Openvpn

Certification Autorithy

Create certificate folder

```
apt-get install easy-rsa
make-cadir /etc/easy-rsa-legnago
cd /etc/easy-rsa-legnago
```

Edit vars and

```
source vars
./clean-all
./build-dh
./pkitool --initca
```

server certificate

```
NAME=legnago-gw
./pkitool --pass --server $NAME # create passphrase here
openssl rsa -in keys/$NAME.key -out keys/$NAME.pem # give passphrase here
chmod 600 keys/$NAME.pem
```

client certificate

```
NAME=nms
./pkitool --pass $NAME
openssl rsa -in keys/$NAME.key -out keys/$NAME.pem
```

Mikrotik server

Upload and import certificates

```
/certificate
import file=server.crt
import file=server.pem
import file=ca.crt
```

~~Simplier method~~

```
openssl genrsa -des3 -out ca.key 4096
```

```
# specify dns name of mikrotik server in common name
openssl req -new -x509 -days 3650 -key ca.key -out ca.crt

# now import in mikrotik ca.crt and after ca.key
```

ip pool

```
/ip pool add name=ovpn-pool ranges=10.15.32.34-10.15.32.38
```

profile and vpn user

```
/ppp profile
add change-tcp-mss=default comment="" local-address=10.15.32.33 \
name="your_profile" only-one=default remote-address=ovpn-pool \
use-compression=default use-encryption=required use-vj-compression=default
```

define vpn user

```
/ppp secret
add caller-id="" comment="" disabled=no limit-bytes-in=0 \
limit-bytes-out=0 name="username" password="password" \
routes="" service=any
```

openvpn instance

```
/interface ovpn-server server
set auth=sha1,md5 certificate=router_cert \
cipher=blowfish128,aes128,aes192,aes256 default-profile=your_profile \
enabled=yes keepalive-timeout=disabled max-mtu=1500 mode=ip netmask=29 \
port=1194 require-client-certificate=no
```

Linux client

```
apt-get install openvpn
```

/etc/openvpn/client1.conf

```
dev tun
proto tcp-client

remote legnago.csgalileo.org 1194

ca  /etc/easy-rsa-legnago/keys/ca.crt
cert /etc/easy-rsa-legnago/keys/nms.crt
key  /etc/easy-rsa-legnago/keys/nms.pem

tls-client
port 1194
```

```
user nobody
group nogroup

#comp-lzo # Do not use compression. It doesn't work with RouterOS (at least
up to RouterOS 3.0rc9)

# More reliable detection when a system loses its connection.
ping 15
ping-restart 45
ping-timer-rem
persist-tun
persist-key

# Silence  the output of replay warnings, which are a common false
# alarm on WiFi networks.  This option preserves the  security  of
# the replay protection code without the verbosity associated with
# warnings about duplicate packets.
mute-replay-warnings

# Verbosity level.
# 0 = quiet, 1 = mostly quiet, 3 = medium output, 9 = verbose
verb 3

cipher AES-256-CBC
auth SHA1
pull

auth-user-pass auth.cfg
script-security 2
up /etc/openvpn/up.sh
```

/etc/openvpn/up.sh (chmod +x)

```
#!/bin/sh

ip route add 10.90.0.0/16 via 10.15.32.33
```

/etc/openvpn/auth.cfg

```
username
password
```

Start service with systemd

```
systemctl start openvpn@client1
systemctl enable openvpn@client1
```

Linux server

/etc/openvpn/server.conf

```
proto tcp
dev tun

ca /etc/easy-rsa/keys/ca.crt
cert /etc/easy-rsa/keys/captive.crt
key /etc/easy-rsa/keys/captive.pem
dh /etc/easy-rsa/keys/dh2048.pem

server 10.4.0.0 255.255.255.0
ifconfig-pool-persist ipp.txt
keepalive 10 120
cipher BF-CBC
max-clients 100
client-config-dir ccd

# It's a good idea to reduce the OpenVPN
# daemon's privileges after initialization.
#
# You can uncomment this out on
# non-Windows systems.
user nobody
group nogroup

persist-key
persist-tun

# Output a short status file showing
# current connections, truncated
# and rewritten every minute.
#status /var/log/openvpn/captive.stats
log /var/log/openvpn/captive.log


# 0 is silent, except for fatal errors
# 4 is reasonable for general usage
# 5 and 6 can help to debug connection problems
# 9 is extremely verbose
#log         openvpn.log
#log-append  openvpn.log
verb 0

# Silence repeating messages.  At most 20
# sequential messages of the same message
# category will be output to the log.
mute 20
```

```
#fragment 1300
mssfix 1300
#link-mtu 1503
#tun-mtu 1460


#client-connect /etc/openvpn/on-client-connect
script-security 2
push "explicit-exit-notify"

management localhost 7505

client-to-client
```

Mikrotik client

Import certificates

```
import file-name=ca.crt
import file-name=galileo.crt
import file-name=galileo.pem
```

LXD

To enable tun inside container

```
lxc config device add <NAME> tun unix-char path=/dev/net/tun
```