Amazon Elastic Compute Cloud

Getting Started Guide

AWS Management Console

Table of Contents

What's New

The following table describes the important changes since the last release of the Amazon EC2 Getting Started Guide.

Change	Description	Release Date
Initial Release	initial release of the AWS Management Console Guide	March 11, 2008

Introduction

Amazon Elastic Compute Cloud (EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing easier for developers.

Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon's proven computing environment. Amazon 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. Amazon EC2 changes the economics of computing by allowing you to pay only for capacity that you actually use. Amazon EC2 provides developers the tools to build failure resilient applications and isolate themselves from common failure scenarios.

This Getting Started Guide is designed to teach System Administrators, Software Developers, and other IT professionals how to utilize the Amazon EC2 service through several easy tutorials. In this guide, we will demonstrate AWS Management Console that allows you to graphically interact with the Amazon EC2 service.

We have organized this Getting Started Guide into five tutorials, ranging from starting a new virtual server (instance) to using our advanced features. Each of these tutorials will provide you with the basics of how to use these components. Users are encouraged to leverage our additional resources (see the Related Resources Section) to learn about more advanced features of our system, like our APIs.

Users with minimal knowledge can interact with Amazon EC2 using AWS Management Console, but it is recommended to have basic understanding of web services. If you need to get familiar with this concept, please go to the W3 Schools Web Services Tutorial.

Amazon Elastic Compute Cloud is often referred to within this guide as "Amazon EC2" or simply "EC2"; likewise the Amazon Simple Storage Service is referred to in this guide as "Amazon S3"; all copyrights and legal protections still apply.

The AWS Management Console is accessed via https://console.aws.amazon.com

Setting up

Setting up an AWS Account

To use Amazon EC2, you must sign up for a AWS Account, sign up for Amazon EC2, and sign up for the Amazon Simple Storage Service (Amazon S3). These are three different actions that must be performed separately. For information on obtaining an AWS Account, go to the Amazon AWS Home Page (http://aws.amazon.com). For information on signing up for Amazon EC2, see Signing up for Amazon EC2. For information on signing up for Amazon S3, see Signing up for Amazon S3.

Signing up for Amazon EC2

To utilize the Amazon EC2 service, you will need to enable your AWS account for use with Amazon EC2. If you don't already have an AWS account, you will be prompted to create one as part of the sign up process. If you already have an Amazon EC2 account, you can skip this step. To sign-up for Amazon EC2 simply perform the following steps:

Go to the Amazon EC2 homepage (http://aws.amazon.com/ec2) in your web browser.

1. Click **Sign Up For Web Service** in the top right of the screen and follow the on-screen instructions.

Signing up for Amazon S3

Amazon EC2 AMIs are stored in and retrieved from Amazon S3. This means you will also need to sign up for Amazon S3. If you already have an Amazon S3 account, you can skip this step.

Go to the Amazon S3 homepage (http://aws.amazon.com/s3) in your web browser.

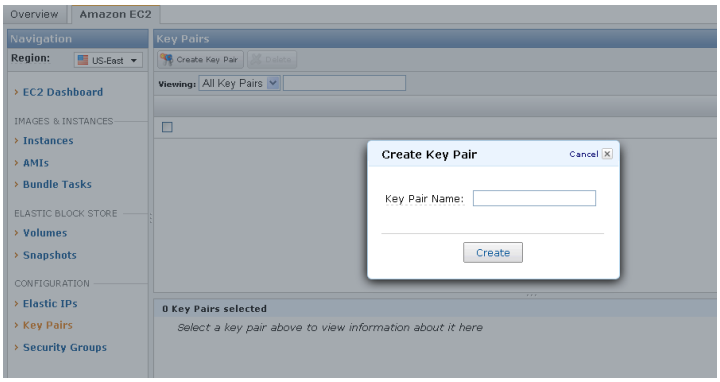
2. Click the **Sign up for this service** button.

Creating a KeyPair

An SSH keypair is used for several purposes including connecting to Linux/OpenSolaris instances and retrieving your Windows Administrator password. To generate a keypair, simply perform the following steps:

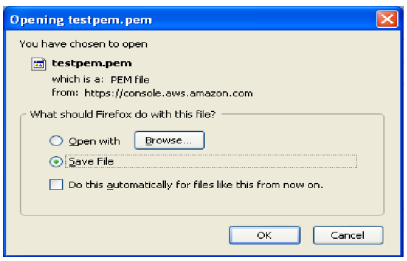
Logon to AWS management console https://console.aws.amazon.com and click on the “Key Pairs” in the navigation menu.

1. Click on “Create Key Pair” button.



3. Type in a name for your KeyPair, and click the “Ok” button.

4. You will be prompted to open or save the .pem file. Select Save and them move to a secure file location



5. Your KeyPair will now be created; remember this name, because you will use it later.

Tutorial #1: Running an Instance

Goal

The goal of this tutorial is to launch an Amazon EC2 virtual server, what we call an instance. This tutorial assumes that you have completed the necessary setup as described above.

Core Concepts

The sections below outline the core concepts used in this tutorial.

Amazon Machine Image (AMI)

An Amazon Machine Image (AMI) is an encrypted file stored in Amazon S3. It contains all the information necessary to boot instances of your software. It is somewhat analogous to a snapshot of the boot partition containing the operating system and installed software running on your server.

Amazon EC2 Instance

A running server instantiated from an AMI is referred to as an instance. All instances launched from the same AMI will create a nearly identical running server (except for the IP range or computer name). Note that an instance is ephemeral and that any information on it is lost when it is terminated or if it fails. The Elastic Block Storage feature described in Creating an Elastic Block Store (EBS) Volume of this tutorial can be used to create long term storage for data produced by the instance.

Security Group

The security group is analogous to a firewall that can block all incoming (ingress) and outgoing (egress) traffic that does not come in on a specific IP (specified by a CIDR) or port number range. For more information on CIDRs, please visit http://en.wikipedia.org/wiki/CIDR. Each EC2 instance can be a member of up to 100 security groups. Group membership cannot be changed while an instance is running, but the rules within the group can be changed, and will take effect immediately. Multiple security groups can be used to create secure, multi-tiered systems. For example, consider the classic three-tier model consisting of web, application, and database servers. Placing each tier of servers in a distinct security group allows the web server to be accessible externally, with controlled access to the other tiers on an as –needed basis.

