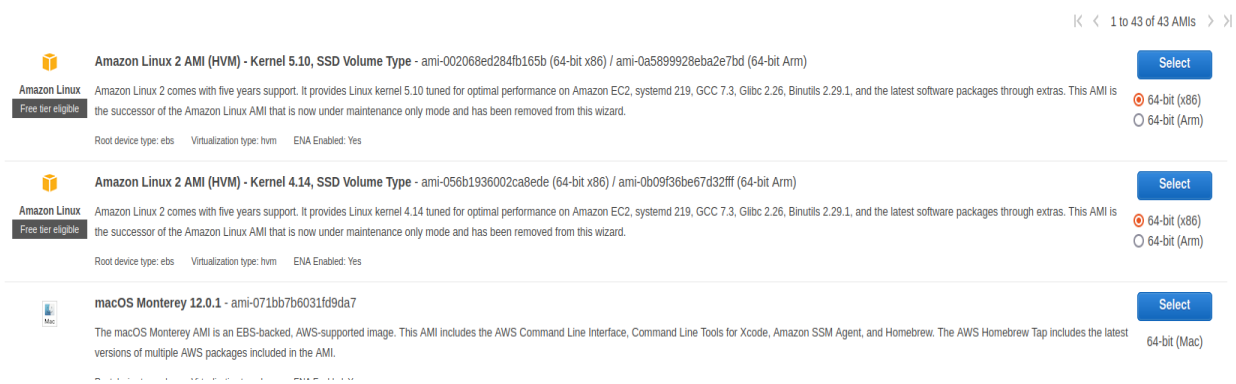
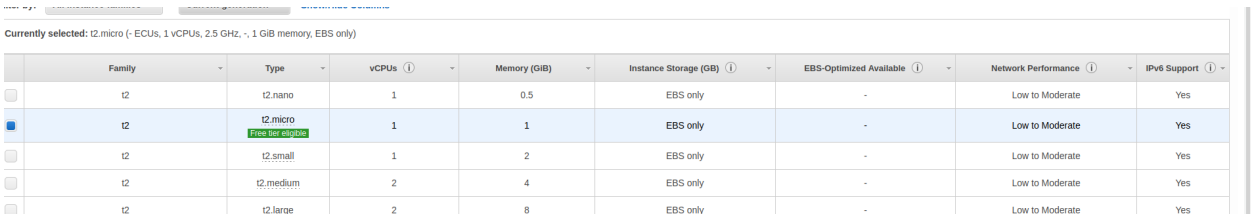
Create EC2 inside Public Subnet

- Allow SSH ingress traffic for your own IPs only
- Spin up simple http server @ 9099 port and verify it is accessible from public
- Install and Setup OpenVPN Server, open ports required to use it for these CIDR ranges 27.43.45.72, 103.110.112.124, 139.150.163.202, 0.0.0.0/8 and your own IPs
- (Optional) Create a OpenVPN Client with Split tunnel to use that Server

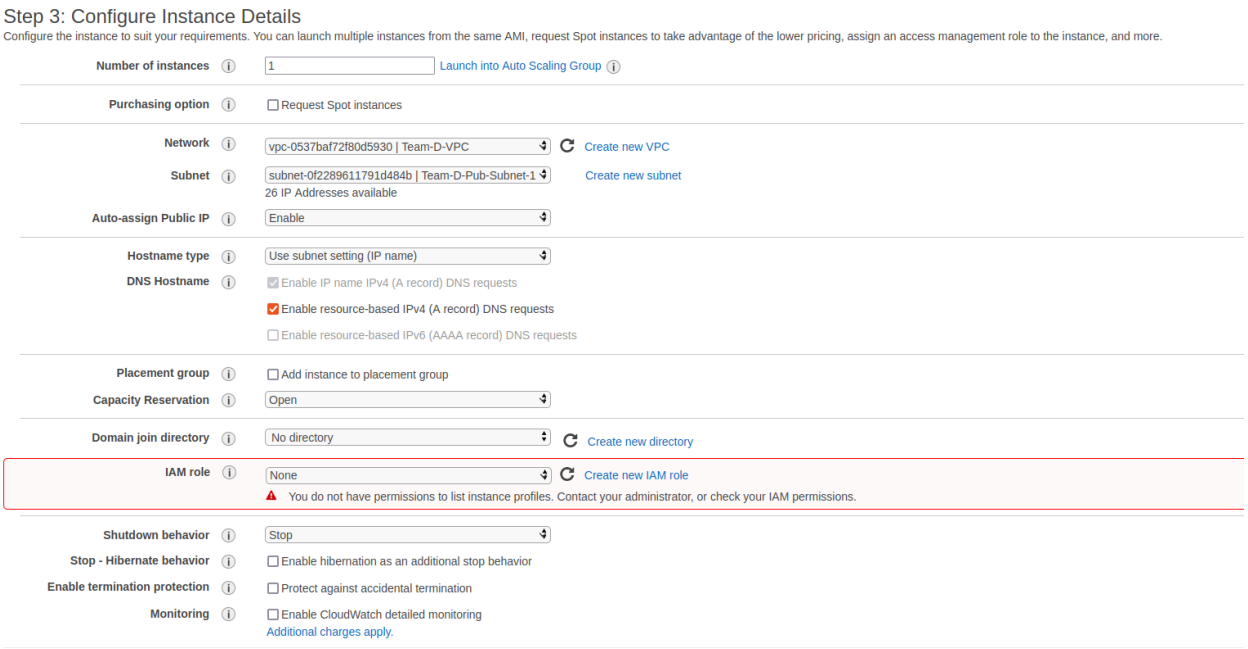
Select the Amazon Linux 2 ami :



