

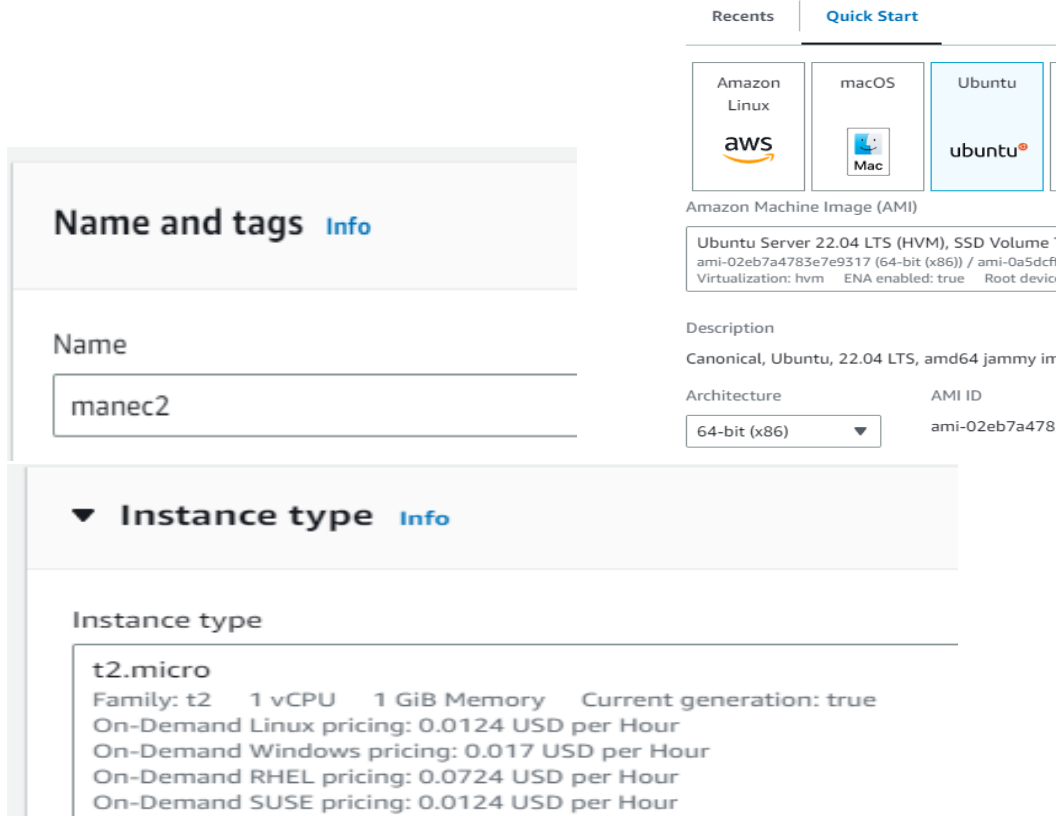
# ASSIGNMENT NO-12

 **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.



**Name and tags** [Info](#)

Name

manec2

**▼ Instance type** [Info](#)

Instance type

t2.micro

Family: t2 1 vCPU 1 GiB Memory Current generation: true

On-Demand Linux pricing: 0.0124 USD per Hour

On-Demand Windows pricing: 0.017 USD per Hour

On-Demand RHEL pricing: 0.0724 USD per Hour

On-Demand SUSE pricing: 0.0124 USD per Hour

**Amazon Machine Image (AMI)**

Ubuntu Server 22.04 LTS (HVM), SSD Volume

ami-02eb7a4783e7e9317 (64-bit (x86)) / ami-0a5dcf

Virtualization: hvm ENA enabled: true Root devic

Description

Canonical, Ubuntu, 22.04 LTS, amd64 jammy ir

Architecture

64-bit (x86)

AMI ID

ami-02eb7a478

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).

## **Firewall (security groups)** [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow sp instance.

☐ Create security group

☒ Select existing security group

## **Security groups** [Info](#)

Select security groups

mansecure1 sg-09f812d1e5b00aebf X

VPC: vpc-044ae7b2b99754d9f

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.)