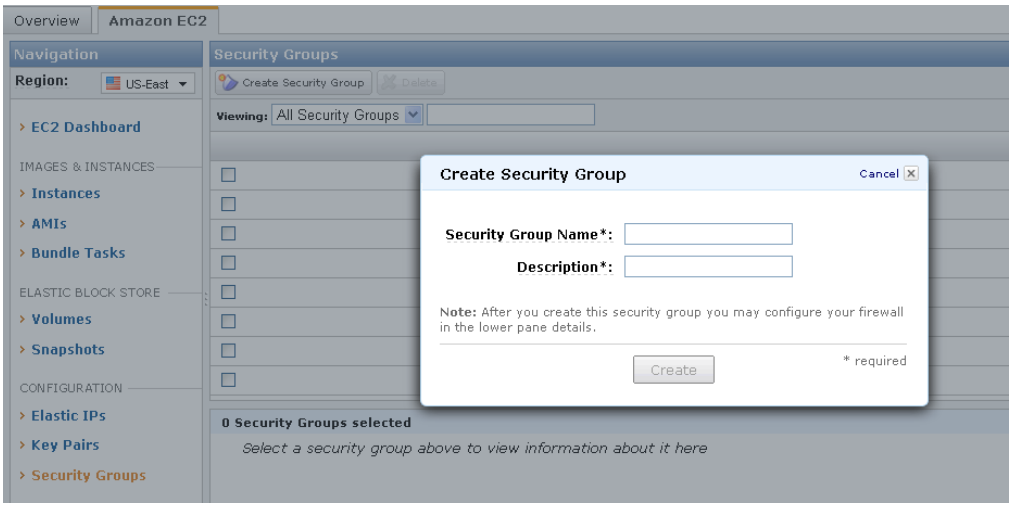
Step 1: Setting up a Security Group

Every launched instance requires that you have a security group defined to specify what network traffic is allowed to reach the instance. By default we do not enable any traffic. If you have already defined a security group to allow network traffic from your address you can skip this step.

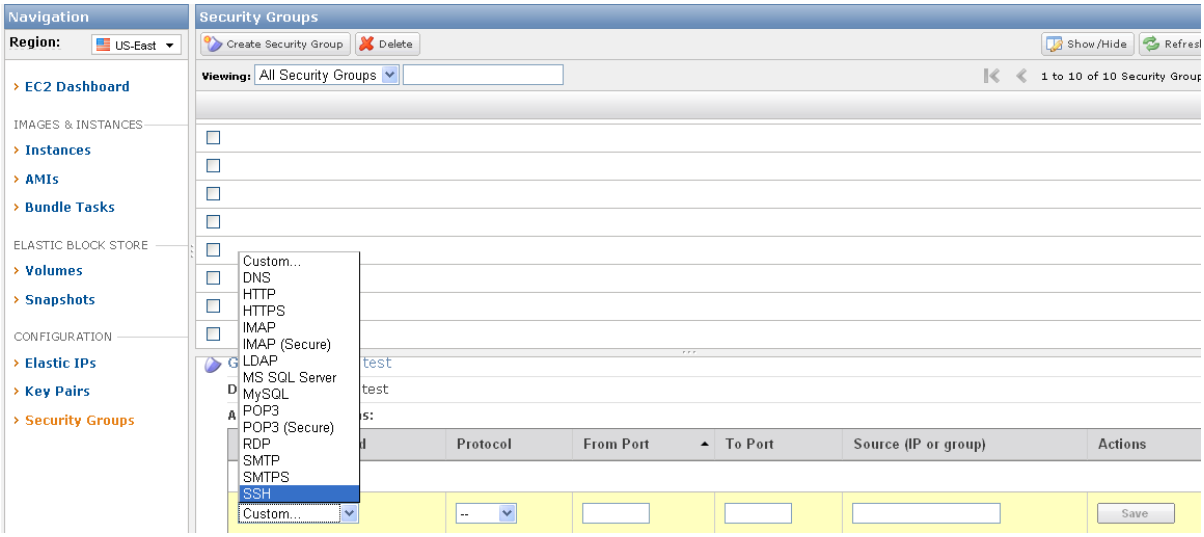
For the purposes of this tutorial, we are going to create a new group called “All Incoming”, that will allow any incoming traffic to connect to any launched instances on SSH port (22), HTTP port (80), and RDP port (3389). This will enable us to connect to either a Linux/UNIX or Windows instance. To open these ports for testing purposes, please complete the following steps:

Launch AWS Management Console by using the browser to go to location https://console.aws.amazon.com.

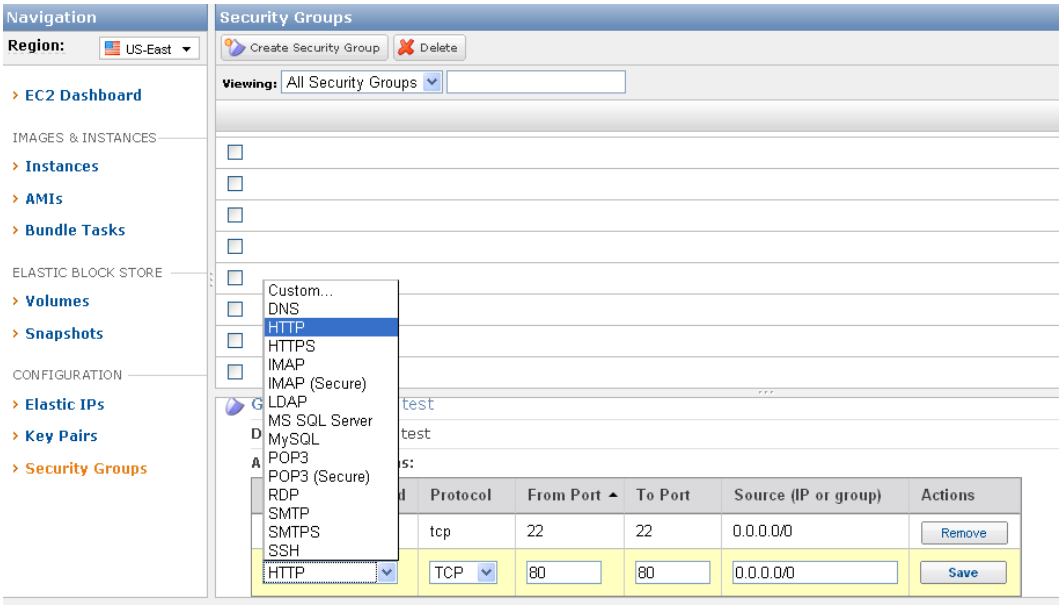
1. Click on the “Security Groups” in the Navigation menu and then press the “Create Security Group” button and enter a name and description.



3. Enter the allowed connections by selecting connection method SSH and press save. This will allow access from an ssh client to logon to the instance.



5. Select the HTTP protocol from the connection method dropdown and press save. This will allow HTTP access via port 80 to the instance.



6. Now you will have your security group setup.

Caution

In this example, you enable access to port 80 of the instance for all hosts on the Internet - "0.0.0.0/0". Although this might be acceptable for testing or demo purposes, it is extremely unsafe for production environments. For production systems, you must obtain your public IP address ranges and grant access to those ranges only. For example, if your IP address is 123.123.123.123, you specify "123.123.123.123/32". For more details on controlling network security groups, see the Amazon EC2 Developer Guide.