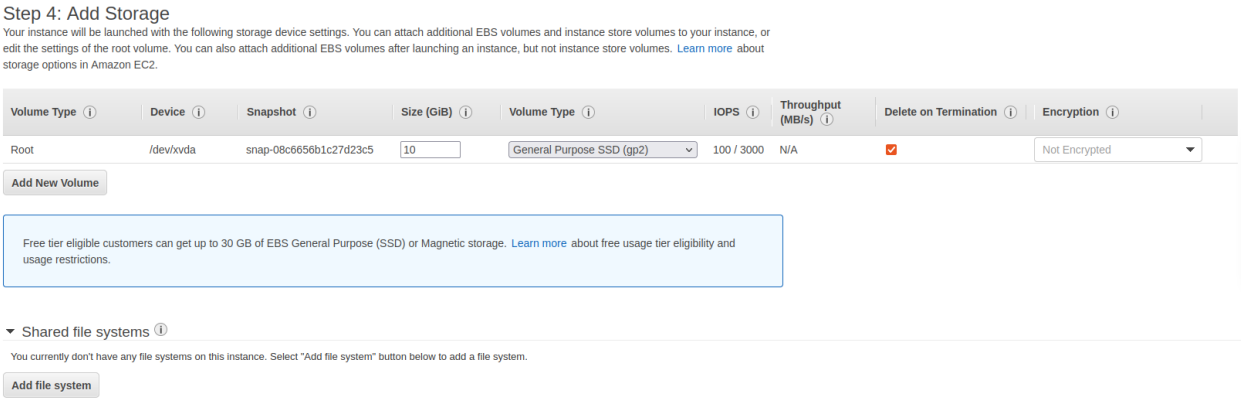
Choose **t2.micro :**



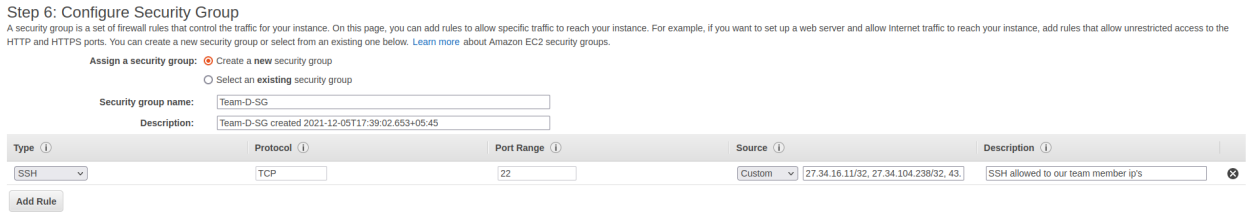
Configure the Instance Details using our **public subnet :**



