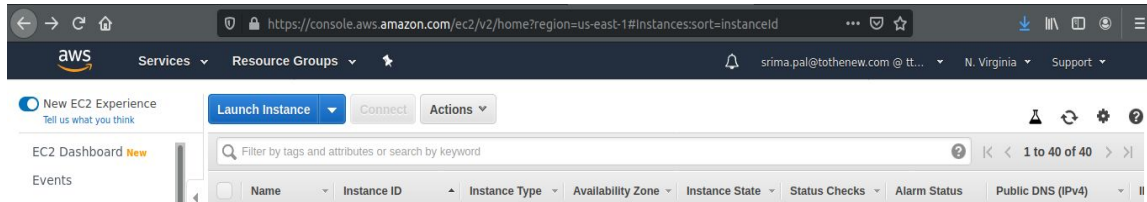


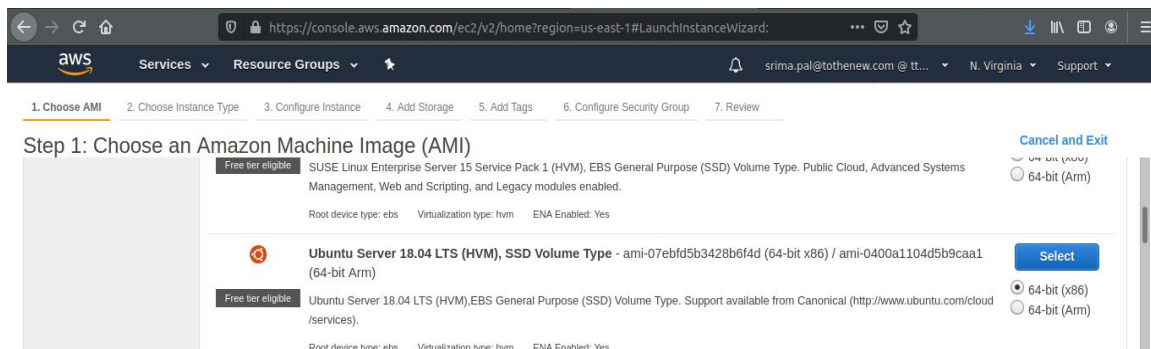
1. Create an EC2 instance (Ubuntu 18.04, T3 nano).(instance A)

Ans) (For free tier eligibility, T2 Micro is used)

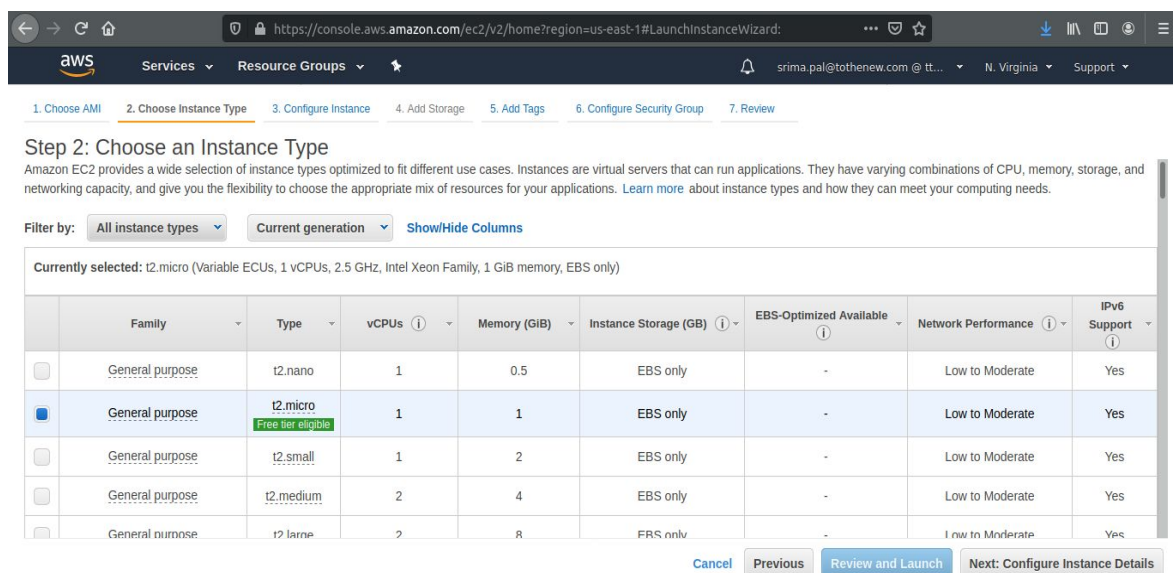
Step 1. Click on Launch Instance.



Step 2. Select the base OS



Step 3. Select the Instance Family and click on configure instance details



Step 4. Chosen the default configurations . Click on add storage.

The screenshot shows the 'Step 3: Configure Instance Details' page in the AWS Management Console. The page is part of the 'Launch Instance Wizard' and is currently on step 3 of 7. The configuration details are as follows:

- Number of instances:** 1
- Purchasing option:** ☐ Request Spot instances
- Network:** vpc-d38d68b7 | default (default) | Create new VPC
- Subnet:** subnet-06680a5b651f104dc | testpusubnet | us-east-1 | 65493 IP Addresses available | Create new subnet
- Auto-assign Public IP:** Use subnet setting (Disable)
- Placement group:** ☐ Add instance to placement group
- Capacity Reservation:** Open | Create new Capacity Reservation
- IAM role:** None | Create new IAM role

At the bottom right, there are four buttons: 'Cancel', 'Previous', 'Review and Launch', and 'Next: Add Storage'.

Step 5. Configure Storage settings(Defaults are selected)

The screenshot shows the 'Step 4: Add Storage' page in the AWS Management Console. The page is part of the 'Launch Instance Wizard' and is currently on step 4 of 7. The storage configuration is as follows:

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/sda1	snap-0e078112eedec9db	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypt

Below the table, there is an 'Add New Volume' button.

A note at the bottom states: "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."

At the bottom right, there are four buttons: 'Cancel', 'Previous', 'Review and Launch', and 'Next: Add Tags'.