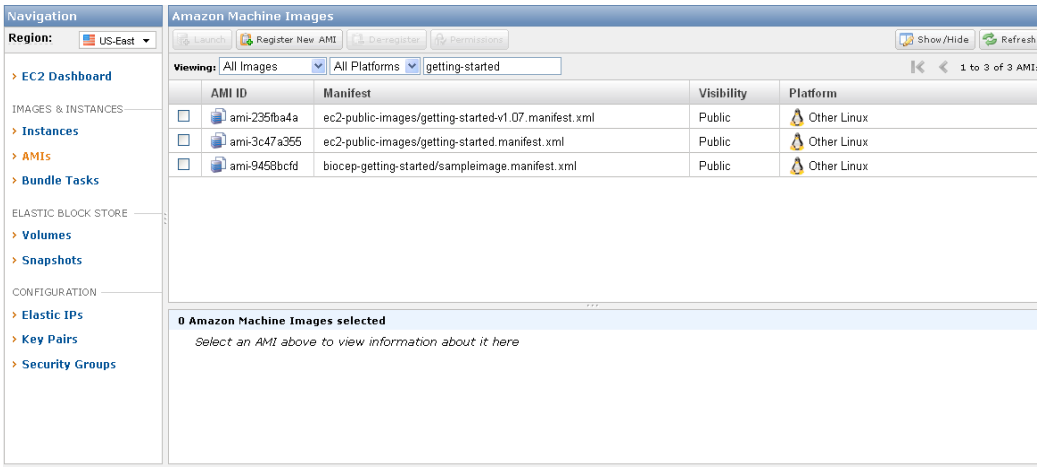
Step 2: Choosing an AMI

Amazon Web Services, software companies, and the community provide many different AMIs to use. You can leverage any of the publicly visible ones to start an instance. A list with more description about many of the paid and public AMIs can be found at: http://developer.amazonwebservices.com/connect/kbcategory.jspa?categoryID=171 and http://developer.amazonwebservices.com/connect/kbcategory.jspa?categoryID=101 respectively. For the purposes of this tutorial, we will use a pre-configured AMI called "ec2-public-images/getting-started".

To select this AMI, please perform the following steps:

1. Click on the “AMIs” in the Navigation menu.

2. In the last text box of the line with label “Viewing”, type in “getting-started” if you want to launch a Linux/UNIX instance or “windows” if you would like a Windows instance. This will search for the image that contains the name “getting-started” or “windows”.

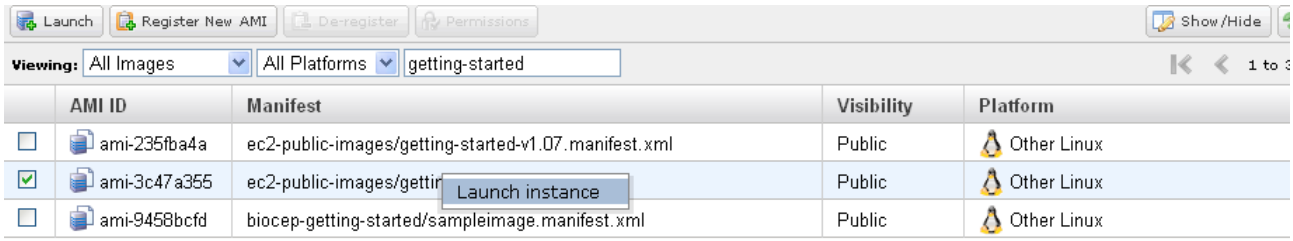


3. Click on the image with the Manifest named “ec2-public-images/getting-started.manifest.xml” for Linux/UNIX or “ec2-public-windows-images/Server2003r2-i386-anon-v1.00.manifest.xml” for Windows.

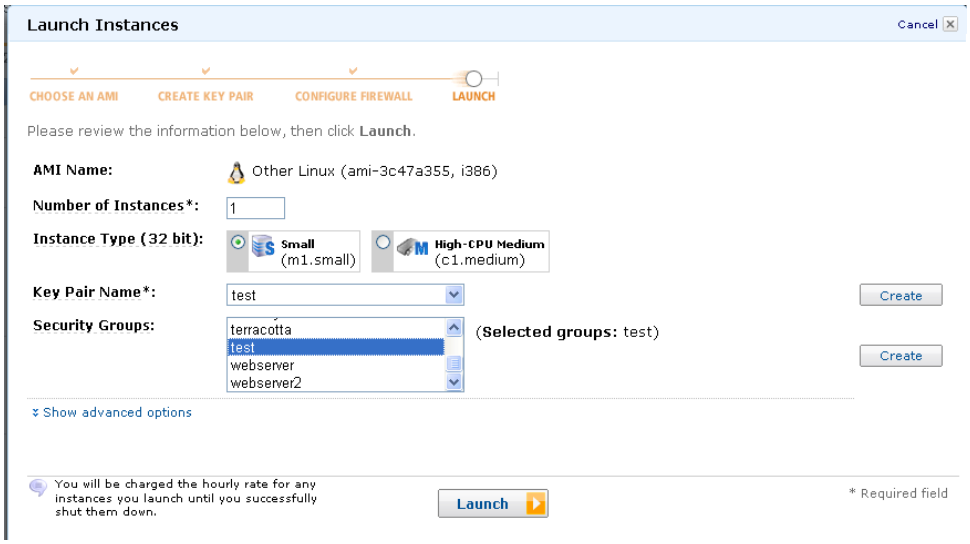
Step 3: Launching an Instance

Once you have the AMI selected, you can easily launch an instance by performing the following:

1. Right click on the selected AMI and click on the “Launch Instance(s) of this AMI”.



2. In the Number of Instances section, enter 1. This enables multiple instances to be started.

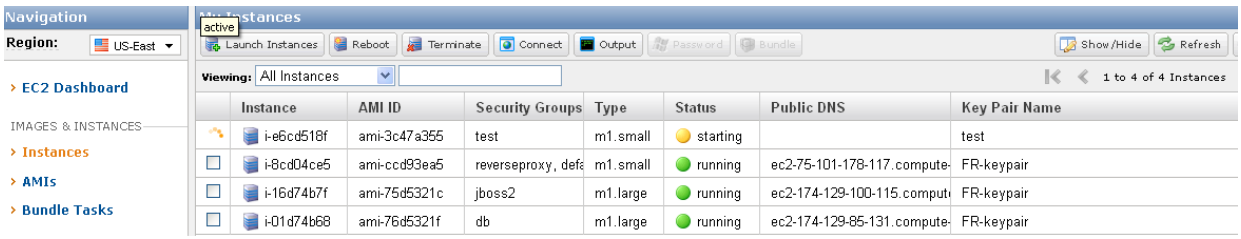


3. In the KeyPair section of the pop-up box, select the KeyPair you created in the setup section. This will associate your security KeyPair with an instance, so that you can connect to it.
4. In the Security Groups section, select the appropriate group and click on the right arrow to move it into the Launch in box.

Note: There are additional options available when launching an instance. These can be set by clicking on “Show advanced options”.

5. Click the “Launch” button.

6. Click “Instances” in the navigation menu and the instance will show up in the list in the “starting” state. After about a minute or two it should reach the “running” state, otherwise press the refresh button. If you are running a Linux/UNIX instance, it will be ready to use. If you are using Windows, please right click on the instance, and select “Show Console Output”. When the text “Windows is ready to use” appears, your instance is ready to use!