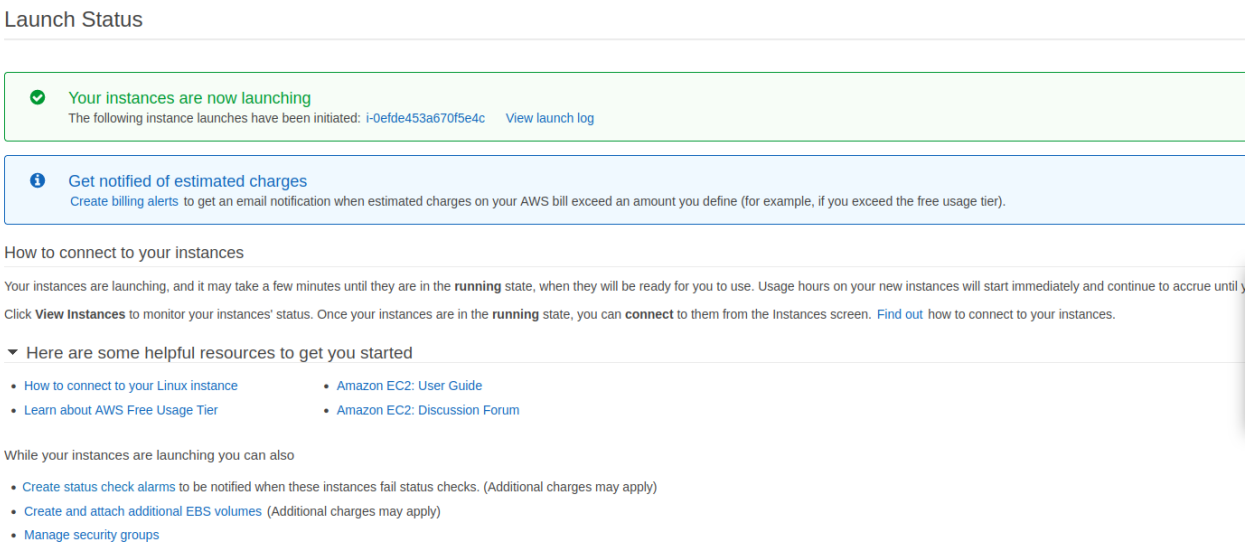
Add the storage



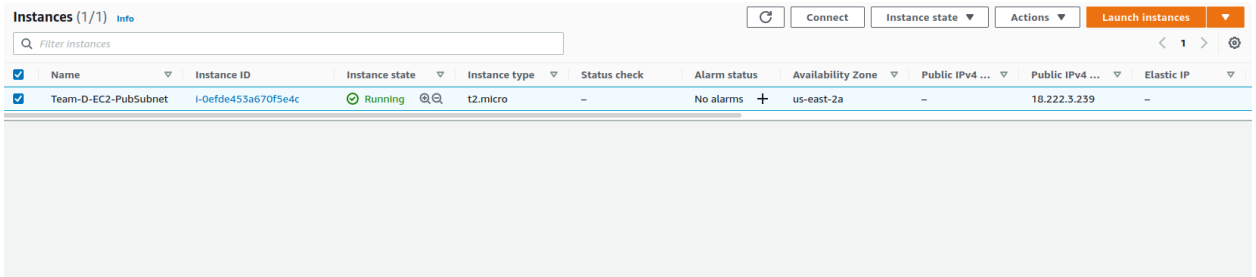
Configure the security grp as below allowing ssh to our ips:



Launch the instance



We can finally see the launched and running instance as below:



SSH to our EC2 instance as below:

```
ec2-user@ip-10-15-32-4:~
lostinserver@lostinserver:~/Downloads$ chmod 400 team-D-key.pem
lostinserver@lostinserver:~/Downloads$ ssh -i team-D-key.pem ec2-user@18.222.3.2
39
The authenticity of host '18.222.3.239 (18.222.3.239)' can't be established.
ECDSA key fingerprint is SHA256:MNi0WYYaXj0LMYHi/InkxFEOmeyLwA1Z7A/iHCdP87U.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '18.222.3.239' (ECDSA) to the list of known hosts.

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-15-32-4 ~]$
```

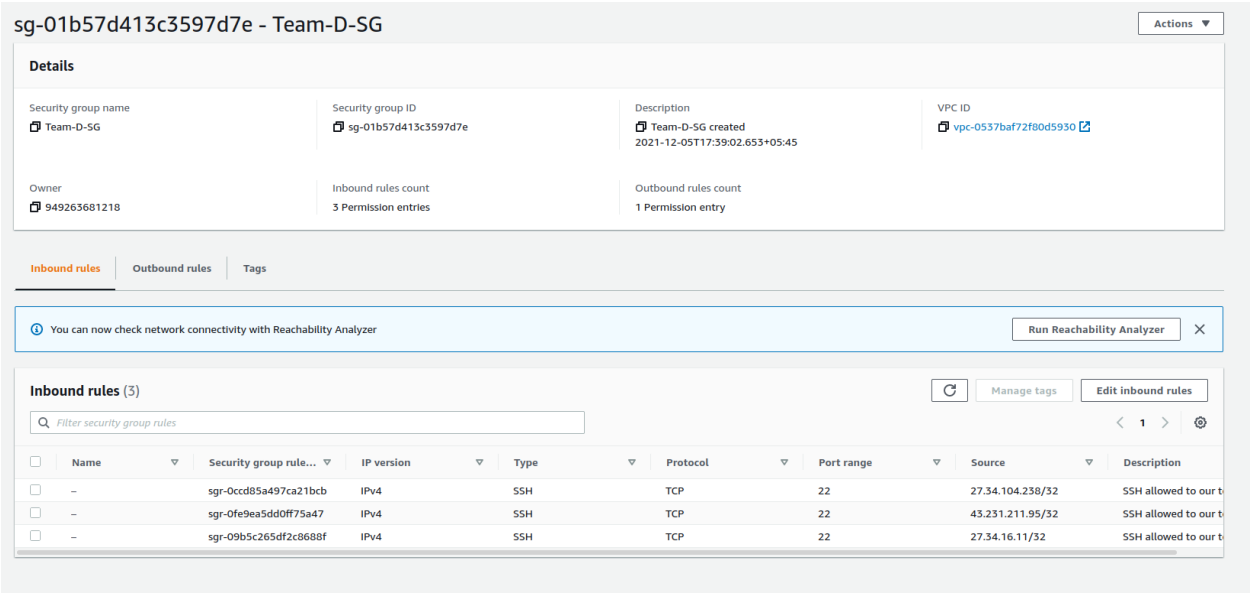
Create a test index page for testing:

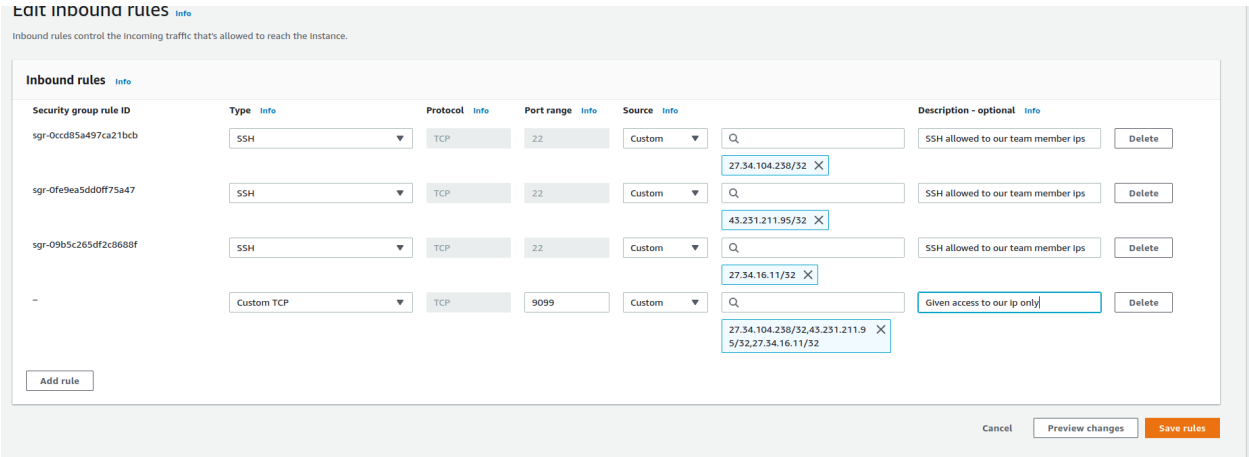
```
  GNU nano 2.9.8                    index.html                    Modified

<h1>Hello from team D</h1>
```

Run the server with our SSHed machine:

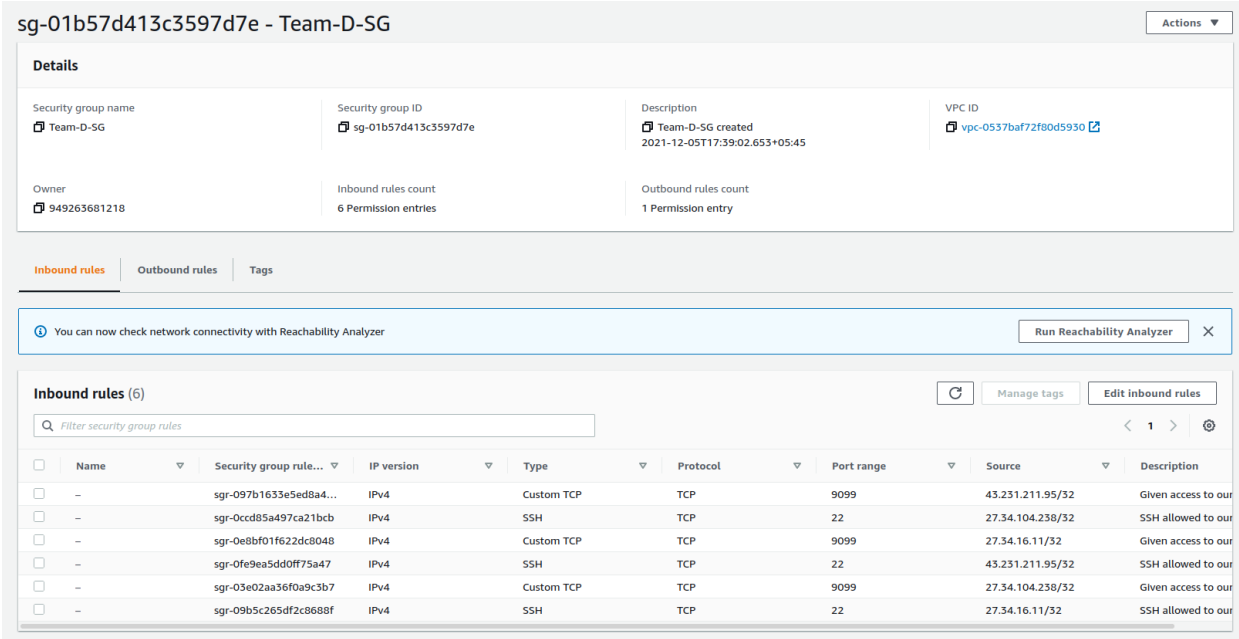
```
ec2-user@ip-10-15-32-4:~
[ec2-user@ip-10-15-32-4 ~]$ python3 -m http.server 9099
Serving HTTP on 0.0.0.0 port 9099 (http://0.0.0.0:9099/) ...
```