Step 6. Add tags

← → ↺ ⌂ https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#LaunchInstanceWizard: srima.pal@tothenew.com @ tt... N. Virginia Support

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 ⓘ
Owner-name	Srima	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Instance	A	<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](#)

Step 7. Configure security group

← → ↺ ⌂ https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#LaunchInstanceWizard: srima.pal@tothenew.com @ tt... N. Virginia Support

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

allow Internet traffic to reach your instance, and 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

[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](#)

Step 8. Review the details

← → ↺ ⌂ https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#LaunchInstanceWizard: srima.pal@tothenew.com @ tt... N. Virginia Support

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-127(Srima), is open to the world.

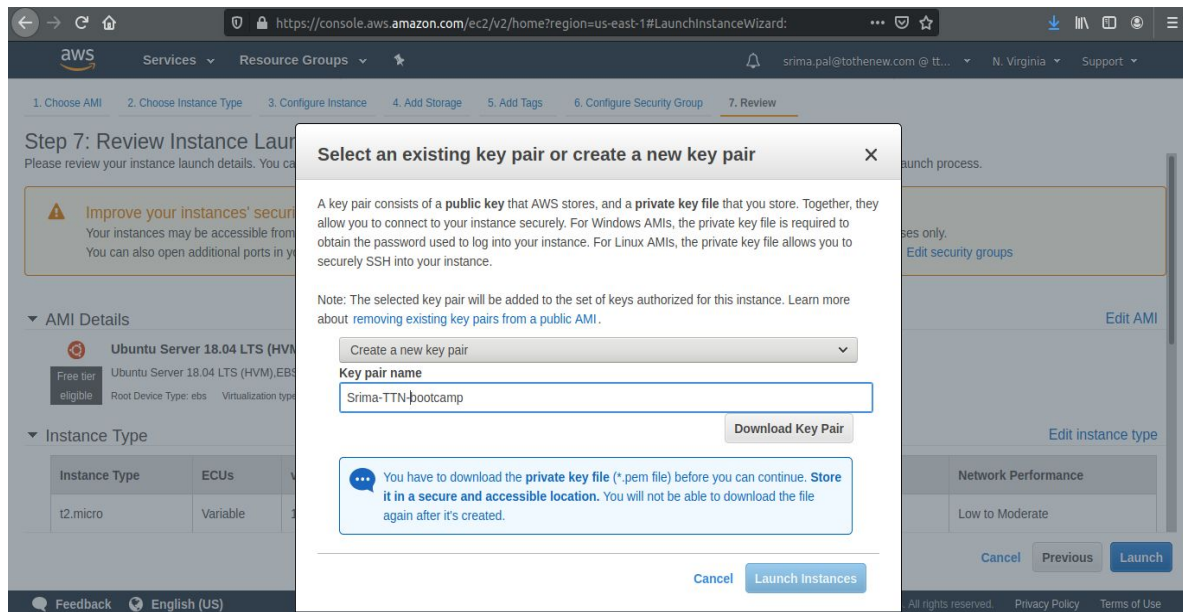
Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only. You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web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

AMI Details

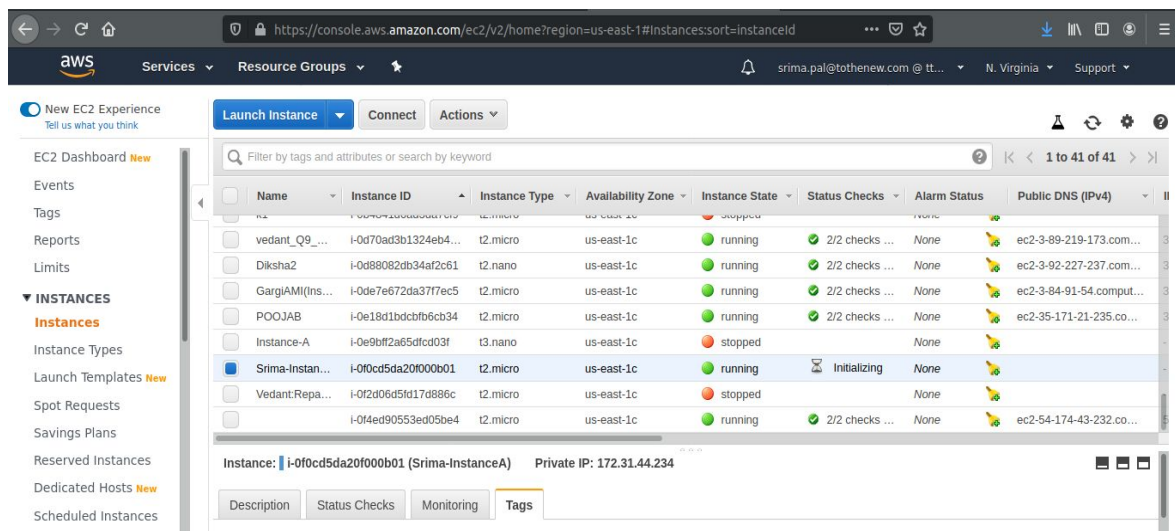
Free tier eligible **Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-07ebfd5b3428b6f4d** [Edit AMI](#)

Ubuntu Server 18.04 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

Step 9. Create a new key-pair and download the .pem file(private key)

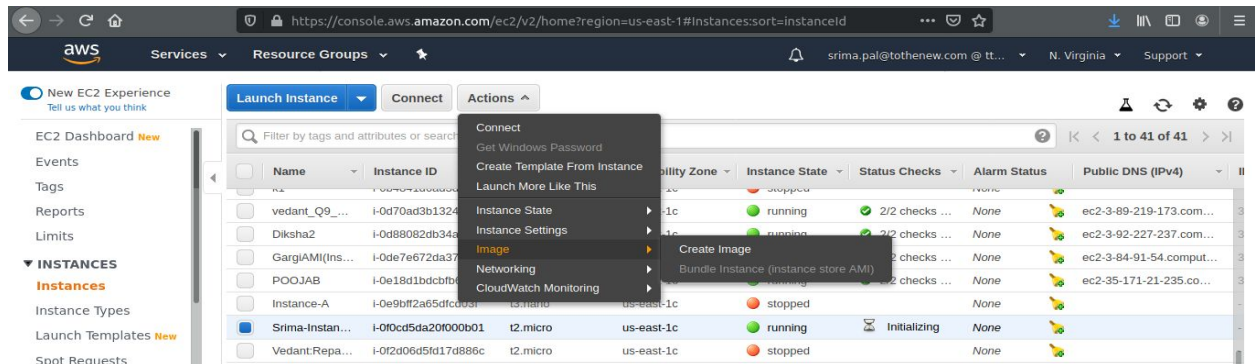


Step 10. Launch the instance.