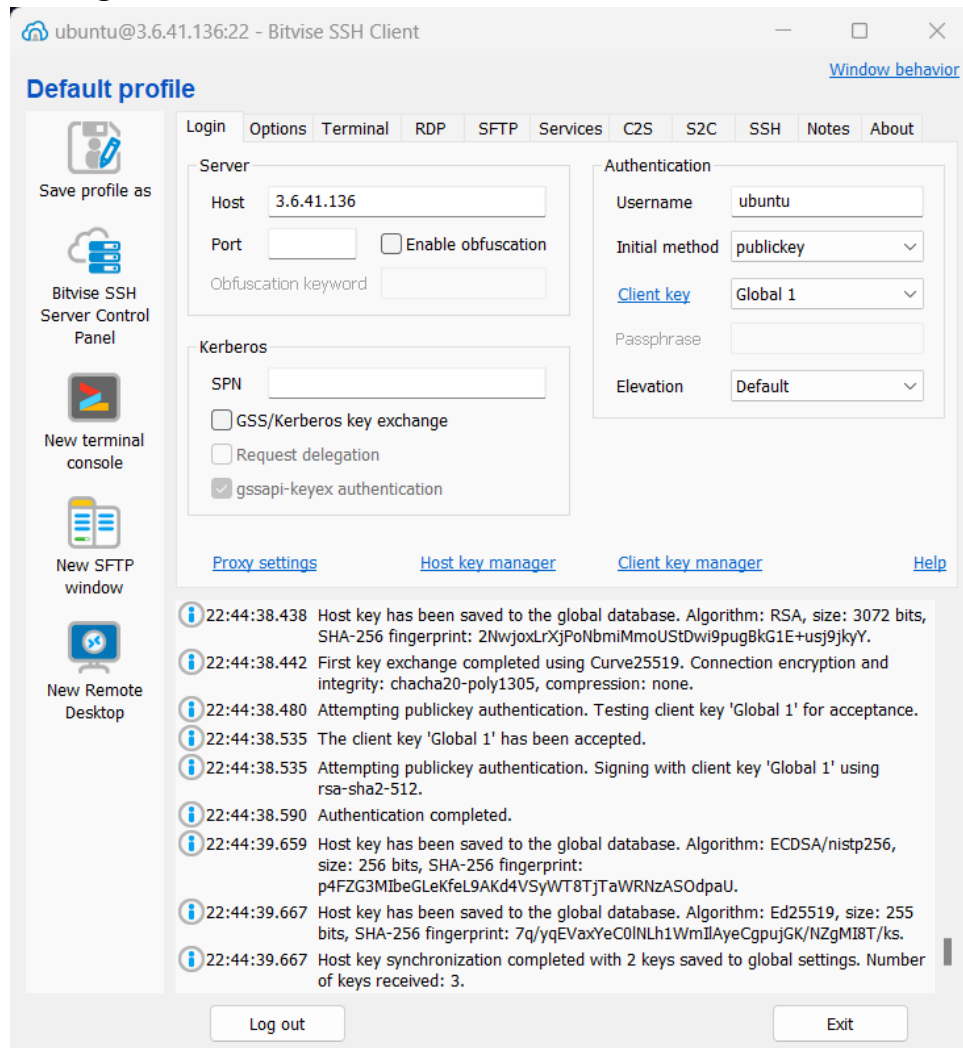
Now, click launch instances and the instance will be created.

Instances (1) <a href="#">Info</a>			
<input type="text" value="Find instance by attribute or tag (case-sensitive)"/>			
<input type="checkbox"/>	Name	Instance ID	Instance state
<input type="checkbox"/>	manec2	i-0c2c88f5a5a68383e	<span>✓ Running</span>

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 _;

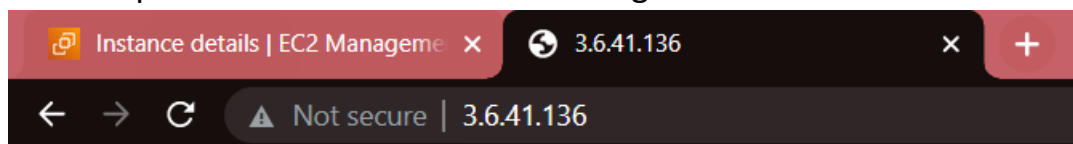
#
location / {
    # First attempt to serve request as file, then
    # as directory, then fall back to displaying a 404.
    try_files $uri $uri/ =404;
#
}
location / {
    proxy_pass http://localhost:4000;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header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