Update the inbound rules in our Security group allowing the Custom TCP port **9099** to our own IPs:

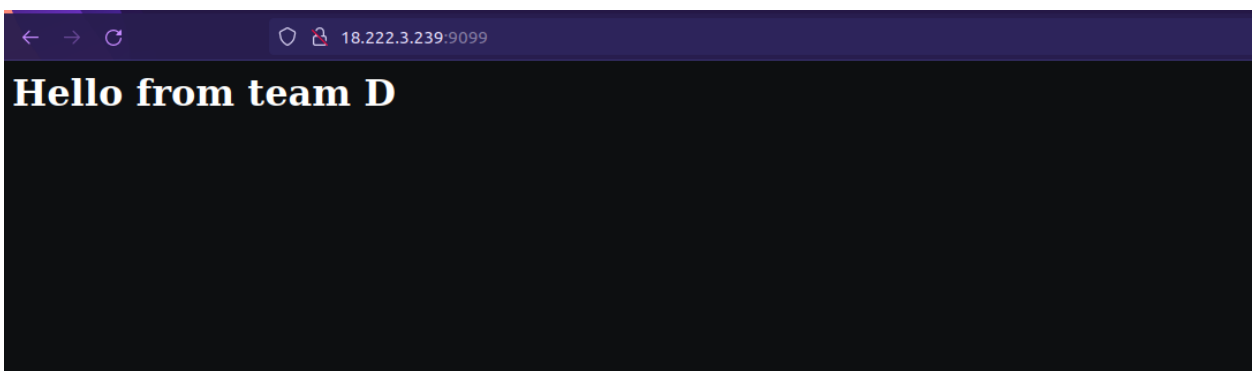




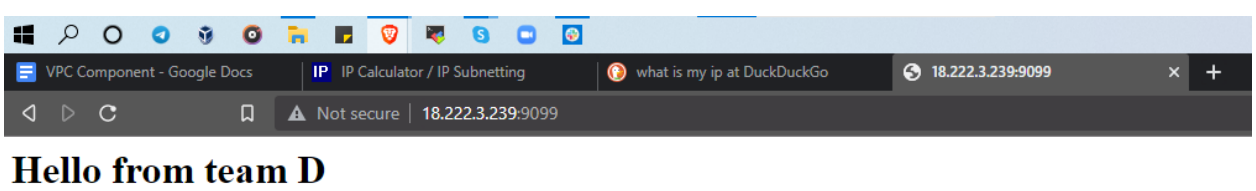
Updated Security Group



Verification that the site is accessible in public(**On Prerit System**):



On Bibek System:



Now we install the openvpn server and open ports for the CIDR ranges:

Firstly install amazon linux extras as :

Sudo amazon-linux-extras install epel -y
Sudo yum-config-manager –enable epel
Sudo yum update

```
[ec2-user@ip-10-15-32-4 ~]$ sudo amazon-linux-extras install epel -y
Installing epel-release
Failed to set locale, defaulting to C
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Cleaning repos: amzn2-core amzn2extra-docker amzn2extra-epel
              : amzn2extra-kernel-5.10
17 metadata files removed
6 sqlite files removed
0 metadata files removed
Failed to set locale, defaulting to C
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                | 3.7 kB     00:00
amzn2extra-docker                                         | 3.0 kB     00:00
amzn2extra-epel                                           | 3.0 kB     00:00
amzn2extra-kernel-5.10                                    | 3.0 kB     00:00
(1/9): amzn2-core/2/x86_64/group_gz                         | 2.5 kB   00:00
(2/9): amzn2-core/2/x86_64/updateinfo                       | 424 kB   00:00
(3/9): amzn2extra-epel/2/x86_64/primary_db                  | 1.8 kB   00:00
(4/9): amzn2extra-kernel-5.10/2/x86_64/updateinfo           |   76 B   00:00
(5/9): amzn2extra-kernel-5.10/2/x86_64/primary_db           | 5.3 MB   00:00
(6/9): amzn2extra-docker/2/x86_64/updateinfo                | 4.7 kB   00:00
(7/9): amzn2extra-docker/2/x86_64/primary_db                |  86 kB   00:00
(8/9): amzn2extra-epel/2/x86_64/updateinfo                  |   76 B   00:00
```

```
[ec2-user@ip-10-15-32-4 ~]$ sudo yum-config-manager --enable epel
Failed to set locale, defaulting to C
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
================================== repo: epel ==================================
[epel]
async = True
bandwidth = 0
base_persistdir = /var/lib/yum/repos/x86_64/2
baseurl =
cache = 0
cachedir = /var/cache/yum/x86_64/2/epel
check_config_file_age = True
compare_providers_priority = 80
cost = 1000
deltarpm_metadata_percentage = 100
deltarpm_percentage =
enabled = True
enablegroups = True
exclude =
failovermethod = priority
ftp_disable_epsv = False
```

```
[ec2-user@ip-10-15-32-4 ~]$ sudo yum update
Failed to set locale, defaulting to C
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
epel/x86_64/metalink                                     | 9.1 kB     00:00
209 packages excluded due to repository priority protections
Resolving Dependencies
--> Running transaction check
---> Package python-lockfile.noarch 1:0.9.1-4.amzn2 will be obsoleted
---> Package python-simplejson.x86_64 0:3.2.0-1.amzn2.0.2 will be obsoleted
---> Package python2-lockfile.noarch 1:0.11.0-17.el7 will be obsoleting
---> Package python2-simplejson.x86_64 0:3.11.1-1.el7 will be obsoleting
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                  Arch         Version                 Repository  Size
================================================================================
Installing:
 python2-lockfile         noarch       1:0.11.0-17.el7         epel        29 k
     replacing  python-lockfile.noarch 1:0.9.1-4.amzn2
 python2-simplejson       x86_64       3.11.1-1.el7            epel       188 k
```

