**ACIT 3640 - Cloud Computing**

**(Lab 5)**

**AMI Volumes, Snapshots, Upgrading, Patching and Resizing**

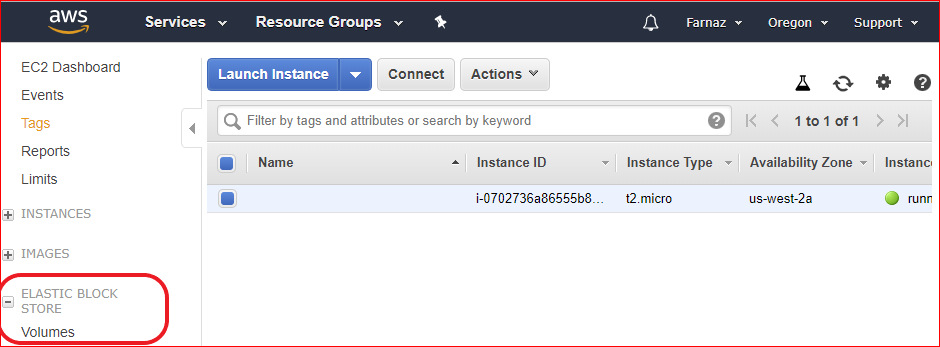
**DO NOT DELETE ANY INSTANCES UNTIL YOU HAVE RECEIVED A MARK FROM YOUR INSTRUCTOR!!!**

Volumes exist on Elastic Block Store (EBS) when an instance (AMI) is created. Volumes are essentially virtual hard disks.

Snapshots exist on Simple Storage Service (S3) and they are created from volumes. A snapshot is a copy of a volume at a particular moment in time.

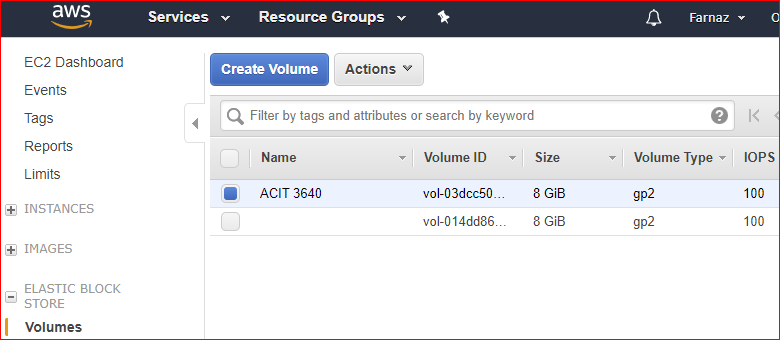
Part 1: Linux – Volumes and Snapshots

1. Launch an Amazon Linux (free tier eligible) instance.
2. Examine the volume that gets created as part of the process of creating an instance.
   1. On the left side of the EC2 Dashboard, under Elastic Block Store, click Volumes.

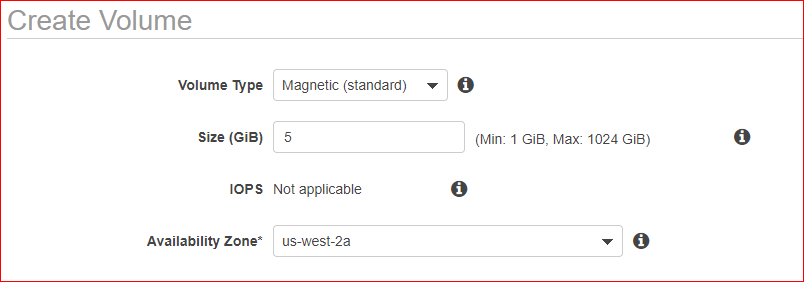


* 1. Under the column Attachment Information, locate the Instance ID of your running Linux instance.
  2. Write down the name of the device. Example: /dev/xvda
  3. Click the checkbox associated with your instance.
  4. Click the Description tab. In what Availability Zone is your volume located? Is it encrypted?

