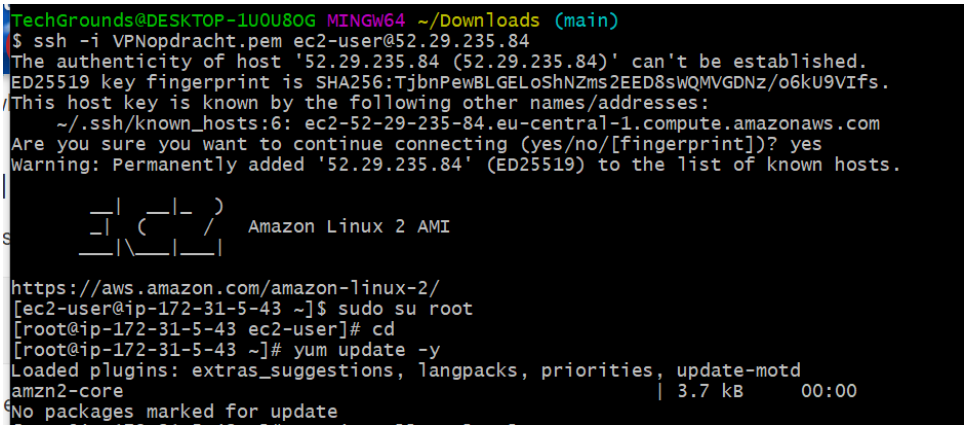
Open VPN assignment

First we logged in, using ssh command

$ sshi –i “VPNopdracht.pem” ec2-user@52.29.235.84

Make sure, the pem document stands inside the download folder AND you are in the present working directory.

1



When connection is made,
We update Centos repository within AWS.

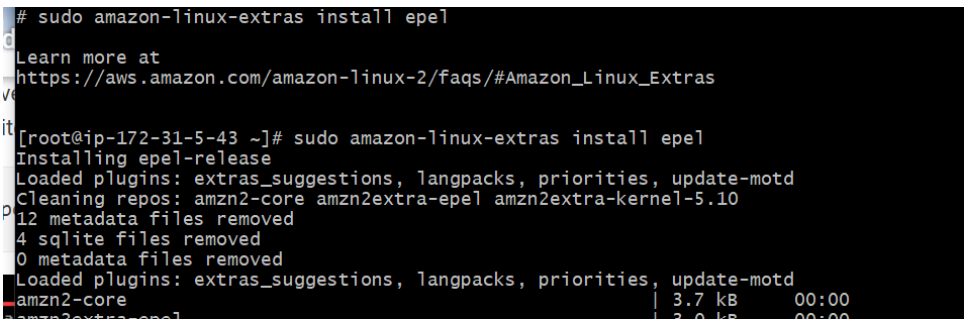
$ yum update -y

We enabled the epel repository

$ yum amazon-linux-extras install epel

(Maybe, the command $ yum install epel-release –y works also)

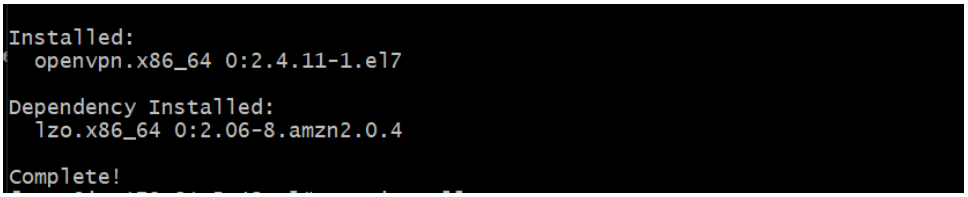
2



3

We installed openvpn

$ yum install –y openvpn



Step 2: Install Easy RSA

Now, we install CLI utility easy RSA for creating and managing PKI Certificate Authority (CA).

