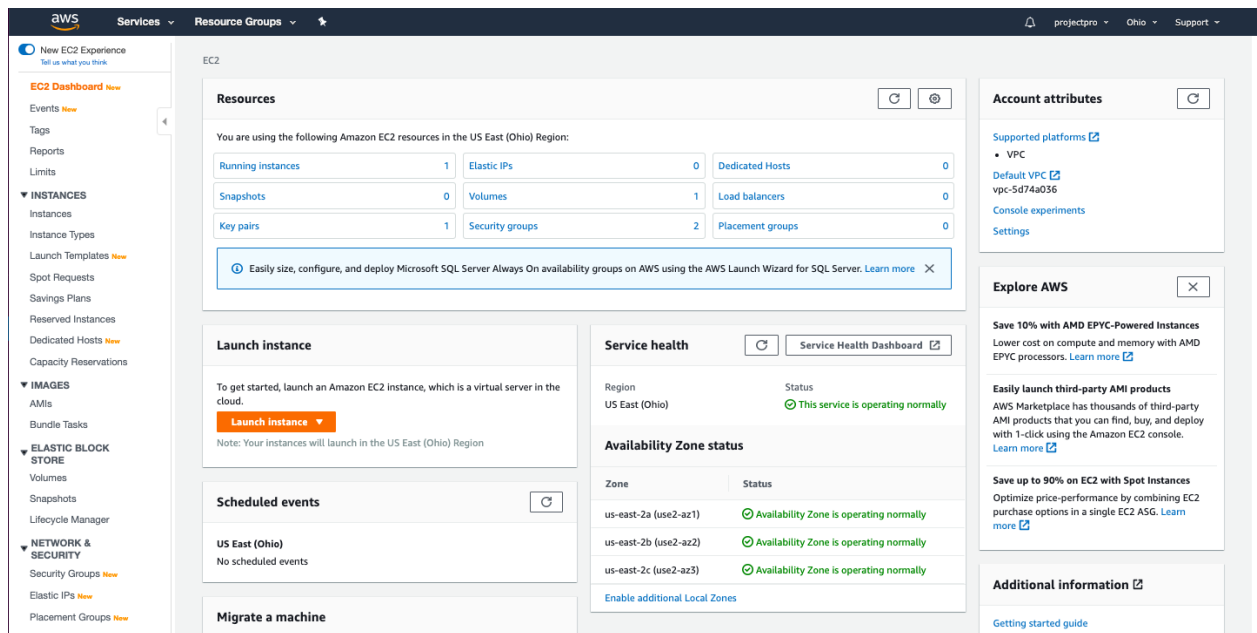
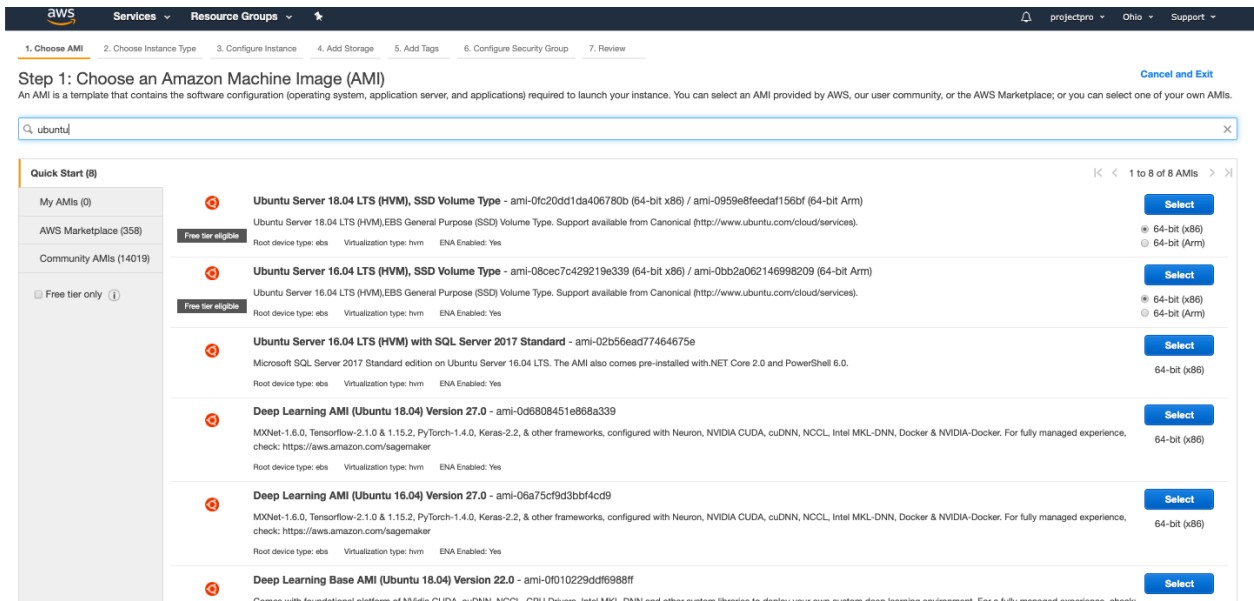


