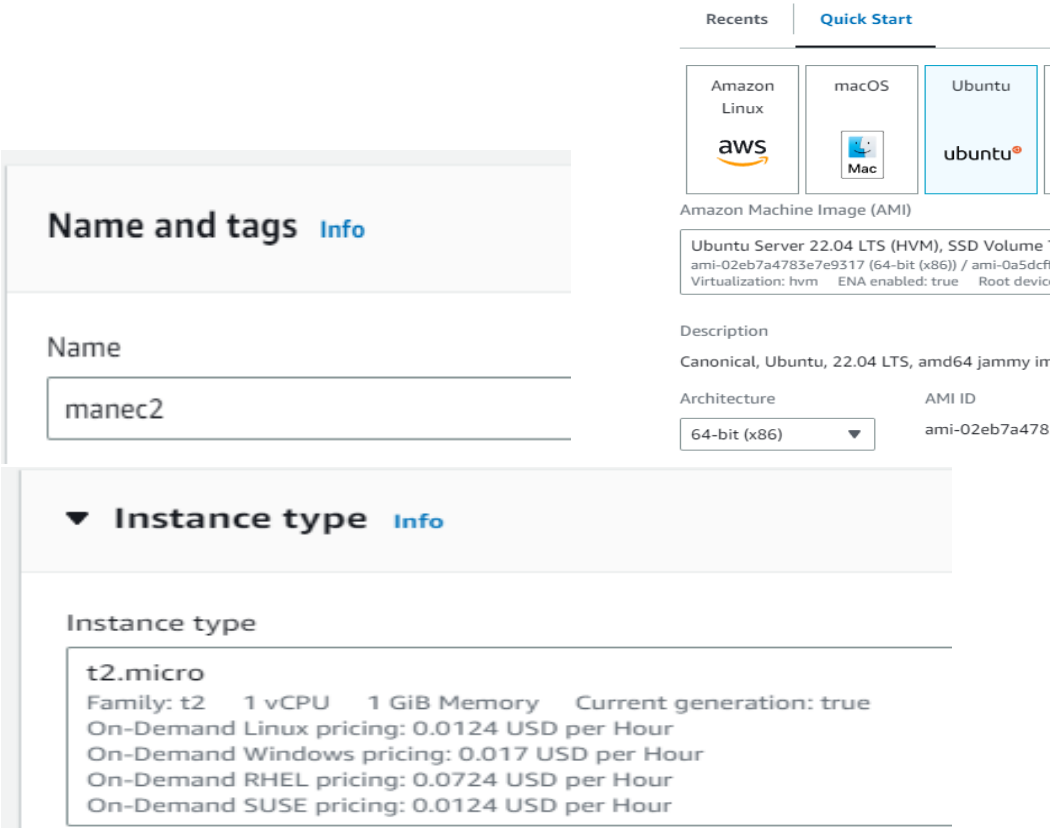
ASSIGNMENT NO-12

Problem Statement:- Deploy and run project in AWS without using port.

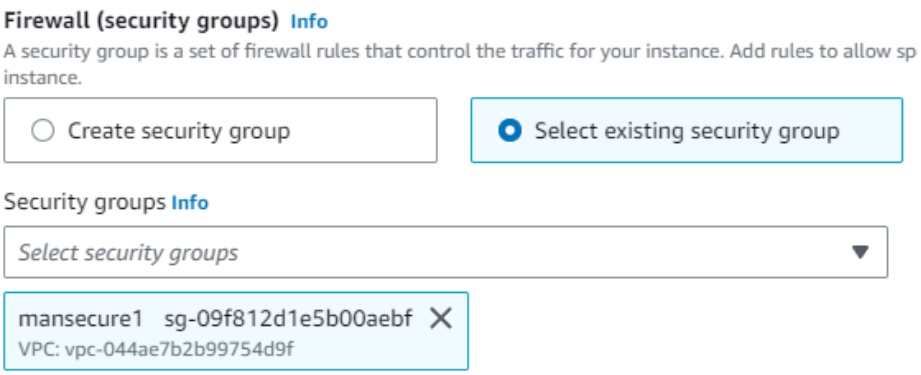
Steps:-

EC2 creation:-

1. In EC2 dashboard click **launch instances**. Give name,select **ubuntu** for hardware, architecture is 64-bit, instance type is t2.micro.