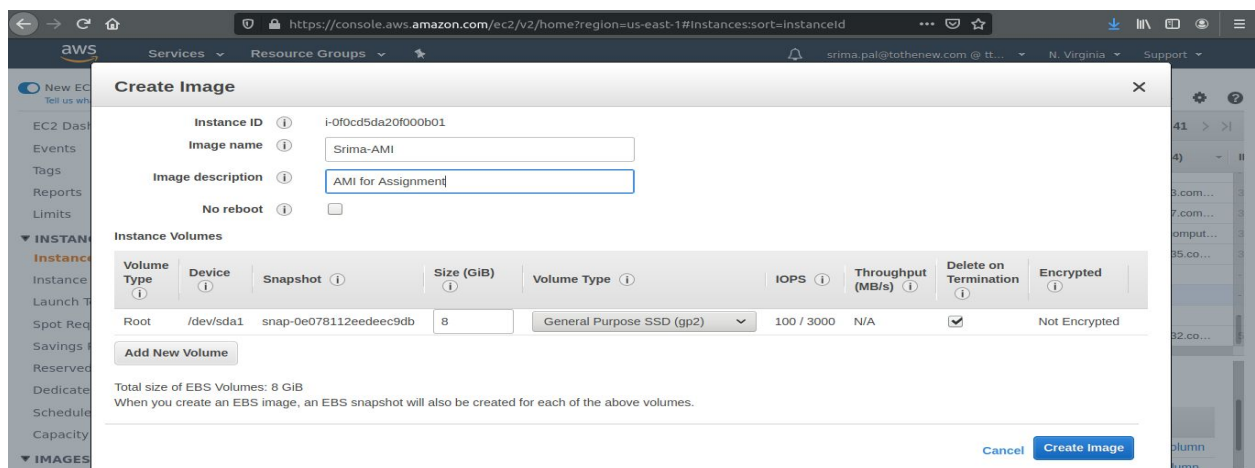


2. Create AMI of above instance and launch it. (instance B)

Step 1. Select your instance, Go to Action, and Select Image and then create Image.



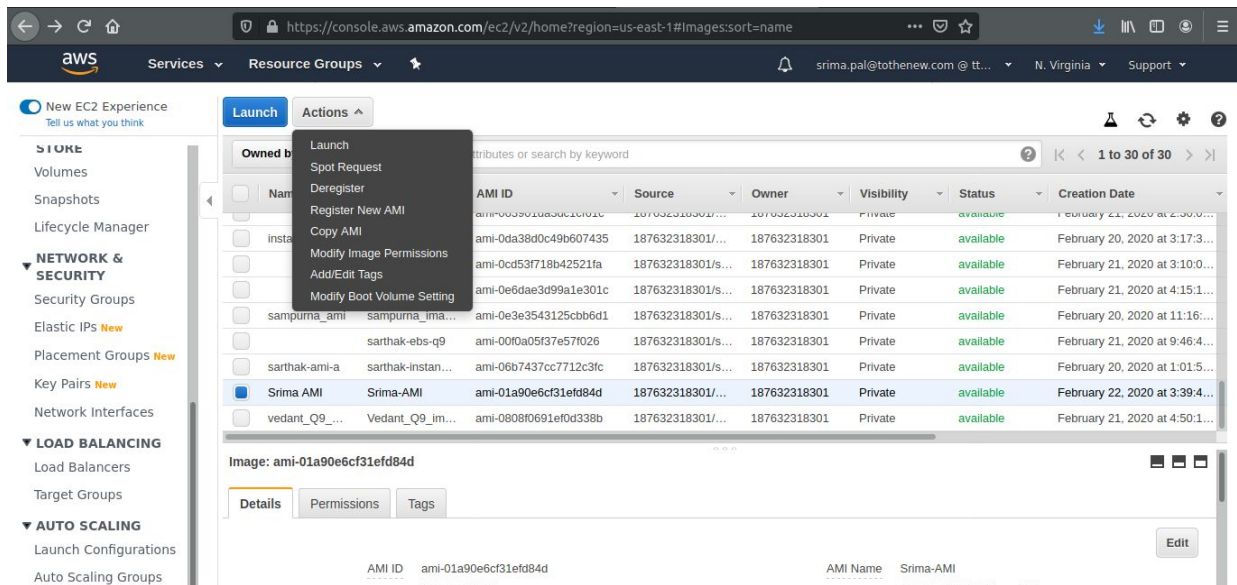
Step 2. Give name and Description for AMI, Select the EBS settings.



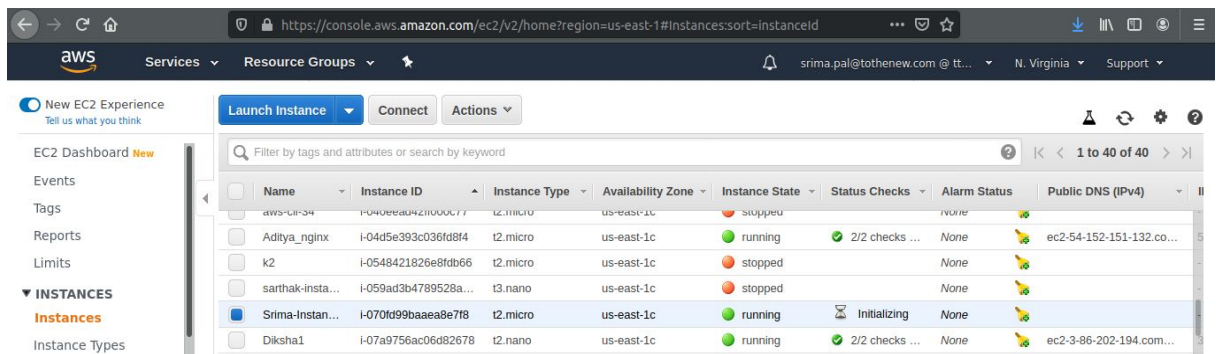
▼ ELASTIC BLOCK STORE		sampurna_ebs...	ami-0e6dae3d99a1e301c	187632318301/s...	187632318301	Private	available	February 21, 2020 at 4:15:1...
▼ VOLUMES		sampurna_ami	sampurna_ima...	ami-0e3e3543125cbb6d1	187632318301/s...	187632318301	Private	February 20, 2020 at 11:16:...
▼ SNAPSHOTS		sarthak_ebs-q9	ami-00f0a05f37e57f026	187632318301/s...	187632318301	Private	available	February 21, 2020 at 9:46:4...
▼ LIFECYCLE MANAGER		sarthak-ami-a	sarthak-istan...	ami-06b7437cc7712c3fc	187632318301/s...	187632318301	Private	February 20, 2020 at 1:01:5...
		Srima AMI	Srima-AMI	ami-01a90e6c31efd84d	187632318301/...	187632318301	Private	February 22, 2020 at 3:39:4...

