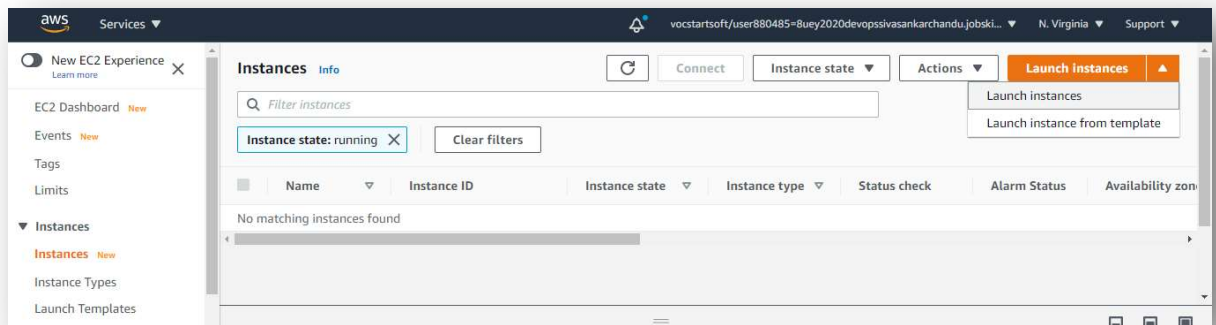
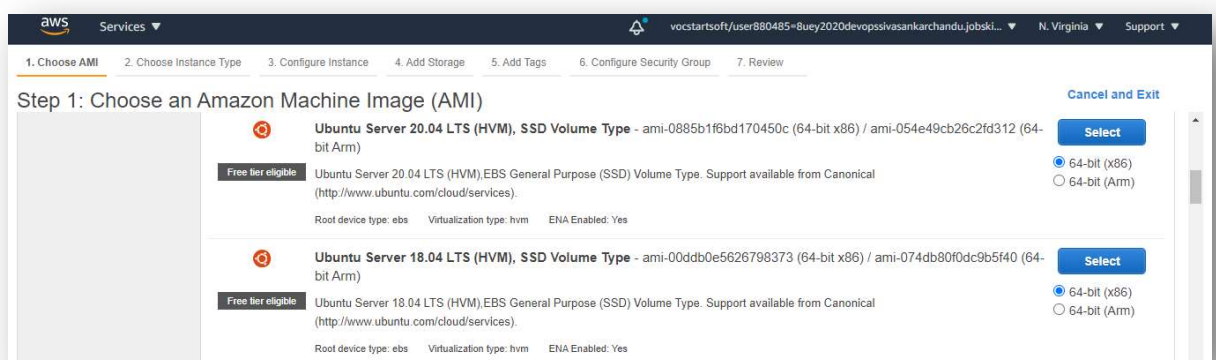


EC2 Instance (Ubuntu) – Git Bash

- Open AWS Console and Search for EC2 Service then click on EC2 service. We directed to EC2 page, click on running instances then we find below page. Now click on Launch Instances.



- Now select Amazon Machine Image (AMI), An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs. Select Ubuntu Server then forwarded to Next step.



- Now, choose instance type. Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. After choosing Instance type based on requirement click Next to Configure Instance Details.

Step 2: Choose an Instance Type
Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: All instance families Current generation Show/Hide Columns

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, ~, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	t2	t2.micro <small>Free tier eligible</small>	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Instance Details](#)

- Now, based on requirement Select the VPC, Subnet, Public IP, shutdown behaviour, stop hibernate, Termination protection, Monitoring and Use bash code in User data to automatically install and run application. Click next to add storage.

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

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 [Launch into Auto Scaling Group](#)

Purchasing option ☐ Request Spot Instances

Network [Create new VPC](#)

Subnet [Create new subnet](#)

Auto-assign Public IP

Placement group ☐ Add instance to placement group

Capacity Reservation

Domain join directory [Create new directory](#)

IAM role [Create new IAM role](#)

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

- Now add storage to EC2 Instance. I am not adding any volume so I am going to next step.