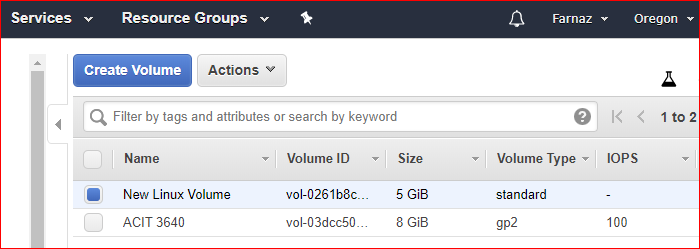
1. Create a new volume by clicking the Create Volume button on the left side of the screen. Make sure you have selected your instance.



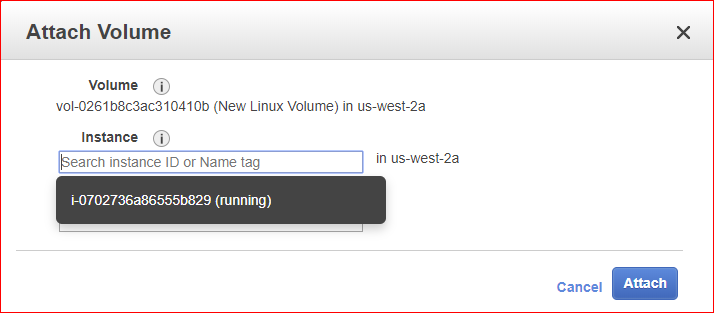
* 1. Select “Magnetic” as the Volume Type.
  2. Set the size to 5 GiB.
  3. Make sure the volume is created in the same Availability Zone as your AMI.



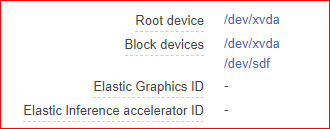
1. Once the volume has been created, give it a name. I called mine “New Linux Volume.”



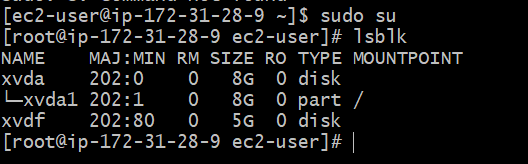
1. Attach the new volume to Linux instance.
   1. Click the checkbox next to the new magnetic volume you created.
   2. Click the Actions button and select Attach Volume.
   3. Click in the Instance field and you should see your AMI listed.
   4. Select it and accept the default value in the Device field.
   5. Click Attach.



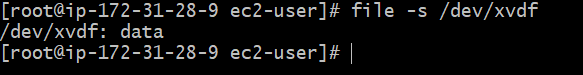
1. Click on the Description for your instance. Notice that you have a new block device associated with your instance.



1. Mount the new drive (volume) on your Linux instance and format it.
   1. Log into the instance using SSH.
   2. Remember: The username (login as) is “ec2-user.”
   3. Change to root user: “sudo su”
   4. Run the “lsblk” command.

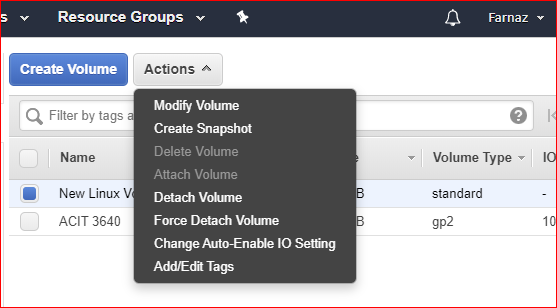


* 1. Check for data on new volume: “file –s /dev/xvdf”
     1. The volume name “xvdf” might be different on your instance.

