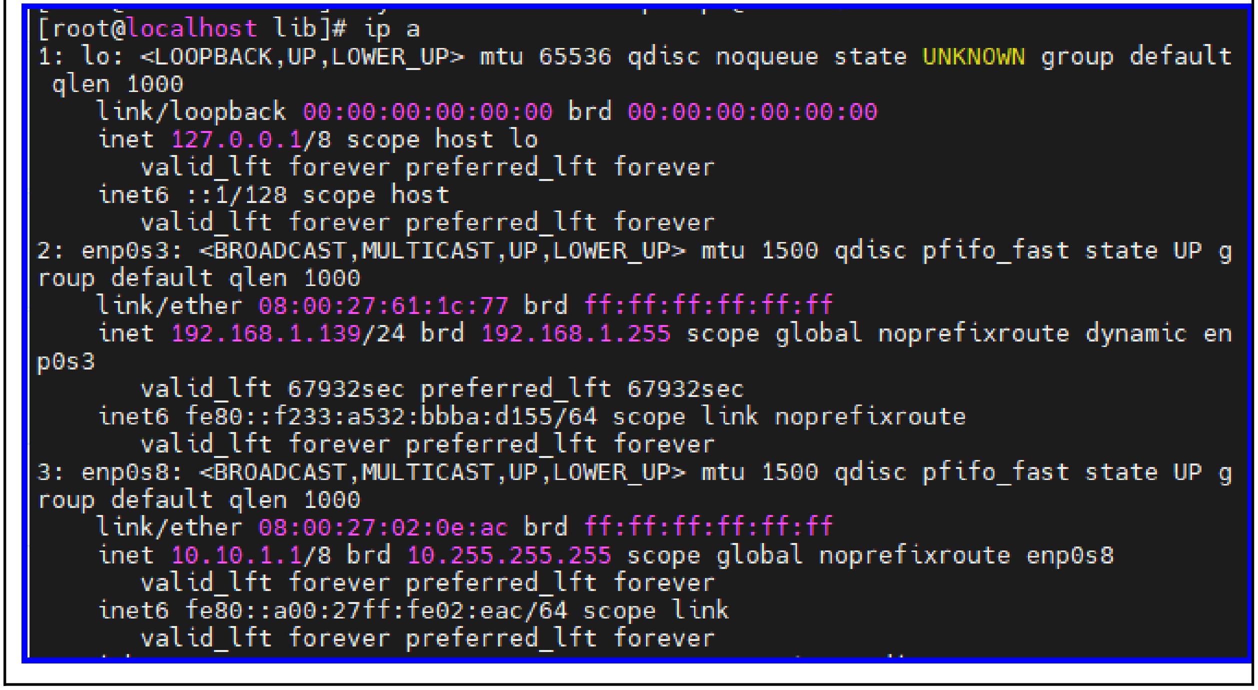
1. **Setup a VPN server in one vm. You can use openvpn for this purpose.**
    a. **You should have two network interfaces one for wan and another for lan. You should set up a vpn server which listens on the WAN interface and provides a LAN interface subnets ip address to the client which connects using openvpn client.**
    b. **You should create certificates files for both server and client to connect to server and export client certificates to the client vm.**

VPN Server Setup on Centos 7

1- a. Answer

We have two Network Interface

WAN - 192.168.1.0/24
LAN - 10.10.1.0/8



```
[root@localhost lib]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
 qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP g
roup default qlen 1000
    link/ether 08:00:27:61:1c:77 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.139/24 brd 192.168.1.255 scope global noprefixroute dynamic en
p0s3
       valid_lft 67932sec preferred_lft 67932sec
    inet6 fe80::f233:a532:bbba:d155/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP g
roup default qlen 1000
    link/ether 08:00:27:02:0e:ac brd ff:ff:ff:ff:ff:ff
    inet 10.10.1.1/8 brd 10.255.255.255 scope global noprefixroute enp0s8
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe02:eac/64 scope link
       valid_lft forever preferred_lft forever
```

VPN setup using OpenVPN

To **install OpenVPN** server on Centos 7 using wget

sudo apt install -y epel-release

sudo apt install -y openvpn wget

We need **Easy-RSA** primarily for key management and also for web certificates.

