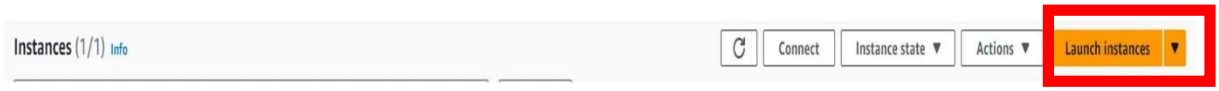
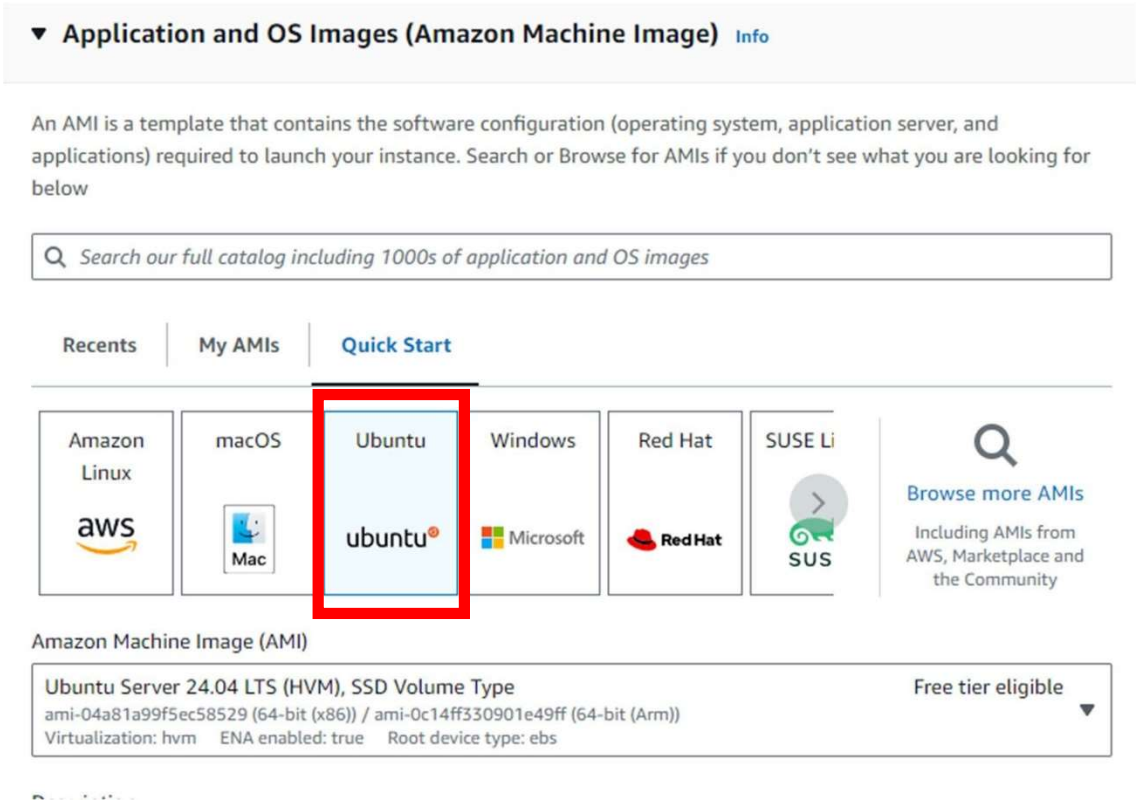
Step 1: Launch an EC2 Instance

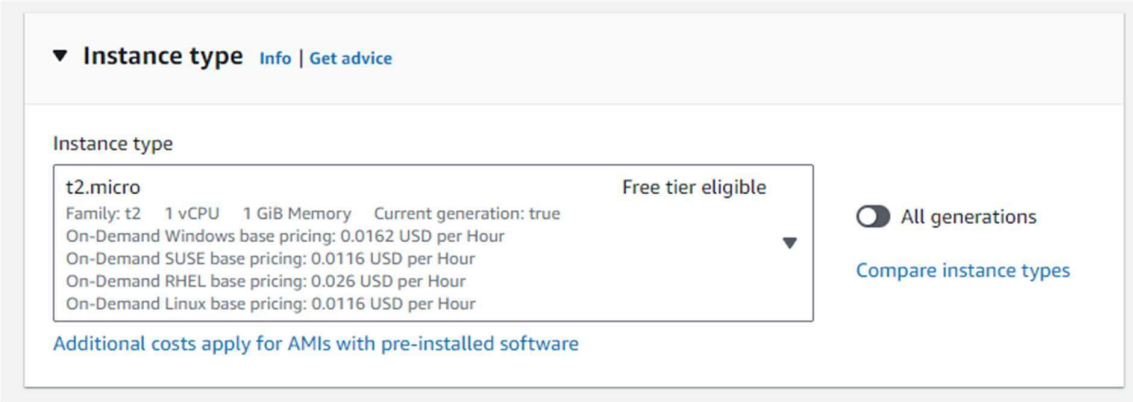
1. **Log in to AWS Management Console**:
    - Go to the AWS Management Console at https://aws.amazon.com/console/
    - Sign in with your AWS credentials.
2. **Navigate to EC2 Dashboard**:
    - In the AWS Management Console, type "EC2" in the search bar and select EC2 to navigate to the EC2 Dashboard.
3. **Launch an Instance**:
    - Click on the "Launch Instance" button.