Steps to create Ec2 instance for the Project

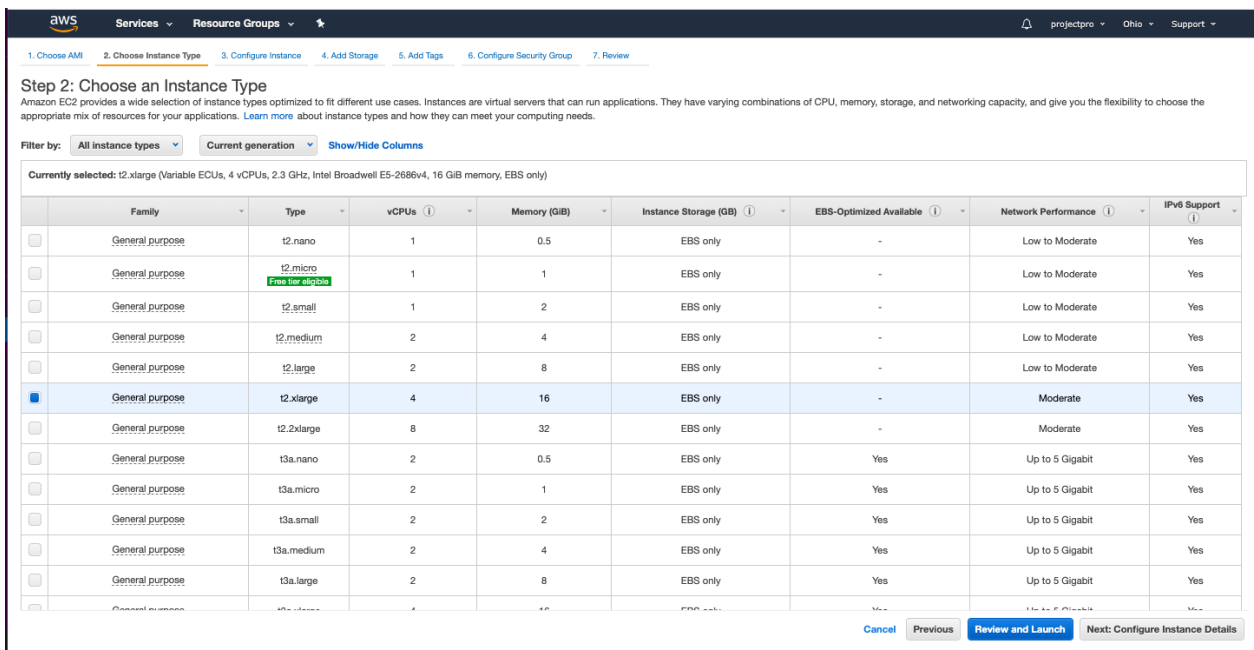
- Sign in a root user into AWS management console (AWS account)
- Click on services -> ec2
- Launch instance



- Choose machine image for your Ubuntu server as Ubuntu Server 16.04 LTS (HVM), SSD Volume Type



- Choose instance type as t2.xlarge and click on Configure instance details



- Do not change anything in the configure instance details - will be useful for multi node cluster configurations

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances: 1 [Launch into Auto Scaling Group](#)

Purchasing option: ☐ Request Spot instances

Network: vpc-5d74a036 (default) [Create new VPC](#)

Subnet: No preference (default subnet in any Availability Zone) [Create new subnet](#)

Auto-assign Public IP: Use subnet setting (Enable)

Placement group: ☐ Add instance to placement group

Capacity Reservation: Open [Create new Capacity Reservation](#)

