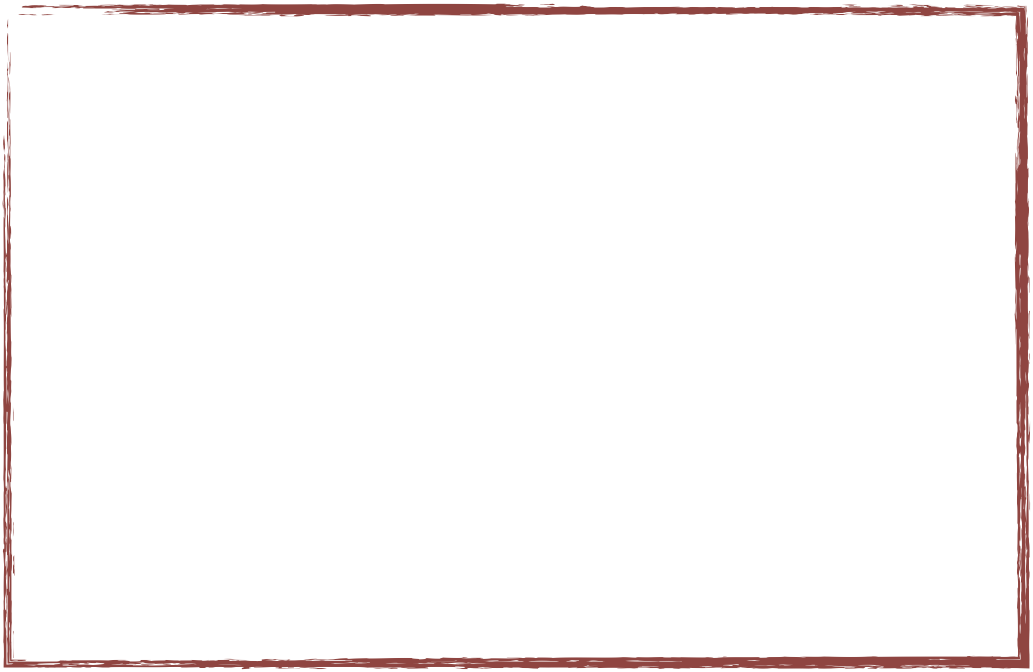


**Computer Science and Engineering Department**



**Group Name**&**Surname:** Mehmet Taha USTA & Burcu ÖZTAŞ

**Identity Numbers:** 21527472 & 21483435

**Course:**  BBM-465 Information Security Lab.

**Experiment:** Assignment 5

**Subject:**  VPNs

**Due Date:** 10/1/2020 - 23:59

**Advisor:**  Dr. Ahmet Selman BOZKIR

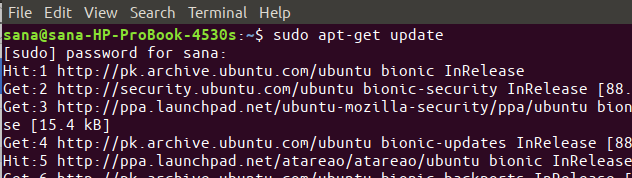
**1.** **OpenSSH Server**

Server and client must be installed to provide SSH Connection.

**1.1 Update Repository Index**

In order to install the latest available version of software from the Internet repositories, your local repository index needs to be in line with them. Run the following command as sudo in order to update your local repository index:

$ sudo apt-get update

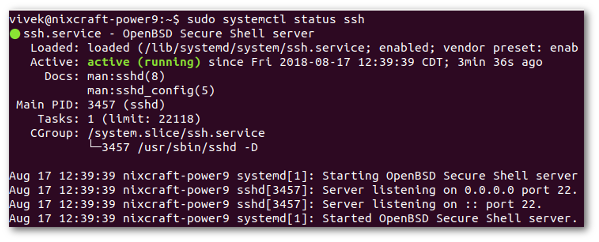


**1.2 Install OpenSSh-Server**

$ sudo apt install openssh-server

**1.3 Verify That SSH Service Running**

$ sudo systemctl status ssh



If not running enable the ssh server and start it as follows by typing the systemctl command:

$ sudo systemctl enable ssh

$ sudo systemctl start ssh

**1.4 Password Reset**

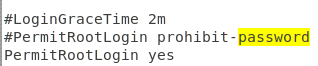
$ sudo passwd

**1.5 Allow Root Authentication**

$ sudo gedit /etc/ssh/sshd\_config

add command to file:

PermitRootLogin yes



$ sudo service ssh restart

**2.** **OpenVPN**

**2.1 Building CA with EasyRSA**

Perform the following steps on your CA machine

First, download the EasyRSA from the project Github repository with the following wget command:

cd && wget <https://github.com/OpenVPN/easy-rsa/releases/download/v3.0.5/> EasyRSA-nix-3.0.5.tgz



Extract the archive:

tar xzf EasyRSA-nix-3.0.5.tgz

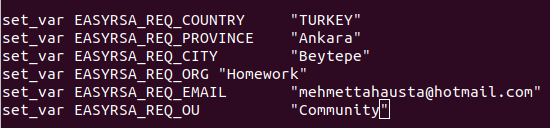


Switch to the EasyRSA directory and create a configuration file named vars by copying the vars.example file:



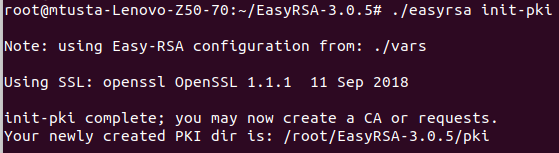
Open the file and uncomment and update the following entries to match your information.

nano ~/EasyRSA-3.0.5/vars



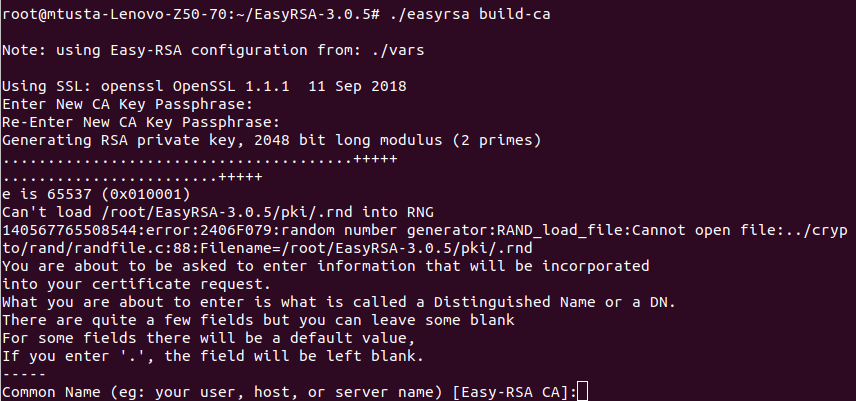
Before generating a CA keypair first we need to initialize a new PKI with:

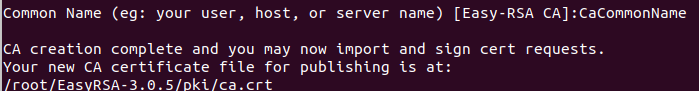
./easyrsa init-pki

****

The next step is to build the CA:

./easyrsa build-ca nopass(optional)

****

****

**2.2 Installing OpenVPN and EasyRSA**

Our next step is to install the OpenVPN package which is available in Ubuntu's repositories and download the latest version of EasyRSA.

sudo apt update

sudo apt install openvpn

Download the EasyRSA from the project Github repository with the following wget command:

cd && wget <https://github.com/OpenVPN/easy-rsa/releases/download/v3.0.5/> EasyRSA-nix-3.0.5.tgz



Extract the archive:

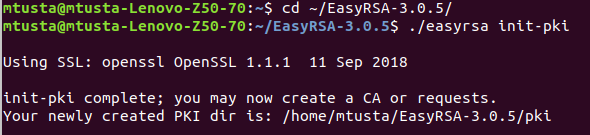
tar xzf EasyRSA-nix-3.0.5.tgz



we need to create a new PKI on the OpenVPN server

cd ~/EasyRSA-3.0.5/

./easyrsa init-pki

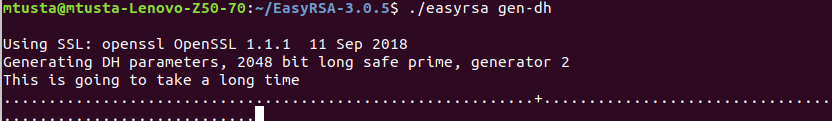


**2.3 Creating Diffie-Hellman and HMAC keys**

First navigate to the EasyRSA directory on your OpenVPN server and Generate a Diffie-Hellman key:

cd ~/EasyRSA-3.0.5/

./easyrsa gen-dh





Copy the dh.pem file to the /etc/openvpn directory:

sudo cp ~/EasyRSA-3.0.5/pki/dh.pem /etc/openvpn/

Generate a HMAC signature:

openvpn --genkey --secret ta.key

Once completed copy the ta.key file to the /etc/openvpn directory:

sudo cp ~/EasyRSA-3.0.5/ta.key /etc/openvpn/



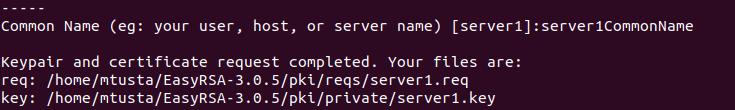
**2.4 Creating Server Certificate and Private Key**

This section describes how to generate a private key and certificate request for the OpenVPN server.

Navigate to the EasyRSA directory on your OpenVPN server and generate a new private key for the server and a certificate request file:

cd ~/EasyRSA-3.0.5/

./easyrsa gen-req server1 nopass(optional)

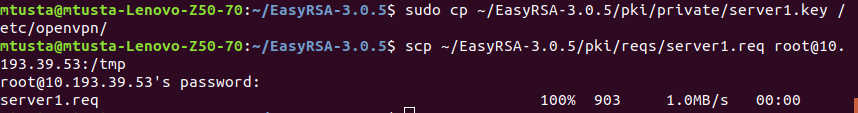


Copy the private key to the /etc/openvpn directory:

sudo cp ~/EasyRSA-3.0.5/pki/private/server1.key /etc/openvpn/

Transfer the certificate request file to your CA machine:

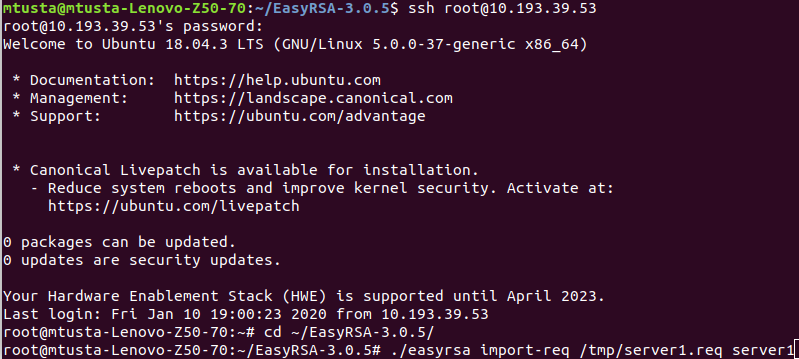
scp ~/EasyRSA-3.0.5/pki/reqs/server1.req root@10.193.39.53:/tmp



Login to your CA machine, switch to the EasyRSA directory and import the certificate request file:

cd ~/EasyRSA-3.0.5

./easyrsa import-req /tmp/server1.req server1

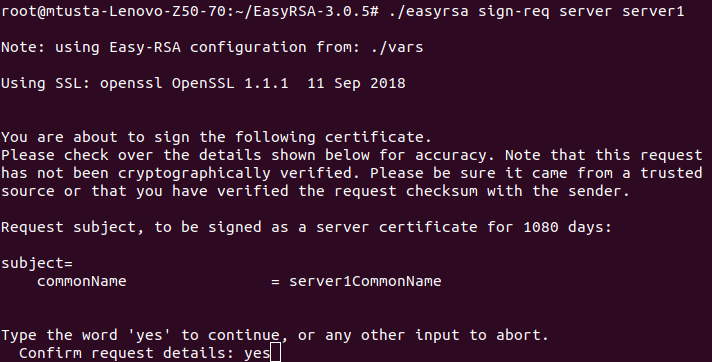


This command just copies the request file into the pki/reqs directory.