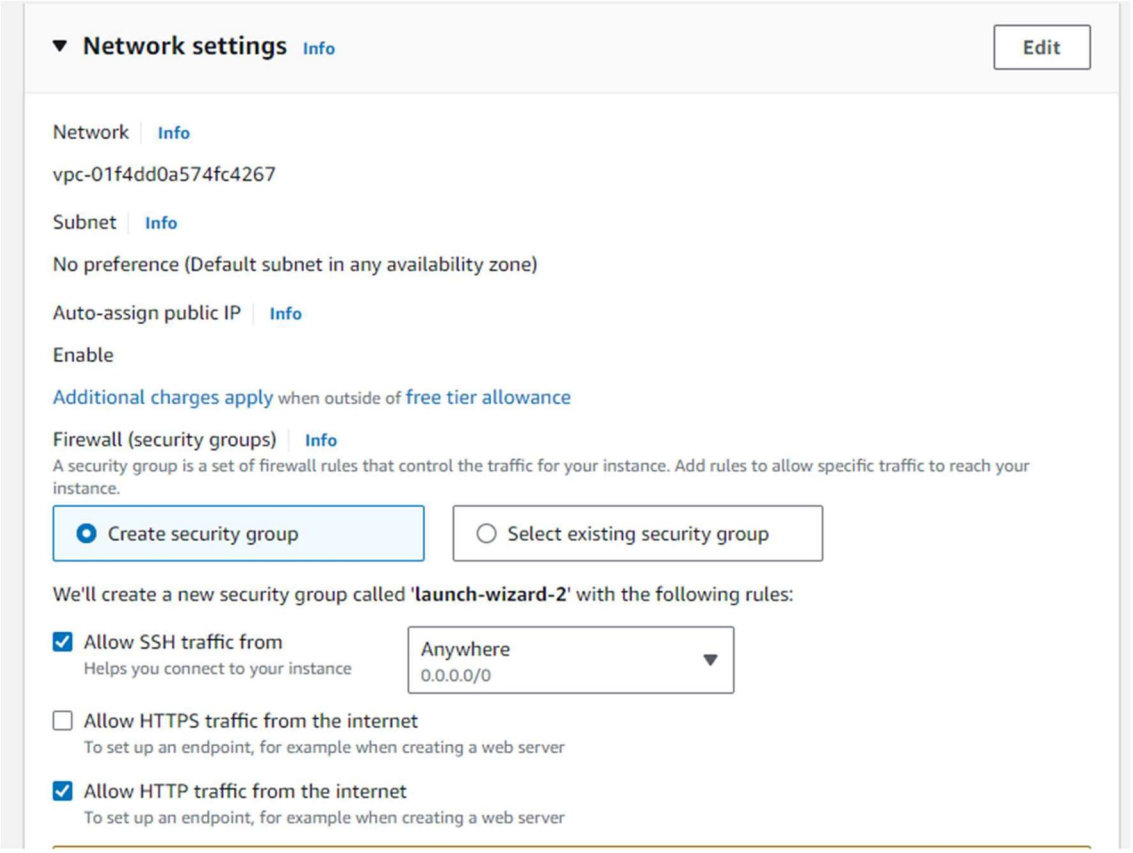
    - Choose an Amazon Machine Image (AMI): Select "Ubuntu Server 20.04 LTS (HVM), SSD Volume Type".



- Choose an Instance Type: Select `t2.micro` (eligible for the free tier).