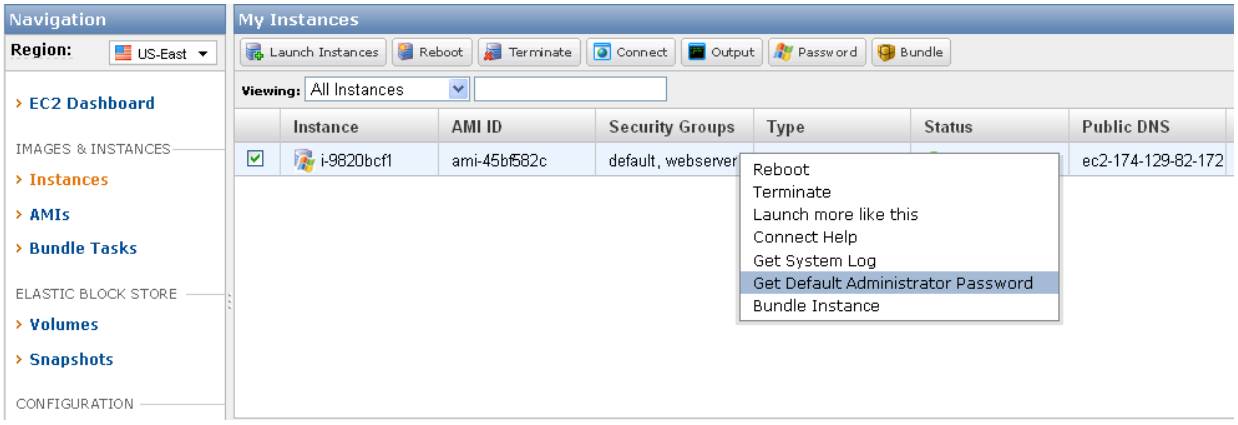
Step 4: Connecting to the Instance

Linux/UNIX Instance

To Connect from a windows client:
an ssh client needs to be installed. See the Appendix 1 on installing, configuring and using PuTTY.
To connect from a linux client:
ssh -i xxx-keypair root@ec2-XXX-XXX-XXX-XXX.z-2.compute-1.amazonaws.com

Windows Instance

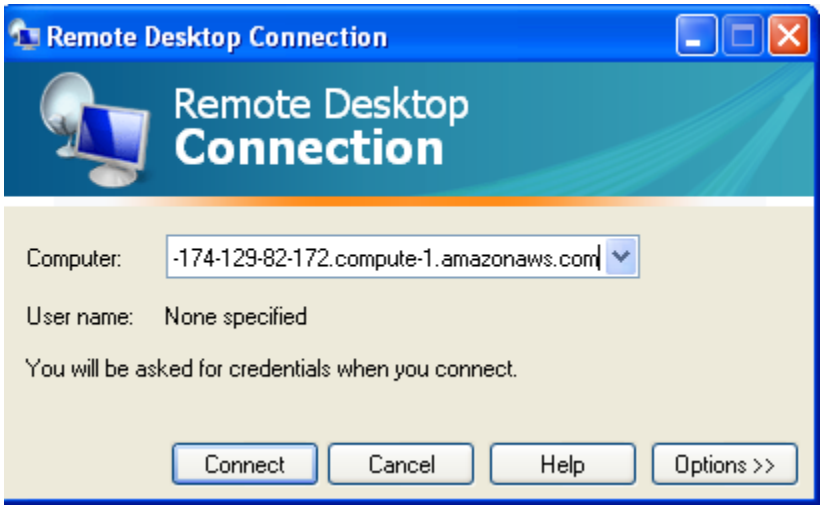
1. Get the Windows Administrator by right clicking on the instance in the AWS Management Console and clicking “Get Default Administrator Password”.



2. Next you need to copy the contents of the keypair.pem file generated earlier and press Decrypt Password to see the unencrypted password.

3. Copy the unencrypted password to somewhere safe.

4. On your windows machine go into remote desktop and logon to the machine using the Public DNS.



5. Login as user “administrator” with the unencrypted password.

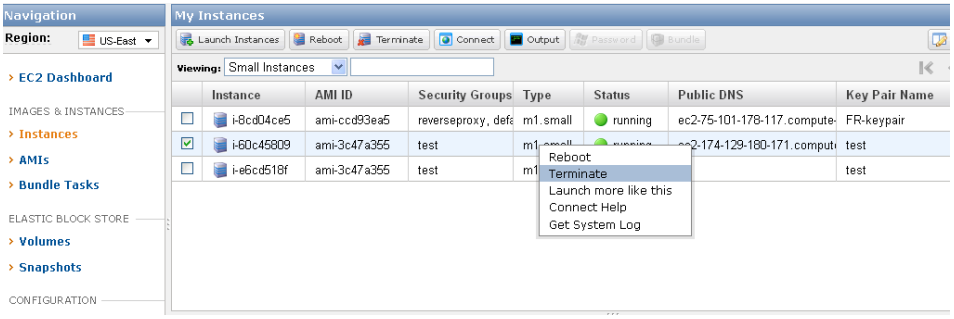
Note: Your security group must allow access to the RDP or SSH port.

At this point, you have now successfully gained access to your new instance!

Cleaning Up: Terminating your Instance

When you are done with your instance, you can simply terminate it by:

1. On the “Instances” navigation menu option, right click on the instance you want to terminate and select “Terminate”.



2. After a minute the state should change to “terminated”.

Tutorial #2: Bundling an Instance into an AMI

Goal

The goal of this tutorial is to bundle an instance into an AMI. This is helpful so that you can customize an instance with any software or changes you may need to make. Then, at a later point you can launch one or more copies of the AMI.

Core Concepts

The sections below outline the core concepts used in this tutorial.

S3 Bucket

S3 is a persistent data store that enables you to store objects, like an AMI. The bucket is similar to a folder on a file system so that you can keep your objects organized. The bucket name though is unique across all S3 users.

Bundling

Bundling is a method of taking a snapshot of the file system, so you can later boot from it. You can make a new AMI by modifying and extending an existing image (such as the one you just booted and logged onto in Tutorial #1), and then bundling it to use later.

Windows Operating Systems

Step 1: Make any Modifications

The first step in creating any AMI is to make any modifications to your running instance. To create a running instance, please follow the steps provided in the tutorial #1. For the purposes of this tutorial, please just add a file named test.txt to your “c:\” directory. Once you have added this file, please proceed to the next step.

We also suggest that you perform the following steps to reduce your startup time, by clearing up any temporary files on your instance, defragmenting your system, and explicitly “zeroing” out your free space using the ‘sdelete’ command. Your startup time is proportional to the size of your AMI.

-We recommend using the Disk Cleanup tool to remove unneeded temporary files from your instance, since it provides easy-to-use wizards that remove the files for you. To access the Disk Cleanup tool, click Start, click Run, and then type “Cleanmgr.exe”.

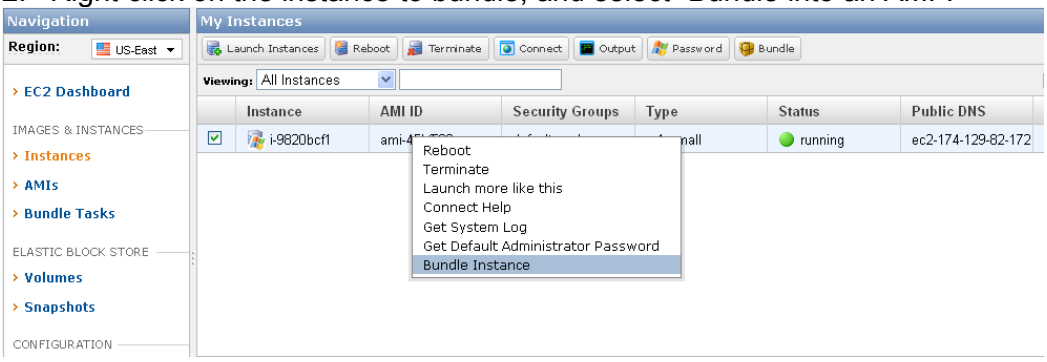
-To defragment your machine, we recommend the Disk Defragmenter utility included with Microsoft Windows. To access that tool you can click Start, click All Programs, click Accessories, click System Tools, click Disk Defragmenter.