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

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/xvda	snap-0299d083f0ce6cd12	8	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 GB 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](#)

- Now assign some Tags to EC2 Instance. Key is Name and Value is First Ubuntu Instance then Click Next to Configure Security Group.

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. A copy of a tag can be applied to volumes, instances or both. Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key (128 characters maximum)	Value (256 characters maximum)	Instances	Volumes
Name	First Ubuntu Instance	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

[Add another tag](#) (Up to 50 tags maximum)

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Security Group](#)

- Now, we can select either new Security Group or existing Security Group. Security Group have set of firewall rules that controls the traffic of EC2 Instance. Select Security Group and Add new rule to it. Click next review the selected options and launch the EC2 Instance.

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

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
HTTP	TCP	80	Custom 0.0.0.0/0, :::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](#)

- Review the EC2 Instance.

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

Improve your instances' security. Your security group, launch-wizard-5, is open to the world.

Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only. You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. [Edit security groups](#)

AMI Details [Edit AMI](#)

Ubuntu Server 20.04 LTS (HVM), SSD Volume Type - ami-0885b1f6bd170450c

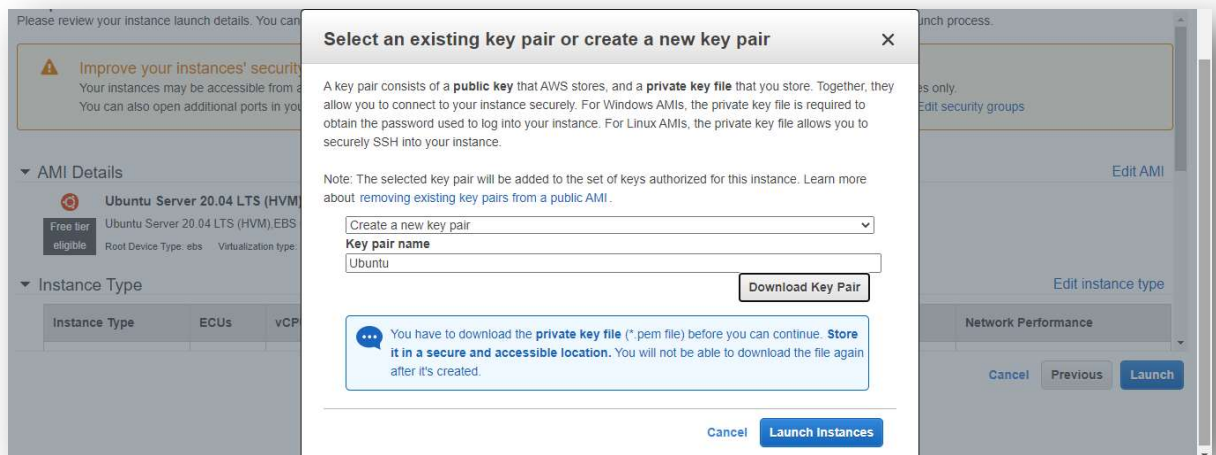
Free tier eligible Ubuntu Server 20.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).
Root Device Type: ebs Virtualization type: hvm

Instance Type [Edit instance type](#)

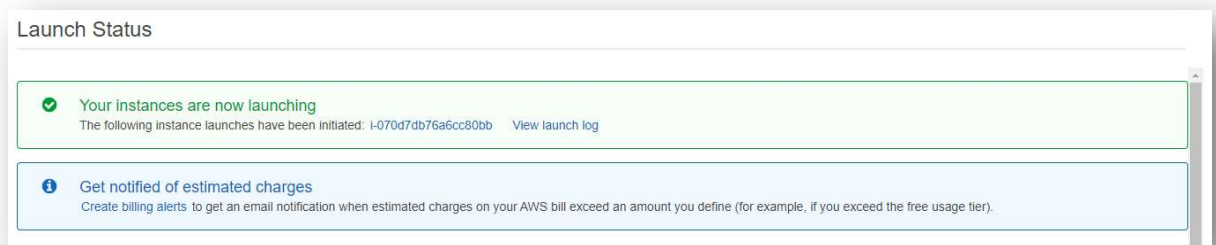
Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	-	1	1	EBS only	-	Low to Moderate