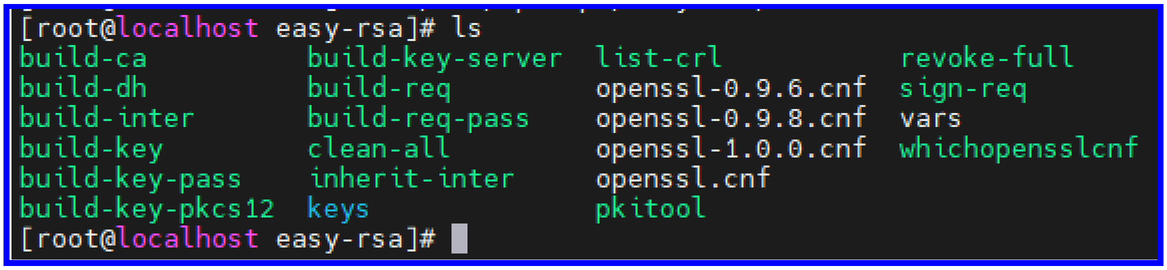
wget -O /tmp/easyrsa https://github.com/OpenVPN/easy-rsa-old/archive/2.3.3.tar.gz

tar xfz /tmp/easyrsa

Creating a sub-directory under **/etc/openvpn** and extracting EasyRSA files over here.

sudo mkdir /etc/openvpn/easy-rsa

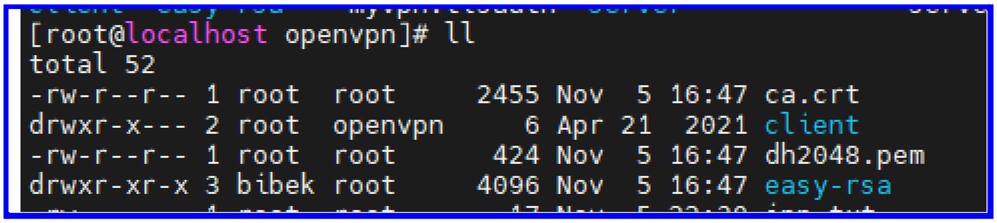
sudo cp -rf easy-rsa-old-2.3.3/easy-rsa/2.0/* /etc/openvpn/easy-rsa



```
[root@localhost easy-rsa]# ls
build-ca           build-key-server   list-crl           revoke-full
build-dh           build-req          openssl-0.9.6.cnf  sign-req
build-inter        build-req-pass     openssl-0.9.8.cnf  vars
build-key          clean-all          openssl-1.0.0.cnf  whichopensslcnf
build-key-pass     inherit-inter      openssl.cnf
build-key-pkcs12   keys               pkitool
[root@localhost easy-rsa]#
```

Changing directory's owner to non-root sudo user

sudo chown bibek /etc/openvpn/easy-rsa/



```
[root@localhost openvpn]# ll
total 52
-rw-r--r-- 1 root  root     2455 Nov  5 16:47 ca.crt
drwxr-x--- 2 root  openvpn     6 Apr 21  2021 client
-rw-r--r-- 1 root  root      424 Nov  5 16:47 dh2048.pem
drwxr-xr-x 3 bibek root     4096 Nov  5 16:47 easy-rsa
```

Configuring OpenVPN

We will use the server example configuration file from its documentation directory.

sudo cp /usr/share/doc/openvpn-2.4.11/sample/sample-config-files/server.conf /etc/openvpn/

sudo nano /etc/openvpn/server.conf and make following changes

- To listen at WAN address

local 192.168.1.139 -- ip address of centos 7 router (WAN)

- Default port

port 1194

- I have enabled both tcp and udp protocol

proto tcp
proto udp

- To created routed IP tunnel

dev tun

- Default Client and server certificate & key names

ca ca.crt
cert server.crt
key server.key

- Default Diffie helmen parameter name

dh dh2048.pem

- Network topology

topology subnet

- To give client address

server 10.10.1.0 255.255.255.0

- Push routes to the client to allow it to reach each other private subnets behind the server

push "route 192.168.10.0 255.255.255.0"

- DNS servers

push "dhcp-option DNS 8.8.8.8"

push "dhcp-option DNS 8.8.4.4"