While still in the EasyRSA directory on CA machine run the following command to sign the request:

./easyrsa sign-req server server1



Next step is to transfer the signed certificate server1.crt and ca.crt files back to your OpenVPN server. Again you can use scp, rsync or any other secure method:

scp ~/EasyRSA-3.0.5/pki/issued/server1.crt mtusta@10.193.39.53:/tmp

scp ~/EasyRSA-3.0.5/pki/ca.crt mtusta@10.193.39.53:/tmp

Login to your OpenVPN server, and move the server1.crt and ca.crt files into the /etc/openvpn/ directory:

sudo mv /tmp/{server1,ca}.crt /etc/openvpn/

Upon completing the steps outlined in this section, you should have the following new files on your OpenVPN server:



**2.5 Configuring the OpenVPN Service**

Start by extracting the configuration file to the /etc/openvpn/ directory:

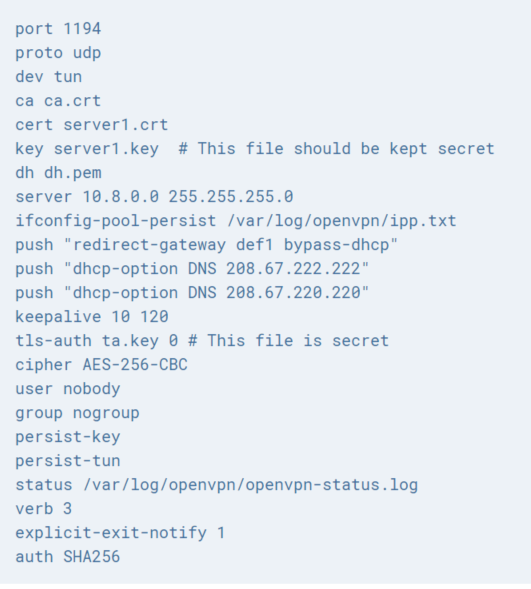
sudo sh -c "gunzip -c /usr/share/doc/openvpn/examples/sample-config-files/server.conf.gz > /etc/openvpn/server1.conf"

Open the file with text editor:

sudo gedit /etc/openvpn/server1.conf

Delete some command line character (# or ;)

Once you are done, the server configuration file (excluding comments) should look something like this:

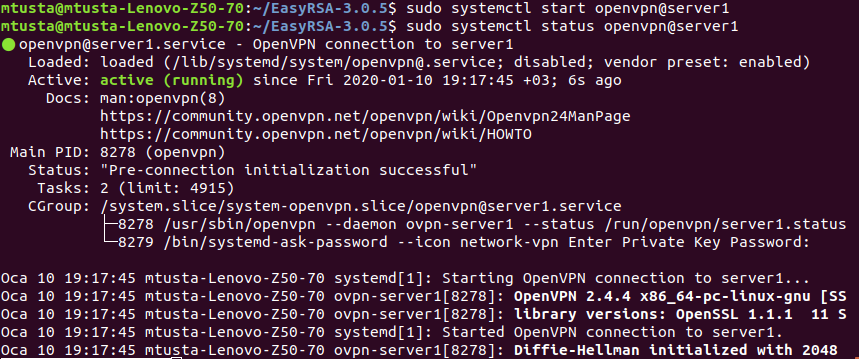


**2.6 Starting OpenVPN Service**

On your OpenVPN server run the following command to start the OpenVPN service:

sudo systemctl start openvpn@server1

sudo systemctl status openvpn@server1



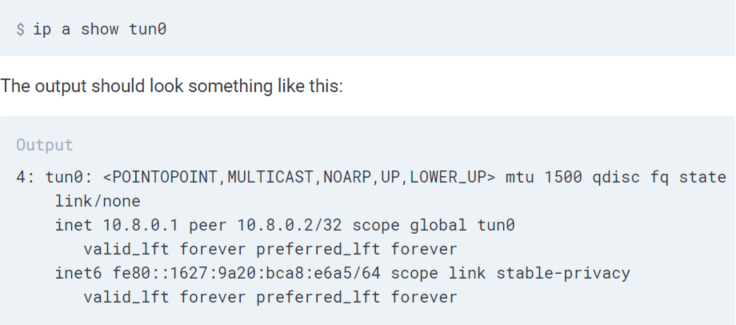
Enable the service to automatically start on boot with:

sudo systemctl enable openvpn@server1



If the OpenVPN service fails to start check the logs with sudo journalctl -u openvpn@server1

The OpenVPN Server will create a new tun device tun0. To check whether the device is available, use the following ip command:



At this point, your OpenVPN server is configured and running properly.

**2.7 Firewall and Server Networking Configuration**

In order to forward network packets properly, we need to enable IP forwarding.

sudo gedit /etc/sysctl.conf

# Uncomment the next line to enable packet forwarding for IPv4

net.ipv4.ip\_forward=1

Save and close the file.