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

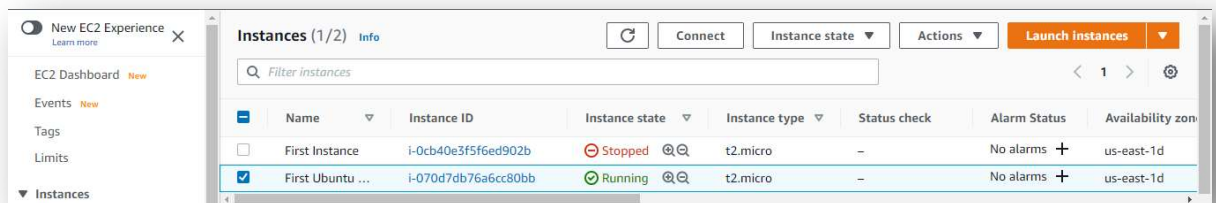
- After click on launch we select the Private key pair. Create new key pair and download it otherwise choose existing if you have any keypair. I am selecting new key pair and key pair downloaded. Now I am click on Launch Instances button to launch my Ubuntu Instance.



- Launch status

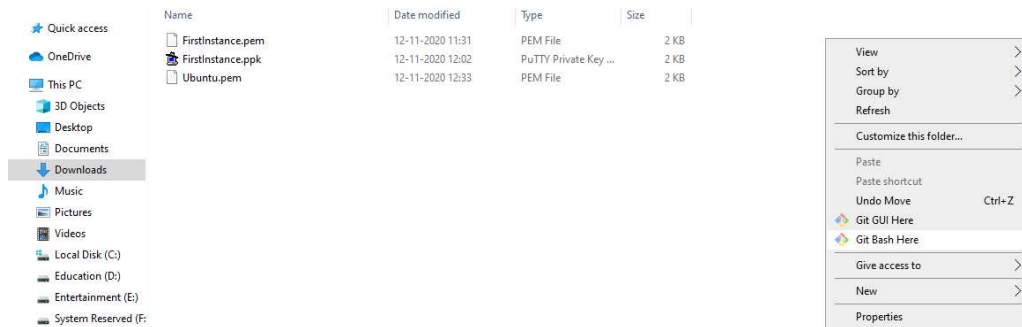


- My Instance state Running. Now connect instance through the Git Bash.



Connecting Instance through Git Bash...

- First Install Git Bash. After installing Git Bash open folder where the pem stored. Right Click on white space and then click on Git Bash here.



- Now paste the code (`ssh -i Ubuntu.pem ubuntu@PublicIP_Address`). Copy the Public IP address from EC2 Instance and Click Enter.

```
ubuntu@ip-172-31-20-214: ~
$ ssh -i Ubuntu.pem ubuntu@34.234.92.161
Load publickey 'Ubuntu.pem': invalid format
The authenticity of host '34.234.92.161 (34.234.92.161)' can't be established.
ECDSA key fingerprint is SHA256:12hGCoqVjv9gakkE4WMOAts5IguKEj1f0sDjm/1LnDU.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '34.234.92.161' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-1029-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Thu Nov 12 07:09:32 UTC 2020

System load: 0.05          Processes: 101
Usage of /: 16.8% of 7.69GB Users logged in: 0
Memory usage: 20%          IPv4 address for eth0: 172.31.20.214
Swap usage: 0%

1 update can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-20-214:~$
```

EC2 Instance Connected...

Now run “**sudo apt-get update**” to apply all the latest updates to EC2 Instance