

# **FSI LEI 2025/2026**

---

## **Practical class #6**

- VPN using OpenVPN**

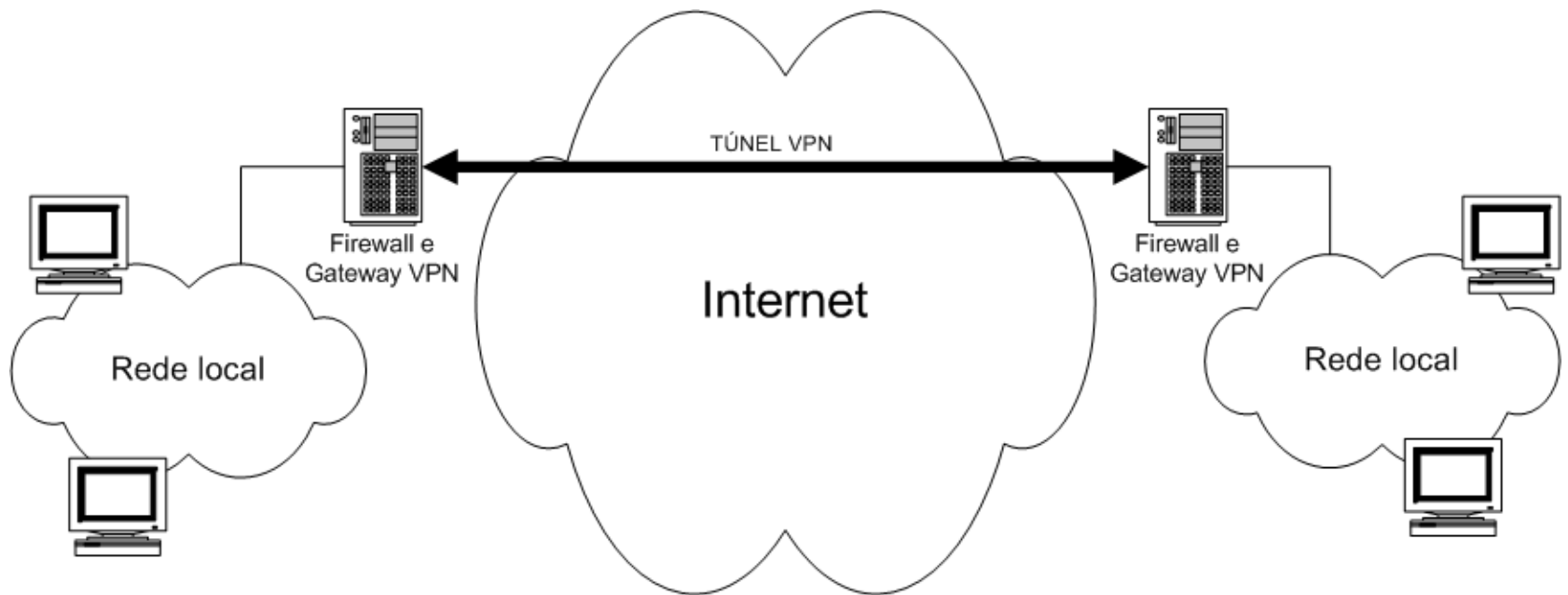
# OpenVPN

- VPN using SSL/TLS
- UDP or TCP
- TAP (Network Tap) or TUN (Network Tunnel) virtual network devices
- Supports common VPN usage scenarios
- Various client authentication methods (e.g. passwords, private keys, smart cards, certificates)