Step 3. Launching Instance using that AMI

Go to AMI's, Select the AMI, Go to Actions and click on launch



Step 4. Configure the instance details as done above and click on launch.



3. Attach EBS (8 GB) on that running instance.

In the left side, In Elastic store Block, Go to volumes.

Step 1. Create Volume

The screenshot shows the 'Create Volume' page in the AWS Management Console. The 'Volume Type' is set to 'General Purpose SSD (gp2)'. The 'Size (GiB)' is set to 8. The 'IOPS' are 100 / 3000. The 'Availability Zone' is set to 'us-east-1a'. The 'Throughput (MB/s)' is 'Not applicable'. The 'Snapshot ID' is 'Select a snapshot'. The 'Encryption' checkbox is unchecked.

Volume Type: General Purpose SSD (gp2)

Size (GiB): 8 (Min: 1 GiB, Max: 16384 GiB)

IOPS: 100 / 3000 (Baseline of 3 IOPS per GiB with a minimum of 100 IOPS, burstable to 3000 IOPS)

Availability Zone*: us-east-1a

Throughput (MB/s): Not applicable

Snapshot ID: Select a snapshot

Encryption: ☐ Encrypt this volume

Step 2. Select the Volume, go to Actions, Click on Attach Volume

The screenshot shows the AWS Management Console with the 'Volumes' page selected. A dropdown menu is open for the 'Srima EBS3' volume, showing the 'Attach Volume' option. Below the menu, a table lists the volumes. At the bottom, the 'Instances' page is visible, showing the 'Srima-Instan...' instance with the 'Root device type' set to 'ebs'.

Actions:

- Modify Volume
- Create Snapshot
- Delete Volume
- Attach Volume
- Detach Volume
- Force Detach Volume
- Change Auto-Enable IO Setting
- Add/Edit Tags

Name	Volume Type	IOPS	Snapshot	Created	Availability Zone	State
Srima EBS3	gp2	100		February 22, 2020 ...	us-east-1a	available
Srima Root E	gp2	100	snap-06f75d9d...	February 22, 2020 ...	us-east-1c	in-use
Srima Root A	gp2	100	snap-0e07811...	February 22, 2020 ...	us-east-1c	in-use
	gp2	100	snap-0cf6af33...	February 21, 2020 ...	us-east-1c	in-use

Instance Name	Instance ID	Instance Type	Availability Zone	State	Checks	Termination Protection	Lifecycle	Monitoring
Srima-Instan...	i-0f0cd5da20f000b01	t2.micro	us-east-1c	running	2/2 checks ...	None	normal	basic
Vedant.Repa...	i-0f2d06d5fd17d886c	t2.micro	us-east-1c	stopped		None		
	i-0f4ed90553ed05be4	t2.micro	us-east-1c	running	2/2 checks ...	None		

Step 3. No mount point right now

```
ubuntu@ip-172-31-140-22: ~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0       7:0      0 89.1M  1 loop /snap/core/8268
loop1       7:1      0  18M   1 loop /snap/amazon-ssm-agent/1480
xvda        202:0    0   8G   0 disk
└─xvda1     202:1    0   8G   0 part /
xvdf        202:80   0   8G   0 disk
ubuntu@ip-172-31-140-22:~$
```

Step 4. Creating file system ext4

```
ubuntu@ip-172-31-140-22:~$ sudo mkfs.ext4 /dev/xvdf
mkfs2fs 1.44.1 (24-Mar-2018)
Creating filesystem with 2097152 4k blocks and 524288 inodes
Filesystem UUID: fb5c90d2-196c-4988-b33e-df46510db15f
Superblock backups stored on blocks:
32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632
Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

Step 5. Temporarily mounting the ebs on /home/ubuntu/srima-ebs

```
ubuntu@ip-172-31-140-22:~$ sudo mkdir srima-ebs
ubuntu@ip-172-31-140-22:~$ sudo mount /dev/xvdf /home/ubuntu/srima-ebs/
```

Step 6. Permanent mount (entry in /etc/fstab)

```
# LABEL=clouding-rootfs / ext4 defaults,discard 0 0
/dev/xvdf /home/ubuntu/srima-ebs ext4 defaults,discard 0 0
```

Step 7. Now /dev/xvdf is mounted on /home/ubuntu/srima-ebs

```
ubuntu@ip-172-31-140-22:~$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0       7:0      0 89.1M  1 loop /snap/core/8268
loop1       7:1      0  18M   1 loop /snap/amazon-ssm-agent/1480
xvda        202:0    0   8G   0 disk
└─xvda1     202:1    0   8G   0 part /
xvdf        202:80   0   8G   0 disk /home/ubuntu/srima-ebs
```

4. Stop, Start, Restart the instance (EBS must be auto-attached).

Step 1. Instance stopped, public IP gone.











Description	Status Checks	Monitoring	Tags
Instance ID	i-021da27c154a91f81		
Instance state	stopped		
Instance type	t2.micro		
Finding	You may not have permission to access AWS Compute Optimizer.		
Public DNS (IPv4)	-		
IPv4 Public IP	-		
IPv6 IPs	-		
Elastic IPs	-		

Step 2. Instance started, IP changed.

Launch Templates New	Srma-Instan...	i-021da27c154a91f81	t2.micro	us-east-1c	running	Initializing	None	ec2-52-71-103-58.com...
Spot Requests	test for fdisk	i-024b1b5ef8cd8aa66	t2.nano	us-east-1c	stopped		None	
Savings Plans	Ekanshu-Te...	i-0263a04fe84e4edd3	t2.nano	us-east-1c	stopped		None	




Description	Status Checks	Monitoring	Tags
Instance ID	i-021da27c154a91f81		
Public DNS (IPv4)	ec2-52-71-103-58.compute-1.amazonaws.com		
Instance state	running		
IPv4 Public IP	52.71.103.58		

Step 3. On restarting instance, Ip remains same

	Srma-Instan...	i-021da27c154a91f81	t2.micro	us-east-1c		running		Initializing	None		ec2-52-71-103-58.com...
	test for fdisk	i-024b1b5ef8cd8aa66	t2.nano	us-east-1c		stopped			None		
	Ekanshu-Te...	i-0263a04fe84e4edd3	t2.nano	us-east-1c		stopped			None		

Instance: **i-021da27c154a91f81 (Srma-Instance-A)**

Public DNS: ec2-52-71-103-58.compute-1.amazonaws.com



Description	Status Checks	Monitoring	Tags
Instance ID	i-021da27c154a91f81		
Public DNS (IPv4)	ec2-52-71-103-58.compute-1.amazonaws.com		
Instance state	running		
IPv4 Public IP	52.71.103.58		

5. Make some mistake in fstab file, stop and start the instance, then troubleshoot it.

Step 1. Made mistake in fstab file