To “zero” out your free space, you can simply use the “sdelete –c C:\” command. For full details on how to use the sdelete command, please see http://technet.microsoft.com/en-us/sysinternals/bb897443.aspx.

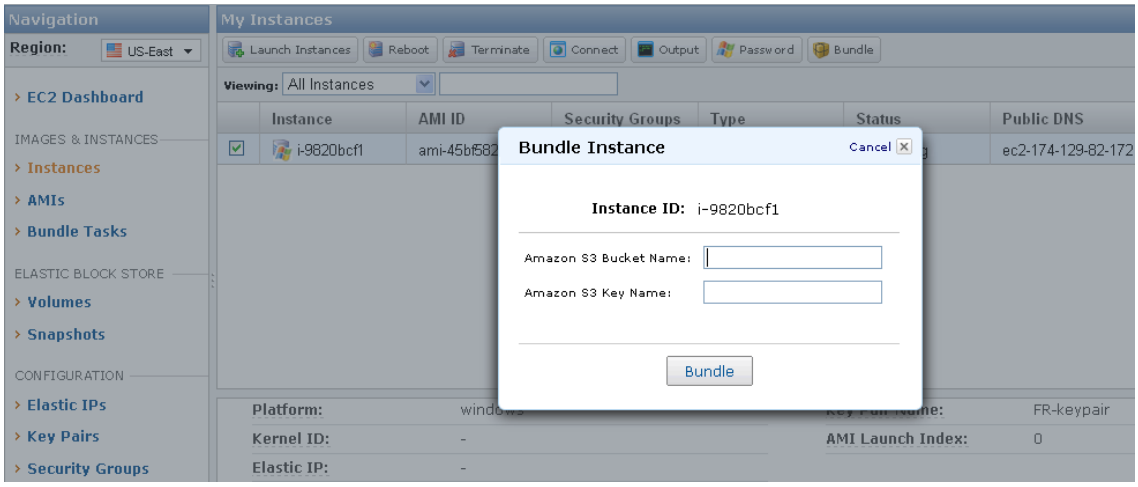
Step 2: Bundle the Image

Once your modifications have been made to a running instance, you can bundle your instance into an AMI. This will automatically shut down your instance, take a snapshot, and restart it for you. To take the snapshot:

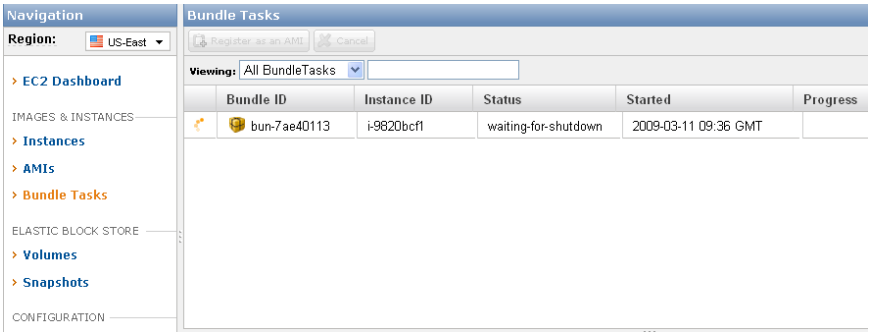
1. Click on “Instances” in the AWS Management Console Navigation Menu
2. Right click on the instance to bundle, and select “Bundle into an AMI”.



3. Enter in a S3 bucket name to store the AMI in and a name for the image, then click “Bundle”.



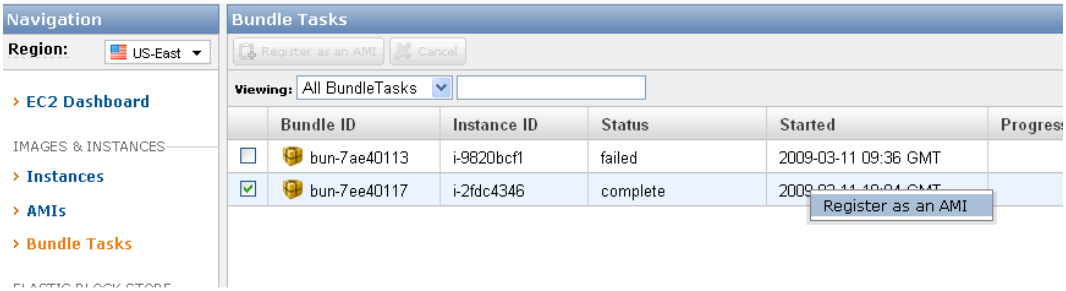
4. Go to Bundle Tasks and wait for the task to complete. This may take a while.



Step 3: Register the Image

Once the image has reached the "completed" state in the "Bundle Tasks" tab, you just need to register the AMI to make it available for you to use. To do this:

1. Right click on the bundle to register, and select "Register as an AMI". Then on the pop-up screen press "Register".



Linux/UNIX Instance

Bundling a Linux/UNIX instance requires the use of the AMI tools, and differs from Windows in that the bundling is performed from within the running EC2 instance.

Step 1: Upload Access Identifiers.

Get you access identifiers from http://aws-portal.amazon.com/gp/aws/developer/account/index.html?action=access-key. Copy the access key and secret access key and download the certificate to your PC.

1. Upload the key (the .pem file from earlier) and the certificate file to your instance and store in the /root folder.

Note: From a Windows PC download WinSCP (http://winscp.net) to use to upload files

Step 2: Bundle the Image

Once your modifications have been made to a running instance, you can bundle your instance into an AMI. In the instance:

1. Create a directory for the bundle: mkdir /mnt/bundle.

2. Create a bundle using the following command:
   ec2-bundle-vol -d /mnt/bundle -k /root/pk-<key>.pem -c /root/cert-<cert>.pem -u <aws user number>
   NOTE: /root/pk-<key>.pem and /root/cert-<cert>.pem were uploaded earlier.

3. Upload the bundle to S3:
   ec2-upload-bundle -b <ami folder>/ -m /mnt/bundle/image.manifest.xml -a <access key> -s <secret access key>
   NOTE: <access key> and <secret access key> copied earlier

Step 3: Register the Image

1. In AWS Management Console Register go to “AMIs” in navigation menu
2. Press “Register New AMI” and enter <ami folder>/image.manifest.xml

Tutorial #3: Creating an Elastic Block Store (EBS) Volume

Goal

The goal of this tutorial is to create a new EBS volume and attach it to your running EC2 instance.

Core Concepts

Amazon Elastic Block Store (EBS)

Amazon Elastic Block Store (EBS) provides block level storage volumes for use with Amazon EC2 instances. It provides highly available, highly reliable storage volumes that can be attached to a running Amazon EC2 instance and exposed as a device within the instance.

Amazon EBS Volume

Amazon EBS volumes are off-instance storage that persists independently from the life of an instance. You can create storage volumes from 1 GB to 1 TB that can be mounted as devices by Amazon EC2 instances. Multiple volumes can be mounted to the same instance. Amazon EBS volumes are placed in a specific Availability Zone, and can then be attached to instances also in that same Availability Zone. Each storage volume is automatically replicated within the same Availability Zone. This prevents data loss due to failure of any single hardware component.

