Amazon S3 Lab
AWS Essentials

Version 3.1

Introduction

Overview

Amazon Simple Storage Service (Amazon S3) is a scalable object storage service designed for the Internet. Amazon S3 allows you to store an unlimited number of files – called objects – in web-accessible containers called buckets. Objects are stored and managed in buckets via HTTP requests, and Amazon S3 provides a number of higher-level interfaces to work with objects, including a web graphical user interface called the AWS Management Console.
Although all objects are private by default, Amazon S3 provides a powerful and flexible security scheme that allows you to make individual objects public, or multiple objects public, by grouping them into folder-like structures.
Additionally, files stored in Amazon S3 may be encrypted automatically, and you can define policies to move specific objects to different storage classes such as Amazon Glacier for long-term archival storage or Amazon S3 Reduced Redundancy.

Topics Covered

The following Amazon S3 topics will be covered in this lab:

- Overview of the Amazon S3 Management Console
- Creating an Amazon S3 bucket and configuring access logging
- Uploading objects and managing object-level permissions
- Accessing objects from a web browser
- Modifying object metadata
- Encrypting objects with Server Side Encryption
- Creating folders and applying bucket-wide security with a bucket policy
- Enabling Life Cycle policies to archive and delete logs

The Scenario

As the operations focused individual in the start-up business, Asperatus Tech, you are tasked with coming up with a low cost, high availability solution for your customer facing website. Your website will host a myriad of documents for your customers, as well as video and static content. A distributed workforce will iterate upon the content. You will start off by looking into the functionality of Amazon S3, and it's functionality.

Using Amazon S3

The AWS Management Console

In this section, you access and configure the AWS Management Console, a web-based GUI provided by AWS for managing AWS services.

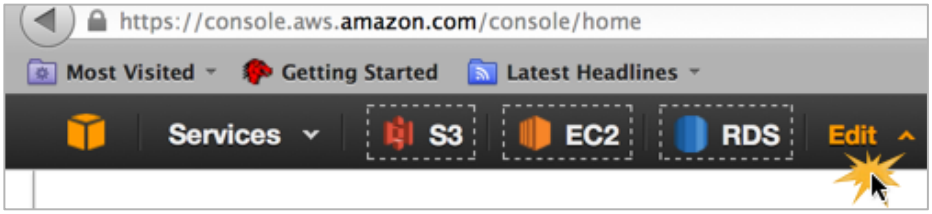
1. Click the Open Console button on the qwiklab Lab Connection page or type the following URL in your browser's Address bar – https://console.aws.amazon.com/console/home – and sign in with qwiklab provided credentials if prompted. The console page will load as shown below:

   (1) The navigation bar displays your AWS account information. Click your provided account name to view account details and to sign out of the console.
   (2) Acces to the services are available within the body of the page in the console.

2. You can customize the navigation bar at the top of the console to include shortcuts to services you frequently use. To add shortcuts to the navigation bar:

   (1) Click **Edit**.
   (2) Drag the **EC2** shortcut into the navigation bar.

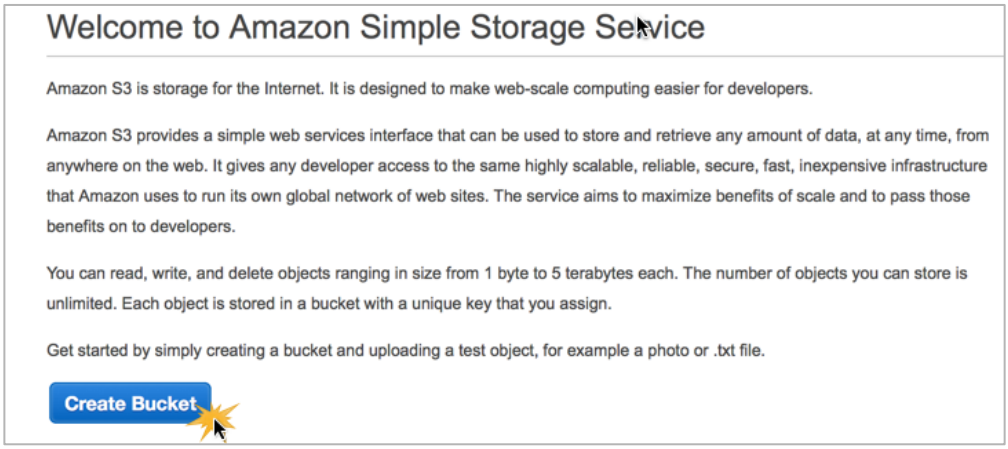
3. Drag the **S3, EC2** and **RDS** shortcuts into your navigation bar.



Amazon S3 Basics

As Asperatus Tech's subject matter expert, you need to get the public facing website up. In this section, you access the Amazon S3 Management Console, create a new Amazon S3 bucket to contain Asperatus Tech's website content, configure logging, upload an object, and then access that object.

1. Open the Amazon S3 Management Console by clicking the **S3** link in the navigation bar.
2. In the Amazon S3 console, click the **Create Bucket** button.



3. In the "Create a Bucket" dialog:

   (3) Type a **Bucket Name** such as **asperatus-text-documents**. The name you choose must be globally unique so add some something at the end, such as your initials to ensure uniqueness.

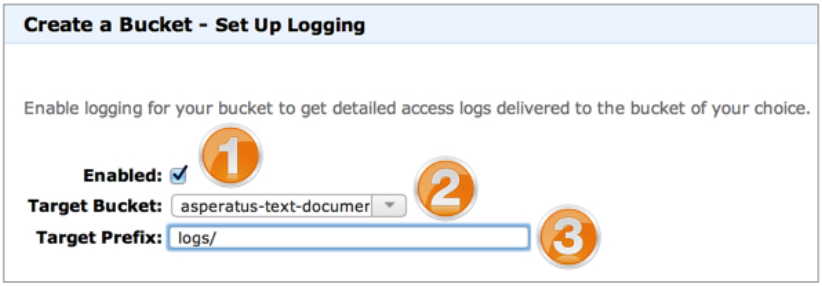
   (4) For **Region**, choose **Oregon**. Specificity here is important as all services must live within the same region to be able to communicate for this lab series.

   (5) Click **Set Up Logging**.

4. In the next "Create a Bucket" dialog:

   (6) Select **Enable Logging**.

(7) Choose the bucket you previously created, **asperatus-text-documents**.
(8) In **Target Prefix**, type **logs/** (be sure to include the trailing **/** in **logs/**).

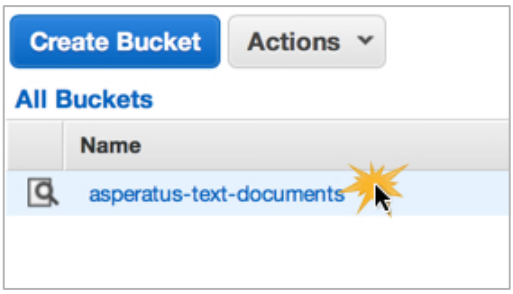


(9) Click **Create**.

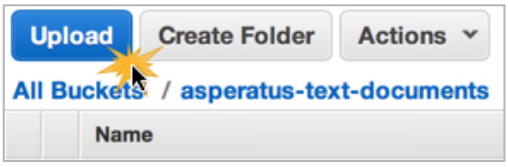
5. When your bucket is created:

(1) The bucket **Properties** panel opens. If not, click **Properties**.
(2) Within this panel you manage the configuration of your bucket such as **Permissions** to the bucket

6. On the left panel, click the bucket's hyperlink to view the contents. You will receive a message indicating the bucket is empty.

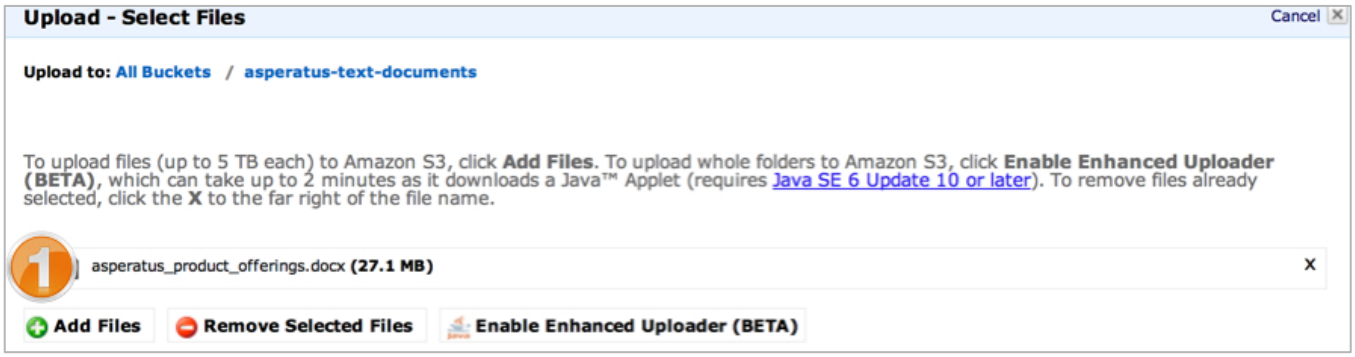


7. Click the **Upload** button to add a new file "object" to your bucket.



8. In the "Upload – Select Files" dialog:

(10) Click **Add Files**.
(11) Select a file from any location on your local machine.
(12) Click **Start Upload**.



9. Upload progress is shown in the **Transfers** panel. When the status changes to **Done**, select your object.

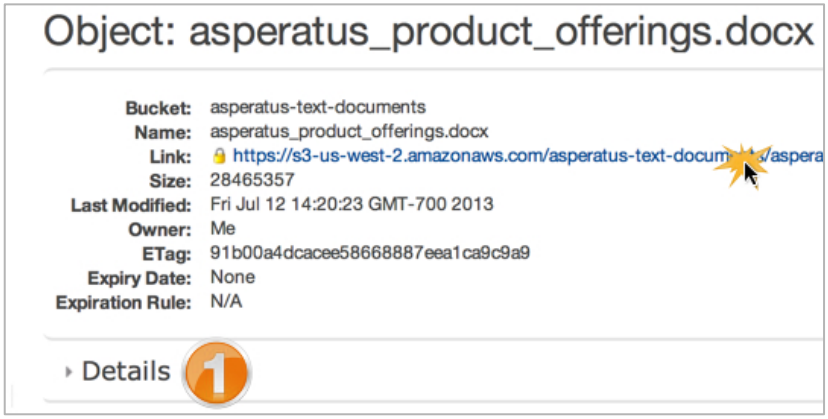
Working with Objects

Data for the Asperatus Tech website will have various phases, permissions and security associated with it. In this section you use the Amazon S3 Management Console to modify object attributes, including: permissions, encryption, and Reduced Redundancy Storage options.

10. After selecting your object, click the **Properties** button. The object details panel appears on the right.

    (13) Click the **Details** section to view object preferences.

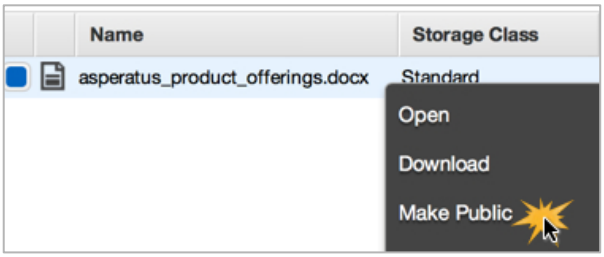
    (14) Click your object's **Link** to open it.