IAM role: None [Create new IAM role](#)

Shutdown behavior: Stop

Stop - Hibernate behavior: ☐ Enable hibernation as an additional stop behavior

Enable termination protection: ☐ Protect against accidental termination

Monitoring: ☐ Enable CloudWatch detailed monitoring [Additional charges apply.](#)

Tenancy: Shared - Run a shared hardware instance [Additional charges will apply for dedicated tenancy.](#)

Elastic Inference: ☐ Add an Elastic Inference accelerator [Additional charges apply.](#)

T2/T3 Unlimited: ☐ Enable [Additional charges may apply](#)

File systems: [Add file system](#) [Create new file system](#)

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

-
- Click on Add storage
- Change size to 16GB and click on Add tags

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/sda1	snap-056d4f1de89dadf9	16	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

Free tier eligible customers can get up to 30 GiB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Tags](#)

-
- Do not do anything in the add tags. Please proceed to add security groups
- Add type as custom TCP as shown in the figure and click on review and launch instance

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: ☒ Create a new security group ☐ Select an existing security group

Security group name:

Description:

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop
Custom TCP	TCP	0	Anywhere 0.0.0.0/0	e.g. SSH for Admin Desktop

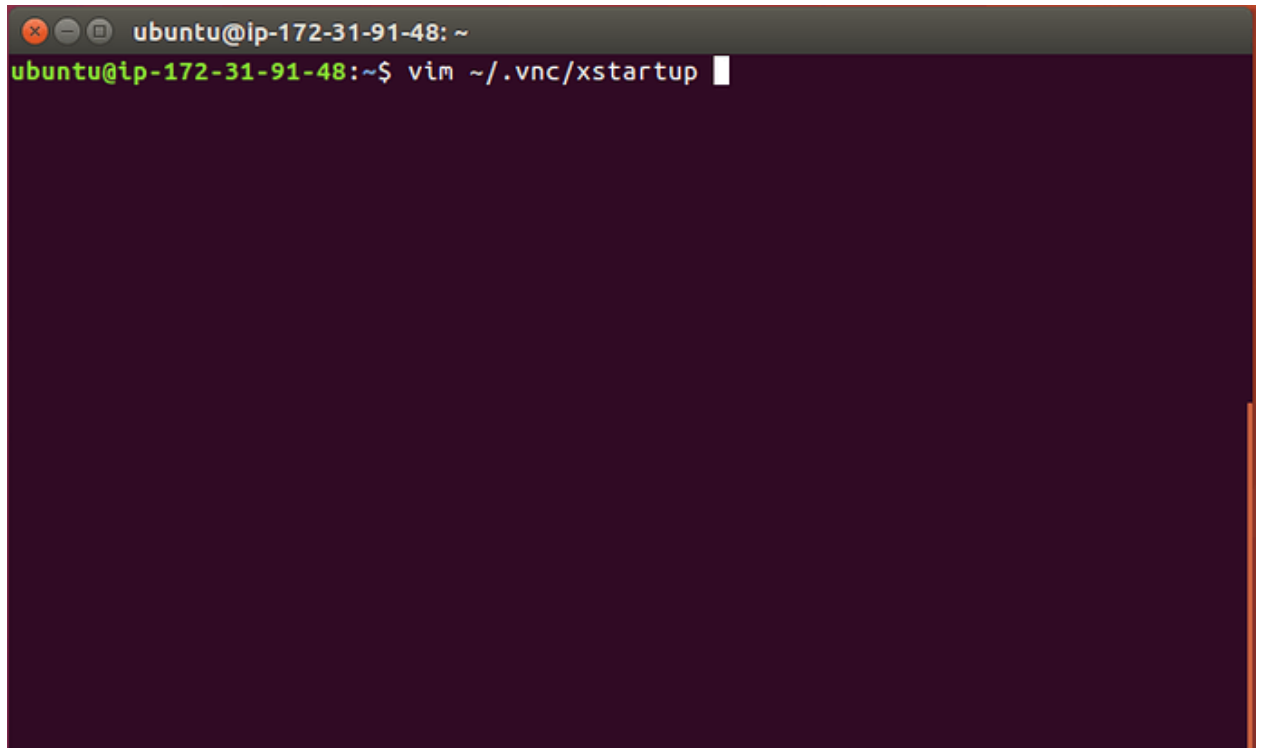
[Add Rule](#)