Apply the new settings by running the following command:

sudo sysctl -p

Output:

net.ipv4.ip\_forward = 1



public network interface of Ubuntu OpenVPN Server.

ip -o -4 route show to default | awk '{print $5}'



Open the UFW configuration file, locate the DEFAULT\_FORWARD\_POLICY key and change the value from DROP to ACCEPT:

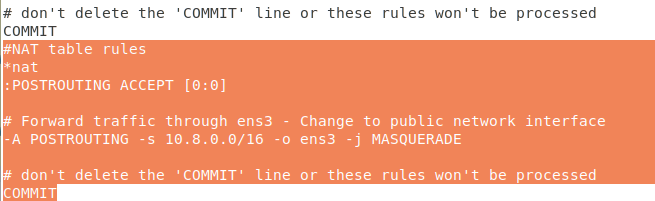
# Set the default forward policy to ACCEPT, DROP or REJECT. Please note that

# if you change this you will most likely want to adjust your rules

DEFAULT\_FORWARD\_POLICY="ACCEPT"

Next, we need to set the default policy for the POSTROUTING chain in the nat table and set the masquerade rule.

sudo gedit /etc/ufw/before.rules



# don't delete the 'COMMIT' line or these rules won't be processed

COMMIT

#NAT table rules

\*nat

:POSTROUTING ACCEPT [0:0]

# Forward traffic through ens3 - Change to public network interface

-A POSTROUTING -s 10.8.0.0/16 -o ens3 -j MASQUERADE

# don't delete the 'COMMIT' line or these rules won't be processed

COMMIT

When you are done, save and close the file.

We also need to open UDP traffic on port 1194

sudo ufw allow 1194/udp

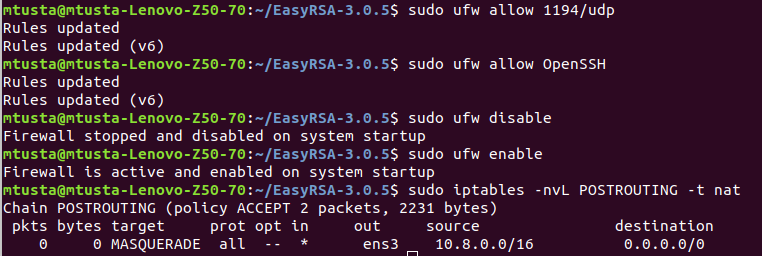
sudo ufw allow OpenSSH

sudo ufw disable

sudo ufw enable

To verify the changes run the following command to list the POSTROUTING rules:

sudo iptables -nvL POSTROUTING -t nat



**2.8 Creating the Client Configuration Infrastructur**

Start by creating a set of directories to store the clients files:

base directory will store the base files and configuration that will be shared across all client files.

configs directory will store the generated client configuration.

files directory will store client-specific certificate/key pair.

mkdir -p ~/openvpn-clients/{configs,base,files}

Copy the ca.crt and ta.key files to the ~/openvpn-clients/base directory:

cp ~/EasyRSA-3.0.5/ta.key ~/openvpn-clients/base/

cp /etc/openvpn/ca.crt ~/openvpn-clients/base/

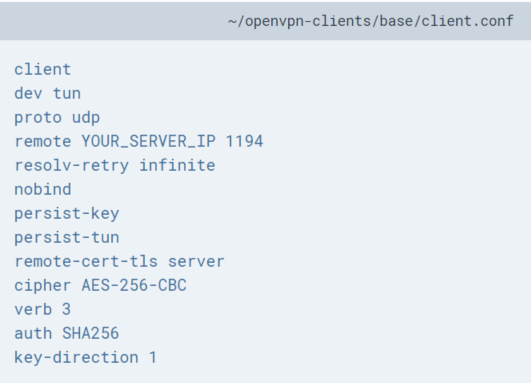
Next copy the sample VPN client configuration file into the client-~/openvpn-clients/base directory. We will use this file as a base configuration:

cp /usr/share/doc/openvpn/examples/sample-config-files/client.conf ~/openvpn-clients/base/

Open the configuration file with your text editor::

gedit ~/openvpn-clients/base/client.conf

The server configuration file should look something like this:



Next, create a simple bash script that will merge the base configuration and files with the client certificate and key, and store the generated configuration in the ~/openvpn-clients/configs directory.