11. Objects in S3 are private by default. You should receive an “access denied” error message after clicking the link.

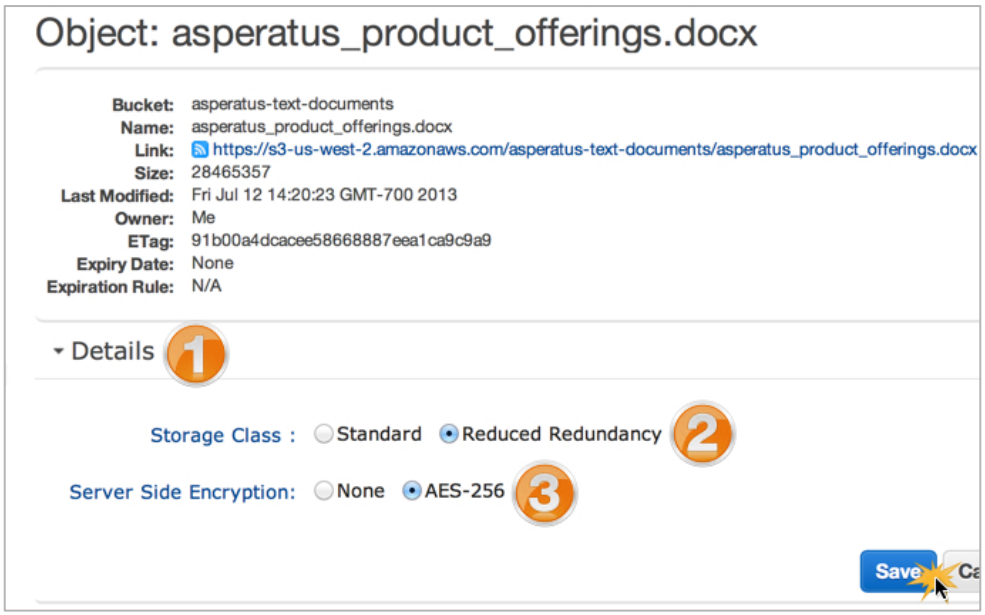
```
– <Error>
    <Code>AccessDenied</Code>
    <Message>Access Denied</Message>
    <RequestId>F00C2D8886C7F48E</RequestId>
  – <HostId>
      0OAsnqeOimABiWeMzVaimBBJZ9PoVNF0mSEQWd97vG73M/0hbyPq12/n1qeWVHjA
    </HostId>
  </Error>
```

12. To make your object publically available, right-click it and choose **Make Public**.

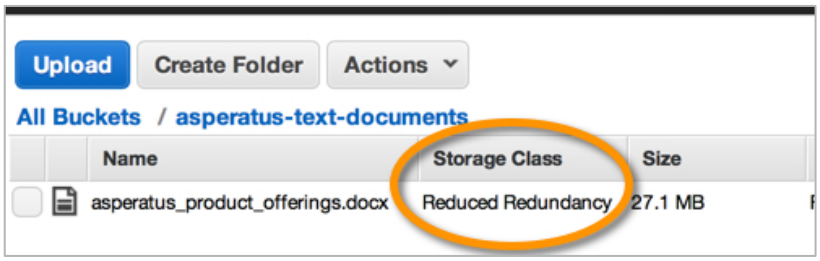


13. Click the object’s **Link** again in the Details panel. It should open without error.
14. Right-click your object and choose **Properties**.
15. In the object “Details” panel:

    (15) Expand the **Details** section.
    (16) For **Storage Class**, select **Reduced Redundancy**.
    (17) For **Server Side Encryption**, select **AES-256**.
    (18) Click **Save.** This changes your object’s storage class to RRS, and S3 automatically encrypts the object.



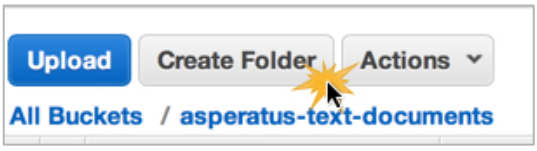
    Notice that your object’s **Storage Class** property has been changed to **Reduced Redundancy**.



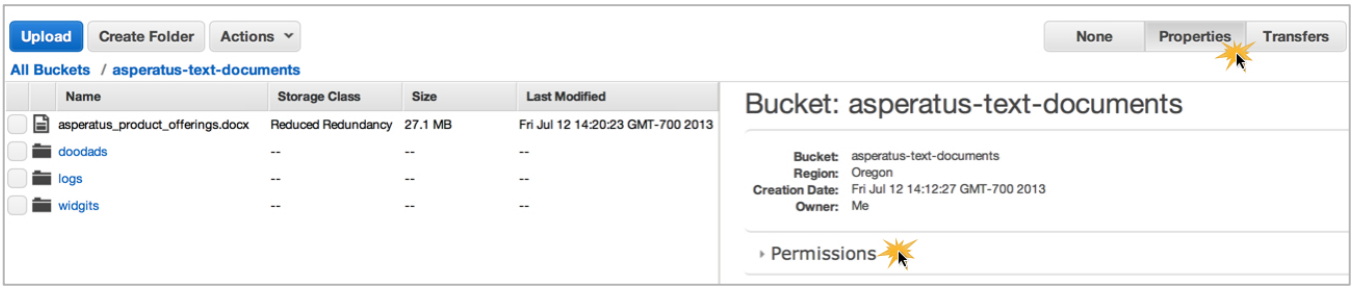
Folders and Bucket Policies

As you are uploading Asperatus documents you begin to realize the need for organization and more granular permissions for simplicity of management. In this section you use the Amazon S3 Management Console to organize your objects into folders. You also create a Bucket Policy that defines object permissions based on folder association.

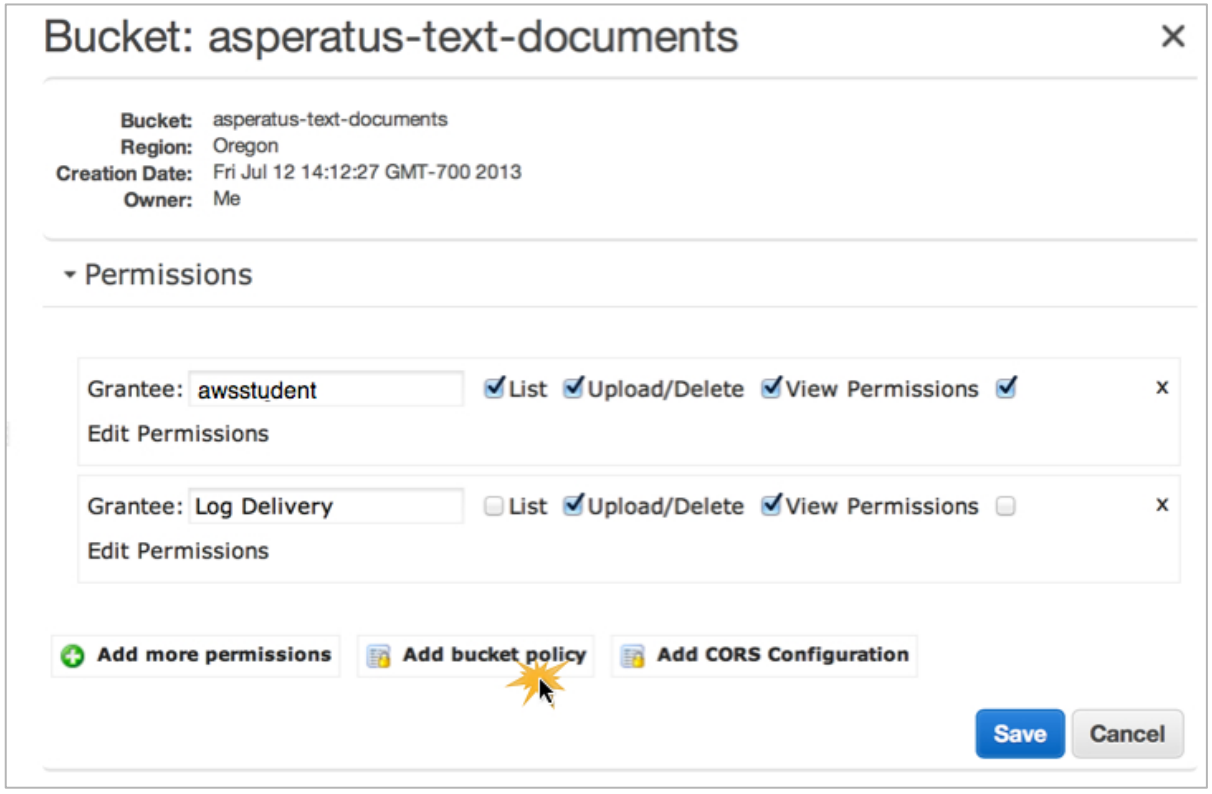
16. Click the **Create Folder** button and create three folders: **widgits**, **doodads**, and **logs**.



17. After creating the folders, click the **Properties** button to view the bucket's properties, and then click **Permissions**.



18. Bucket policies are the essence of the permissions structure for S3. Click **Add bucket policy**.



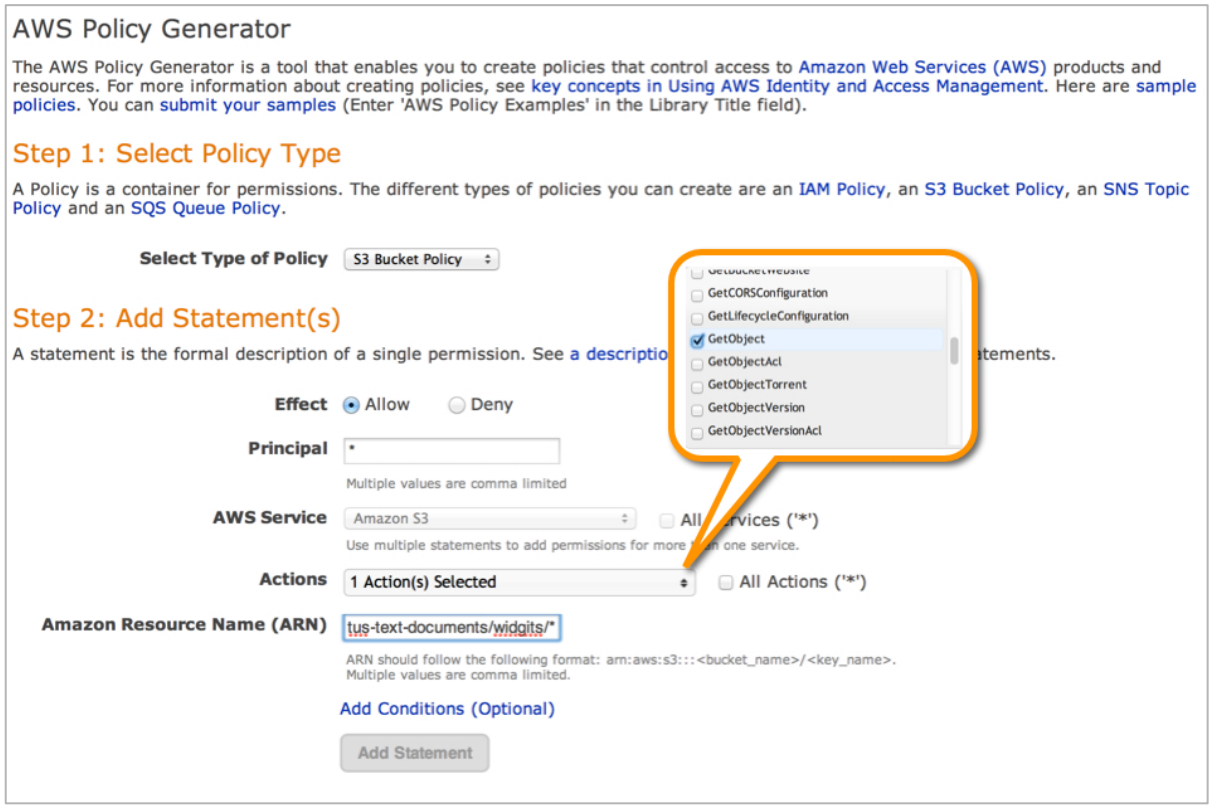
19. You can manually enter a policy document or use the Policy Generator. In this lab, you use the AWS Policy Generator to assist in constructing the policy. Click the **AWS Policy Generator** link to open the tool.

20. It is far easier to allow the Policy Generator to build a policy which we can either use or use as a guide to building a policy. In the AWS Policy Generator:

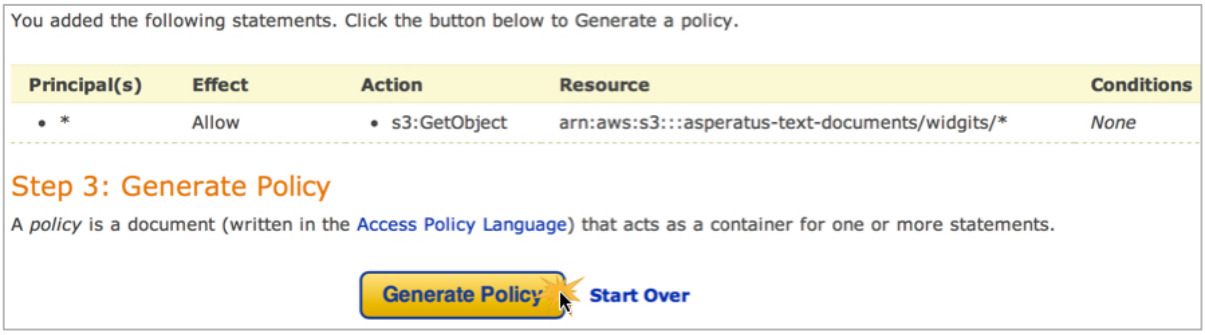
    (19)For **Select Type of Policy**, choose **S3 Bucket Policy**.
    (20)For **Effect**, select **Allow**.
    (21)For **Principal**, type ***.**
    (22)For **AWS Service**, choose: **Amazon S3.**
    (23)For **Actions**, select **GetObject**.
    (24)For **Amazon Resource Name (ARN)**, type **arn:aws:s3:::asperatus-text-documents/widgits/***

    Important: Replace *asperatus-text-documents* in the statement with your bucket's name.