Warning
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

[Cancel](#) [Previous](#) [Review and Launch](#)

- Steps to create keypair and connect to your EC2 instance from Windows/Mac OS:
- While creating EC2 instance , you are suggested to create new KeyPair for logging into the EC2 instance .
- Download the .pem file .
- If Mac, directly use .pem file to login into instance
- `chmod 400 **/path/to/your/key/**EDAMAME.pem`
- `ssh -i **/path/to/your/key/**EDAMAME.pem ubuntu@ec2-**UNIQUE SET OF NUMBERS**.compute-1.amazonaws.com`
- If Windows, convert .pem to .ppk file and login into instance
- <https://docs.aws.amazon.com/quickstarts/latest/vmlaunch/step-2-connect-to-instance.html>
- Once we login into Ubuntu instance, we are supposed to set up Ubuntu remote desktop
- Steps to connect to Ubuntu remote desktop :
- Setting up TightVNC on AWS

- Let's install Ubuntu Desktop and TightVNC on your EC2 instance. After logging in to your EC2 instance using the terminal, enter the following commands to install the tools that will be required to run Ubuntu desktop :
- While installing VNC Server you'll be required to setup a password for the server. So remember this since it will be needed later to connect to our VNC server.
- `sudo apt update`
- `sudo apt install ubuntu-desktop`
- `sudo apt install tightvncserver`
- `sudo apt install gnome-panel gnome-settings-daemon metacity nautilus gnome-terminal`
- After completion, your machine is ready with GUI support but needs some configuration to be done.
- Configuring the VNC server
- In your terminal type the following command to launch VNC server to create an initial configuration file:
- `vncserver :1`
- Open the configuration file in vim:
- `vim ~/.vnc/xstartup`
 -

A terminal window with a dark background and light text. The title bar shows 'ubuntu@ip-172-31-91-48: ~'. The command prompt shows 'ubuntu@ip-172-31-91-48:~\$ vim ~/.vnc/xstartup' followed by a cursor. The terminal area is mostly empty, representing the vim editor in normal mode.

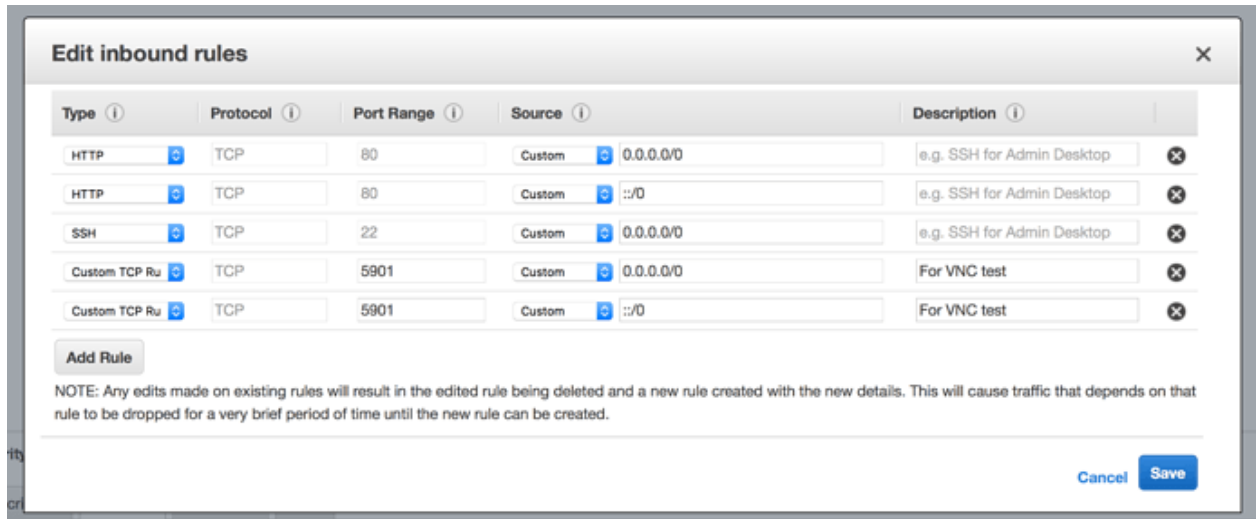
```
ubuntu@ip-172-31-91-48: ~
ubuntu@ip-172-31-91-48:~$ vim ~/.vnc/xstartup
```