```
ubuntu@ip-172-31-140-22: ~  
ubuntu@ip-172-31-140-22: ~ 151x39  
LABEL=clouding-rootfs / ext4 defaults,discard 0 0  
/dev/xv /home/ubuntu/srma-ebs ext4 defaults,discard 0 0  
ubuntu@ip-172-31-140-22:~$ sudo mount -a  
mount: /home/ubuntu/srma-ebs: special device /dev/xv does not exist.  
ubuntu@ip-172-31-140-22:~$
```

Step 2. Trying to re ssh in the instance

```
srma@srma:~/Downloads$ sudo ssh -i Srma-TTN-bootcamp.pem ubuntu@52.71.103.58  
ssh: connect to host 52.71.103.58 port 22: Connection refused  
srma@srma:~/Downloads$
```

Step 3. Detach Root volume from instance A and attach it as a secondary volume of instance B

The screenshot shows the AWS Management Console interface. On the left, the navigation menu includes 'New EC2 Experience', 'Reserved Instances', 'Dedicated Hosts', 'Scheduled Instances', 'Capacity Reservations', 'IMAGES', and 'AMIs'. The main content area displays a table of EBS volumes. A dropdown menu is open under the 'Actions' column for the volume 'Srima-Root-A', showing options like 'Modify Volume', 'Create Snapshot', 'Delete Volume', 'Attach Volume', 'Detach Volume', 'Force Detach Volume', 'Change Auto-Enable IO Setting', and 'Add/Edit Tags'. The 'Attach Volume' dialog box is open, showing the volume 'vol-0a71af33fa7dd606f (Srima-Root-A) in us-east-1c' being attached to instance 'i-021da27c154a91f81' in us-east-1c. The device path is set to '/dev/sdg'. A note at the bottom states: 'Note: Newer Linux kernels may rename your devices to /dev/xvdf through /dev/xvdp internally, even when the device name entered here (and shown in the details) is /dev/sdf through /dev/sdp.'

Step 4. After mounting instance A's root volume on instance B

```

disk identifier: 0xb32e823c
Device Boot Start      End  Sectors  Size Id Type
/dev/xvda1 *SIGNM 2048 16777182 16775135    8G 83 Linux

File Edit View Tools Help Last edit was made yesterday at 3:11 PM by Chirag

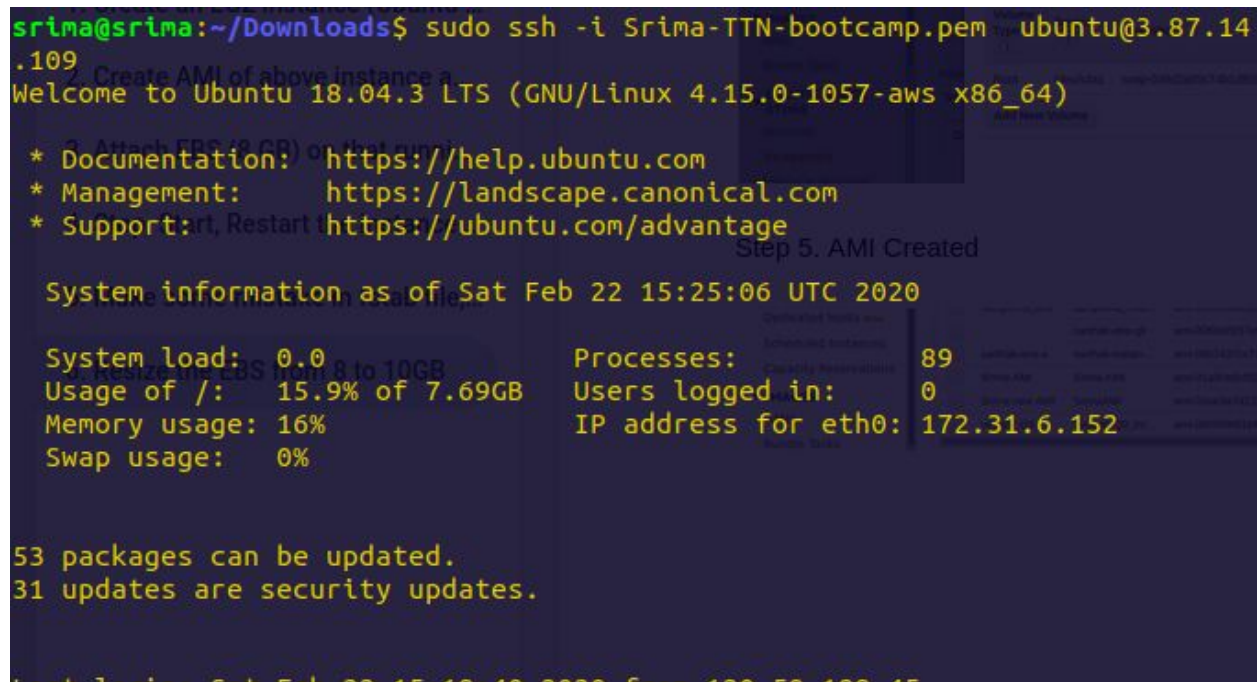
disk /dev/xvdf: 8 GiB, 8589934592 bytes, 1677216 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xb32e823c