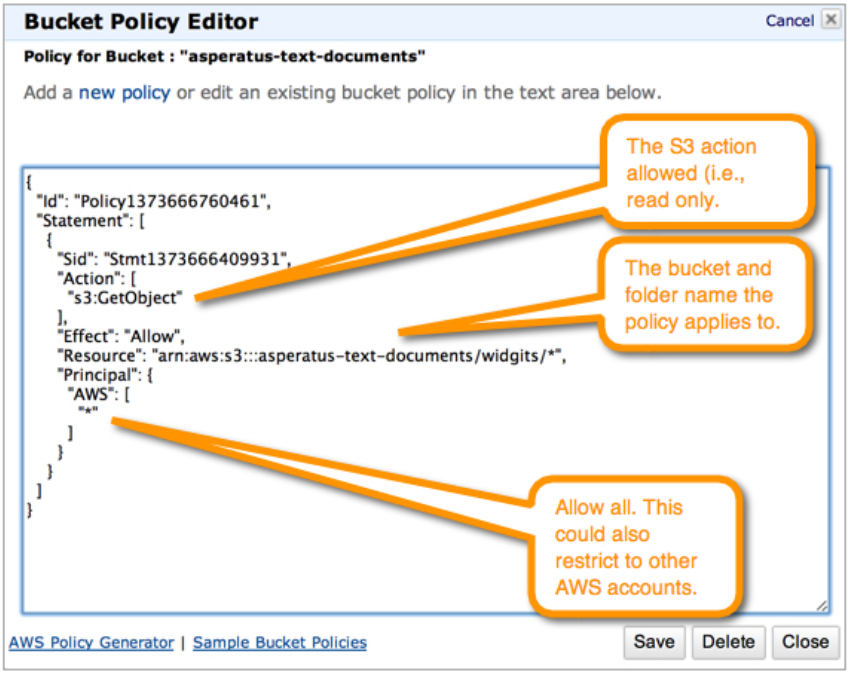
21. Click **Add Statement** to apply the newly statement to the policy editor.



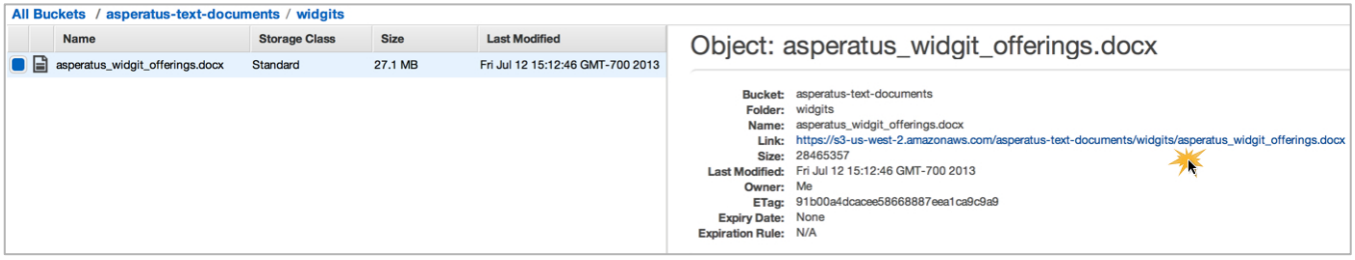
22. After adding the statement, click **Generate Policy** and copy the text to the clipboard so you can transfer it to the **Bucket Policy Editor**.



23. Return to the S3 Management Console, paste the policy into the “Bucket Policy Editor” dialog, and click **Save**.



24. It is time to put your first object into S3. Upload a new object from your computer into the **widgits** folder.
25. View the properties of that object from within the right hand panel. Click the **Link** to open the URL.

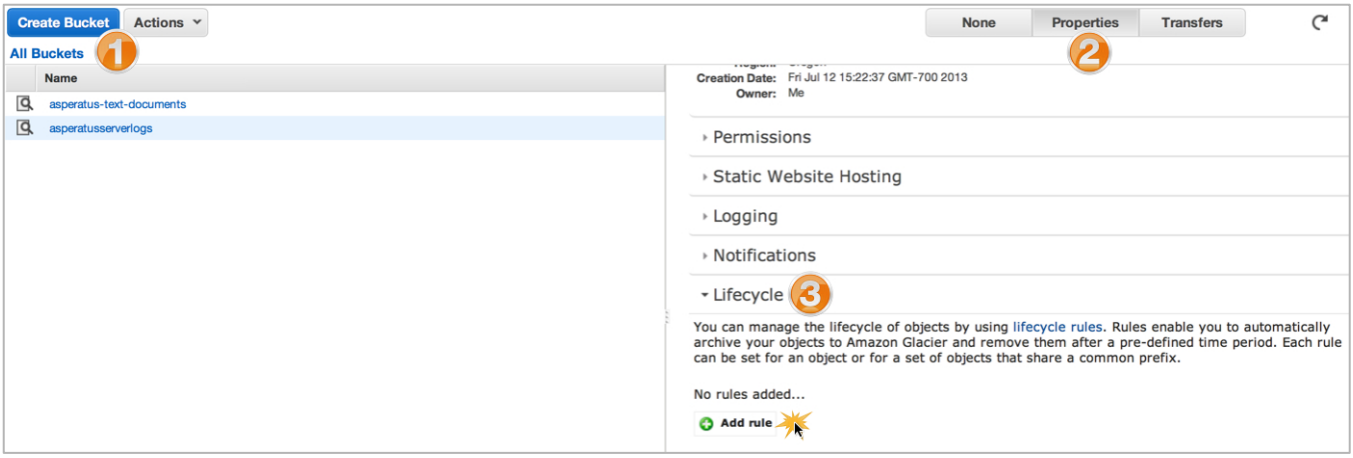


Lifecycle Policies

Now that you have Asperatus’ business needs taken care of, you want to take care of the technical needs. It is important to have logs stored in an easy location, but it is also important to move them to a lower cost location for long-term storage and then eventual removal in certain circumstances. In this section you use the Amazon S3 Management Console to define Lifecycle Rules for your bucket. The rules move files from your **logs/** directory to Glacier storage after 30 days. After 90 days, the files are deleted.

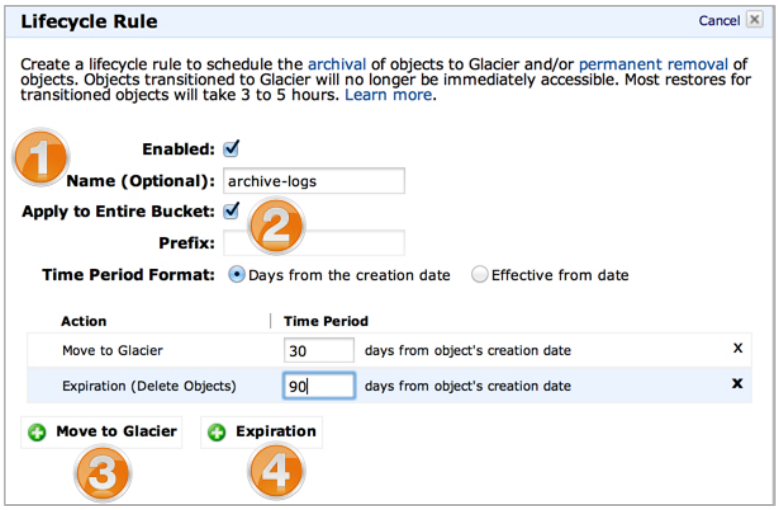
26. Create a new **bucket** called **asperatusserverlogs** (with some unique identifier such as your initials), in Oregon region.
27. Click the link to your bucket and click **Properties**. On the bucket’s “Properties” panel, add a lifecycle rule:

(25) Click **Lifecycle**.
(26) Click **Add rule**.



28. In the "Lifecycle Rule" dialog:

(27) Type a **Name** for the rule such as archive-logs.
(28) Check **Apply to Entire Bucket**.
(29) Click the **Move to Glacier** and **Expiration** buttons.
(30) Configure the rules to move items to Glacier after 30 days, and to expire them after 90 days.
(31) Click **Save**.



29. Click **OK** to confirm the rule.

Conclusion

Congratulations! You now have successfully:

- Created an S3 bucket and configured access logging
- Uploaded objects and managed object-level permissions
- Accessed objects from a web browser

- Modified object metadata
- Encrypted objects with Server Side Encryption
- Used folder and bucket policies to restrict access to objects
- Enabled life cycle policies to archive and delete logs

Please return to the course to complete the module.

For feedback, suggestions and corrections to this lab, please email aws-course-feedback@amazon.com.

AWS Essentials - Amazon EC2 Lab
Windows Version

Version 3.1

Introduction

Overview

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. EC2 reduces the time required to obtain and boot new server instances to minutes, allowing you to quickly scale capacity, both up and down, as your computing requirements change.

Topics Covered

The following Amazon EC2 topics will be covered in this lab:

- Overview of the EC2 Management Console
- Creating an EC2 key pair and a security group to allow RDP
- Choosing a correct Windows AMI and launching an instance
- Creating an AMI with customizations
- Modifying object metadata
- Creating, attaching, detaching and migrating an EBS volume
- Managing an instance's lifecycle, including: Termination protection, starting, stopping, resizing and terminating the instance

The Scenario

As the operations focused individual in the start-up business, Asperatus Tech, you have created a distribution point within Amazon S3. Your next job is to deploy a web server presence within Amazon EC2, to begin to host your content

Using Amazon EC2

The AWS Management Console

Please review the instructions included within the first lab for opening and configuring the console

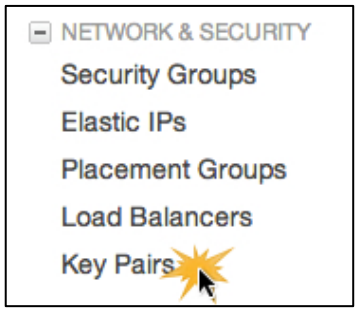
Amazon EC2 Key Pairs and Security Groups

Your operations hat now requires you to deploy a server to host your website. In this section, you access the Amazon EC2 Management Console, create a new key pair, and create a security group to allow RDP access to your new web server. The key pair will be used to retrieve your administrator account password. For Windows systems, this key pair is only used to retrieve your password and is required to do so. If it is not open, click the **EC2** link to open the EC2 Management Console.

Note: Mac users will need to download RDC.

1. Open the EC2 Management console by clicking the EC2 link in the navigation bar you created previously.

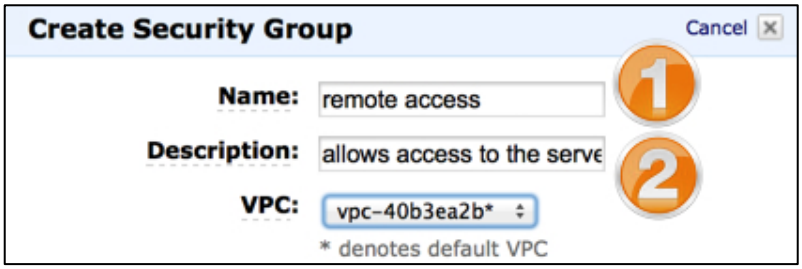
2. In the left navigation pane, click on **Key Pairs** listed under the **Network & Security**.



3. Click **Create Key Pair**.
4. In the "Create Key Pair" dialog, type a key pair name such as **asperatus _key_pair** and then click **Yes**. This will download a **[key pair name].pem** file to your computer, where [key pair name] is the name you typed in the "Create Key Pair" dialog.
5. Click **OK** to save the file to the **/Downloads** folder when the file is generated and then click **Close**.
   Note: Do not open the pem file.
6. Next, you will look at Security Groups and their usage. Click **Security Groups** under **Network & Security** to create a new group.