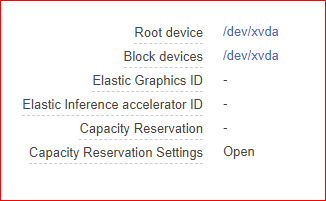


* 1. Create a file system on the new volume: “mkfs -t ext4 /dev/xvdf”
  2. Make a directory as a mount point for the volume: “mkdir /fileserver”
  3. Mount the new volume to the directory: “mount /dev/xvdf /fileserver”
  4. Navigate to the directory: “cd /fileserver”
  5. Perform a file listing: “ls”
  6. Create a couple of files on the new volume:
     1. “touch helloworld.txt”
     2. “touch index.html”
     3. Perform a file listing (ls) to verify your files were created.
     4. DO NOT DELETE THESE FILES!
  7. Navigate to the root directory of the root user: “cd /”
  8. Now, unmount the volume: “umount /dev/xvdf”
     1. No letter ‘n’ in ‘umount.’
  9. Then, “cd /fileserver” and perform an “ls.”
     1. Are your files still there? Why or why not?
     2. Take a screen shot of your bash and save it as part1-a. Show it to your instructor when you finished all parts of labs.

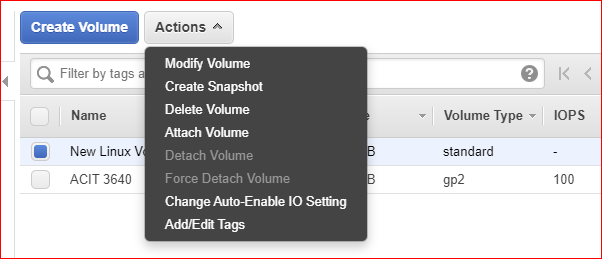
1. Go back to AWS Console and navigate to the Volumes screen.
   1. Click the checkbox next to the volume you created earlier (New Linux Volume).
   2. From the Actions button, select Detach Volume.

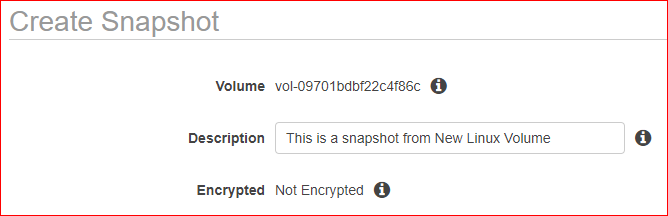


* 1. Verify the volume has been detached by looking the Attachment Information column. Notice that the block device is no longer attached to the instance.

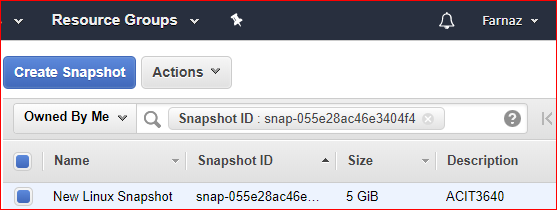


1. Create a Snapshot of the detached volume.
   1. Click the checkbox next to the volume.
   2. From the Actions button, select Create Snapshot.



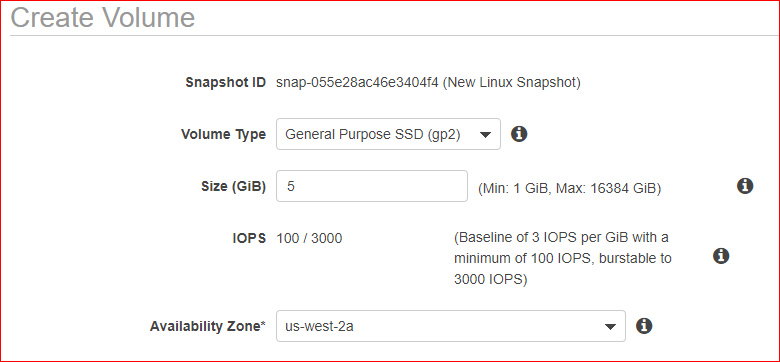


* 1. Click Snapshots on the left menu under Elastic Block Store. You should see your snapshot being created.
  2. Give the snapshot a name you will remember.



* 1. After your snapshot has been created, go back Volumes. Select your volume (New Linux Volume) and delete it. From the Actions button, select Delete Volume.
  2. Go back to Snapshots.