$ yum install –y wget

$ wget https://github.com/OpenVPN/easy-rsa/archive/v3.0.8.tar.gz

4

```
Location: https://codeload.github.com/OpenVPN/easy-rsa/tar.gz/v3.0.8 [following]
--2021-11-20 19:07:25--  https://codeload.github.com/OpenVPN/easy-rsa/tar.gz/v3.0.8
Resolving codeload.github.com (codeload.github.com)... 140.82.121.9
Connecting to codeload.github.com (codeload.github.com)|140.82.121.9|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3864366 (3.7M) [application/x-gzip]
Saving to: ‘v3.0.8.tar.gz’

100%[======================================================================================>

2021-11-20 19:07:25 (23.0 MB/s) - ‘v3.0.8.tar.gz’ saved [3864366/3864366]
```

We extract the downloaded archive

$ tar –xf v3.0.8.tar.gz

Now, switch into a new openvpn directory

$ cd /etc/openvpn/

We create a subdirectory

$ mkdir /etc/openvpn/easy-rsa

We move the file
Be in the right directory!

$ mv /root/easy-rsa-3.0.8 /ect/openvpn/easy-rsa

5

```
[root@ip-172-31-5-43 ~]# cd /etc/openvpn/
[root@ip-172-31-5-43 openvpn]# mkdir /etc/openvpn/easy-rsa
[root@ip-172-31-5-43 openvpn]# mv /root/easy-rsa-3.0.8 /etc/openvpn/easy-rsa
[root@ip-172-31-5-43 openvpn]# cd /etc/openvpn/easy-rsa
[root@ip-172-31-5-43 easy-rsa]# ls
easy-rsa-3.0.8
[root@ip-172-31-5-43 easy-rsa]# cd easy-rsa-3.0.8
[root@ip-172-31-5-43 easy-rsa-3.0.8]# ls
build      COPYING.md  doc       KNOWN_ISSUES  op_test.orig  README.md            release-keys  wop_test.sh
ChangeLog  distro      easyrsa3  Licensing     op_test.sh    README.quickstart.md  wop_test.bat
```

We list the content in easy-rsa.

Step3: Configure OpenVPN

We copy a sample file from openvpn documentation directory.

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

Be shure to have the right version of this:
$ find / -name server.conf

Open the copied configuration file by the following command:

$ vi /etc/openvpn/server.conf

We uncomment severall lines.
Uncomment means to erase the **# of ;** in front of the command.

- **`topology subnet`** (makes the OpenVPN installation function as a subnetwork)
- **`push "redirect-gateway def1 bypass-dhcp"`** (instructs the client to redirect traffic through the OpenVPN server)
- **`push "dhcp-option DNS 208.67.222.222"`** (uses an OpenDNS resolver to connect to OpenVPN)
- **`push "dhcp-option DNS 208.67.220.220"`** (uses an OpenDNS resolver to connect to OpenVPN)
- **`user nobody`** (runs OpenVPN with no privileges)
- **`group nobody`** (runs OpenVPN with no privileges)

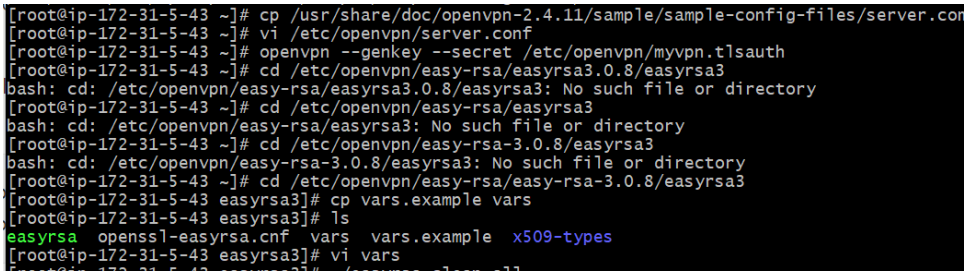
Then, generate a static encryption key to enable TLS authentication.
Add the next line to the file:

tls-crypt myvpn.tlsauth

We generate the static encryptionkey:

$ openvpn –genkey –secret /etc/openvpn/myvpn.tlsauth

6



Step 4: Generate Keys and Certificates

We move into the easyrsa3 directory:

$ cd /etc/openvpn/easy-rsa/easyrsa3

We list the content to know the content of the directories

$ cp vars.example vars

We list the files with $ ls

7

```
[root@ip-172-31-5-43 ~]# cd /etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3
[root@ip-172-31-5-43 easyrsa3]# cp vars.example vars
[root@ip-172-31-5-43 easyrsa3]# ls
easyrsa  openssl-easyrsa.cnf  vars  vars.example  x509-types
[root@ip-172-31-5-43 easyrsa3]# vi vars
```

We opened vars with vi editor: $ vi vars
We uncommented the following lines by erasing the **#**

#set_var EASYRSA_REQ_COUNTRY "**US**"
#set_var EASYRSA_REQ_PROVINCE "**California**"
#set_var EASYRSA_REQ_CITY "**San Francisco**"
#set_var EASYRSA_REQ_ORG "**Copyleft Certificate Co**"
#set_var EASYRSA_REQ_EMAIL "**me@example.net**"
#set_var EASYRSA_REQ_OU "**My Organizational Unit**"

We replaces the default values with our own information.