7. Click **Create Security Group**.
8. In the "Create Security Group" dialog:

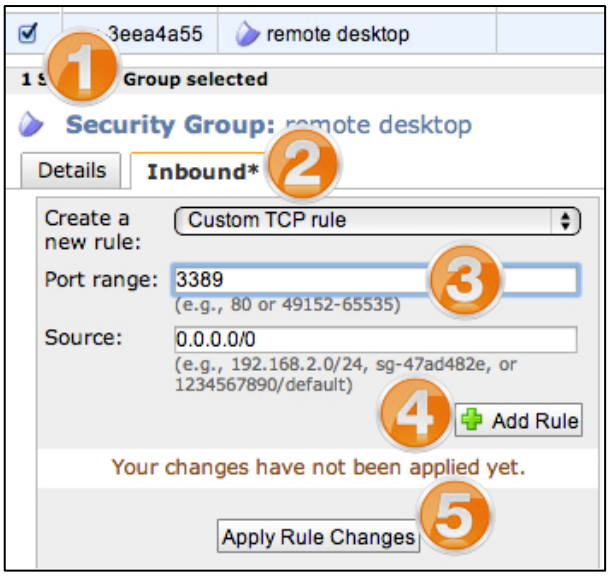
   (3) Type a group **Name** such as **remote access**.
   (4) Type a **Description** such as **allows access to server**.
   Note: Do not change the VPC.



   (5) Click **Yes, Create**.

9. In the EC2 Management Console:

   (6) If not already checked, click the check box to select the newly created security group.
   (7) Click the **Inbound** tab to show the properties in the lower panel.
   (8) In the **Port range** field, type **3389** (or use the **Create a new rule** drop down box and select **RDP**).
   (9) Click **Add Rule**, and repeat the previous step to add port **80** (or use the **Create a new rule** drop down box and choose **http**).
   (10) Click **Apply Rule Changes**.



This shows how to utilize security groups to allow inbound connections to your instances. In this example, you allowed 3389 (RDP) and 80 (http) to your Asperatus Windows server. You are now ready to retrieve your Windows Administrator password and to use RDC to connect to the Windows instance you create in the next step.

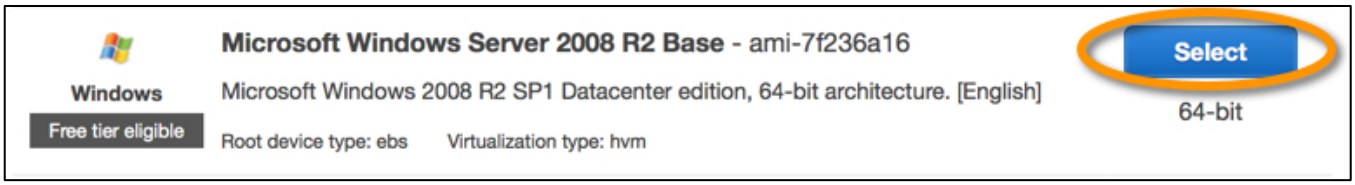
Note: It is a best practice to not enable RDP directly to production servers. Rather, to use a bastion server (or jump server) as the entry point for RDP and SSH connections.

Launching an Amazon EC2 Windows Instance

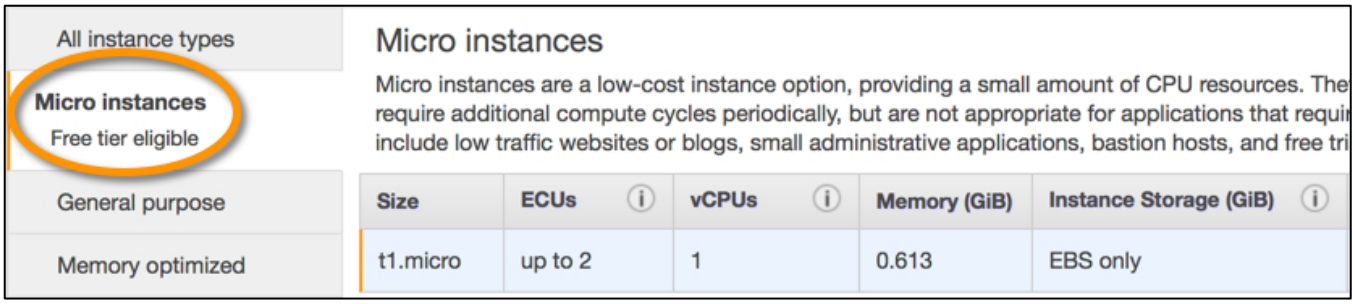
Now that you have your key pair and security groups defined, it's time to launch your web server. In this section, you will launch a Windows instance from an AMI, apply user data to customize and bootstrap the launching of the instance with tools you need to manage your Asperatus web server, and connect to the instance and validate that the instance launched with Asperatus customizations applied.

10. In the EC2 Management Console, click **Instances** listed under **Instances**.
11. Click **Launch Instance**.

12. At the first screen, **Step 1:Choose an Amazon Machine Image (AMI)** panel, scroll to **Microsoft Windows Server 2008 R2 Base** and click **Select**, located to the right of the instance name. This instance only supports 64 bit architectures.



13. Then, at **Step 2: Choose an Instance Type**, ensure that **Micro instances, t1.micro** is selected, and click **Next: Configure Instance Details.**



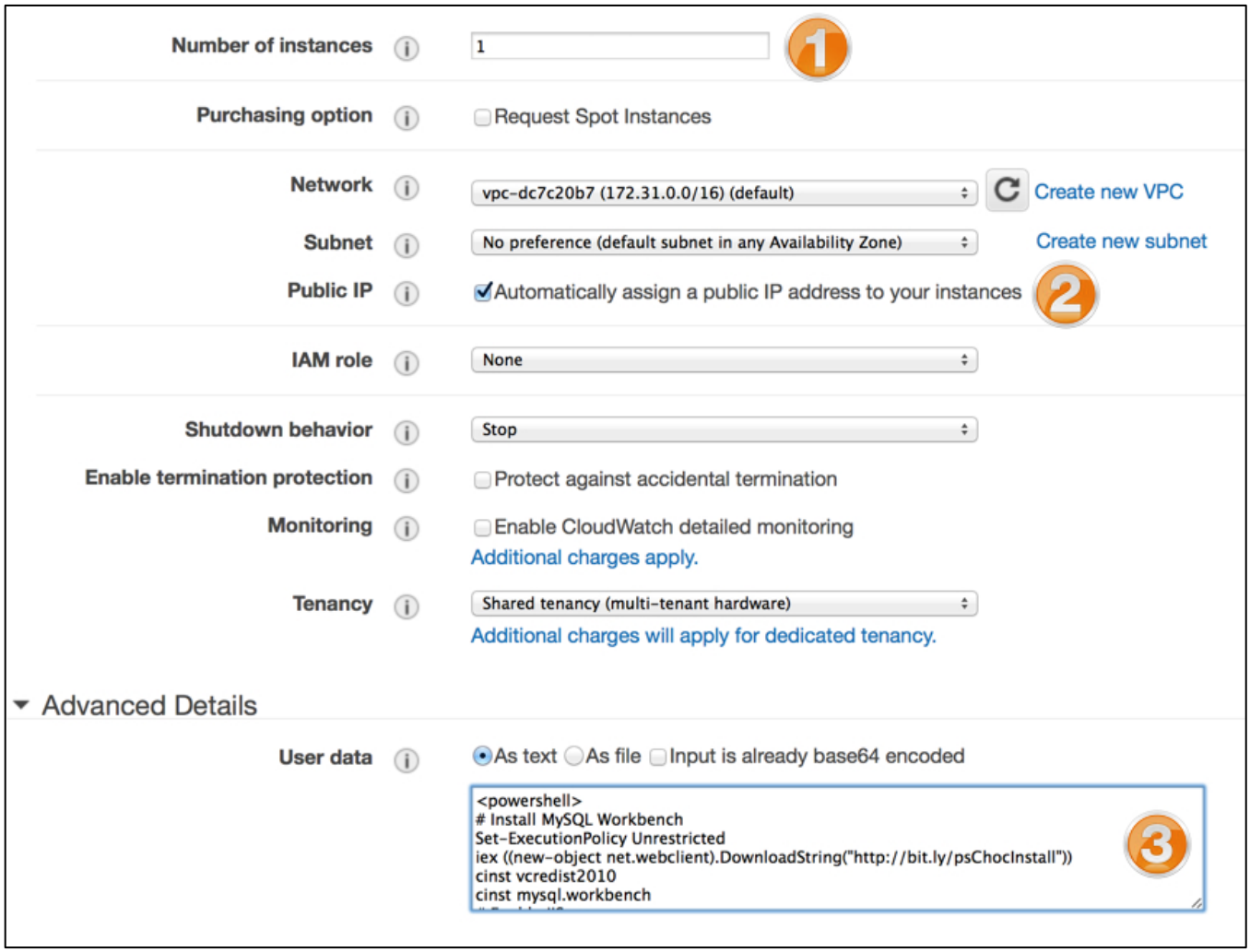
14. Next, **Step 3: Configure Instance Details**. Here,

    (11) Ensure the number of instances is set to **1**.

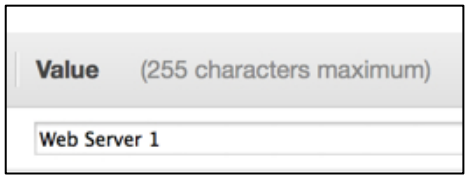
    (12) Scroll down to ensure the checkbox for **Automatically assign a public IP address to your instances** is **checked.**

    (13) Input the following data into the user data section by expanding the **Advanced Details** to expose the input. These customizations install MySQL Workbench, and the IIS packages to run a website. Just what you need to test connectivity to your database and get this web server running!

```
<powershell>
# Install MySQL Workbench
Set-ExecutionPolicy Unrestricted
iex ((new-object net.webclient).DownloadString("http://bit.ly/psChocInstall"))
cinst vcredist2010
cinst mysql.workbench
# Enable IIS
$packages = "IIS-WebServerRole;" +
   "IIS-WebServer;" +
   "IIS-CommonHttpFeatures;" +
   "IIS-StaticContent;" +
   "IIS-DefaultDocument;" +
"IIS-ManagementConsole;" +
   "IIS-ManagementService;" +
   "IIS-LegacySnapIn;" +
   "WAS-NetFxEnvironment;" +
   "WAS-ConfigurationAPI"
 Start-Process "pkgmgr" "/iu:$packages"
(servermanagercmd -install Web-Server -restart)
</powershell>
```

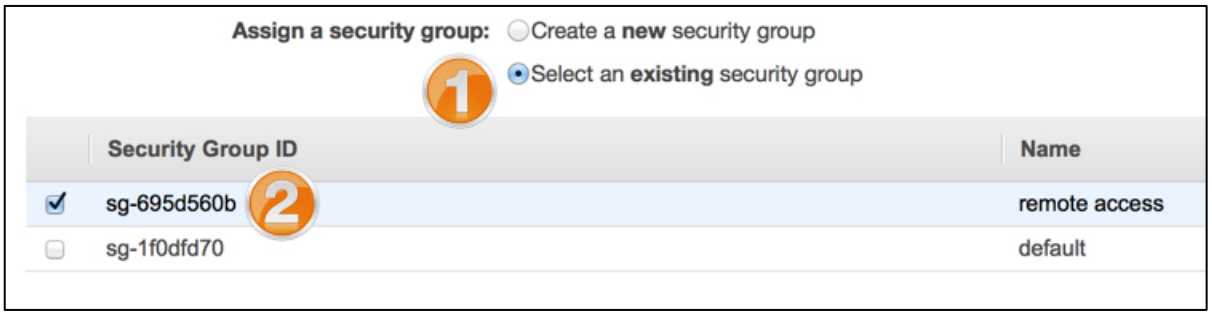


15. After the customizations are complete, click **Next: Add Storage**.
16. At **Step 4: Add Storage**, accept the default values by clicking **Next: Tag Instance**.
17. In **Step 5: Tag Instance**, type a meaningful name in the **Value** field.