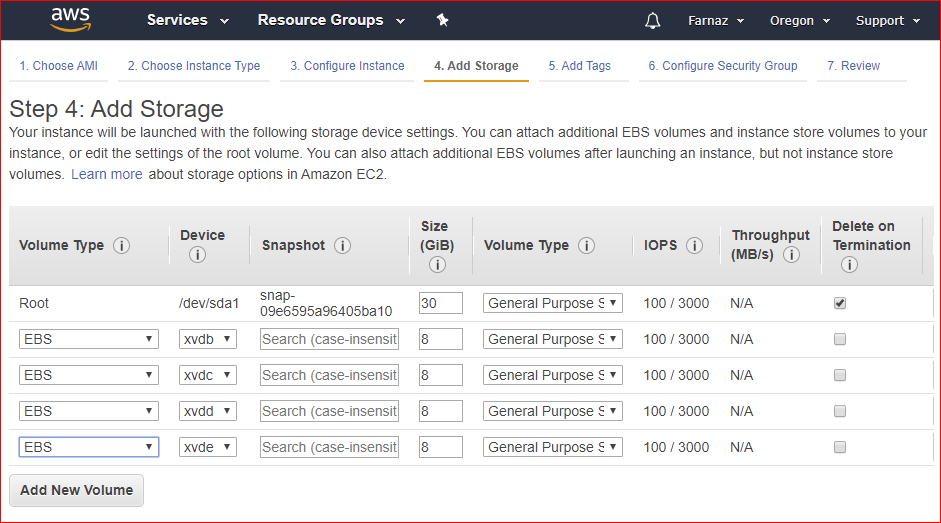
1. Create a new volume from the snapshot.
   1. Click the checkbox next to the Snapshot.
   2. From the Actions button, select Create Volume.
   3. For Volume Type select General Purpose SSD (gp2).
   4. Make sure the volume is created in the same Availability Zone as your AMI.



* 1. Go back to Volumes and attach (from action menu) the volume to your Linux instance again.
  2. Log back into your instance and remount the volume.
     1. Open a new SSH connection to your instance
     2. Change to root user: “sudo su”
     3. Run the “lsblk” command.
     4. Check for data on new volume: Run “file -s /dev/xvdf”
     5. Mount the new volume to the directory: “mount /dev/xvdf /fileserver”
     6. Navigate to the directory: “cd /fileserver”
     7. Run “ls.” Are your files still there?
     8. Take a screen shot of your bash and save it as part1-b. Show it to your instructor when you finished all parts of labs.

Part 2: Windows – RAID Volume

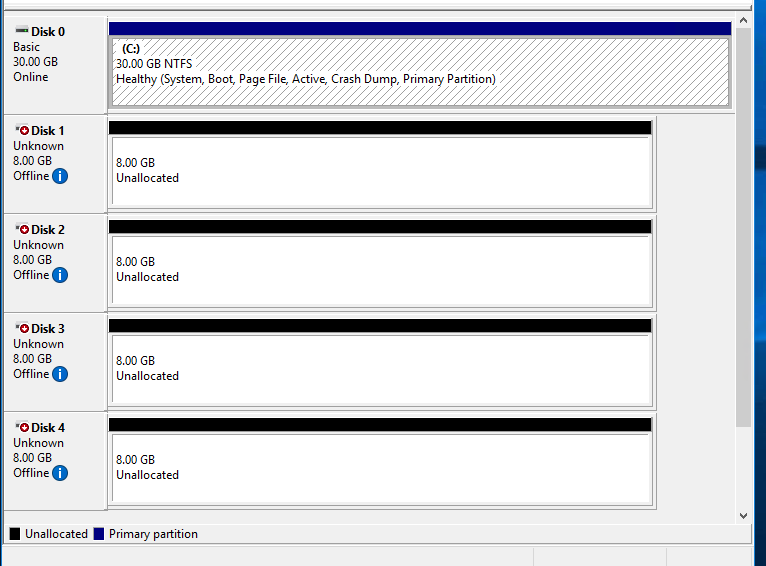
1. What is RAID (Redundant Array of Independent Disks)? See Wikipedia: <https://en.wikipedia.org/wiki/RAID> You’ll learn about this in your Windows course.
2. Read this AWS article: http://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/raid-config.html
3. Launch a Microsoft Windows Server 2016 Base (free tier eligible) instance.
4. On “Step 4: Add Storage,” add four (4) EBS volumes (8 GB) by clicking the “Add New Volume” button. Accept the other defaults.