- To allow different clients to see each other

client-to-client

- For extra security beyond that provided by SSL/TLS, create an "HMAC firewall" (block DoS attack and UDP port flooding)

tls-crypt myvpn.tlsauth

- For non-windows system

user nobody
group nobody

- To append log at specific location

log   /var/log/openvpn.log

- To notify client when the server restarts

explicit-exit-notify 1

- For tls web client authentication

remote-cert-eku "TLS Web Client Authentication"

- For user password authentication

plugin /usr/lib64/openvpn/plugins/openvpn-plugin-auth-pam.so openvpn

- Generating static encryption key

```
sudo openvpn --genkey --secret /etc/openvpn/myvpn.tlsauth
```

1- b. Answer

Creating certificates files for both server and client to connect to server

- Creating keys directory where Easy-RSA will store any keys and certs we generate

```
sudo mkdir /etc/openvpn/easy-rsa/keys
```

- Default certificate variables are set in vars file in /etc/openvpn/easy-rsa

```
sudo nano /etc/openvpn/easy-rsa/vars
```

Leaving others as default change the following parameters as per required

```
export KEY_COUNTRY="NP"
export KEY_PROVINCE="KTM"
export KEY_CITY="Kathmandu"
export KEY_ORG="LFTechnology"
export KEY_EMAIL="root@example.com"
export KEY_EMAIL=root@example.com
export KEY_CN=192.168.1.139
export KEY_NAME="EasyRSA"
export KEY_OU=LFTechnology
```

Save and exit

- To start generating keys, move to easy-rsa directory and source in the new variables

```
cd /etc/openvpn/easy-rsa
source ./vars
```

- Clean any keys and certificates already in the folder

```
./clean-all
```

- Build certificate authority. We have already set variables in the vars file, so we can press ENTER to accept the defaults for each one

./build-ca - *this script generates ca.key used to sign your server and client's certificates*

- Creating key and certificate for the server

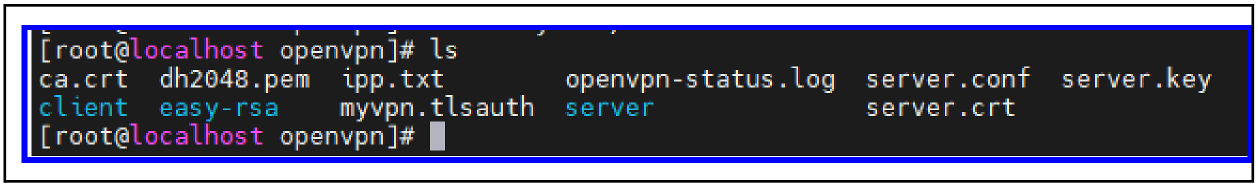
```
 ./build-key-server server
```

- Creating diffie helmen key exchange file

./build-dh - this can take few minutes to complete

- Now copy the server keys and certificates from **keys** directory to **openvpn** directory

```
cd /etc/openvpn/easy-rsa/keys
sudo cp dh2048.pem ca.crt server.crt server.key /etc/openvpn
```

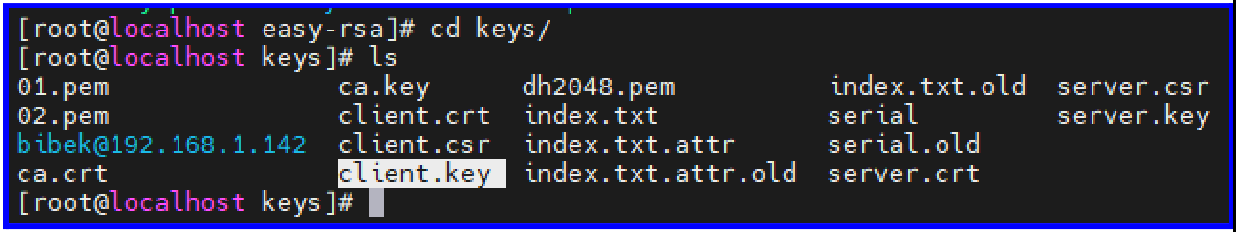


```
[root@localhost openvpn]# ls
ca.crt   dh2048.pem  ipp.txt         openvpn-status.log  server.conf  server.key
client   easy-rsa    myvpn.tlsauth   server              server.crt
[root@localhost openvpn]#
```

