phase

After configuring these rules, we have to verify that the IPsec tunnel was established successfully and that the intended traffic is passing through securely. First of all, in the log view interface we see that packets actually pass through the ipsec tunnel. For example, if we try to access the web server from kali, passing through the proxy:



The Journey of Three Brave IT Professionals

Once upon a time, in a land where networks were vast and cyber threats loomed large, there were three young IT professionals. These three were chosen for an extraordinary mission that would test their skills, determination, and courage.

The task was given by the god of firewalls, known as Prof. A.S. He summoned the trio and assigned them the formidable duty of configuring the ultimate firewall. The young professionals, eager to prove their mettle, accepted the challenge without hesitation.

Their journey was fraught with countless challenges. They encountered complex configurations, baffling network topologies, and relentless cyber threats. Each obstacle seemed insurmountable, but with teamwork and perseverance, they overcame each one.

Days turned into nights, and the trio worked tirelessly. They delved deep into the realms of network security, learning and adapting. The god of firewalls watched over them, guiding them through the darkest hours.

Finally, after a series of adventures and near-miraculous breakthroughs, the three young professionals succeeded. They had configured the firewall to perfection, securing the network against all threats. Their final task was to present their work to the great god of firewalls, Prof. A.S.

And then we check the logs for the IPsec interface:

	Interface		Time	Source	Destination	Proto	Label
▶	IPsec	→	2024-06-02T20:40:16	100.100.2.100:41528	100.100.6.3:3128	tcp	CCP - allow CLIENTS to DMZ through tunnel
▶	IPsec	→	2024-06-02T20:39:50	100.100.2.100:54730	100.100.6.3:3128	tcp	CCP - allow CLIENTS to DMZ through tunnel
▶	IPsec	→	2024-06-02T20:39:46	100.100.2.100:54722	100.100.6.3:3128	tcp	CCP - allow CLIENTS to DMZ through tunnel
▶	IPsec	→	2024-06-02T20:39:43	100.100.2.100:54708	100.100.6.3:3128	tcp	CCP - allow CLIENTS to DMZ through tunnel

we can see that the tunnel is up and running.
Next, we check that the encryption process is actually working:

5 Conclusion

The key takeaways from our configuration include:

- Security: By implementing strong encryption protocols.
- Testing: the testing phase was crucial for the correct configuration of the two tunnels.
- User Access Control: Differentiating access levels for operators and employees. Operators have unrestricted access, while employees are restricted from accessing sensitive internal networks.