Device Boot Start      End  Sectors  Size Id Type
/dev/xvdf1 *          2048 16777182 16775135    8G 83 Linux
ubuntu@ip-172-31-117-45:~$ sudo mount /dev/xvdf1 /home/ubuntu/srima-ebs-instanceA/
ubuntu@ip-172-31-117-45:~$ cd srima-ebs-instanceA/
ubuntu@ip-172-31-117-45:~/srima-ebs-instanceA$ ls
bin  home  lib64  opt  sbin  tmp  vmlinuz.old
boot  initrd.img  lost+found  proc  snap  usr
dev  initrd.img.old  media  root  srv  var
etc  lib  mnt  run  sys  vmlinuz
  
```

Step 5 . Correcting the changes in fstab

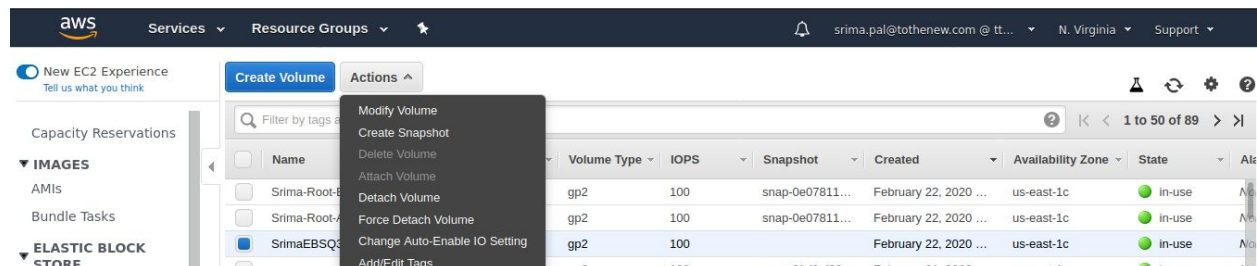


Detach the volume from instance B and Attach the volume to Instance A and you are able to login to the instance again

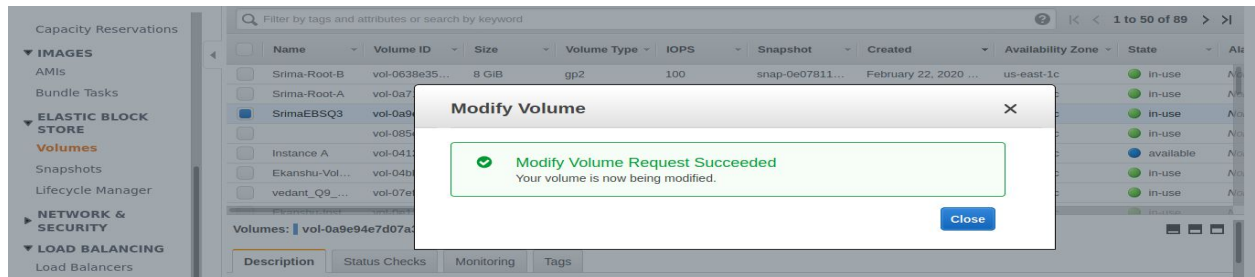
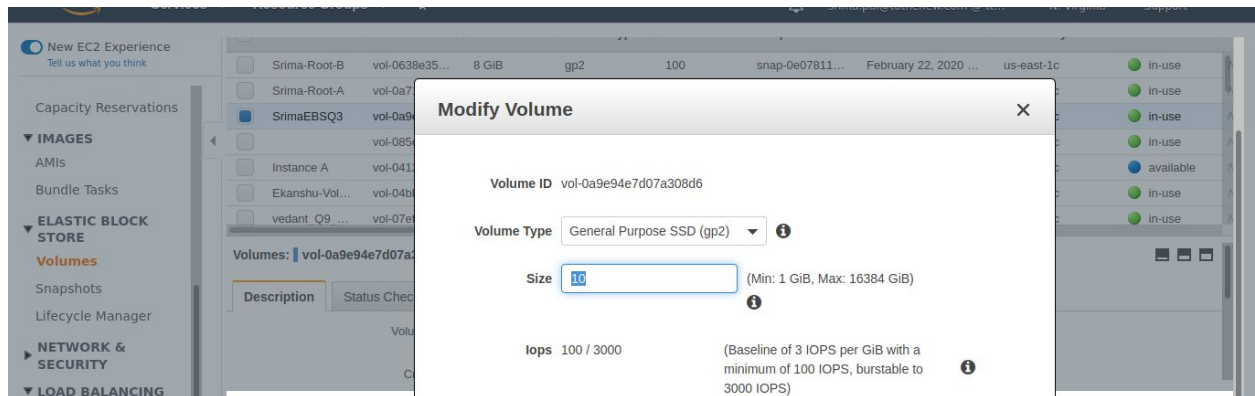


6. Resize the EBS from 8 to 10GB

Step 1. Select the volume, go to Actions and click on modify volume

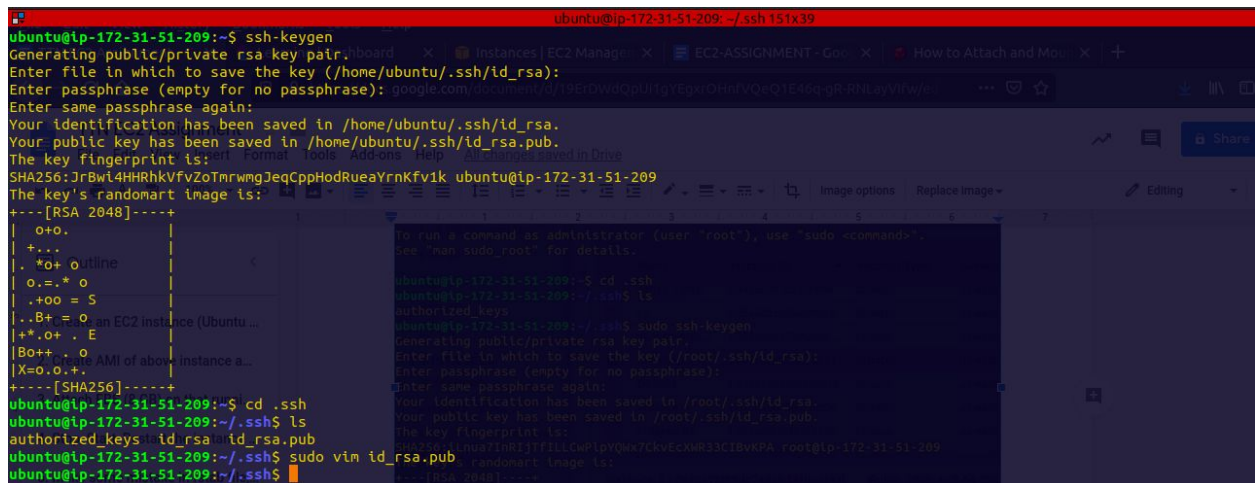


Step 2. Change the size



7. SSH from one instance A to instance B

Step 1. Login to instance A and generate SSH keys



Step 2. Copy the id_rsa.pub of instance A to authorized keys of instance B