Press the 'i' key on your keyboard to get into the insert mode which will allow you to enter text into the file. Edit the file to look like so :

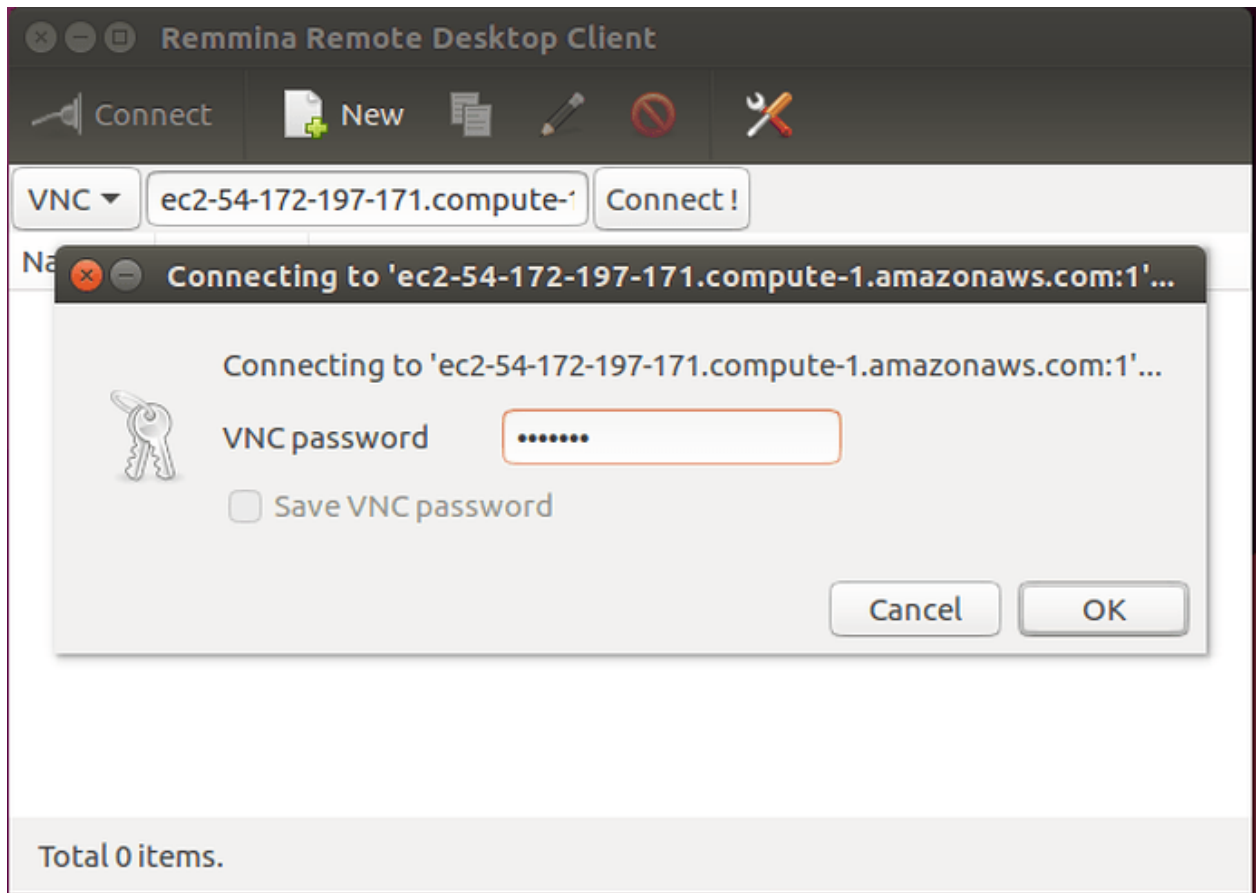
```
#!/bin/sh
    export XKL_XMODMAP_DISABLE=1
unset SESSION_MANAGER
unset DBUS_SESSION_BUS_ADDRESS
[ -x /etc/vnc/xstartup ] && exec /etc/vnc/xstartup
[ -r $HOME/.Xresources ] && xrdp $HOME/.Xresources
xsetroot -solid grey
    vncconfig -iconic &
gnome-panel &
gnome-settings-daemon &
metacity &
nautilus &
gnome-terminal &
```