Next, we added the line: export KEY_NAME=”server”
because it was NOT present in the file.

The same with the next line: export KEY_CN=openvpn.yourdomain.com
Fill in the DNS IPv4 address.

Now, clean up any previous key and generate the certificate authority:

$ ./easyrsa clean-all

9

```
[root@ip-172-31-5-43 easyrsa3]# ./easyrsa clean-all

Note: using Easy-RSA configuration from: /etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/vars

init-pki complete; you may now create a CA or requests.
Your newly created PKI dir is: /etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/pki
```

We build the certificate authority:

$ ./easyrsa build-ca

Then, set a CA Key Passphrase and common name for your CA.

10

```
[root@ip-172-31-5-43 easyrsa3]# ./easyrsa build-ca

Note: using Easy-RSA configuration from: /etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/vars
Using SSL: openssl OpenSSL 1.0.2k-fips  26 Jan 2017

Enter New CA Key Passphrase:
Re-Enter New CA Key Passphrase:
Generating RSA private key, 2048 bit long modulus
.......................+++
.................+++
e is 65537 (0x10001)
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Common Name (eg: your user, host, or server name) [Easy-RSA CA]:Marieke
```

Create a key and certificate for the server:

$ ./easyrsa build-server-full server

11

```
[root@ip-172-31-5-43 easyrsa3]# ./easyrsa build-server-full server

Note: using Easy-RSA configuration from: /etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/vars
Using SSL: openssl OpenSSL 1.0.2k-fips  26 Jan 2017
Generating a 2048 bit RSA private key
..............................................................................................+++
.................+++
writing new private key to '/etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/pki/easy-rsa-4924.pcC104/tmp.Jh1rp9'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
Using configuration from /etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/pki/easy-rsa-4924.pcC104/tmp.gFjBlO
Enter pass phrase for /etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/pki/private/ca.key:
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName            :ASN.1 12:'server'
Certificate is to be certified until Feb 23 19:35:28 2024 GMT (825 days)

Write out database with 1 new entries
Data Base Updated
```

Generate a Diffie-Hellman key exchange file:

$ ./easyrsa gen-dh

12

```
[root@ip-172-31-5-43 easyrsa3]# ./easyrsa gen-dh

Note: using Easy-RSA configuration from: /etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/vars
Using SSL: openssl OpenSSL 1.0.2k-fips  26 Jan 2017
Generating DH parameters, 2048 bit long safe prime, generator 2
This is going to take a long time
.............................+.........................+.........................+.....
```

```
.....++*++*

DH parameters of size 2048 created at /etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/pki/dh.pem
```

We create a certificate and key for client1:

$ ./easyrsa build-client-full client1

13

```
[root@ip-172-31-5-43 easyrsa3]# ./easyrsa build-client-full client1

Note: using Easy-RSA configuration from: /etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/vars
Using SSL: openssl OpenSSL 1.0.2k-fips  26 Jan 2017
Generating a 2048 bit RSA private key
.........................+++
.........................+++
writing new private key to '/etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/pki/easy-rsa-1543.4ODn8N/tmp.qOusUu'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
Using configuration from /etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/pki/easy-rsa-1543.4ODn8N/tmp.fqkkHr
Enter pass phrase for /etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/pki/private/ca.key:
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName            :ASN.1 12:'client1'
Certificate is to be certified until Feb 23 19:39:13 2024 GMT (825 days)

Write out database with 1 new entries
Data Base Updated
```

We navigate to the PKI directory:

$ cd /etc/openvpn/easy-rsa/easyrsa3/pki

We copy the following 4 files into the open VPN directory:

ca.crt
dh.pem
ca.key
server.key

$ cp ca.crt dh.pem /etc/openvpn

We move into a subdirectory:

$ cd private

$ cp ca.key server.key/etc/openvpn

14

```
[root@ip-172-31-5-43 easyrsa3]# cd
[root@ip-172-31-5-43 ~]# cd /etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/pki
[root@ip-172-31-5-43 pki]# ls
ca.crt            dh.pem     index.txt.attr      index.txt.old  openssl-easyrsa.cnf  renewed  revoked              serial
certs_by_serial  index.txt  index.txt.attr.old  issued         private              reqs     safessl-easyrsa.cnf  serial.old
[root@ip-172-31-5-43 pki]# cp ca.crt dh.pem /etc/openvpn
[root@ip-172-31-5-43 pki]# cd private
[root@ip-172-31-5-43 private]# cp ca.key server.key /etc/openvpn
```

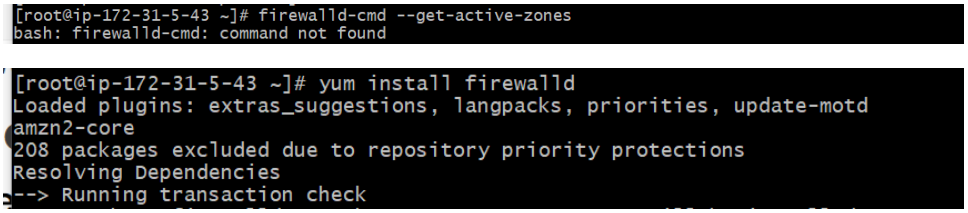
Step 5: Firewall and Routing Configuration

We check our firewalld zone

It did not work, so we reinstalled the firewalld:

$ yum install firewalld

15



We enabled firewalld:

$ systemctl enable firewalld

We checked the firewalld status:
$ sudo systemctl status firewalld

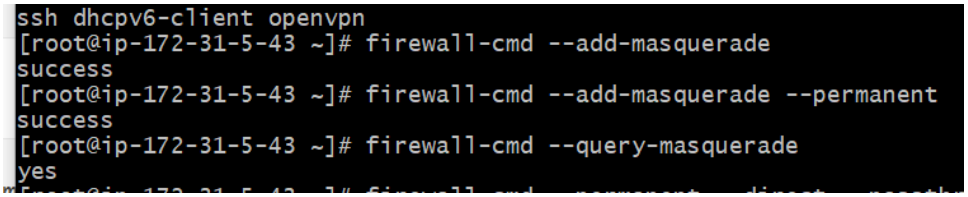
Firewall was active.

16

Next command to run:
$ firewall-cmd –add-masquerade
$ firewall-cmd –add-masquerade –permanent
$ firewall-cmd –query-masquerade

The output is here:

17



We route to our OpenVPN subnet.

$ VAR=$(ip route get 208.67.222.222 | awk ‘NR==1 {print $NF-2)}’)

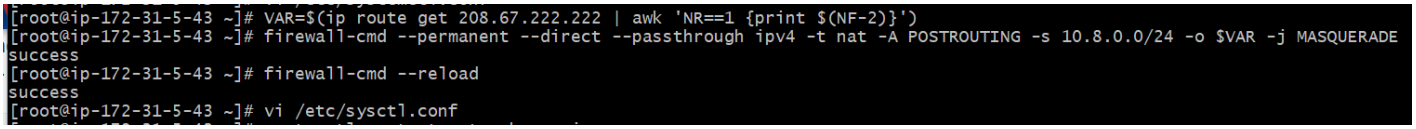
$ firewall-cmd –permanent –direct –direct –passthrough ipv4 –t nat –A POSTROUTING –s 10.8.0.0/24 -o $VAR –j MASQUERADE

(one command in one line)

We reload the firewalld:

$firewall-cmd --reload

18



We enable IP forwarding with vi and the file sysctl.conf.
We open the vi editor:

$ vi /etc/sysctl.conf

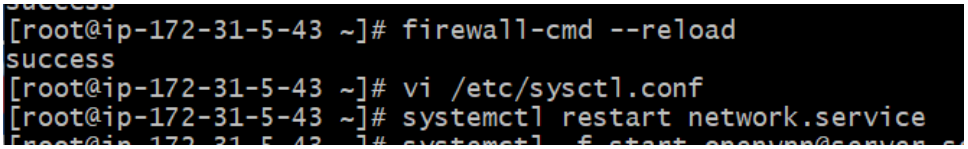
We add the following line:

Net.ipv4.ip_forward = 1

We restart the service:

$ systemctl restart network.service

19



Step 6: Start OpenVPN

We start the openvpn service with:

$ systemctl –f start openvpn@server.service

We got an error as seen in the image below:

19

```
[root@ip-172-31-5-43 ~]# systemctl -f start openvpn@server.service
Job for openvpn@server.service failed because the control process exited with error code. See "systemctl status openvpn@server.service" and
[root@ip-172-31-5-43 ~]# sysctl -f start openvpn@server.service
sysctl: cannot open "start": No such file or directory
sysctl: cannot open "openvpn@server.service": No such file or directory
[root@ip-172-31-5-43 ~]# sudo systemctl -f start openvpn@server.service
Job for openvpn@server.service failed because the control process exited with error code. See "systemctl status openvpn@server.service" and
```

We checked the status of our openvpn server.
The active status was ‘failed’.

We do not now the solution.
We have found another website with instructions how to create an OpenVPN.
Many differences occured between this solution and that page.

Configuring OpenVPN on AWS EC2 (Update: Jun 2019) (zealfortechnology.com)

(https://www.zealfortechnology.com/2018/08/configuring-openvpn-on-aws-ec2-update.html)

We were not able to test these commands because of a lack of time.

This line is a possible solution to our problem:

If server is running well with no error, but the client is still not able to connect; then disable tls-auth. Comment them out. Please note that tls-auth is not working for some version of OpenVPN, use tls-crypt instead.

20

```
The configuration file has been written to /home/ec2-user/mmvdh.ovpn.
Download the .ovpn file and import it in your OpenVPN client.
[root@ip-172-31-5-43 ~]# systemctl status openvpn@server.service
● openvpn@server.service - OpenVPN Robust And Highly Flexible Tunneling Application On server
   Loaded: loaded (/usr/lib/systemd/system/openvpn@.service; disabled; vendor preset: disabled)
   Active: failed (Result: exit-code) since Sat 2021-11-20 20:21:56 UTC; 10min ago
  Process: 4692 ExecStart=/usr/sbin/openvpn --cd /etc/openvpn/ --config %i.conf (code=exited, status=1/FAILURE)
 Main PID: 4692 (code=exited, status=1/FAILURE)

Nov 20 20:21:56 ip-172-31-5-43.eu-central-1.compute.internal systemd[1]: Starting OpenVPN Robust And Highly Flexible Tunneling Application On se
Nov 20 20:21:56 ip-172-31-5-43.eu-central-1.compute.internal openvpn[4692]: Options error: --dh fails with 'dh2048.pem': No such file or directo
Nov 20 20:21:56 ip-172-31-5-43.eu-central-1.compute.internal openvpn[4692]: Options error: --cert fails with 'server.crt': No such file or direc
Nov 20 20:21:56 ip-172-31-5-43.eu-central-1.compute.internal openvpn[4692]: Options error: Please correct these errors.
Nov 20 20:21:56 ip-172-31-5-43.eu-central-1.compute.internal openvpn[4692]: Use --help for more information.
Nov 20 20:21:56 ip-172-31-5-43.eu-central-1.compute.internal systemd[1]: openvpn@server.service: main process exited, code=exited, status=1/FAIL
Nov 20 20:21:56 ip-172-31-5-43.eu-central-1.compute.internal systemd[1]: Failed to start OpenVPN Robust And Highly Flexible Tunneling Applicatio
Nov 20 20:21:56 ip-172-31-5-43.eu-central-1.compute.internal systemd[1]: Unit openvpn@server.service entered failed state.
Nov 20 20:21:56 ip-172-31-5-43.eu-central-1.compute.internal systemd[1]: openvpn@server.service failed.
[root@ip-172-31-5-43 ~]# systemctl -f start openvpn@server.service
Job for openvpn@server.service failed because the control process exited with error code. See "systemctl status openvpn@server.service" and "jou
[root@ip-172-31-5-43 ~]# cd /etc/openvpn/easy-rsa/easy-rsa-3.0.8/easyrsa3/pki
[root@ip-172-31-5-43 pki]# cd private
[root@ip-172-31-5-43 private]# ls
ca.key  client1.key  server.key
```