```
ubuntu@ip-172-31-7-196: ~/.ssh 75x39
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQBAQC142D1o1Z3uUEkL7z9vtJYST2mZn+uy5nwbMB
M07X4wRbqVxVlcykF8BNFz8SevoGfXuMeZBSwwYrbIDheh4cgSgyWURCuH++N7bsqQZ3W89Pft
sOXso6tLzDui5UV0LuzoRB/RU1fgjjmgk3bj4qBMQIEJjjkyzN5/G5Fgr+SHs55X/k9/Ay3dbI6
omuMg2XeybiLHuM8nxA/IiGWFPpOweUZKORl/i5AaptBEHPvUjJ76Zp3ezjLGe3mzb/Jg1JudNX
FtonF1BYHOziaPu74nhoJp2x7dPUJo+BF4q/tbfppjYoytFWkOtslew5RmSsbQov9ZnPMUGNeIW
tnNkx SrIma-TTN-bootcamp
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDp1XNuXnL2eZmNg0iIlh14+/gPrwoelU2Tyaz
hYMFJbtTA+ufj7Rpkh01lIf75Si3GSypCku1lI1LZX+gE2CLMjLqa+1z8nrWHnLPHqWfZIdSCJ4
3HaHG6SjhIVI0J3WxzKudh/Tv891EWgTvJpLd+fTuXSSTpzWINBnfp2hqP1dYh7FeCchWf0wWcA
usehNqeTbFRURSZR0GguNqy2Me569zB+w0AIFZCXVJMOAQH+cu1sxnIES13C02lkYLLbRwISnqK
/GeRYR4jzRUQ4VMXW09Pq9N70+wIbX7izHho75mBBY9uNzr70MrNX250Ui7FCh7Ar2RSxtH98fs
y0MU7 ubuntu@ip-172-31-51-209
```

Step 3. Ssh from instance A to instance B

```
ubuntu@ip-172-31-7-196: ~/.ssh 75x39
ubuntu@ip-172-31-7-196:~$ cd .ssh
ubuntu@ip-172-31-7-196:~/.ssh$ sudo vim authorized_keys
ubuntu@ip-172-31-7-196:~/.ssh$ sudo vim authorized_keys
ubuntu@ip-172-31-7-196:~/.ssh$

Step 3. Ssh from instance A to instance B

ubuntu@ip-172-31-51-209: ~ 75x39
ubuntu@ip-172-31-51-209:~$ cd .ssh
ubuntu@ip-172-31-51-209:~/.ssh$ sudo vim id_rsa.pub
ubuntu@ip-172-31-51-209:~/.ssh$ cd
ubuntu@ip-172-31-51-209:~$ ssh ubuntu@18.207.214.2
The authenticity of host '18.207.214.2 (18.207.214.2)' can't be established
ECDSA key fingerprint is SHA256:pFD0X0JJd1EKfY0fzgvtMWAEnPOh3E/qMc4jUc+36eQ
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '18.207.214.2' (ECDSA) to the list of known host
s.
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-1057-aws x86_64)

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

System information as of Sat Feb 22 14:49:23 UTC 2020
System load:  0.0% (load average over the last 1 minute)
Processes:    90
Usage of /:    13.8% of 7.69GB
Memory usage: 15%
Swap usage:   0%

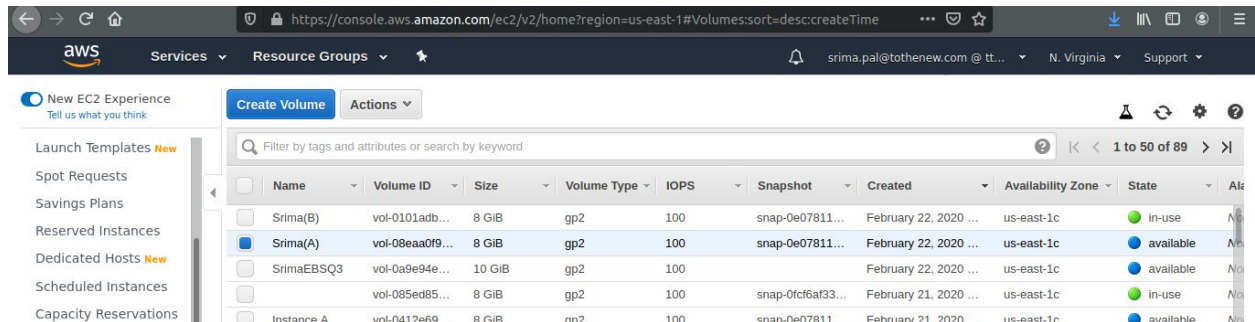
0 packages can be updated.
0 updates are security updates.