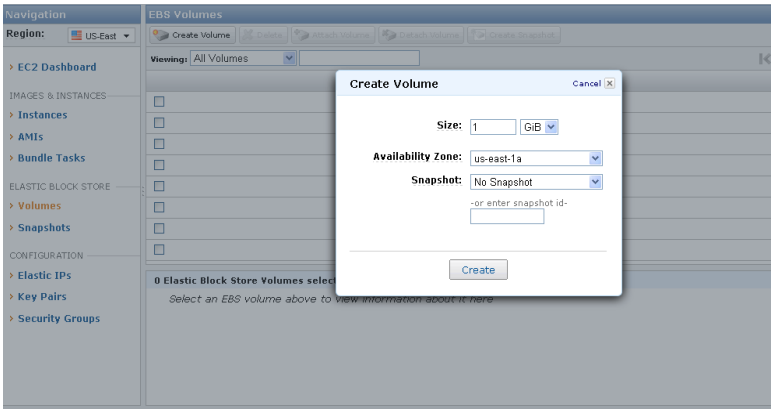
Amazon EBS Snapshot

Amazon EBS also provides the ability to create point-in-time snapshots of volumes, which are persisted to Amazon S3. These snapshots can be used as the starting point for new Amazon EBS volumes, and protect data for long-term durability. The same snapshot can be used to instantiate as many volumes as you wish.

Step 1: Create a New Amazon EBS Volume

To create a new Amazon EBS Volume, please perform the following steps:

1. Click “Volumes” in the navigation menu and press the “Create Volume” button.



2. Type in the size in gigabytes (GB) of the volume you would like between 1 and 1000. For the purposes of this demo, please use “50” to specify a 50 GB drive.

3. Specify the availability zone and “no snapshot”.

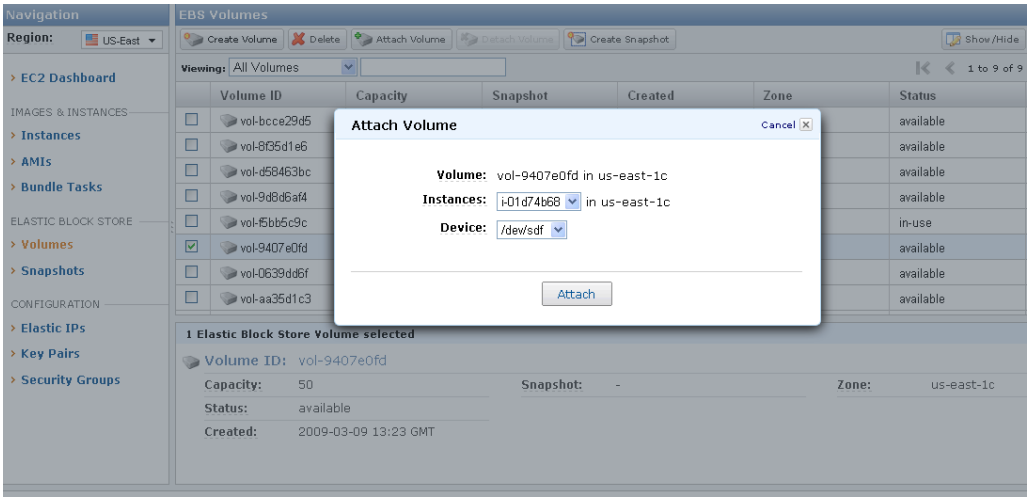
Note: A volume must be in the same availability zone as an instance. You can optionally specify an Availability Zone if desired or the snapshot id to create a volume from. If you accidentally create a volume in the wrong zone, you can either create a new blank volume in the proper zone, or create a new volume based on a snapshot of a volume in another zone.

4. Click the “Create” button, and your Amazon EBS volume will be created. Take note of the volume id of the Amazon EBS volume you just created, because we will use it later. Also notice the volume goes from “creating” status to available” status.

Step 2: Attach the Amazon EBS Volume with an Instance

If you do not have an instance running, please start one as defined in Tutorial #1. Once you have created the EBS volume, you can attach it to one of the instances by performing the following steps:

1. Click “Volumes” in the navigation menu, select the volume and press attach.



2. Select the instance Id of the running instance.

3. In the “device” box, you will need to input the device you would like to attach to the volume.

   The device must be in the format of “/dev/sdh” for Linux/UNIX. The device is automatically selected for Windows (in most cases, the Device field is disabled and displays the text “windows_device”). Linux/UNIX instances currently support devices “sdf” to “sdh”. Any device that is not reserved can be attached to an Amazon EBS volume. For a list of devices that are reserved by the instance stores, see Instance Storage in the developer guide.

4. Press “Attach” when completed. This will attach the new Amazon EBS volume to your specified instance.

Step 3: Formatting an Amazon EBS Volume

Windows

To format a volume for Amazon EC2 running Windows, please perform the following steps.

1. Log in to your instance using Remote Desktop.
2. Select Start and click Run.
3. Type diskmgmt.msc and click OK. The Disk Management utility opens.
4. Right-click the Amazon EBS volume, select Initialize, and follow the on-screen prompts.

Linux/UNIX

To format your volume for Amazon EC2 running Linux/UNIX, please perform the following steps.

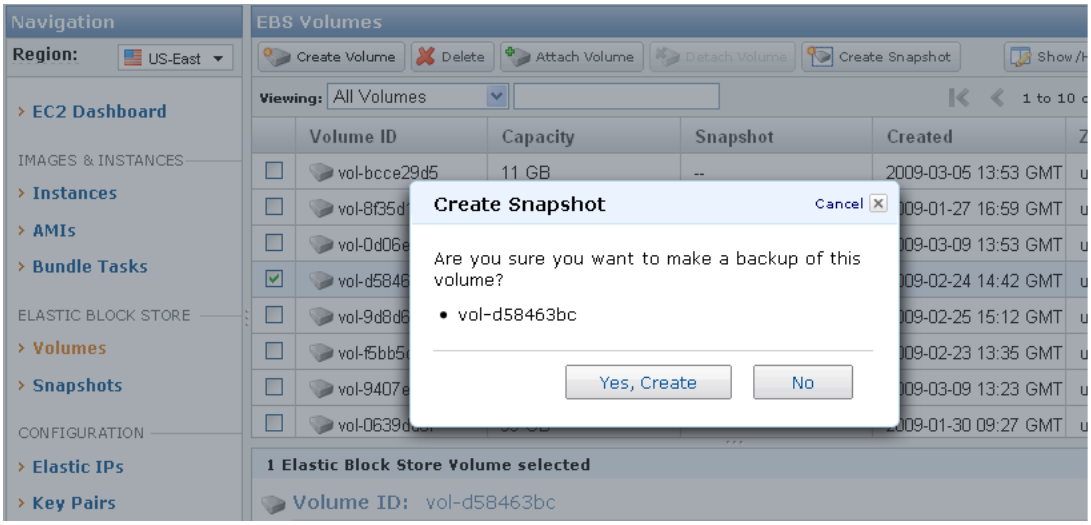
1. Log into your instance using ssh.
2. Use the mk2fs tool to format the EBS volume.

