OpenVPN

Installation

- Put client configuration into `/etc/openvpn/client/`
- Start openvpn services

```
systemctl start openvpn-client@config-name
systemctl status openvpn-client@config-name
systemctl enable openvpn-client@config-name
```

NOTE: `openvpn-client@` service doesn't contain `restart`. The result of failed openvpn daemon looks like:

```
systemctl status openvpn-client@config-name
...
   Active: activating (auto-restart) since Mon 2020-10-19 15:50:36 CEST; 15s
ago
     Docs: man:openvpn(8)
           https://community.openvpn.net/openvpn/wiki/Openvpn24ManPage
           https://community.openvpn.net/openvpn/wiki/HOWTO
 Main PID: 19630 (code=exited, status=0/SUCCESS)
...
```

To make sure your VPN is running:

```
systemctl edit openvpn-client@config-name
```

and enter following config:

```
[Service]
Restart=always
RestartSec=300
```

```
systemctl daemon-reload
```

issue

```
openvpn[281925]: Failed to query password: Timer expired
openvpn[281924]: ERROR: Failed retrieving username or password
```

Solution:

/etc/systemd/system/openvpn-client@.service.d/askpass.conf

```
[Service]
ExecStart=
ExecStart=/usr/sbin/openvpn --suppress-timestamps --askpass --nobind --
config
%i.conf
```

Deprecated

- Put client configuration into /etc/openvpn/client.conf
- Enable autostart ALL or specified configs in `/etc/default/openvpn`
- Generate systemd services from openvon configs

```
systemctl daemon-reload
```

- Start openvpn services

```
systemct start openvpn
```

Certifcates

- CA has to be with

```
X509v3 Key Usage: Certificate Sign, CRL Sign
```

. Without `CRL Sign` latest version of OpenVPN doesn't allow to use CRL.

    - basicConstraints = CA:TRUE (critical)
    - nsCertType = sslCA # restrict the usage
    - keyUsage = keyCertSign, cRLSign
    - subjectKeyIdentifier = hash
    - authorityKeyIdentifier = keyid:always,issuer:always
- OpenVPN Server
    - basicConstraints = CA:FALSE
    - subjectKeyIdentifier = hash
    - authorityKeyIdentifier = keyid,issuer
    - nsCertType = server # restrict the usage
    - keyUsage = digitalSignature, keyEncipherment
    - extendedKeyUsage = serverAuth # restrict the usage
- OpenVPN Client
    - basicConstraints = CA:FALSE
    - subjectKeyIdentifier = hash
    - authorityKeyIdentifier = keyid,issuer
    - nsCertType = client # restrict the usage
    - keyUsage = digitalSignature # restrict the usage
    - extendedKeyUsage = clientAuth

Configuration

Routing

route directive adds normal routes to the Kernel table. It routes the packet from kernel to OpenVPN.
iroute directive adds routes to internal OpenVPN table. It routes the packets to specified clients.

Subnets behind client

In normal scenario, each VPN client is the final endpoint. But sometimes, there are additional networks behind client.

- Client side (or CCD directory - per client). There are networks **192.168.22.0/24** and **fcaa::/64** behind client:

```
iroute 192.168.22.0/24
iroute-ipv6 fcaa::/64
```

* Server configuration

```
route 192.168.22.0/24
route-ipv6 fcaa::/64
```

Username support

To easily distinguish clients with the same cert.
Server configuration

/etc/openvpn/auth-accept.sh

```
#!/bin/sh
exit 0
```

/etc/openvpn/server.conf

```
duplicate-cn
auth-user-pass-verify /etc/openvpn/auth-accept.sh via-env
auth-user-pass-optional
#username-as-common-name
```

Client configuration
Create file with username in 1st line, and password in 2nd

/etc/openvpn/devicename

```
client_A
fakepassword
```

/etc/opencvpn/client.conf

```
auth-user-pass /etc/openvpn/devicename
```

IPv6

- https://community.openvpn.net/openvpn/wiki/IPv6
- http://silmor.de/ipv6.openvpn.php
- https://superuser.com/questions/1151539/routing-problems-with-ipv6-over-openvpn
- https://www.digitalocean.com/community/questions/openvpn-ipv6-works-only-in-local-network

Troubleshooting

Error: “write to TUN/TAP : Invalid argument (code=22)”.
Cause: one side use LZO compression, second side not.
Solution: “comp-lzo no” on both sides.
Note:
this is a bug: the server pushes out 'comp-lzo' to the client but this is not picked up, because the client does not have 'comp-lzo' configured in the client config (all according to man page). The bug is , that when the client reconnects that it then does honor the 'comp-lzo' pushed out from the server. The client should either consistently refuse 'comp-lzo' or it should consistently accept this option as pushed out by the server.

Error: Cannot open TUN/TAP dev /dev/net/tun: Permission denied (errno=13).
Exiting due to fatal error
Use persist-key and persist-tun. **Cause**: on VPS platform /dev/net/tun has only root permisstion. So openvpn should be started as root user.

Error: unsupported protocol **Cause**: Modern OpenSSL (like 1.1.1) config forbids TLSv1 **Solution**:

/etc/ssl/openssl.cnf

```
MinProtocol = TLSv1
```

Error: File transfer stuck **Cause**: File transfer are using maximum packet size, which probably cannot fit to MTU limitataions **Solution**: Not tested, try params like:

```
# On one side of connection
mssfix 1400
```

```
# MTU on tunX interface
# has to be set on both sides
tun-mtu 1400
```

More:

- https://community.openvpn.net/openvpn/wiki/271-i-can-ping-through-the-tunnel-but-any-real-work-causes-it-to-lock-up-is-this-an-mtu-problem
- Setting correct MTU for OpenVPN

rsyslog

/etc/rsyslog.d/20-ovpn.conf

```
if $programname startswith 'ovpn-' then /var/log/openvpn/ovpn.log
& ~
```

```
mkdir /var/log/openvpn
chown syslog /var/log/openvpv
```

/etc/logrotate.d/openvpn

```
/var/log/openvpn/*.log {
        weekly
        size 100M
        rotate 4
        compress
        delaycompress
        missingok
        notifempty
        create 640 syslog adm
}
```

Create p12 package for android

```
openssl pkcs12 -export -in user.crt -inkey user.key -certfile ca.crt -name user -out user.p12
```