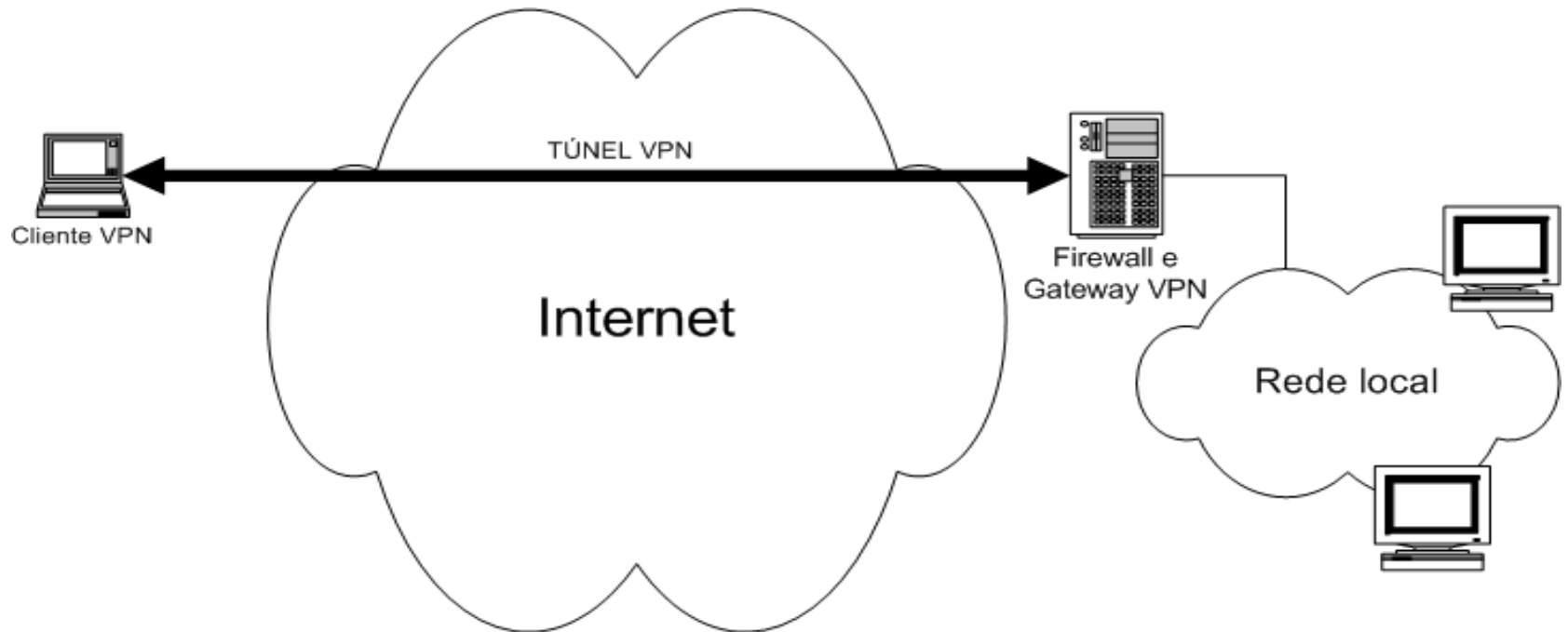
# OpenVPN (Usage Scenarios)

- Gateway-to-gateway (VPN) tunnel



# OpenVPN (Usage Scenarios)

- Road warrior (user remote access)



# OpenVPN (Installation & basics)

Install OpenVPN

```
# yum install epel-release
```

```
# yum install openvpn
```

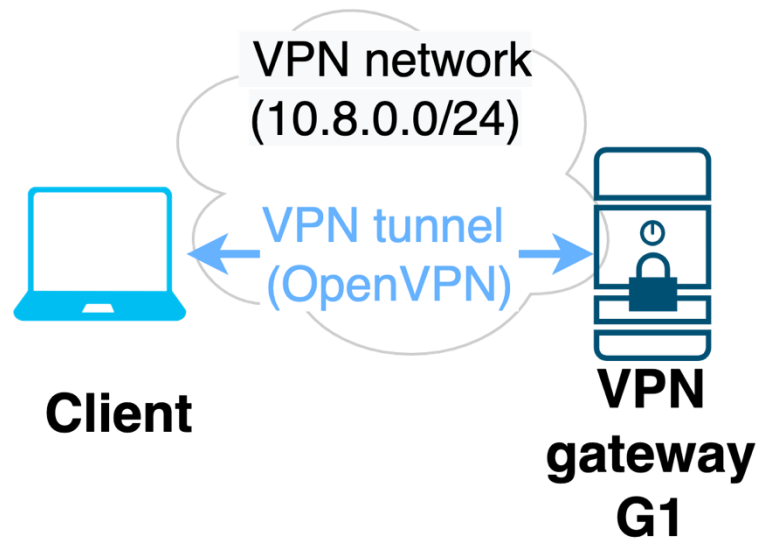
Start service

```
# systemctl start openvpn
```

Example documented configuration files can be found at:

```
/usr/share/doc/openvpn/sample/sample-config-files/
```

# OpenVPN – exercise scenario



# OpenVPN – Exercise in 3 phases

1. **OpenVPN authentication with X.509 certificates**
2. **OpenVPN authentication with passwords**
3. **Check OpenVPN functioning and security hardening**

# Phase 1 - OpenVPN Configuration

## # Example server configuration

local 192.168.1.1 # Modify according your settings.

port 1194

proto udp

dev tun

ca ca.crt # Consider the settings of the previous class (private CA)

cert gw-vpn.crt # Certificate generated for server

key gw-vpn.key # Private key of server

dh dh2048.pem # Diffie Hellman parameters

server 10.8.0.0 255.255.255.0

## # Start server (for testing)

# openvpn --config /etc/openvpn/server.conf

## # Check output of server



# Phase 1- OpenVPN Configuration (client)

## # Example client configuration

```
client
dev tun
proto udp
remote <IP/hostname> 1194 #Modify accordingly
persist-tun
persist-key
ca ca.crt
cert cliente_vpn.crt
key cliente_vpn.key
```

## # Start client with (for testing, this blocks terminal)

```
# openvpn --config /etc/openvpn/client.conf
```

## # Check output of client

# Phase 1- Checking connectivity

# on the server (look for tun0 interface)

# ifconfig

```
tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1500  
    inet 10.8.0.1 netmask 255.255.255.255 destination 10.8.0.2
```