Step 4: Taking a Amazon EBS Volume Snapshot

Once you have formatted a volume and made any necessary changes, you can take a snapshot of the volume so that you have a point in time record of the block storage. The snapshot occurs asynchronously and the volume's status indicates "pending" until it completes. To complete this snapshot, please perform the following steps:

1. Click on “Volumes” in the navigation menu.

2. Right click on the EBS volume that corresponds to the device and instance id you recorded, and select “Create a snapshot from volume”. Then press “Yes,Create” in the pop-up. This will kick off a snapshot task, which will go to “completed” status when done. You may need to click the refresh button to get an updated status.



Clean-up

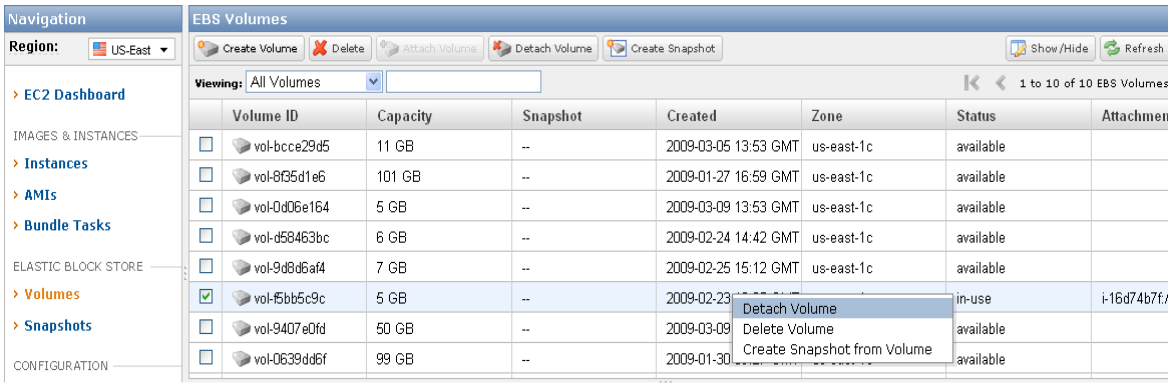
To ensure that you remove any components created from this tutorial, please shutdown any instances as defined in Tutorial #1 and perform the following two steps.

Step 1: Detaching and Deleting the EBS Volume

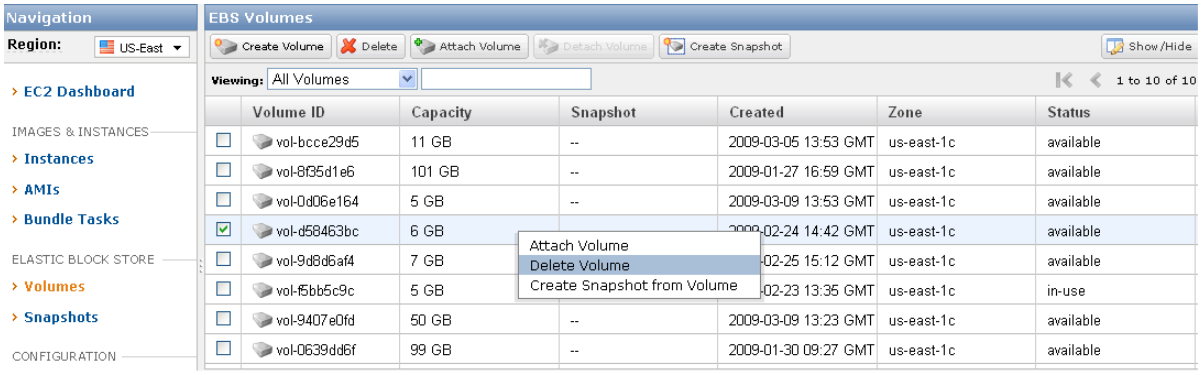
Once you have completed the tutorial you will want to detach and your volume from your EC2 instance. To accomplish this, please perform the following steps:

1. Click on “Volumes” in the navigation menu.

2. Right click on the Amazon EBS volume that you want to detach, and select the “Detach this volume” option and press “Yes, Detach” in the pop-up. This will disassociate the Amazon EBS volume from the instance.



3. Right click on the Amazon EBS volume that you want to delete, and select the "Delete Volume" option. This will delete the Amazon EBS volume.

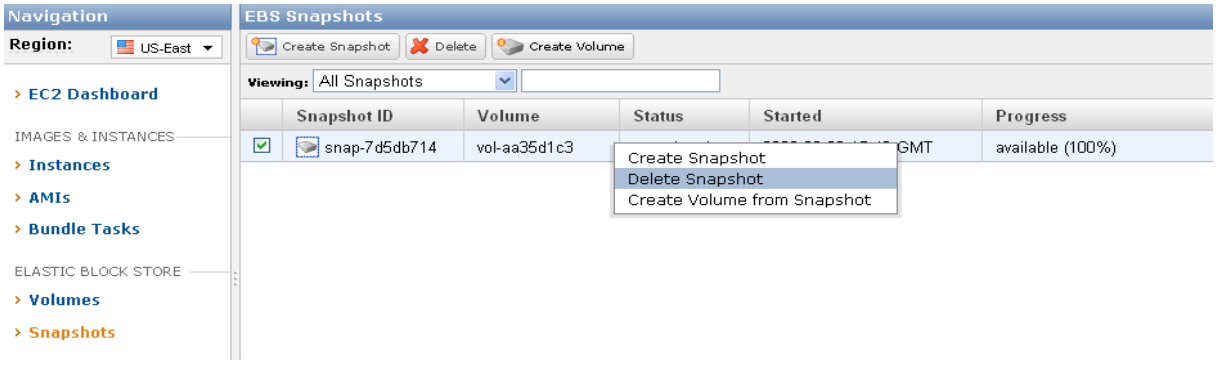


Step 2: Deleting any EBS Snapshots

To delete the Amazon EBS snapshot, please perform the following steps:

1. Click "Snapshots" in the navigation menu.

2. Right click on the snapshot you would like to delete, and select "Delete Snapshot".



Tutorial #4: Associating an Elastic IP (EIP) with an instance

Goal

The goal of this tutorial is to create a new elastic IP (EIP) and map it to an instance. Elastic IPs are important to provide a static IP associated with an EC2 instance.

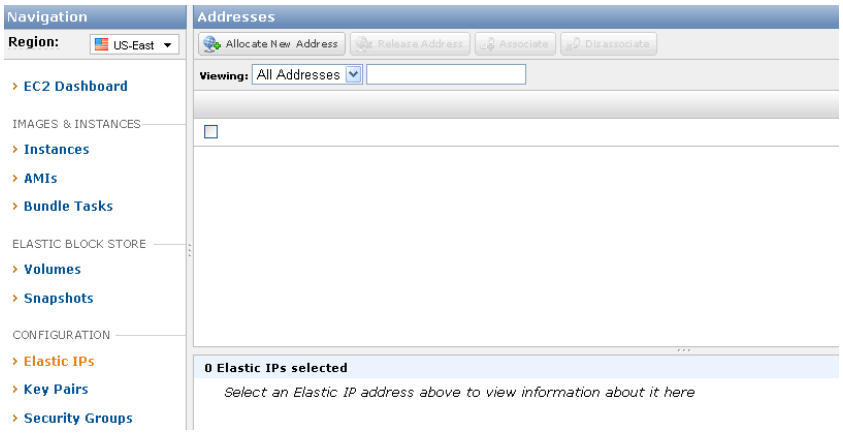
Core Concepts

Elastic IP (EIP)

Elastic IP addresses are static IP addresses designed for dynamic cloud computing. Elastic IP addresses are associated with your account, not specific instances. Any elastic IP addresses that you associate with your account remain associated with your account until you explicitly release them. Unlike traditional static IP addresses, however, elastic IP addresses allow you to mask instance or availability zone failures by rapidly remapping your public IP addresses to any instance in your account. You can only associate one elastic IP address with one instance at a time.