nano ~/openvpn-clients/gen\_config.sh

#!/bin/bash

FILES\_DIR=$HOME/openvpn-clients/files

BASE\_DIR=$HOME/openvpn-clients/base

CONFIGS\_DIR=$HOME/openvpn-clients/configs

BASE\_CONF=${BASE\_DIR}/client.conf

CA\_FILE=${BASE\_DIR}/ca.crt

TA\_FILE=${BASE\_DIR}/ta.key

CLIENT\_CERT=${FILES\_DIR}/${1}.crt

CLIENT\_KEY=${FILES\_DIR}/${1}.key

# Test for files

for i in "$BASE\_CONF" "$CA\_FILE" "$TA\_FILE" "$CLIENT\_CERT" "$CLIENT\_KEY"; do

if [[ ! -f $i ]]; then

echo " The file $i does not exist"

exit 1

fi

if [[ ! -r $i ]]; then

echo " The file $i is not readable."

exit 1

fi

done

# Generate client config

cat > ${CONFIGS\_DIR}/${1}.ovpn <<EOF

$(cat ${BASE\_CONF})

<key>

$(cat ${CLIENT\_KEY})

</key>

<cert>

$(cat ${CLIENT\_CERT})

</cert>

<ca>

$(cat ${CA\_FILE})

</ca>

<tls-auth>

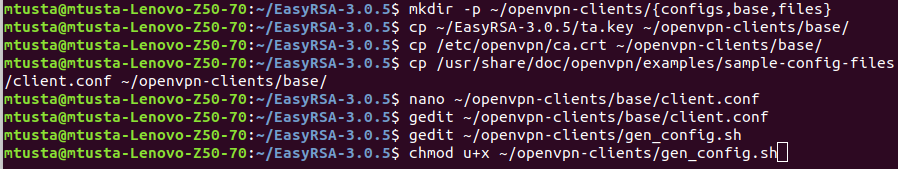
$(cat ${TA\_FILE})

</tls-auth>

EOF

Save the file and make it executable by running the following chmod command:

chmod u+x ~/openvpn-clients/gen\_config.sh

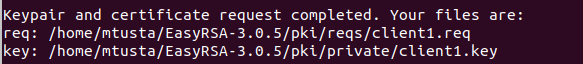


**2.9 Creating Client Certificate Private Key and Configuration**

Navigate to the EasyRSA directory on your OpenVPN server and generate a new private key and a certificate request file for the client:

cd ~/EasyRSA-3.0.5/

./easyrsa gen-req client1 nopass(optional)



Copy the private key client1.key to the ~/openvpn-clients/files directory you created in the previous section:

cp ~/EasyRSA-3.0.5/pki/private/client1.key ~/openvpn-clients/files/

Transfer the certificate request file to your CA machine:

scp ~/EasyRSA-3.0.5/pki/reqs/client1.req root@10.193.39.53:/tmp

Login to your CA machine, switch to the EasyRSA directory and import the certificate request file:

cd ~/EasyRSA-3.0.5

./easyrsa import-req /tmp/client1.req client1



From within the EasyRSA directory on CA machine run the following command to sign the request:

./easyrsa sign-req client client1



Next, transfer the signed certificate client1.crt file back to your OpenVPN server. You can use scp, rsync or any other secure method:

scp ~/EasyRSA-3.0.5/pki/issued/client1.crt mtusta@10.193.39.53:/tmp

Login to your OpenVPN server, and move the client1.crt file into the ~/openvpn-clients/files directory:

mv /tmp/client1.crt ~/openvpn-clients/files

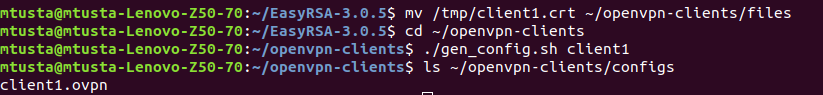
The final step is to generate a client configuration using the gen\_config.sh script. Switch to the ~/openvpn-clients directory and run the script using the client name as an argument:

cd ~/openvpn-clients

./gen\_config.sh client1

The script will create a file named client1.ovpn in the ~/client-configs/configs directory. You can check by listing the directory:

ls ~/openvpn-clients/configs



At this point the client configuration is created. You can now transfer the configuration file to the device you intend to use as a client.

**2.10 Connecting Clients**

Install OpenVPN on Ubuntu

sudo apt update

sudo apt install openvpn

Once the package is installed, to connect to the VPN server use the openvpn command and specify the client configuration file:

sudo openvpn --config client1.ovpn