2. Give key pair(in case if u have it give existing one). In Network settings Firewall section click **select existing security group** and select the security group which is created previously(ex-mansecure1).



3. In Advance details , User data section type the following commands-

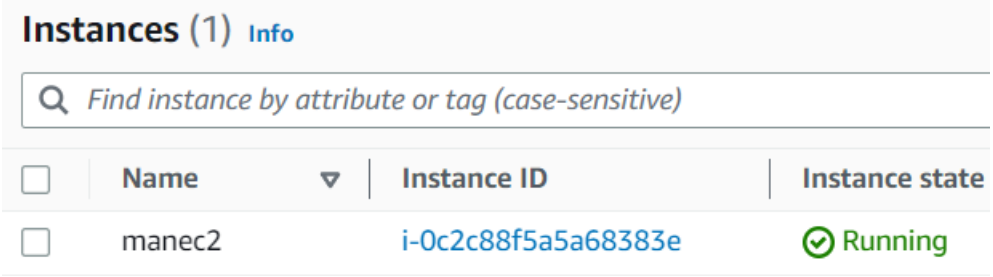
User data - *optional* Info

Enter user data in the field.

```
#!/bin/bash
apt-get update
apt-get install -y nginx
systemctl start nginx
systemctl enable nginx
apt-get install -y git
curl -sl https://deb.nodesource.com/setup_18.x | sudo -E bash -
apt-get install -y nodejs
git clone https://github.com/manas003884/repo2.git
cd repo2
npm install
node index.js
```

(In git clone line user can give his repo name and cd line the repo name will be according to his repo name.)

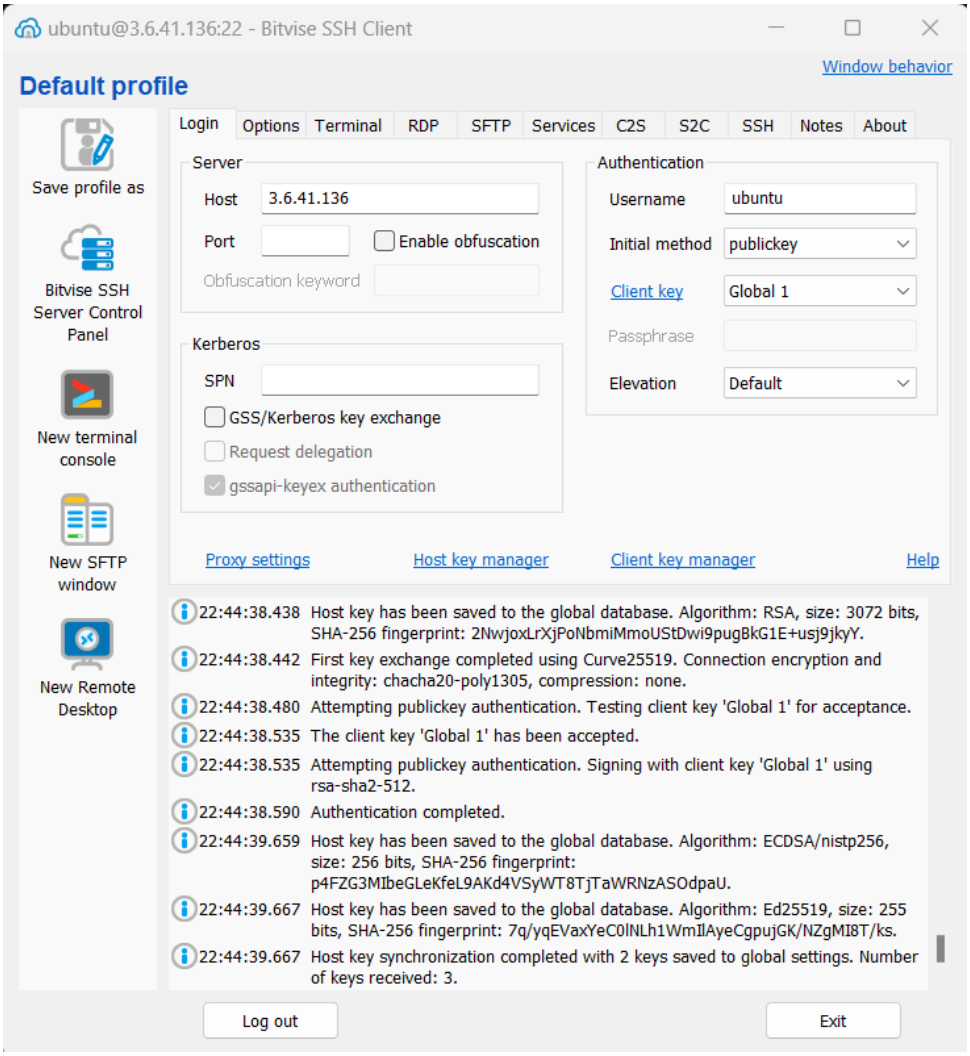
Now, click launch instances and the instance will be created.