18. Then click Next: Configure **Security Group**.
19. You will configure the security group that the instance operates within in **Step 6: Configure Security Group**.

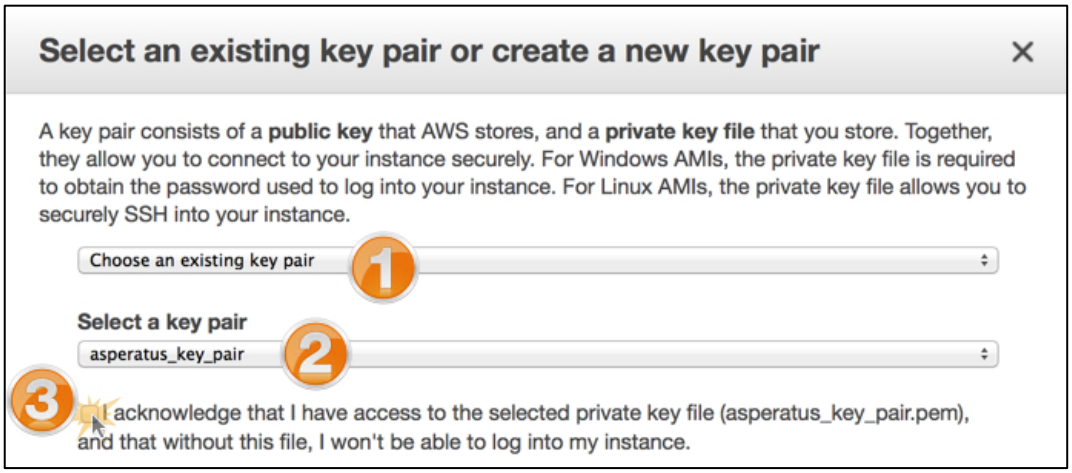
    (14) Select the radio button for **Select an existing security group.**
    (15) Place a check mark next to the security group you created earlier.



Click **Review and Launch.**

20. At **Step 7: Review Instance Launch**, you can review all of the settings that you have configured through this wizard. Click **Launch** to continue.
21. You are presented with a window **Select an existing key pair or create a new key pair.**

    (16) Ensure **Choose an existing key pair** is selected and that
    (17)[your_keypair_name] is selected,
    (18) And then check the box stating you have the private key.
    (19) Click, **Launch Instances**.

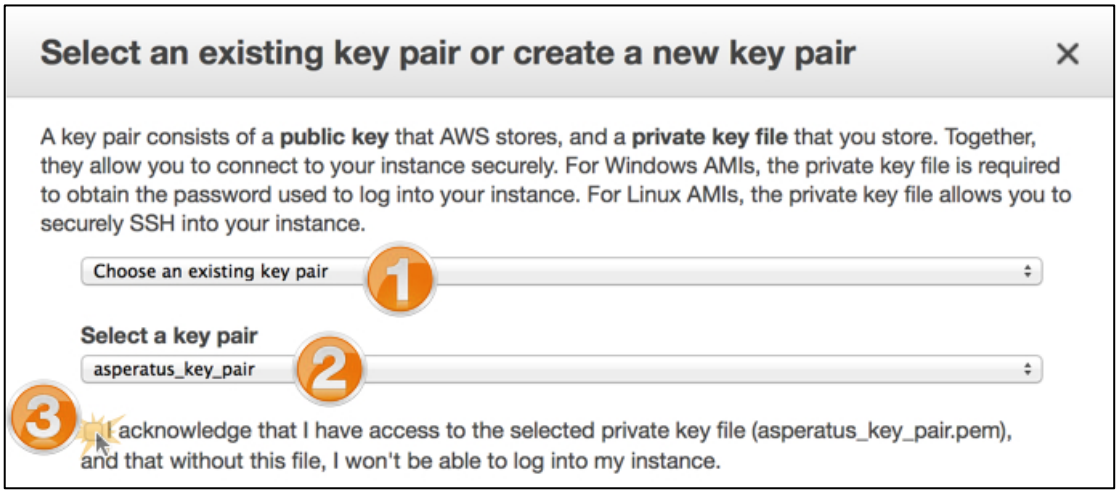


22. In the, **Launch Status** window, click **View Instances** to return to the instances view of the EC2 control panel.
23. When the instance is fully provisioned, you will be able to select the newly created Windows instance in the EC2 console. To retrieve instance information:

    (20) Right-click the instance name.
    (21) Choose **Get Windows Password** to retrieve your logon information. This will also give you the external DNS name to use in the remote desktop client.
    *Note: It can take 15-30 minutes, so now may be a good time to skip below to the **Using EBS Volumes** section while you wait for the instance to provision.*

24. This option launches a wizard to retrieve the default windows administrator password, account name, and instance connection link. The wizard requires the **.pem** file you created previously. In the wizard:

    (22) Ensure **Choose an existing key pair** is selected.
    (23) Ensure the key pair you created is selected (not the *qwik*LAB™ keypair).
    (24) Check the box acknowledging that you have access to the private key.
    Note: This is the file you downloaded earlier in this lab!

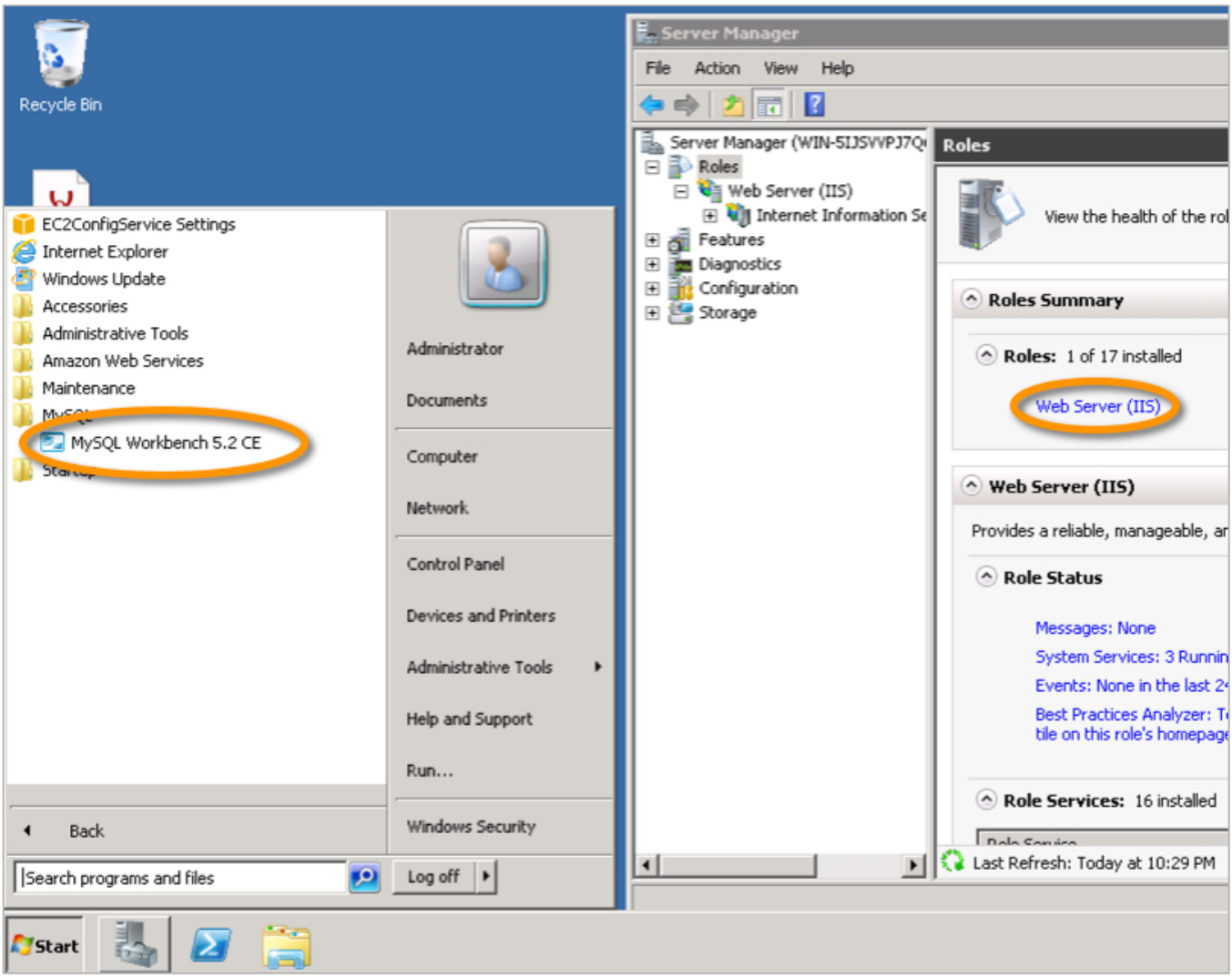


    Next, click **Decrypt Password**.

25. The **Retrieve Default Windows Administrator Password** dialog displays the information used by the remote desktop client to connect to your image. Type or copy the information using a text editor such as Notepad:

    (25) The public IP address of the instance
    (26) The administrator password (make note of this, as you will need it later!)

26. Click **Close** when you are finished reviewing the information.
27. Using Remote Desktop Connection and the information from the previous steps, connect to your instance. You may get an invalid server certificate warning while connecting; this error can be safely ignored at this time.
28. Once connected to the instance using the remote desktop client, verify the MySQL tools are installed by clicking **Start > All Programs > MySQL**. Also, verify the IIS roles by clicking **Start > Administrative Tools > Server Manager > Roles**. This is also a good time to change your administrator account password. This process demonstrates how you can use scripts to easily install tools and services at instance creation.

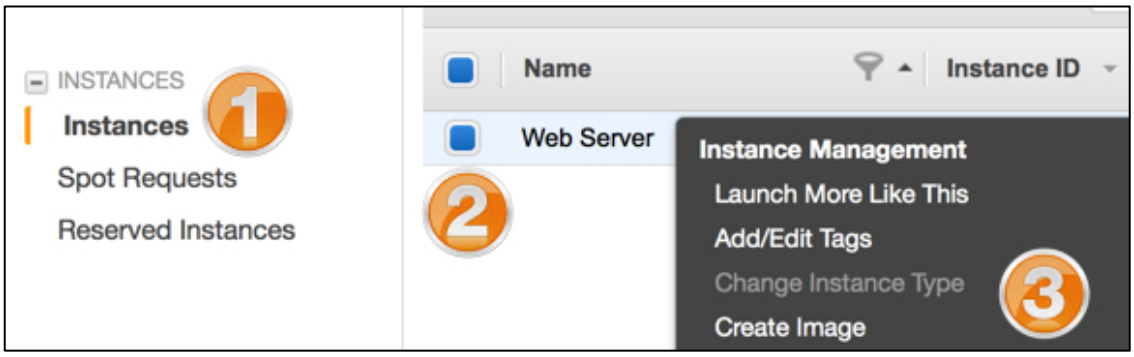


29. From your local computer's web browser, input the IP address of your instance that you collected in the previous steps. You should see the Welcome to IIS default Windows website.
30. When you are finished, close Remote Desktop Connection.

Creating an AMI

You successfully started your Windows instance and validated its functionality! Great! Now, you will need to make this a “golden image” to allow for quick and easy deployments in the future. That’s where the custom AMI comes into play. In this section you use the Amazon EC2 management console to create an AMI from your existing EC2 instance. Using this process, the AMI you generate becomes a template for generating future EC2 instances with IIS and MySQL Workbench pre-installed. You also learn some of the use cases for building custom AMIs.

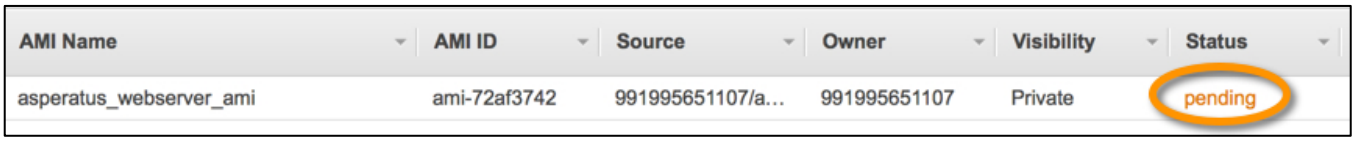
31. Using the instance created in the previous section, “Launching an Instance”, create an AMI. In the EC2 Management Console:
    (27) In the left navigation pane, expand **Instances** and click **Instances**.
    (28) Right-click your instance name.
    (29) Choose **Create Image**.