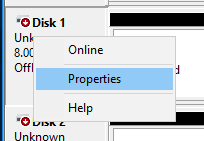
1. When adding a security group, make sure RDP is enabled.
2. Launch your instance. Note: On Step 7, you will receive a warning message that says “Your instance configuration is not eligible for the free usage tier.” Why did this happen?
3. After you launch the instance, click on the Description tab. Notice you have five block devices associated with your instance.



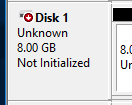
1. Connect (RDP) into your Windows instance.
2. Right click on the Windows Start button and select Disk Management. You should see a screen that looks like this. The four new disks are offline.



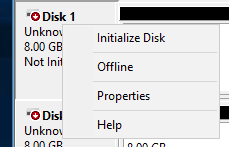
1. Right click on the left area and select Online. It may take several minutes for the disk to come online.

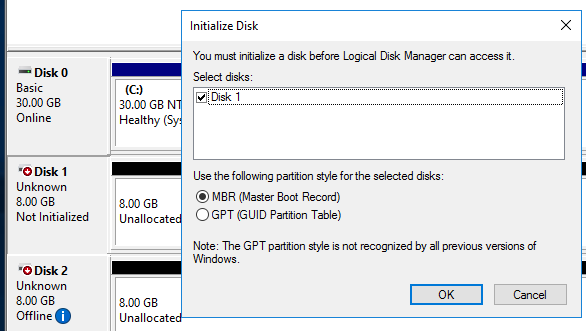


1. Now it says the disk is Not Initialized.

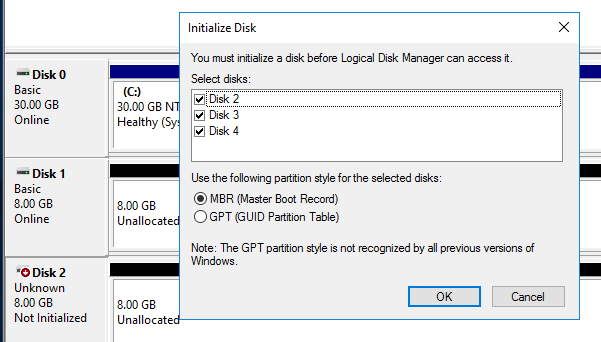


1. Right click again and select Initialize Disk.

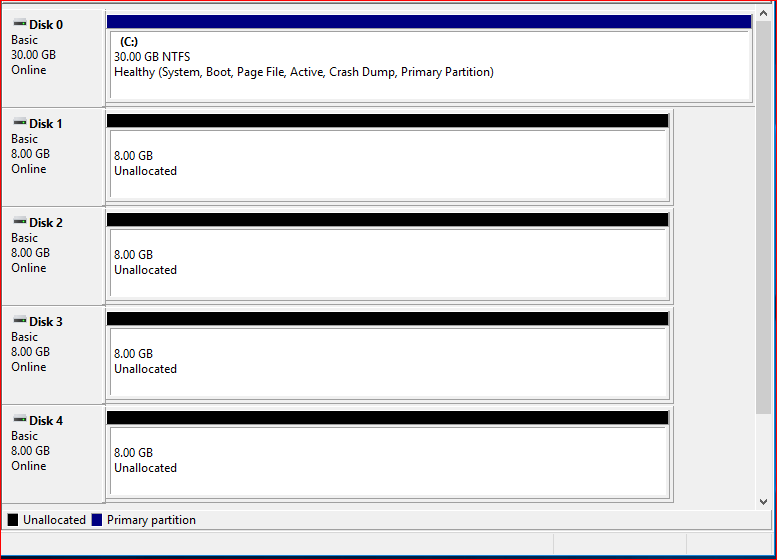




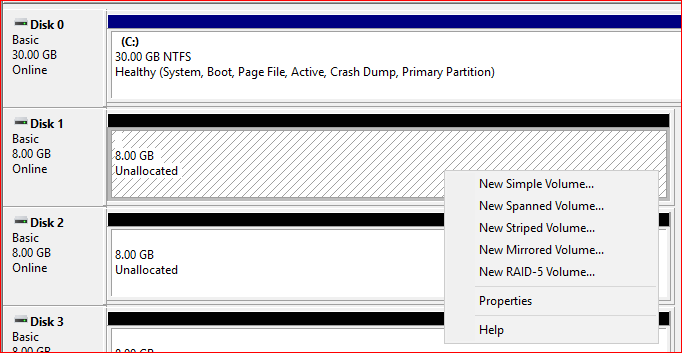
1. Do this for the remaining three disks.



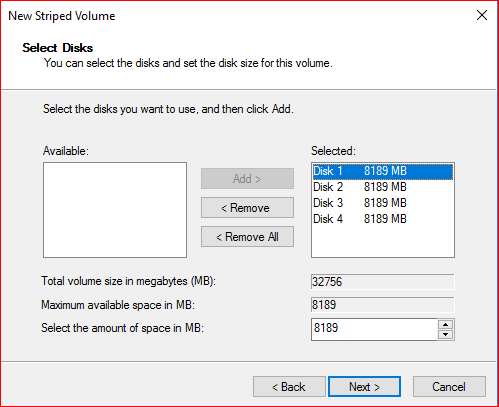
1. Your screen should now look like this with Disks 1 – 4 being Unallocated:



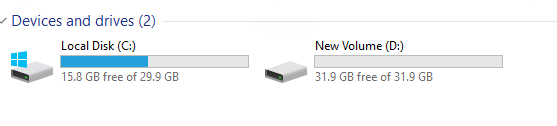
1. Right click on one of the unallocated disks and select new Striped Volume (RAID 0).



1. Move the remaining three disks from Available to Selected. You should have a screen that looks like this with four Disk in the right hand column:



1. Click Next twice.
2. On the Format Volume screen, accept the defaults and check the box that says “Perform a quick format.” You should now see a single disk (drive letter). My drive letter is D: It may take several minutes to format the new drives.
3. Launch File Explorer. You should now see a new D:\ drive. You should see one drive that is 31.9 GB.

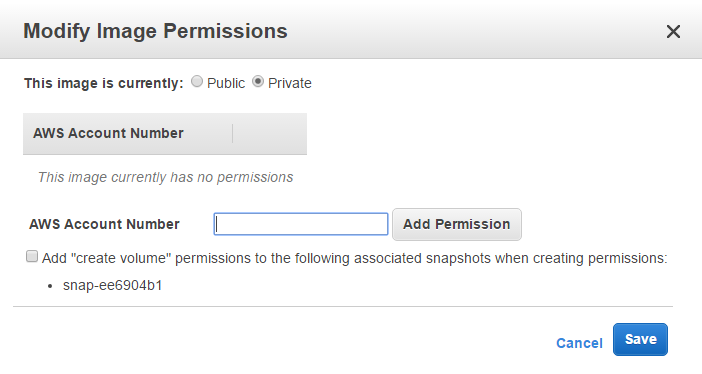


1. Don’t disconnect from your Windows instance! Keep your RDP connection and show your work when you finished all parts.

Part3: Create an AMI (Amazon Machine Image)

You are creating an image that can be published to the public or shared with other people. AMIs are regional, but they can be copied to other regions. An AMI is a template for the root volume of an instance. Permissions related to the AWS accounts that can use the AMI are saved as part of the template.