# on the client (look for tun0 interface)

# ifconfig

```
tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1500  
    inet 10.8.0.6 netmask 255.255.255.255 destination 10.8.0.5
```

# client should be able to ping server IP address used in tunnel

```
sti@node2:~$ ping 10.8.0.1  
PING 10.8.0.1 (10.8.0.1) 56(84) bytes of data.  
64 bytes from 10.8.0.1: icmp_seq=1 ttl=64 time=0.567 ms  
64 bytes from 10.8.0.1: icmp_seq=2 ttl=64 time=0.639 ms  
...
```

# server should be able to ping client IP address used in tunnel

```
sti@debian:~$ ping 10.8.0.6  
PING 10.8.0.6 (10.8.0.6) 56(84) bytes of data.  
64 bytes from 10.8.0.6: icmp_seq=1 ttl=64 time=0.576 ms  
64 bytes from 10.8.0.6: icmp_seq=2 ttl=64 time=0.892 ms
```

# Phase 2 – Modify OpenVPN Configurations

Relevant information on:

`/usr/share/doc/openvpn/README.auth-pam`  
`man openvpn`

## # Server configuration (to add)

; Need to use plugin **openvpn-plugin-auth-pam.so** with login service type

## # Client configuration (to add)

; Need to activate option `authenticate` with server using `username/password`

; Password should be requested to the user

```
2022-02-25 20:22:21 OpenVPN 2.5.1 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO]
2022-02-25 20:22:21 library versions: OpenSSL 1.1.1k 25 Mar 2021, LZO 2.10
🔒 Enter Auth Username: sti
🔒 Enter Auth Password: *****
2022-02-25 20:22:26 WARNING: user was using user /group /server /server without
```

# Phase 3 – Hardening OpenVPN configuration (tls-auth)

Relevant information on:

<https://openvpn.net/community-resources/hardening-openvpn-security/>

man openvpn

**Allows to add HMAC signature to SSL/TLS handshake packets for integrity verification**

## **# Server configuration (to add)**

- ; Need to generate key and secret
- ; Need to transfer information to clientes
- ; Need to activate configuration tls-auth key 0

## **# Client configuration (to add)**

- ; Need to activate configuration tls-auth ta.key 1

# Phase 3 – Hardening OpenVPN configuration (tls-auth)

udp.port == 1194

No.	Time	Source	Destination	Protocol	Length	Info
491	156.304017318	192.168.1.74	192.168.1.63	OpenVPN	94	MessageType: P_ACK_V1
492	156.304017346	192.168.1.74	192.168.1.63	OpenVPN	116	MessageType: P_DATA_V2
494	158.239237170	192.168.1.63	192.168.1.74	OpenVPN	116	MessageType: P_DATA_V2
495	160.766390571	192.168.1.74	192.168.1.63	OpenVPN	116	MessageType: P_DATA_V2
505	169.725987423	192.168.1.74	192.168.1.63	OpenVPN	116	MessageType: P_DATA_V2
507	173.856375050	192.168.1.63	192.168.1.74	OpenVPN	116	MessageType: P_DATA_V2

▶ User Datagram Protocol, Src Port: 1194, Dst Port: 1194

▼ OpenVPN Protocol

- ▶ Type: 0x20 [opcode/key\_id]  
Session ID: 5278228046625990171  
HMAC: 776d82d309f9c61bc4d14dc637f63c2768dd0dc1  
Packet-ID: 6  
Net Time: Feb 25, 2022 21:34:40.000000000 WET  
Message Packet-ID Array Length: 1
- ▶ Packet-ID Array

```
0000  00 00 00 01 00 06 00 0c 29 fe 74 85 e4 16 08 00  .....).t.....
0010  45 00 04 b0 1f d7 40 00 40 11 92 8c c0 a8 01 4a  E.....@.@.....J
0020  c0 a8 01 3f 04 aa 04 aa 04 9c d6 99 20 49 40 08  ...?.....I@.
0030  71 72 a5 66 1b 77 6d 82 d3 09 f9 c6 1b c4 d1 4d  qr.f.wm.....M
0040  c6 37 f6 3c 27 68 dd 0d c1 00 00 00 06 62 19 4b  .7.<'h.....b.K
0050  70 01 00 00 00 03 6d 61 0f 9f f0 d1 eb f1 00 00  p.....ma.....
0060  00 02 14 03 03 00 01 01 17 03 03 07 fe ec c4 d2  .....
0070  24 30 0c 5a 7d ee a7 25 76 fd f6 57 01 03 35 5c  $0.Z}..%v.W.5\
0080  b5 ce 8f 49 bc af b8 5b 40 69 14 ee c5 0f 57 e0  ...I...[ @i...W.
```

wireshark\_anyXZX2H1.pcapng      Packets: 507 · Displayed: 26 (5.1%)      Profile: Default