32. In the “Create Image” dialog:

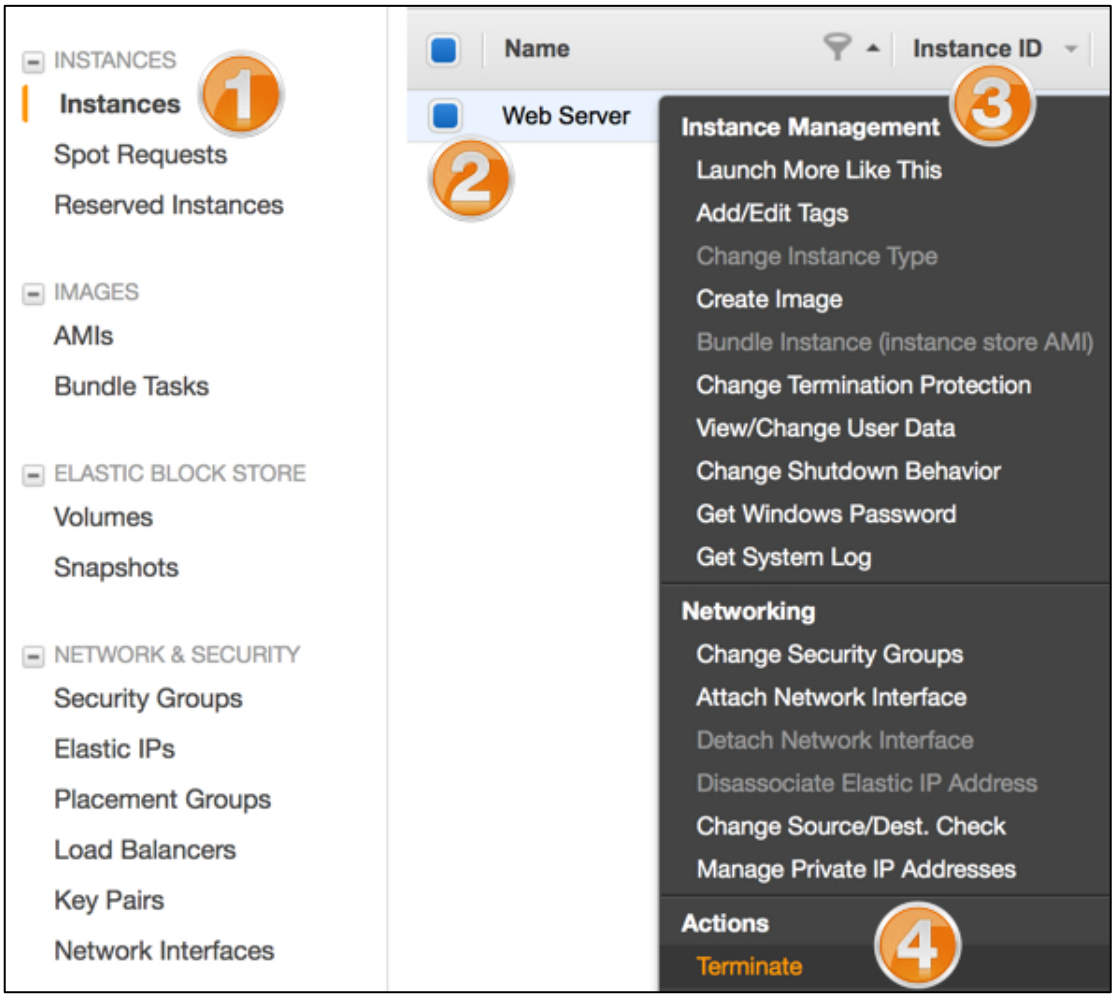
    (30) Type a name in the **Image Name** field such as **asperatus-webserver-ami**.
    (31) Type an optional description in the **Image Description** field such as **Asperatus web server ami**.
    (32) Click the **No Reboot** box.
    (33) Accept the remaining default values and click **Yes, Create**.

33. Creating the image takes 5-15 minutes to complete. In the “Create Image” dialog, click the **View pending image…** link to view AMI creation progress. Alternatively, click **Images > AMIs** in the EC2 management console and notice the **Status**. It will change from pending to available.



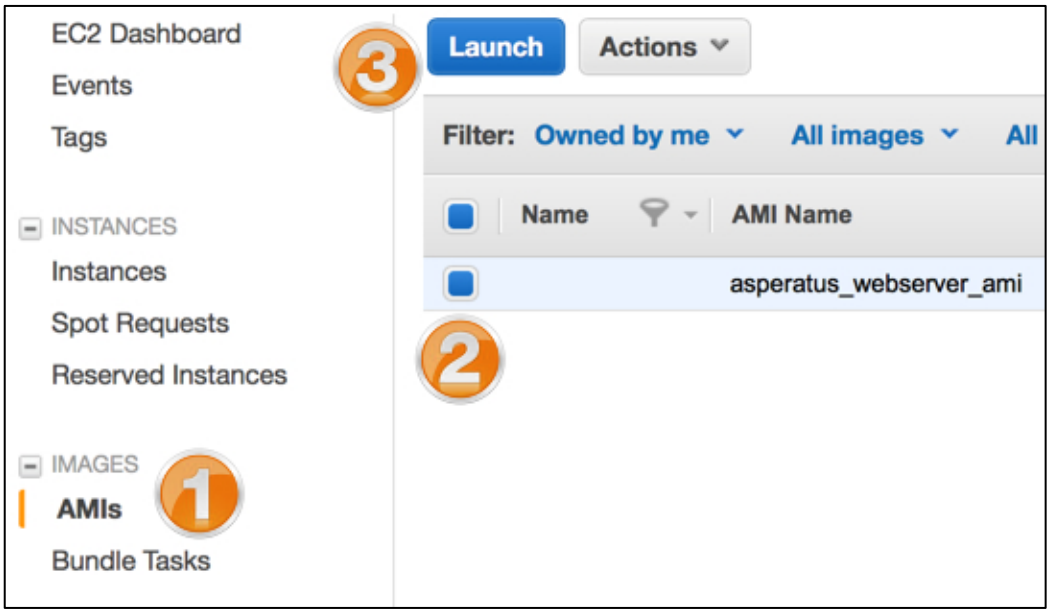
34. Next, you will terminate your current EC2 instance and re-deploy using the AMI you created. **Be sure you have the Windows Administrator password handy for the following step!** This demonstrates that the AMI image behaves like a template containing the configuration you specified using the Powershell script (pasted into the "User Data as text" field). To terminate the existing EC2 instance and re-deploy an instance from the AMI that includes the IIS roles and MySQL tools:

    (34) In the EC2 Management Console, click **Instances > Instances**.
    (35) Select the box to the left of the Windows instance you created to select it.
    (36) Right click **the instance**.
    (37) Choose **Terminate** from the drop-down list.

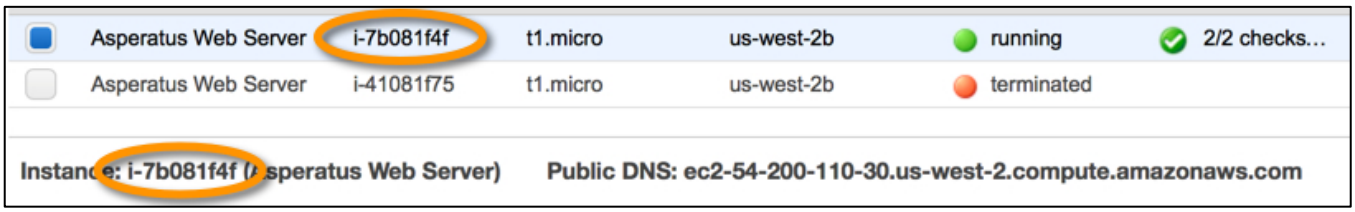


35. In the "Terminate instances" dialog, click **Yes, Terminate**.
36. To launch the AMI:

    (38) In the EC2 Management Console, click **Images > AMIs**.
    (39) Select the new image by clicking the box to the left of the image name.
    (40) Click **Launch**.



37. Use the default options for all screens until **Step 6:Configure Security Group.**
38. At **Step 6,** choose **Select an existing security group,** and pick the remote access security group you created.
39. Click **Review and Launch.**
40. Click **Launch**.
41. Similarly to how you selected your key pair when you launched the instance in previous steps, select your asperatus_key_pair in the **Select an existing key pair** dialogue window.
42. Click **View your instances on the Instances page**.
43. When complete, a terminated instance and your newly deployed AMI appear in the EC2 management console. Note: Your instance names and AMI IDs may differ from the image below.



44. After a few minutes, click the new instance. Retrieve the DNS name from the lower pane.
45. Connect to the instance via Remote Desktop Connection using the password from the previous instance, and confirm that the MySQL tools and IIS roles are both present and that IIS is running properly (use previous steps as a guide if you need assistance).
46. When you are finished, close Remote Desktop Connection.
47. Repeat the process for deploying an AMI to create a second instance.

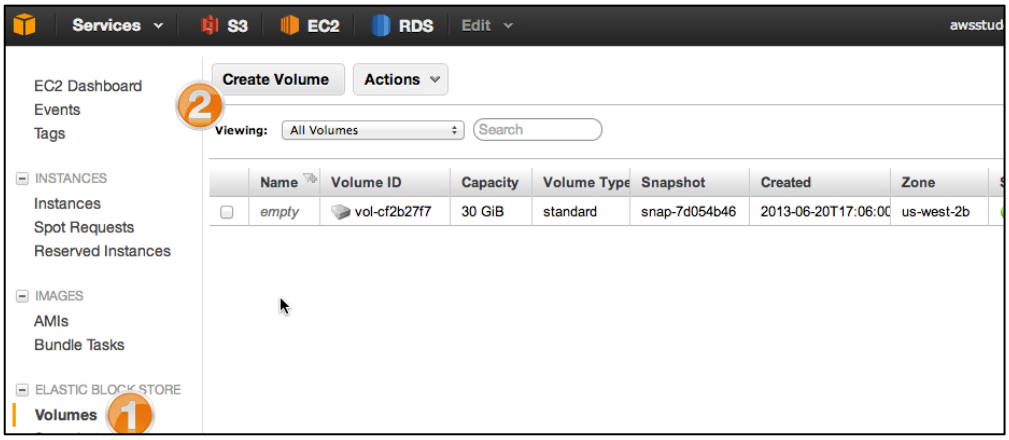
Using EBS Volumes

Now that you, the Asperatus employee in charge of spearheading the infrastructure, have a pair of web servers, and S3 buckets configured, you will take a look at EBS volumes and some ways in which they are used. In this section you use the Amazon EC2 management console and Windows disk management utility to create, attach, detach, migrate and manipulate an EBS volume.

48. In the EC2 Management Console:

    (41) In the left navigation pane, click **Volumes** under **Elastic Block Store**.

    (42) Note the pre-existing volume from the Windows server instance you created earlier (specifically the Zone) and click **Create Volume**.



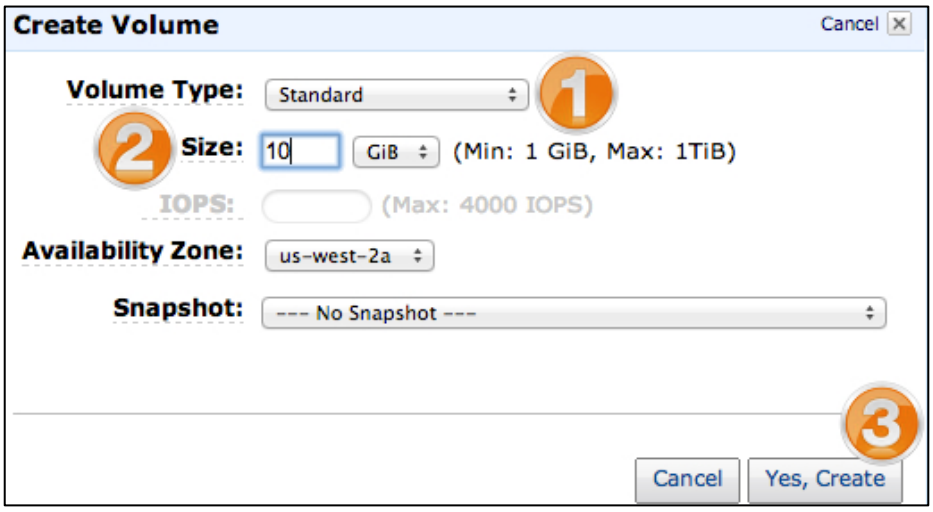
49. In the "Create Volume" dialog:

    (43) For **Volume Type**, choose **Standard**.

