CHULETA: OPENVPN usando OpenSSL (sin easy-rsa)
Estructura de trabajo (crear directorio PKI)

```
sudo mkdir -p /etc/openvpn/pki/{certs,crl,newcerts,private}
sudo touch /etc/openvpn/pki/index.txt
echo 1000 | sudo tee /etc/openvpn/pki/serial
cd /etc/openvpn/pki
```

---

1) Crear CA (Autoridad Certificadora)

```
# clave de CA (con passphrase o sin ella; si la usas, necesitarás introducirla al firmar)
openssl genrsa -aes256 -out private/ca.key.pem 4096
# certificado raíz (auto-firmado)
openssl req -x509 -new -key private/ca.key.pem -sha256 -days 3650 -out certs/ca.crt.pem
# comprobar
openssl x509 -noout -text -in certs/ca.crt.pem
```

---

2) Crear clave y CSR del servidor

```
openssl genrsa -out private/server.key.pem 4096
openssl req -new -key private/server.key.pem -out server.csr.pem
# usa CN=vpn.trun.local o la IP pública interna según tu topología
```

---

3) Firmar el certificado del servidor con la CA (añadir SAN)

```
# Linux bash: usar proceso subshell para pasar extfile inline
openssl x509 -req -in server.csr.pem -CA certs/ca.crt.pem -CAkey private/ca.key.pem \
-CAcreateserial -out certs/server.crt.pem -days 825 -sha256 \
-extfile <(printf "subjectAltName=DNS:vpn.trun.local,IP:172.22.0.80")
# Verificar
openssl verify -CAfile certs/ca.crt.pem certs/server.crt.pem
```

---

4) Generar Diffie-Hellman y tls-auth key

```
openssl dhparam -out dh.pem 2048
openvpn --genkey secret ta.key
```

---

5) Crear certificados de cliente (por cada cliente)

```
openssl genrsa -out private/cliente01.key.pem 4096
openssl req -new -key private/cliente01.key.pem -out cliente01.csr.pem -subj "/CN=cliente01"
openssl x509 -req -in cliente01.csr.pem -CA certs/ca.crt.pem -CAkey private/ca.key.pem \
-CAserial serial -out certs/cliente01.crt.pem -days 825 -sha256
# (opcional) Empaquetar en PKCS#12 si necesitas .p12
openssl pkcs12 -export -in certs/cliente01.crt.pem -inkey private/cliente01.key.pem -out cliente01.p12 -name "cliente01"
```

---

6) Lista de revocación (CRL)

```
# Usando openssl ca requiere index.txt y configuración; si tienes openssl.cnf preparado:
openssl ca -config /etc/ssl/openssl.cnf -revoke certs/cliente01.crt.pem -keyfile private/ca.key.pem -cert certs/ca.crt.pem
openssl ca -config /etc/ssl/openssl.cnf -gencrl -out crl.pem
openssl crl -noout -text -in crl.pem
```

---

7) Estructura final (colocar en servidor/cliente)

Servidor (/etc/openvpn/server/):

- ca.crt.pem
- server.crt.pem
- private/server.key.pem
- dh.pem
- ta.key
- server.conf

Cliente (/etc/openvpn/client/ o /home/user/):

- ca.crt.pem
- cliente01.crt.pem

- private/cliente01.key.pem
- ta.key
- cliente01.conf

---

8) Ejemplo server.conf (mínimo)

```
port 1194
proto udp
dev tun
ca /etc/openvpn/server/ca.crt.pem
cert /etc/openvpn/server/certs/server.crt.pem
key /etc/openvpn/server/private/server.key.pem
dh /etc/openvpn/server/dh.pem
tls-auth /etc/openvpn/server/ta.key 0
server 172.16.0.0 255.255.255.0
ifconfig-pool-persist ipp.txt
push "redirect-gateway def1 bypass-dhcp" # opcional: forzar salida por VPN
keepalive 10 120
cipher AES-256-GCM
user nobody
group nogroup
persist-key
persist-tun
status openvpn-status.log
verb 3
```

---

9) Ejemplo cliente .conf (mínimo)

```
client
dev tun
proto udp
remote 40.40.40.30 1194
resolv-retry infinite
nobind
persist-key
persist-tun
ca /etc/openvpn/client/ca.crt.pem
cert /etc/openvpn/client/cliente01.crt.pem
key /etc/openvpn/client/private/cliente01.key.pem
tls-auth /etc/openvpn/client/ta.key 1
cipher AES-256-GCM
verb 3
```

---

10) Comandos útiles y comprobaciones

```
# Iniciar/activar servicio
sudo systemctl start openvpn-server@server
sudo systemctl enable openvpn-server@server
sudo systemctl status openvpn-server@server
# Interfaces / tun
ip addr show tun0
ifconfig tun0
# Comprobar tráfico
sudo tcpdump -i eth0 -n udp port 1194
sudo tcpdump -i tun0 -n
# NAT y forwarding (si el servidor hace de gateway)
sudo sysctl -w net.ipv4.ip_forward=1
sudo iptables -t nat -A POSTROUTING -s 172.16.0.0/24 -o eth0 -j MASQUERADE
# Verificar certificados
openssl x509 -in certs/server.crt.pem -noout -text
openssl verify -CAfile certs/ca.crt.pem certs/cliente01.crt.pem
```

---

11) Notas prácticas rápidas

- Protege la clave de CA (private/ca.key.pem). Si se filtra, revocar todo.
- Usa SAN (subjectAltName) para evitar warnings de TLS.
- dhparam 2048 es suficiente en práctica; en producción se recomienda 3072+ o ECDH.
- Mantén copia de serial/crl/index.txt en backup.
- Para revocar: `openssl ca -revoke ...` y regenerar crl.pem; copiar crl.pem al servidor OpenVPN (opcional: usar `crl-verify` en server.conf).

VENTAJAS de usar OpenSSL directo:
- Control total sobre extensiones (SAN, keyUsage, etc.)
- No dependes de easy-rsa ni scripts; más didáctico.
- Fácil integración con infra PKI existente.