Generating client keys

- We called it client, but you can give more descriptive name

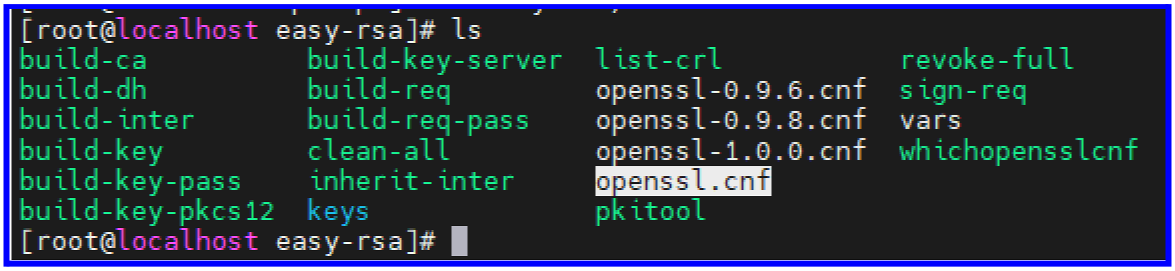
cd /etc/openvpn/easy-rsa
./build-key client



```
[root@localhost easy-rsa]# cd keys/
[root@localhost keys]# ls
01.pem               ca.key      dh2048.pem          index.txt.old  server.csr
02.pem               client.crt  index.txt           serial         server.key
bibek@192.168.1.142  client.csr  index.txt.attr      serial.old
ca.crt               client.key  index.txt.attr.old  server.crt
[root@localhost keys]#
```

- Copy versioned OpenSSL configuration file to versionless name to load configuration

cp /etc/openvpn/easy-rsa/openssl-1.0.0.cnf /etc/openvpn/easy-rsa/openssl.cnf



```
[root@localhost easy-rsa]# ls
build-ca           build-key-server  list-crl           revoke-full
build-dh           build-req         openssl-0.9.6.cnf  sign-req
build-inter        build-req-pass    openssl-0.9.8.cnf  vars
build-key          clean-all         openssl-1.0.0.cnf  whichopensslcnf
build-key-pass     inherit-inter     openssl.cnf
build-key-pkcs12   keys              pkitool
[root@localhost easy-rsa]#
```

Giving instructions to OpenVPN about where to send incoming web traffic (ROUTING)

- Adding openvpn service permanently to external active zone

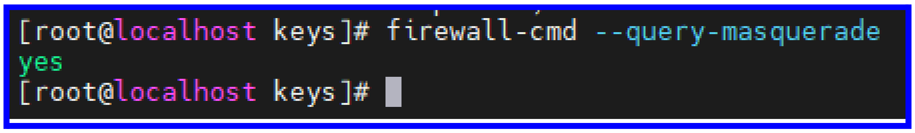
sudo firewall-cmd --zone=external --add-service openvpn --permanent

- Adding masquerade to all future instances with --permanent

sudo firewall-cmd --permanent --add-masquerade

- Check that the masquerade was added correctly

sudo firewall-cmd --query-masquerade - *output must be* ***yes***



```
[root@localhost keys]# firewall-cmd --query-masquerade
yes
[root@localhost keys]#
```

- Forwarding routing to OpenVPN subnet
- Creating variable SHARK which will represent the primary network interface

SHARK=$(ip route get 8.8.8.8 | awk 'NR==1 {print $(NF-2)}')

- Using SHARK variable to permanently add the routing rule to our subnet

sudo firewall-cmd --permanent --direct --passthrough ipv4 -t nat -A POSTROUTING -s 10.10.1.0/24 -o $SHARK -j MASQUERADE