    (44) For **Size**, type **10** and choose **GiB** from the drop-down list.

    (45) Choose an **Availability Zone** value that differs from the zone for your EC2 instance (noted in the previous step).

    (46) Accept the remaining default values and click **Yes, Create**.

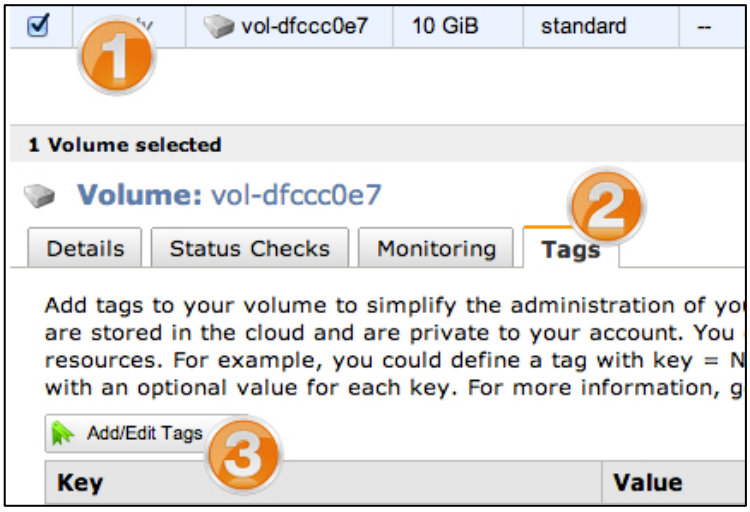


50. Name your new volume:

    (47) Check the box to the left of the volume name to select it, if it is not already selected.

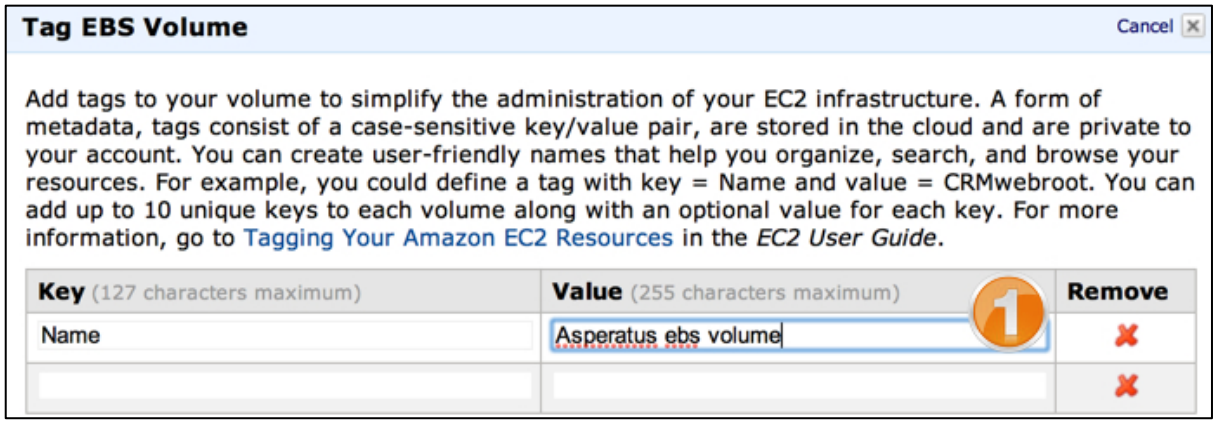
    (48) Click the **Tags** tab.

(49)Click **Add/Edit Tags**.

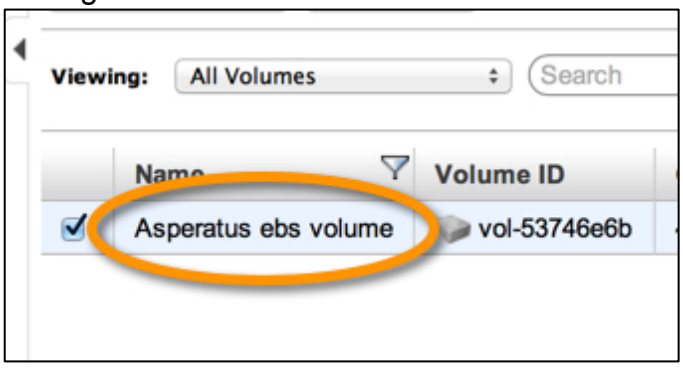


51. In the "Tag EBS Volume" dialog:

(50)For the **Name** key, type **Asperatus ebs volume** in the **Value** column.
(51)Click **Save Tags**.



Note: The name will now be displayed in the "Name" column in the EC2 Management Console.



"Availability zones" also called "zones", define EC2 instances and EBS volumes. By separating zones, you can provide fault tolerance, high availability and segmentation. However, if your volume is in a different zone than the instance

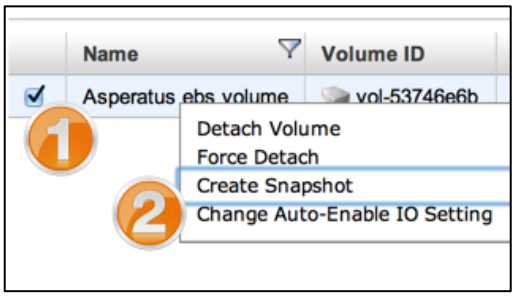
using it, the EC2 instance will not be able to attach to the volume. In the previous steps, the volume you created was placed in a different zone from the instance. In the example shown in this section, the EC2 instance is running in us-west-2c, but the EBS volume is running in us-west-2a. Because the volume is in a different zone from the instance, the volume must be migrated to the same zone as the instance. The following example shows how to take a snapshot of a volume, deploy that snapshot, and migrate it. To migrate your volume:

52. First, verify the zone for your instance.

    (52) In the EC2 Management Console, click **Instances > Instances**.
    (53) Select your running instance, and in the **Description** tab below, note the **zone**.

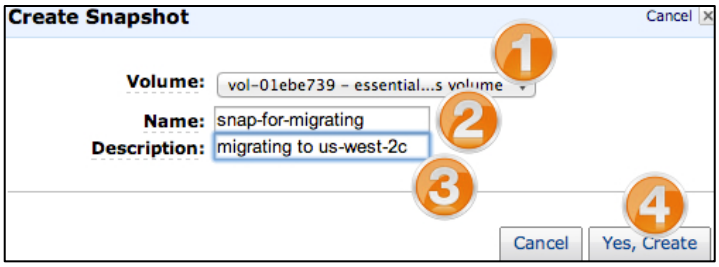
53. Next, take a snapshot of the volume.

    (54) Right-click the **essentials lab ebs** volume.
    (55) Click **Snapshots** listed under **Elastic Block Store.**
    (56) Choose **Create Snapshot**.

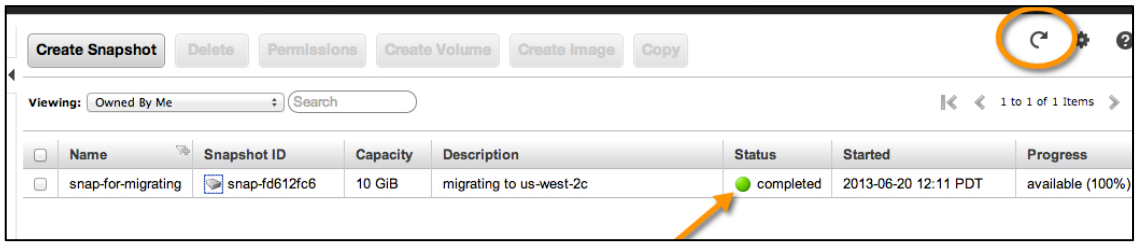


54. In the “Create Snapshot” dialog:

    (57) Verify the correct volume is selected.
    (58) For **Name**, type a value such as **snap-for-migrating**.
    (59) Type a **Description** such as **migrating to us-west-2c**.
    (60) Click **Yes, Create**. The snapshot will take some time to complete.

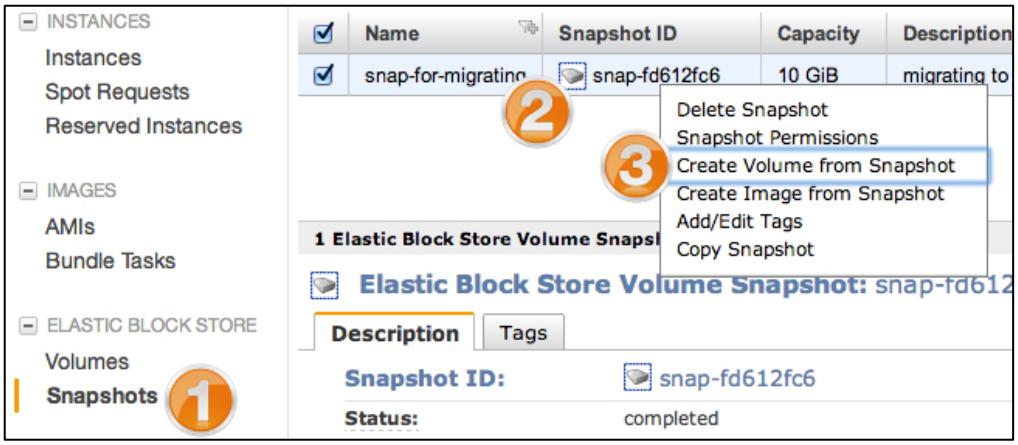


55. Click **Elastic Blockstore** > **Snapshots**.
56. Click the **Refresh** icon and verify the Status is “completed” before continuing.



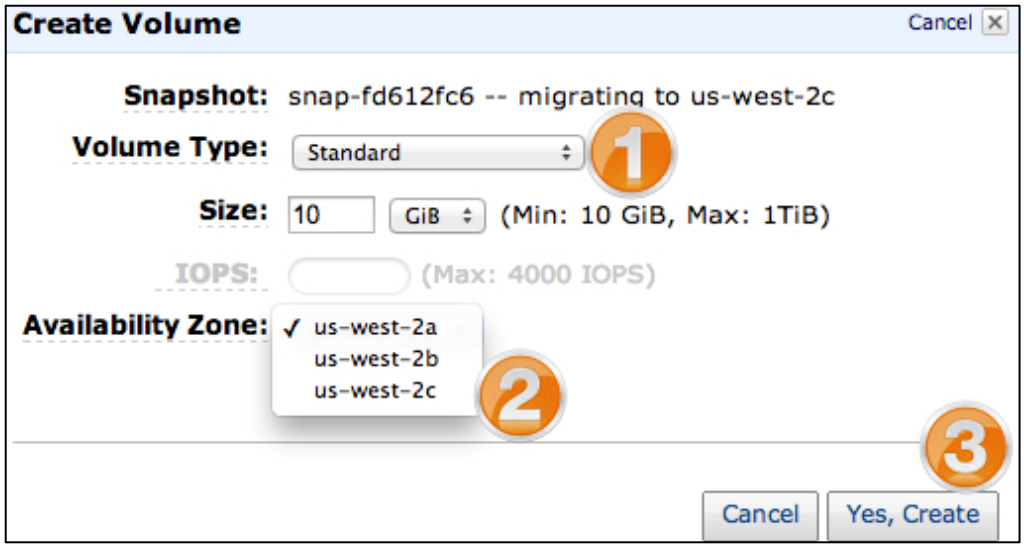
57. Deploy the snapshot to a new volume in the proper zone.

(61) Click **Snapshots** listed under **Elastic Block Store.**
(62) Right-click the **snap-for-migrating** snapshot.
(63) Choose **Create Volume from Snapshot**.



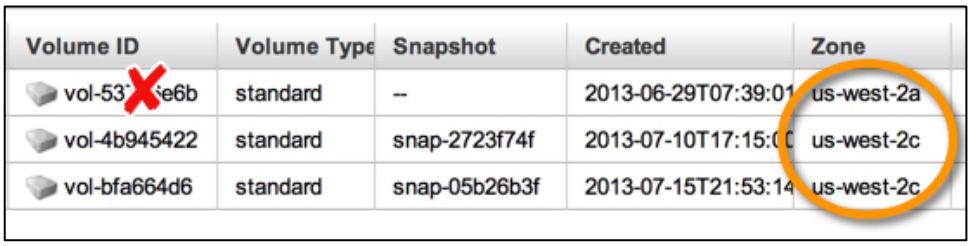
58. In the "Create Volume" dialog:

(64) For **Volume Type**, choose **Standard**.
(65) For **Availability Zone**, choose the zone that matches your EC2 instance's zone. Note: An example would be us-west-2c.
(66) Click **Yes, Create**.