- AWS Configuration
 - We need to make sure that the AWS instance has inbound rules setup to allow connection using VNC. So, head over to your AWS EC2

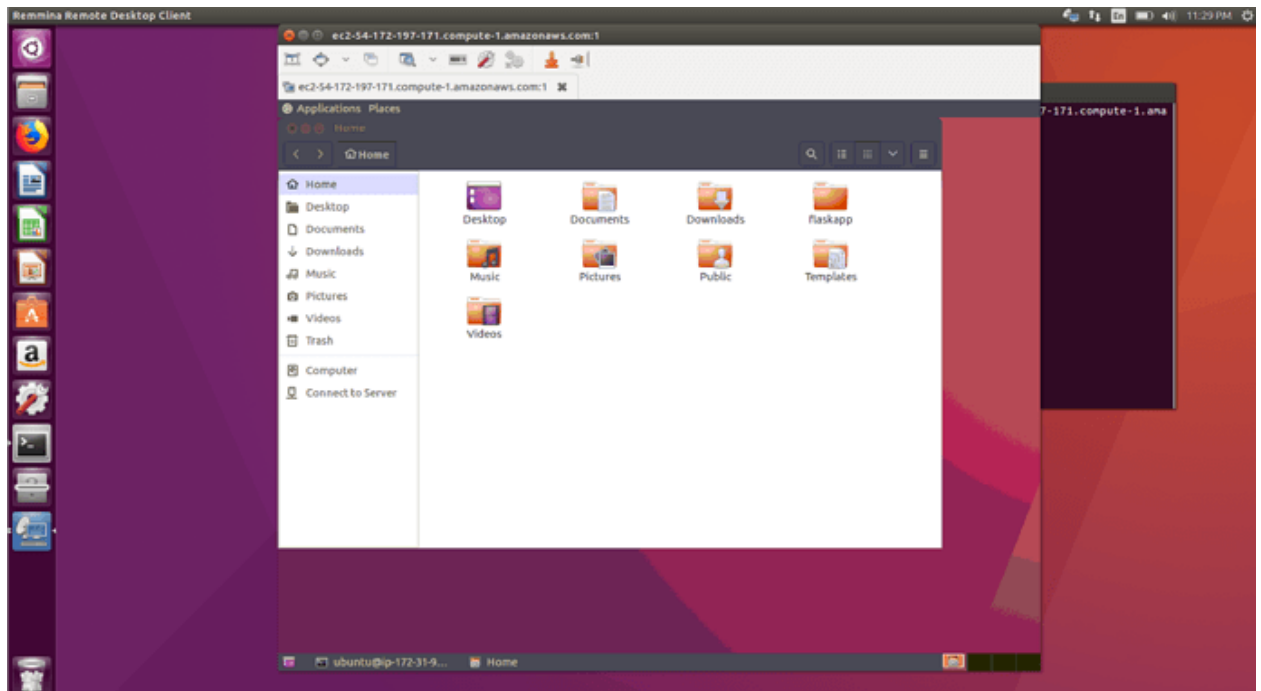
console and modify the inbound-rules. Add the entry : Custom TCP Rule | TCP | 5901 | Custom | 0.0.0.0/0 | VNC Connect



- Download VNC server for your OS from the link below:
 - <https://www.realvnc.com/en/connect/download/viewer/macos/>
- Connect to your remote by following the steps below:
 - Connecting to Ubuntu Desktop:
 - Once vnc server is installed on your ec2 instance, we need to start vnc server in the ec2 instance by giving command
\$vncserver :1
 - Then open the vncserver app which we install in local machine , then specify ec2 instance URL:1 and click on continue .
 - Then give the password which was setup during installation on the vnc server . Then click on continue .
 - The Remmina desktop client had to be installed on the local machine or tightvnc app also can be used locally to login into ec2 instance UI
 - Once we have the software installed on the local system , we open it and given Ec2 url , port number as 1 and connect to Ubuntu GUI .
- ec2-54-172-197-171.compute-1.amazonaws.com:1



- 3. Enter the password you provided during the installation of the VNC Server. 4. Connect!
- Congratulations, you've successfully configured your EC2 instance to run Ubuntu Desktop GUI Support.



- The above has to be done in VNC server .
- <https://ubuntu.com/tutorials/tutorial-ubuntu-desktop-aws#1-overview> from Ubuntu official website might help.