- Reloading firewall-cmd

sudo firewall-cmd --reload

- We have to enable **ip_forwarding=1**

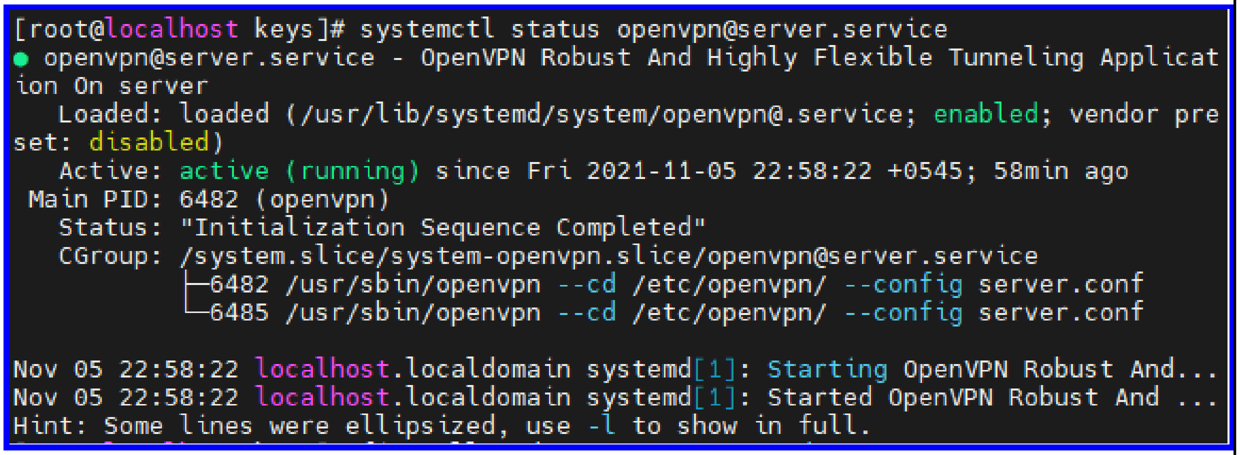
We have done it previously permanently, configuring Centos as a router

- Restart Network service

sud0 systemctl restart network

Now we are ready to start **openvpn service**

sudo systemctl -f enable openvpn@server.service
sudo systemctl start openvpn@server.service
sudo systemctl status openvpn@server.service



```
[root@localhost keys]# systemctl status openvpn@server.service
● openvpn@server.service - OpenVPN Robust And Highly Flexible Tunneling Applicat
ion On server
   Loaded: loaded (/usr/lib/systemd/system/openvpn@.service; enabled; vendor pre
set: disabled)
   Active: active (running) since Fri 2021-11-05 22:58:22 +0545; 58min ago
 Main PID: 6482 (openvpn)
   Status: "Initialization Sequence Completed"
   CGroup: /system.slice/system-openvpn.slice/openvpn@server.service
           ├─6482 /usr/sbin/openvpn --cd /etc/openvpn/ --config server.conf
           └─6485 /usr/sbin/openvpn --cd /etc/openvpn/ --config server.conf

Nov 05 22:58:22 localhost.localdomain systemd[1]: Starting OpenVPN Robust And...
Nov 05 22:58:22 localhost.localdomain systemd[1]: Started OpenVPN Robust And ...
Hint: Some lines were ellipsized, use -l to show in full.
```

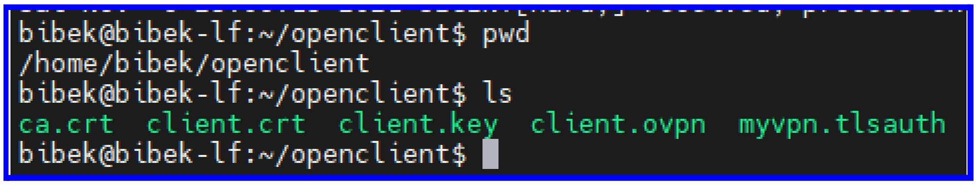
To transfer client certificate to client machine, I used rsync command

- The keys to transfer to client machine are **ca.crt, client.crt, client.key(all three are in keys directory) & myvpn.tlsauth (is in openvpn directory)**
- Change directory path to keys and use rsync command

cd /etc/openvpn/easy-rsa/keys

sudo rsync ca.crt client.crt client.key ../../myvpn bibek@192.168.1.142:/home/bibek/openclient

And provided password for bibek user of 192.168.1.142 server



```
bibek@bibek-lf:~/openclient$ pwd
/home/bibek/openclient
bibek@bibek-lf:~/openclient$ ls
ca.crt  client.crt  client.key  client.ovpn  myvpn.tlsauth
bibek@bibek-lf:~/openclient$
```