1. Select the Volume of one of your running instance (for example Windows instance) and create a Snapshot.
2. Wait for the Snapshot to be created.
3. Highlight the Snapshot and go Actions -> Create Image
4. Give the Image a Name. Enter your name or A00 number. Accept the default values of the rest of the fields, and click Create.
5. On the left side of your EC2 Dashboard, under Images, select AMIs.
   1. Note: It may take several minutes before you see your AMI listed. Have patience! You can always click the Refresh icon if you want to.
6. Select the AMI and change AMI image permissions.
   1. Click Actions -> Modify Image Permissions.
   2. Leave the image Private. Share the image with your neighbor by entering THEIR AWS Account Number (ID), not your account number. Click Save.

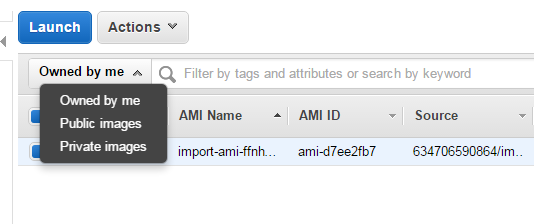


* 1. If you are going to make your image public, please read the following information. You are not going to make your AMI public

<https://aws.amazon.com/articles/9001172542712674>

<http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/building-shared-amis.html>

* 1. Ask your neighbor to go to their EC2 Dashboard and click on AMIs under Images. Your neighbor should see your image and they can Launch an instance from that image. Ask your neighbor to email you the screen shot, save it as part3 and show it to your instructor when you finished all parts.
  2. Click “Owned by Me” and select “Private Images.” (You don’t have to actually launch one.)



Part 4: AWS Instance Patching

In this exercise, we are creating an AMI as a temporary backup of the original instance prior to doing an upgrade. This is common procedure that is done when applying updates to a production Windows or Linux instance. It also applies when updating an application, such as a database, running on an instance. If the upgrade fails, then we can go back to our original image by launching the original instance from the AMI.

1. launch a Linux instance and name it as part4.
   1. Are volumes and snapshots automatically created?
2. Create an AMI from an instance.
   1. Highlight or select the instance.
   2. From the Actions button, select Image -> Create Image.
   3. On the left hand side, under Images, select AMIs. It will take some time to create the AMI.
   4. Examine the snapshots that get created. This doesn’t happen when creating a new instance.
3. SSH into the original instance. Update the original instance, run “sudo yum update -y”
4. Restart the instance. SSH into the original instance and verify that appropriate services, like Apache, are running. If Apache are not installed, *don’t* install them!
   1. To see if apache is running, type “ps aux | grep apache2”
5. If the upgrade was successful, then Deregister the AMI.
6. After the AMI has been deregistered, delete the Snapshots.

Part 5: AWS Instance Resizing

Instance resizing is the process of changing the Instance Type to a larger instance type. This usually means the workloads on an instance have increased and more memory and faster CPU is needed.

1. Using a Linux instance from part4, go to the Actions button and select Instance State -> Stop.
2. After the instance has stopped, go Actions -> Instance Settings -> Change Instance Type.
3. Select a “t2.small” from the list of Instance Types and click Apply.
4. Restart you instance. Go Actions -> Instance State -> Start.
   * Did the Public IP address change?
5. SSH into the instance and verify that appropriate services, Apache, are running.

Example: “ps aux | grep apache2”

DON’T DO THE CLEANUP UNTIL YOUR INSTRUCTOR HAS MAREKD YOUR LAB

Show your work for all the parts to your instructor.

Cleanup

1. Terminate all running instances.
2. Deregister all AMIs.
   1. You will not be able to deregister your neighbor’s AMI, only the AMIs you created. After you neighbor deregisters their AMI, you will not see it on your dashboard.
3. Delete Snapshots.
4. Delete all Volumes.