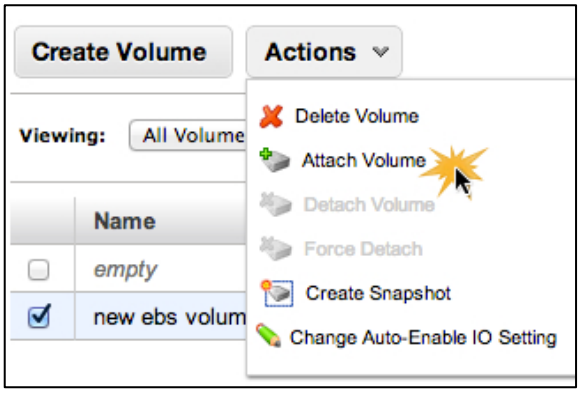
59. There are now 2 volumes in different zones. Delete the volume in the incorrect zone:

(67) Right-click the volume and choose **Delete Volume**.
(68) Click **Yes, Delete** to confirm.



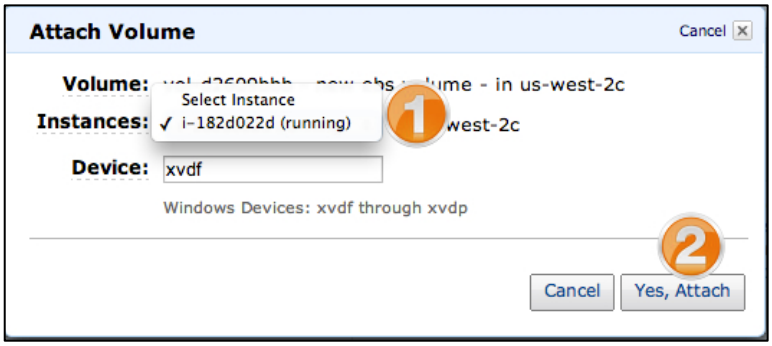
60. Attach the remaining volume to the instance.

    (69) Click **Elastic Block Store** > **Volumes**.
    (70) Select the new volume available.
    (71) Click **Actions**.
    (72) Click **Attach Volume**.



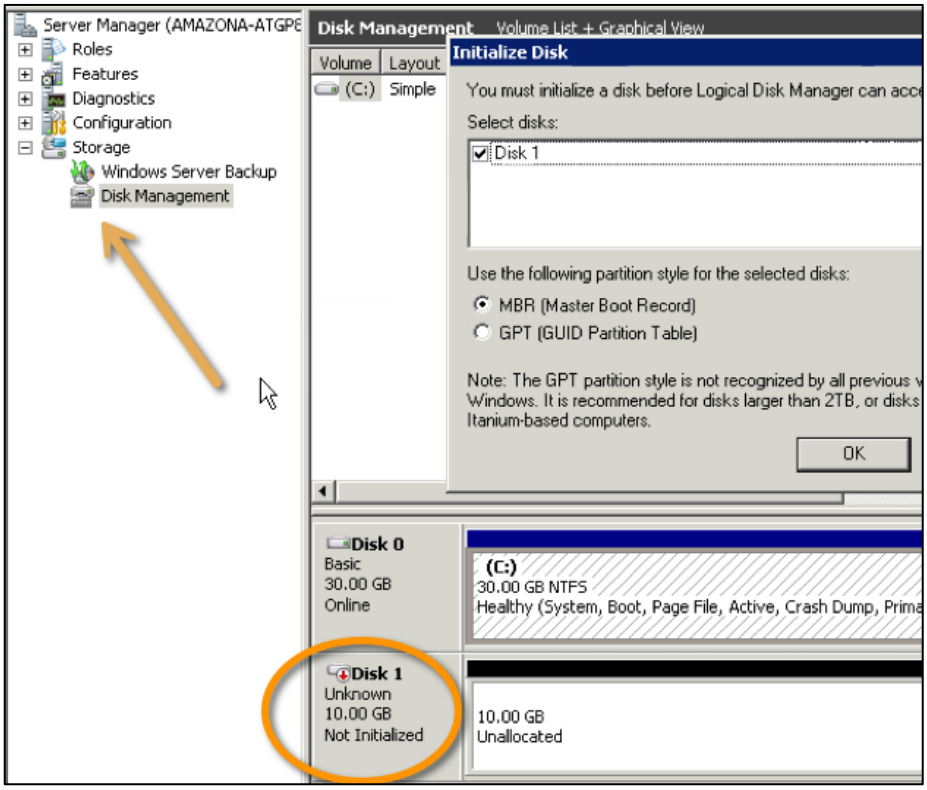
61. In the “Attach Volume” dialog:

    (73) Select your instance
    (74) Click **Yes, Attach**.

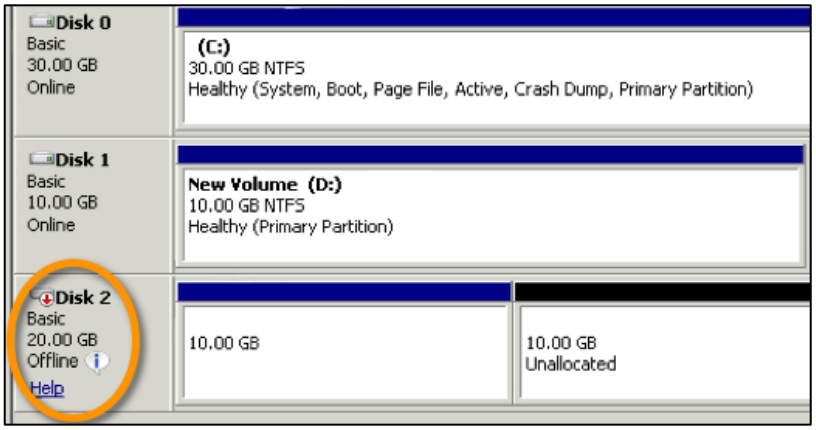


62. Validate the volume is attached to the instance by viewing the Attachment Information column in the Volumes view.

63. In Remote Desktop Connection, log on to Windows, open the disk management utility, and add and format the new disk. Further test the disk by creating a file on it.



64. Using previous steps, create a new snapshot of the disk and deploy a new volume from the snapshot. Increase the volume size to 20GB. Attach the new volume to your instance.
65. Logon to your Windows instance using Remote Desktop Connection and view the new volume in the disk management utility.

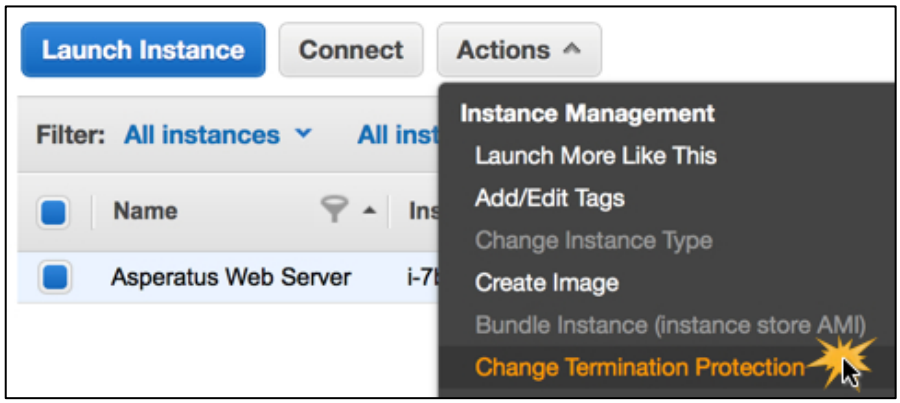


66. Extend the volume and explore it.

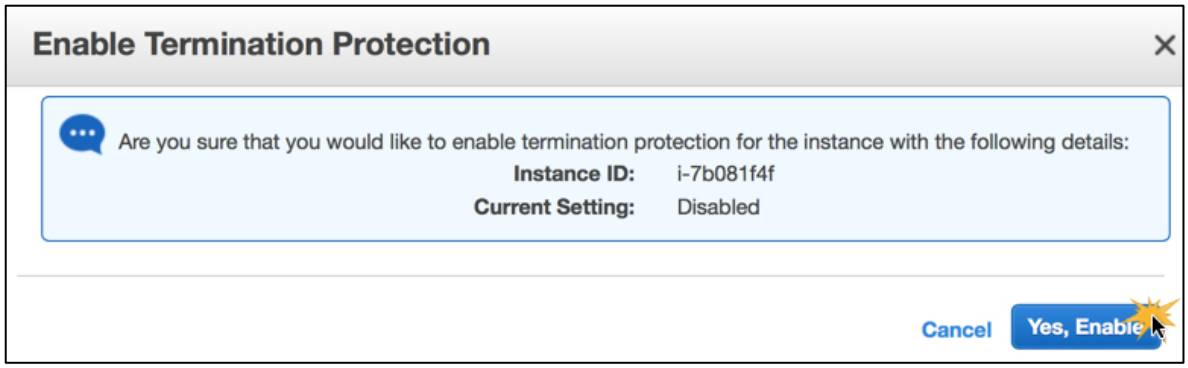
The Instance Lifecycle

Now that you, an Asperatus engineer, have your instances created and configured, images created and EBS volumes ready, you will need to know how to keep those instances from being deleted by accident. You will also want to know how to change the instance type, should you need to increate the amount of CPU and memory available to your server. In this section you change the EC2 Instance Type and protect it from termination.

67. If powered on, power off your instance either by shutting down from within Windows or by right-clicking your instance and choosing **Stop**.
68. Click **Yes, Stop** to confirm.
69. Right-click the instance, or click **Actions** with the instance selected, and click **Change Termination Protection**.



70. In the "Termination Protection" dialog, click **Yes, Enable** to enable termination protection.



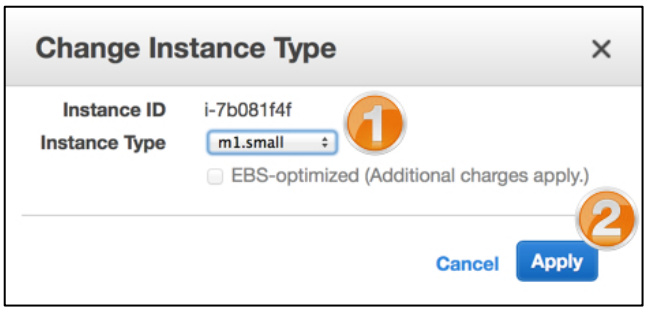
71. In the EC2 Management Console, select your instance and switch to the **Description** tab in the lower pane.
72. Note that **Termination Protection** is set to **Enabled**. Tip: You may need to scroll through the "Description" tab to view the termination protection settings.

Now, you are noticing that the instance is not as responsive as you would like. You need to increase it's performance. To do so, increase the instance size from micro to small:

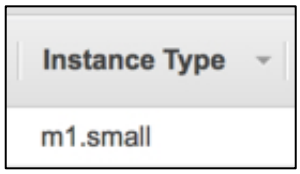
73. Right-click your instance and choose **Actions > Stop,** (or log in to the instance using previous steps as a guide, and shut it down within the operating system).
74. Again, right-click your instance and choose **Instance Management > Change Instance Type**.

75. In the “Change Instance Type” dialog:

    (75)For **Instance Type**, choose **m1.small**.
    (76)Click **Apply**.



76. In the EC2 Management Console, verify the **Type** column value is **m1.small**.



77. Right-click your instance and choose **Actions > Start.**
78. On the subsequent window, choose **Yes, Start.**
79. Also, in the EC2 Management Console, note the new instance DNS name.

Conclusion

Congratulations! You now have successfully:

- Created custom key pairs and security groups.
- Deployed a preexisting AMI with customizations and attached to it via remote desktop.
- Created a custom AMI.
- Created, attached, detached, migrated and took snapshots of EBS volumes.
- Modified an instance type, and worked with instance core functionality.

For feedback, suggestions, or corrections, please email: aws-course-feedback@amazon.com