Install the OpenVPN as :

wget

https://raw.githubusercontent.com/angristan/openvpn-install/master/openvpn-install.sh

sudo chmod +x openvpn-install.sh

sudo bash openvpn-install.sh

```
What protocol do you want OpenVPN to use?
UDP is faster. Unless it is not available, you shouldn't use TCP.
   1) UDP
   2) TCP
Protocol [1-2]: 1

What DNS resolvers do you want to use with the VPN?
   1) Current system resolvers (from /etc/resolv.conf)
   2) Self-hosted DNS Resolver (Unbound)
   3) Cloudflare (Anycast: worldwide)
   4) Quad9 (Anycast: worldwide)
   5) Quad9 uncensored (Anycast: worldwide)
   6) FDN (France)
   7) DNS.WATCH (Germany)
   8) OpenDNS (Anycast: worldwide)
   9) Google (Anycast: worldwide)
   10) Yandex Basic (Russia)
   11) AdGuard DNS (Anycast: worldwide)
   12) NextDNS (Anycast: worldwide)
   13) Custom
DNS [1-12]: 9

Do you want to use compression? It is not recommended since the VORACLE attack makes use of it.
Enable compression? [y/n]: n

Do you want to customize encryption settings?
Unless you know what you're doing, you should stick with the default parameters provided by the scri
pt.
Note that whatever you choose, all the choices presented in the script are safe. (Unlike OpenVPN's d
efaults)
See https://github.com/angristan/openvpn-install#security-and-encryption to learn more.

Customize encryption settings? [y/n]: n

Okay, that was all I needed. We are ready to setup your OpenVPN server now.
You will be able to generate a client at the end of the installation.
Press any key to continue...
```

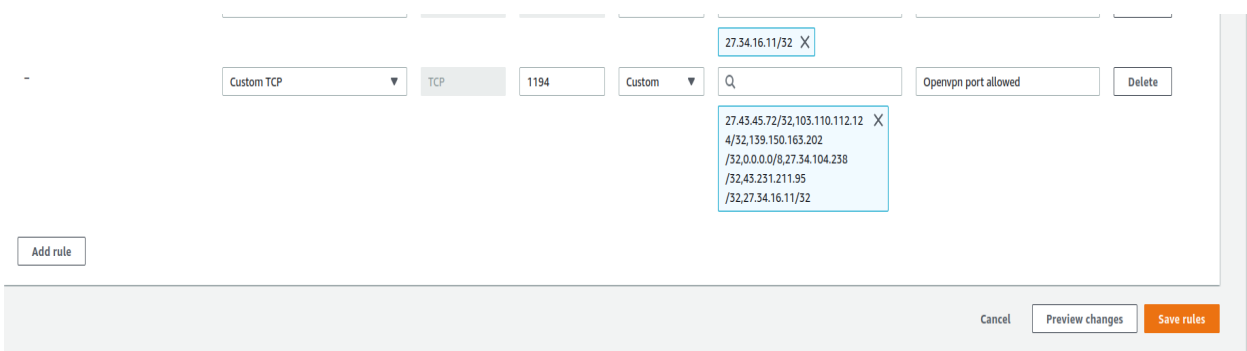
```
commonName             :ASN.1 12:'server_H1IjE3QWbw6ORxWJ'
Certificate is to be certified until Feb  5 14:01:21 2024 GMT (825 days)

Write out database with 1 new entries
Data Base Updated


Note: using Easy-RSA configuration from: /etc/openvpn/easy-rsa/vars
Using SSL: openssl OpenSSL 1.0.2k-fips  26 Jan 2017
Using configuration from /etc/openvpn/easy-rsa/pki/easy-rsa-18871.riswtP/tmp.cMZthJ

An updated CRL has been created.
CRL file: /etc/openvpn/easy-rsa/pki/crl.pem


* Applying /usr/lib/sysctl.d/00-system.conf ...
* Applying /usr/lib/sysctl.d/10-default-yama-scope.conf ...
kernel.yama.ptrace_scope = 0
* Applying /usr/lib/sysctl.d/50-default.conf ...
kernel.sysrq = 16
kernel.core_uses_pid = 1
kernel.kptr_restrict = 1
net.ipv4.conf.default.rp_filter = 1
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.default.accept_source_route = 0
net.ipv4.conf.all.accept_source_route = 0
net.ipv4.conf.default.promote_secondaries = 1
net.ipv4.conf.all.promote_secondaries = 1
fs.protected_hardlinks = 1
fs.protected_symlinks = 1
* Applying /etc/sysctl.d/99-openvpn.conf ...
net.ipv4.ip_forward = 1
* Applying /etc/sysctl.d/99-sysctl.conf ...
* Applying /etc/sysctl.conf ...
Created symlink from /etc/systemd/system/multi-user.target.wants/openvpn-server@server.service to /e
tc/systemd/system/openvpn-server@.service.
```

```
* Applying /etc/sysctl.d/99-sysctl.conf ...
* Applying /etc/sysctl.conf ...
Created symlink from /etc/systemd/system/multi-user.target.wants/openvpn-server@server.service to /e
tc/systemd/system/openvpn-server@.service.
Created symlink from /etc/systemd/system/multi-user.target.wants/iptables-openvpn.service to /etc/sy
stemd/system/iptables-openvpn.service.

Tell me a name for the client.
The name must consist of alphanumeric character. It may also include an underscore or a dash.
Client name: client

Do you want to protect the configuration file with a password?
(e.g. encrypt the private key with a password)
   1) Add a passwordless client
   2) Use a password for the client
Select an option [1-2]: 1

Note: using Easy-RSA configuration from: /etc/openvpn/easy-rsa/vars
Using SSL: openssl OpenSSL 1.0.2k-fips  26 Jan 2017
Generating a 256 bit EC private key
writing new private key to '/etc/openvpn/easy-rsa/pki/easy-rsa-19040.kEfDfz/tmp.Z4LOQB'
-----
Using configuration from /etc/openvpn/easy-rsa/pki/easy-rsa-19040.kEfDfz/tmp.o4Mu1q
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName             :ASN.1 12:'client'
Certificate is to be certified until Feb  5 14:01:44 2024 GMT (825 days)

Write out database with 1 new entries
Data Base Updated

Client client added.

The configuration file has been written to /root/client.ovpn.
Download the .ovpn file and import it in your OpenVPN client.
```