Last login: Sat Feb 22 14:45:03 2020 from 120.59.188.45
ubuntu@ip-172-31-7-196:~$ exit
logout
Connection to 18.207.214.2 closed...
```

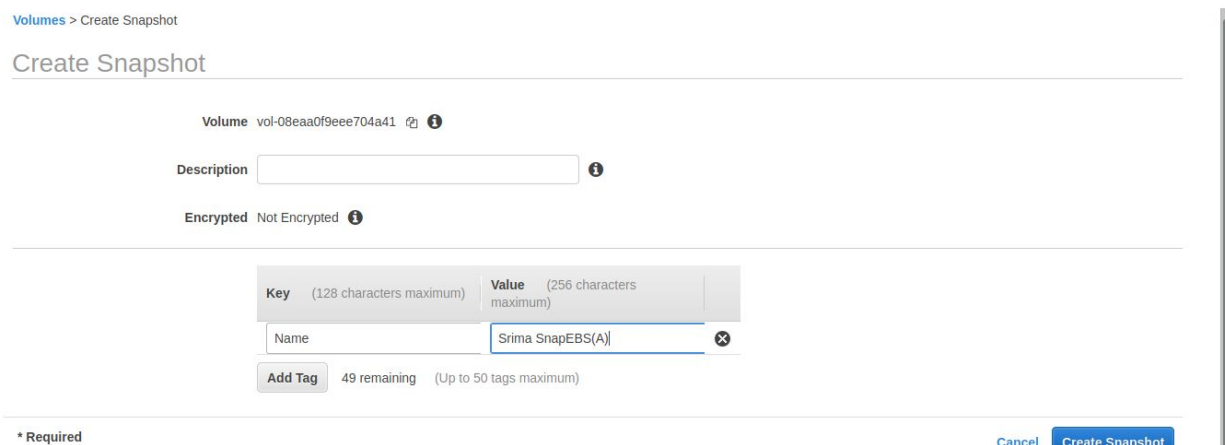
8. Copy the EBS in different region(oregon).

9. Detach the root EBS, create its snapshot, then create the AMI and run it as instance such that nginx should be pre installed at the boot time of instance.

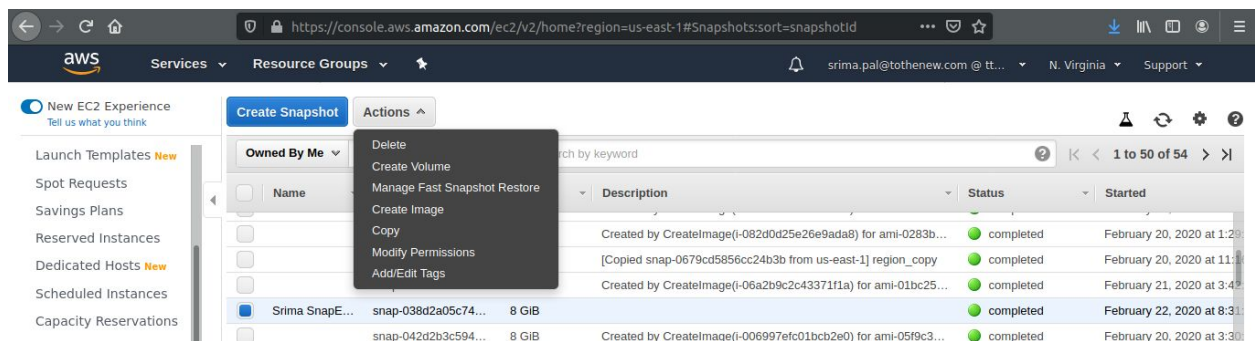
Step 1. Stop the instance and detach the root volume



Step 2. Select the volume, go to Actions, Create Snapshot, Add tags, click on create snapshot



Step 3. Go to snapshots , Select the snapshot, go to actions and then create image



Step 4. Give AMI details

The screenshot shows the 'Create Image from EBS Snapshot' dialog in the AWS Management Console. The dialog is titled 'Create Image from EBS Snapshot' and has a close button (X) in the top right corner. It contains the following fields and sections:

- Name:** 'Srima new AMI'
- Description:** (empty)
- Architecture:** 'x86_64' (dropdown)
- Virtualization type:** 'Hardware-assisted virtualization' (dropdown)
- Root device name:** '/dev/sda1' (dropdown)
- Kernel ID:** 'Use default' (dropdown)
- RAM disk ID:** 'Use default' (dropdown)
- Block Device Mappings:** A table with columns: Volume Type, Device, Snapshot, Size (GiB), Volume Type, IOPS, Throughput (MB/s), Delete on Termination, and Encrypted. The table contains one row for the root device.

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-038d2a05c74b1d801	8	General Purpose	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

Below the table is an 'Add New Volume' button. At the bottom right of the dialog are 'Cancel' and 'Create' buttons.

Step 5. AMI Created

The screenshot shows the 'AMIs' section in the AWS Management Console. A table lists the available AMIs. The 'Srima new AMI' is highlighted in blue.

AMI ID	Source	Owner	Visibility	Status	Creation Date
ami-00f0a05f37e57f026	187632318301/s...	187632318301	Private	available	February 21, 2020 at 9:46:4...
ami-06b7437cc7712c3fc	187632318301/s...	187632318301	Private	available	February 20, 2020 at 1:01:5...
ami-01a90e6c31efd84d	187632318301/...	187632318301	Private	available	February 22, 2020 at 3:39:4...
ami-01ee3a7d122f548e3	187632318301/...	187632318301	Private	available	February 22, 2020 at 8:37:4...
ami-0808f0691ef0d338b	187632318301/...	187632318301	Private	available	February 21, 2020 at 4:50:1...

Step 6. Select the AMI and launch the instance

The screenshot shows the 'Launch' button in the AWS Management Console. A dropdown menu is open, showing the following options: Launch, Spot Request, Deregister, Register New AMI, Copy AMI, Modify Image Permissions, Add/Edit Tags, and Modify Boot Volume Setting. The 'Srima new AMI' is selected in the table below.

AMI ID	Source	Owner	Visibility	Status	Creation Date
ami-0e6dae3d99a1e301c	187632318301/s...	187632318301	Private	available	February 21, 2020 at 4:15:1...
ami-0e3e3543125cbb6d1	187632318301/s...	187632318301	Private	available	February 20, 2020 at 11:16:...
ami-00f0a05f37e57f026	187632318301/s...	187632318301	Private	available	February 21, 2020 at 9:46:4...
ami-06b7437cc7712c3fc	187632318301/s...	187632318301	Private	available	February 20, 2020 at 1:01:5...
ami-01a90e6c31efd84d	187632318301/...	187632318301	Private	available	February 22, 2020 at 3:39:4...
ami-01ee3a7d122f548e3	187632318301/...	187632318301	Private	available	February 22, 2020 at 8:37:4...

Step 7. Give the bootstrap script for installing nginx in the instance while launching it

Add Device

▼ Advanced Details

User data ⓘ ☒ As text ☐ As file ☐ Input is already base64 encoded

```
#!/bin/bash
sudo apt-get update
sudo apt-get -y install nginx
sudo systemctl start nginx
```

Step 8. Login to the instance and check status of nginx

```
ubuntu@ip-172-31-6-152: ~ 75x39
ubuntu@ip-172-31-6-152:~$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled;
   Active: active (running) since Sat 2020-02-22 15:19:15 UTC;
     Docs: man:nginx(8)
  Main PID: 1904 (nginx)
    Tasks: 2 (limit: 1152)
   CGroup: /system.slice/nginx.service
           └─1904 nginx: master process /usr/sbin/nginx -g dae
              └─1905 nginx: worker process
```