Now, copy public IPv4 address and run it with :4000 and we will get the website.

Connect with Bitvise SSH:-

1. Copy public IPv4 address(ex- 3.6.41.136) and paste it on Bitvise SSH Client. Give username ubuntu, initial method publickey, in client key manager import that same existed key pair .pem file(ex-key003.pem) and click Global1 in Client key.and click log in.



2. Now ,in terminal type the following commands-

```
ubuntu@ip-172-31-37-112:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-37-112:~$ cd /
ubuntu@ip-172-31-37-112:/$ pwd
/
ubuntu@ip-172-31-37-112:/$ cd /etc/nginx/sites-available/
ubuntu@ip-172-31-37-112:/etc/nginx/sites-available$ sudo nano default
```

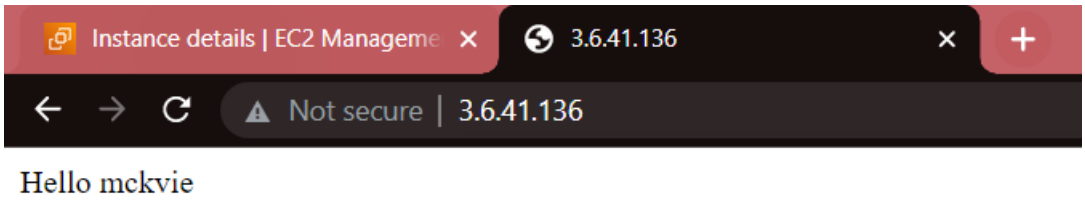
3. In the default file, edit the location to following-

```
 GNU nano 6.2                                          default *
        # Note: You should disable gzip for SSL traffic.
        # See: https://bugs.debian.org/773332
        #
        # Read up on ssl_ciphers to ensure a secure configuration.
        # See: https://bugs.debian.org/765782
        #
        # Self signed certs generated by the ssl-cert package
        # Don't use them in a production server!
        #
        # include snippets/snakeoil.conf;

        root /var/www/html;

        # Add index.php to the list if you are using PHP
        index index.html index.htm index.nginx-debian.html;

        server_name _;

#       location / {
                # First attempt to serve request as file, then
                # as directory, then fall back to displaying a 404.
#               try_files $uri $uri/ =404;
#       }
location / {
                proxy_pass http://localhost:4000;
                proxy_http_version 1.1;
                proxy_set_header Upgrade $http_upgrade;
                proxy_set_header Connection 'Upgrade';
                proxy_set_header Host $host;
                proxy_cache_bypass $http_upgrade;
        }
```

4. Now start the server by giving the following command-**"sudo systemctl restart nginx".**

```
ubuntu@ip-172-31-37-112:/etc/nginx/sites-available$ sudo systemctl restart nginx
```

5. Now copy that IPv4 address and paste it in another tab and we can see that without port number the server is running.



Hello mckvie