Systemctl start openvpn@server.service
Systemctl status openvpn@server.service

```
[root@ip-10-15-32-4 easy-rsa]# systemctl status openvpn@server.service
● openvpn@server.service - OpenVPN Robust And Highly Flexible Tunneling Applica
tion On server
   Loaded: loaded (/usr/lib/systemd/system/openvpn@.service; disabled; vendor p
reset: disabled)
   Active: inactive (dead)
[root@ip-10-15-32-4 easy-rsa]# systemctl enable openvpn@server.service
Created symlink from /etc/systemd/system/multi-user.target.wants/openvpn@server
.service to /usr/lib/systemd/system/openvpn@.service.
[root@ip-10-15-32-4 easy-rsa]# systemctl start openvpn@server.service
[root@ip-10-15-32-4 easy-rsa]# systemctl status openvpn@server.service
● openvpn@server.service - OpenVPN Robust And Highly Flexible Tunneling Applica
tion On server
   Loaded: loaded (/usr/lib/systemd/system/openvpn@.service; enabled; vendor pr
eset: disabled)
   Active: active (running) since Sun 2021-12-05 16:52:17 UTC; 2s ago
 Main PID: 1324 (openvpn)
   Status: "Initialization Sequence Completed"
   CGroup: /system.slice/system-openvpn.slice/openvpn@server.service
           └─1324 /usr/sbin/openvpn --cd /etc/openvpn/ --config server.conf

Dec 05 16:52:17 ip-10-15-32-4.us-east-2.compute.internal systemd[1]: Startin...
Dec 05 16:52:17 ip-10-15-32-4.us-east-2.compute.internal systemd[1]: Started...
```

Add the inbound rules in the security grp for CIDR ranges 27.43.45.72, 103.110.112.124, 139.150.163.202, 0.0.0.0/8 and our own IPs



Inbound rules (10)

Manage tags | Edit inbound rules

Filter security group rules

Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
–	sgr-04e2f1d5fd342bd09	IPv4	Custom TCP	TCP	1194	139.150.163.202/32	Openvpn port allo
–	sgr-0e744c328bfced217	IPv4	Custom TCP	TCP	1194	27.34.16.11/32	Openvpn port allo
–	sgr-0ccd85a497ca21bcb	IPv4	SSH	TCP	22	27.34.104.238/32	SSH allowed to our
–	sgr-01b85d7a2dfdb60...	IPv4	Custom TCP	TCP	1194	103.110.112.124/32	Openvpn port allo
–	sgr-0c644457633db9...	IPv4	Custom TCP	TCP	1194	43.231.211.95/32	Openvpn port allo
–	sgr-0a9d3b3f05fe94285	IPv4	Custom TCP	TCP	1194	27.34.104.238/32	Openvpn port allo
–	sgr-07d3fcb2946cb3124	IPv4	Custom TCP	TCP	1194	0.0.0.0/8	Openvpn port allo
–	sgr-0fe9ea5dd0ff75a47	IPv4	SSH	TCP	22	43.231.211.95/32	SSH allowed to our
–	sgr-0122b17eba0467...	IPv4	Custom TCP	TCP	1194	27.43.45.72/32	Openvpn port allo
–	sgr-09b5c265df2c8688f	IPv4	SSH	TCP	22	27.34.16.11/32	SSH allowed to our