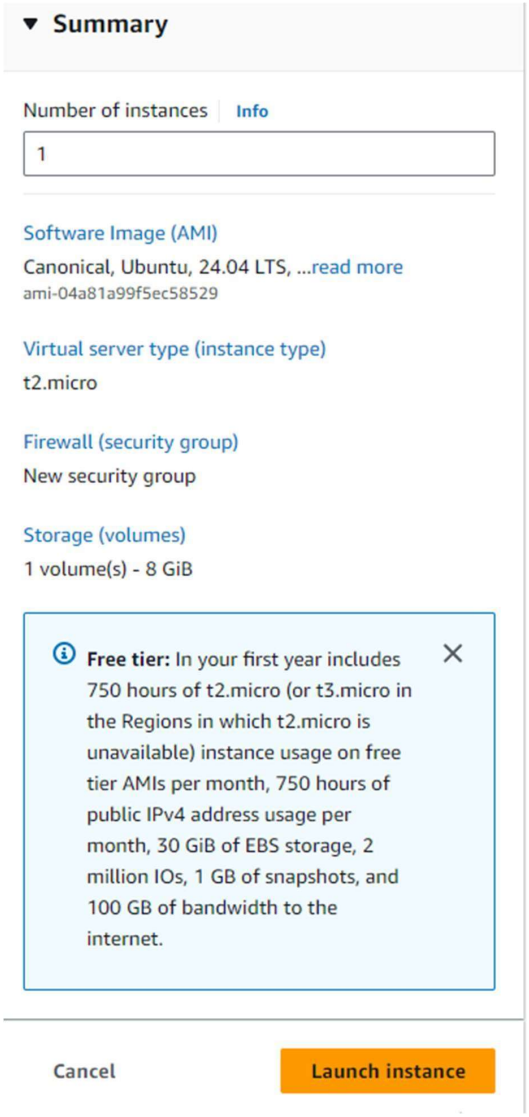


- Configure Instance:
  - Select an existing key pair or create a new one.
  - Network: Choose the default VPC.
  - Subnet: Choose a subnet in the US-East-1 (N. Virginia) region.
  - Enable Auto-assign Public IP.



- Add Storage: Keep the default settings.
- Add Tags: Add a tag to identify your instance (e.g., Key: Name, Value: Nginx).

4. **Review and Launch**:
    - Review your instance settings and click "Launch".



5. **Configure Security Group:**
    - Add a new security group with the following rules:
        - Type: HTTP, Protocol: TCP, Port Range: 80, Source: 0.0.0.0/0
        - Type: SSH, Protocol: TCP, Port Range: 22, Source: 0.0.0.0/0

Step 2: Connect to Your Instance

1. **Connect to the EC2 Instance**:
    - In the EC2 Dashboard, select your instance.
    - Click on "Connect" and follow the instructions to connect to your instanceusing SSH.

Step 3: Install Apache and PHP

1. **Update the package index**:

   sudo apt update -y

2. **Install Apache**:

   sudo apt install apache 2 -y

3. **Start Apache**:

   sudo systemctl start apache2

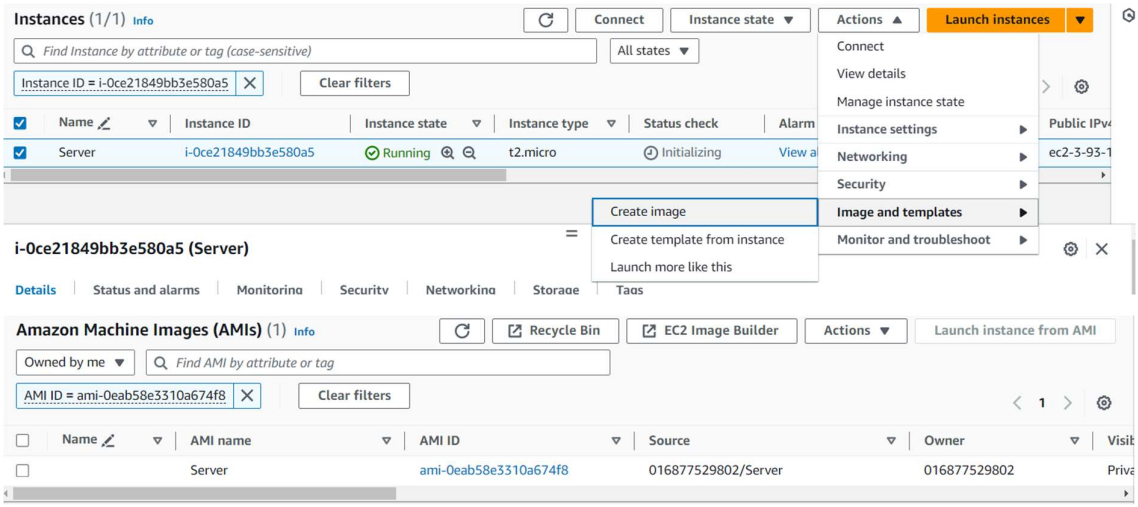
   sudo systemctl enable apache2

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-39-186:/home/ubuntu# ^[[200~sudo systemctl start apache2~^C
root@ip-172-31-39-186:/home/ubuntu# sudo systemctl start apache2
root@ip-172-31-39-186:/home/ubuntu# systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Fri 2024-07-26 03:06:02 UTC; 1min 25s ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2429 (apache2)
      Tasks: 55 (limit: 1130)
     Memory: 5.4M (peak: 5.6M)
        CPU: 41ms
     CGroup: /system.slice/apache2.service
             ├─2429 /usr/sbin/apache2 -k start
             ├─2432 /usr/sbin/apache2 -k start
             └─2433 /usr/sbin/apache2 -k start
```

4. **Restart Apache**:

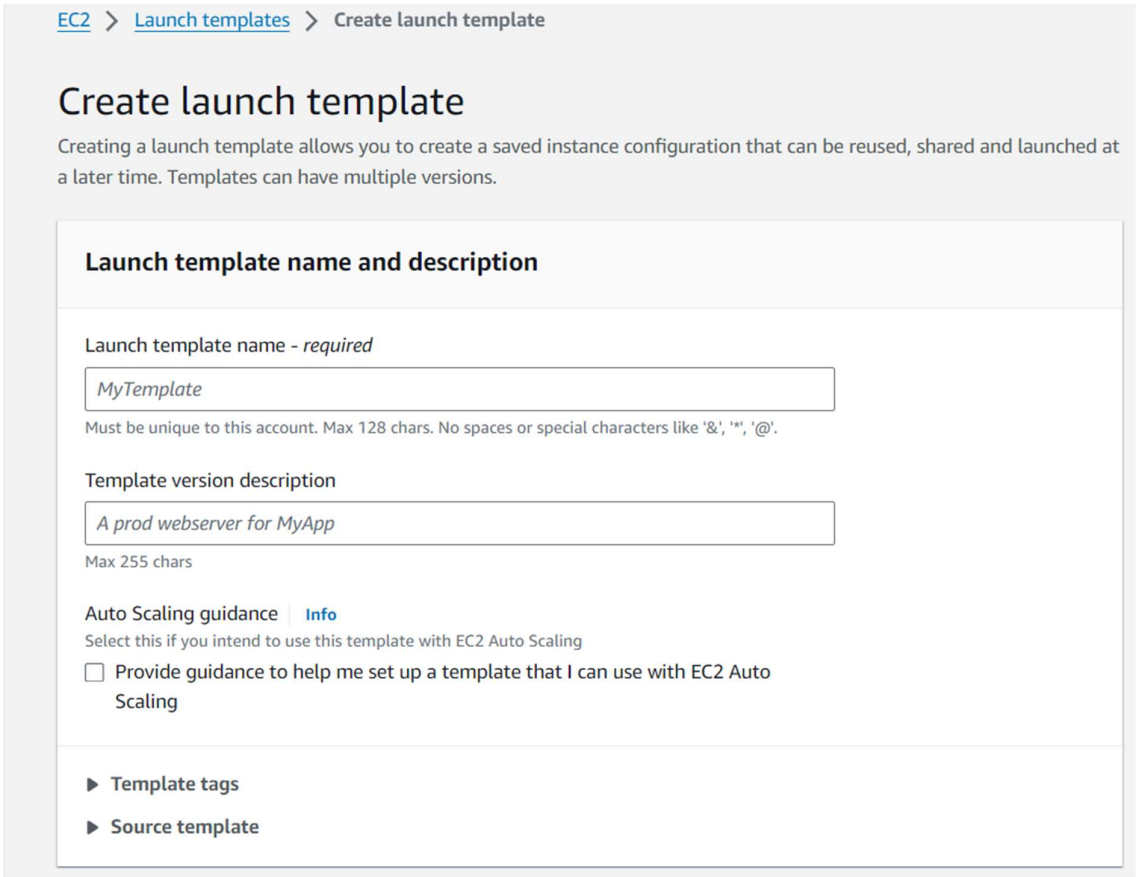
   sudo systemctl restart apache2

5. **Create an AMI:**
    - After your instance is up and running in US-East-1, go to the EC2 Dashboard, right-click on the instance, and select "Create Image".
    - Specify details and create the AMI.



Step 4: Create a Launch Template

1. Navigate to **Launch Templates** in the EC2 dashboard.
2. Click on **Create launch template**.



3. Fill in template details and instance configuration.
4. Ensure to use the same AMI, instance type, and security group as your manually launched instance.

Step 5: Create an Auto Scaling Group:

1. Navigate to **Auto Scaling Groups**.
2. Click on **Create Auto Scaling group**.
3. Choose your launch template.
4. Set the desired capacity to 2, minimum capacity to 1, and maximum capacity to 3.
5. Configure network and subnets.
6. Set up scaling policies (optional).