We install the openvpn in our client and add the keys which we can see the client keys and certificates through rsync. The certificates and keys are as :

```
lostinserver@lostinserver:~/openclient$ ls
ca.crt  client.crt  client.key  myvpn.tlsauth
lostinserver@lostinserver:~/openclient$
```

Add the udp for port **1194** as,

sgr-05882eed085d60d3f	Custom UDP	UDP	1194	Custom	27.34.16.11/32	udp allow	Delete

We have our ovpn file as,

```
lostinserver@lostinserver: ~/openclient
lostinserver@lostinserver:~/openclient$ ls
ca.crt  client.crt  client.key  client.ovpn  dh2048.pem  myvpn.tlsauth
lostinserver@lostinserver:~/openclient$ cat client.ovpn
client
tls-client
ca ca.crt
cert client.crt
key client.key
tls-crypt myvpn.tlsauth
remote-cert-eku "TLS Web Server Authentication"
proto tcp
remote 18.222.3.239 1194 udp
pull
dev tun
topology subnet
user nobody
group nobody
route-nopull
route 10.15.32.0/22 255.255.240.0
lostinserver@lostinserver:~/openclient$
```

Now start the openvpn service as ,

Sudo openvpn –config client.ovpn

```
lostinserver@lostinserver:~/openclient$ sudo openvpn --config client.ovpn
Sun Dec  5 23:59:01 2021 WARNING: file 'client.key' is group or others accessible
Sun Dec  5 23:59:01 2021 WARNING: file 'myvpn.tlsauth' is group or others accessible
Sun Dec  5 23:59:01 2021 OpenVPN 2.4.7 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] built on Jul 19 2021
Sun Dec  5 23:59:01 2021 library versions: OpenSSL 1.1.1f  31 Mar 2020, LZO 2.10
Sun Dec  5 23:59:01 2021 WARNING: you are using user/group/chroot/setcon without persist-tun -- this may cause restarts to fail
Sun Dec  5 23:59:01 2021 WARNING: you are using user/group/chroot/setcon without persist-key -- this may cause restarts to fail
Sun Dec  5 23:59:01 2021 TCP/UDP: Preserving recently used remote address: [AF_INET]18.222.3.239:1194
Sun Dec  5 23:59:01 2021 UDP link local (bound): [AF_INET][undef]:1194
Sun Dec  5 23:59:01 2021 UDP link remote: [AF_INET]18.222.3.239:1194
Sun Dec  5 23:59:01 2021 NOTE: UID/GID downgrade will be delayed because of --client, --pull, or --up-delay
Sun Dec  5 23:59:03 2021 WARNING: 'link-mtu' is used inconsistently, local='link-mtu 1541', remote='link-mtu 1557'
Sun Dec  5 23:59:03 2021 WARNING: 'cipher' is used inconsistently, local='cipher BF-CBC', remote='cipher AES-256-CBC'
Sun Dec  5 23:59:03 2021 WARNING: 'keysize' is used inconsistently, local='keysize 128', remote='keysize 256'
Sun Dec  5 23:59:03 2021 [server] Peer Connection Initiated with [AF_INET]18.222.3.239:1194
Sun Dec  5 23:59:04 2021 TUN/TAP device tun0 opened
Sun Dec  5 23:59:04 2021 /sbin/ip link set dev tun0 up mtu 1500
Sun Dec  5 23:59:04 2021 /sbin/ip addr add dev tun0 10.15.32.98/27 broadcast 10.15.32.127
RTNETLINK answers: File exists
Sun Dec  5 23:59:04 2021 ERROR: Linux route add command failed: external program exited with error status: 2
Sun Dec  5 23:59:04 2021 GID set to nobody
Sun Dec  5 23:59:04 2021 UID set to nobody
Sun Dec  5 23:59:04 2021 WARNING: this configuration may cache passwords in memory -- use the auth-nocache option to prevent this
Sun Dec  5 23:59:04 2021 Initialization Sequence Completed
```

We can see the initialization sequence is completed and vpn is running.

We can see the tunneling with **ip a** command as:

```
tun0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST>  mtu 1500
        inet 10.15.32.98  netmask 255.255.255.224  destination 10.15.32.98
        inet6 fe80::ca34:1255:3157:9e1d  prefixlen 64  scopeid 0x20<link>
        unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  txqueuelen 100  (UNSPEC)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 4  bytes 192 (192.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

We can see the local system(client) accessing the openvpn server created inside the ec2 instance.