Step 1: Creating a new EIP

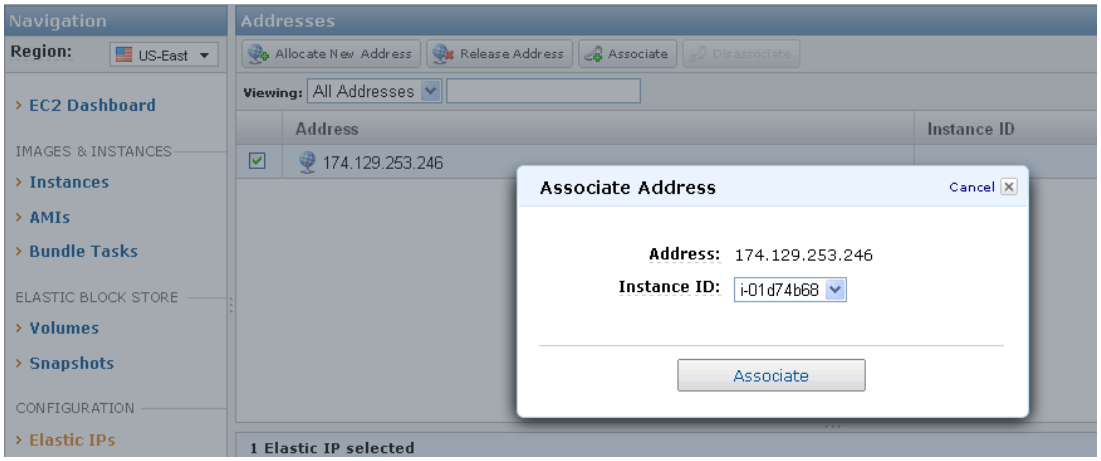
Select “Elastic IPs” from navigation menu and press the “Allocate New Address” button.



Step 2: Associating the EIP with a new instance

If you do not have an instance running, please start one as outlined in Tutorial #1. Once you have created an EIP, you will want to associate the IP with an instance. To accomplish this, please perform the following steps:

1. Select “Elastic IPs” from navigation menu, then select the IP address and press the “Associate” button.

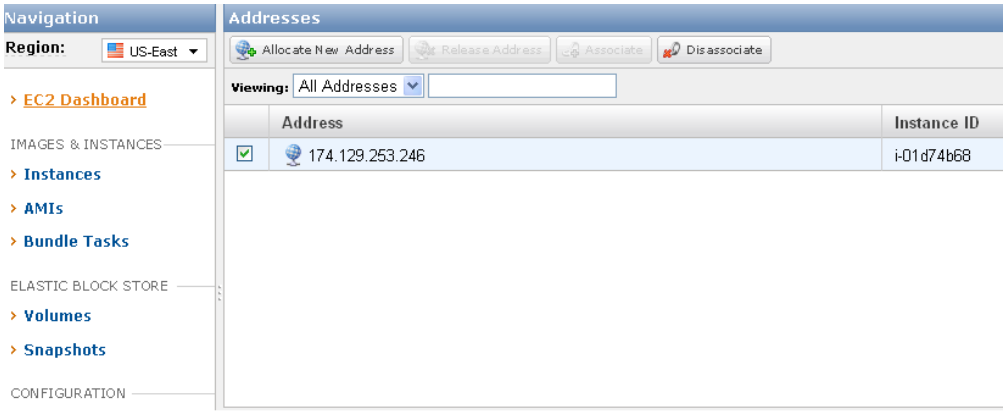


2. Choose the instance ID that is to have the Elastic IP Address and press Associate button

Clean-up

To ensure that you remove any components created from this tutorial, please shutdown any instances as defined in Tutorial #1 and perform the following steps:

1. Select “Elastic IPs” from navigation menu, then select the IP address and press the “Disassociate” button.



2. Select the IP Address again and press the “Release Address” button. This will relinquish your IP address, so that it can be reclaimed by any other customer.

Appendix #1: Putty

Introduction

PuTTY is a free SSH client for Windows. Other tools that form part of the PuTTY suite are PuTTYgen, a key generation program, and pscp, a secure copy command line tool. This guide outlines the additional steps required to use PuTTY with Amazon EC2.

Step 1: Install Putty

Download putty from http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html.
Download the zip file or the windows installer file as puttygen and Putty are both required.

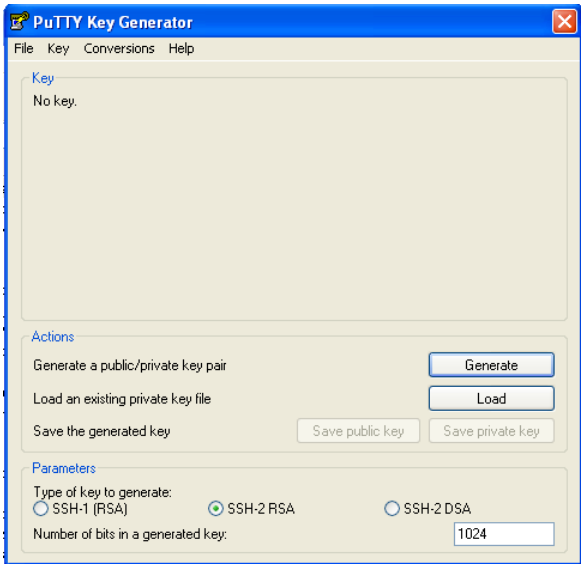
1. Unzip or install in windows.

Step 2: Generate Private Key Format

PuTTY does not natively support the private key format generated by Amazon EC2. Fortunately, PuTTY has a tool called PuTTYgen, which can convert keys to its internal format.

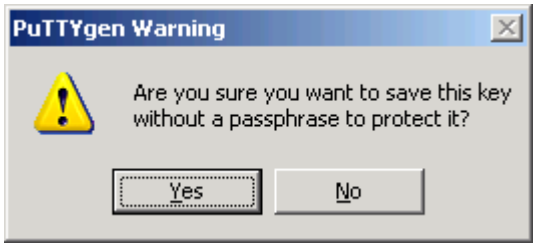
Note: You should have generated a .pem file in the section ***Creating a KeyPair*** And downloaded it to your windows machine and saved it in a file something like test.pem.

1. Launch PuTTYgen and press load and select the .pem file.



2. PuTTYgen should pop-up the following message:

3. PuTTYgen displays a lot of information regarding the key that has been loaded, such as the public key, the key passphrase, the type and the number of bits in the generated key. The keys generated by Amazon EC2 are 1024 bit SSH-2 RSA keys. They are also passphraseless. A passphrase on a private key is an extra layer of protection, so even if your private key is discovered it will not be usable without the passphrase. The downside is that it makes automation harder as human intervention is needed to log on to an instance, or copy files to an instance.



4. Save the key in PuTTY's format. You can either select **Save** from the **File** menu or click **Save private key**. Save the key as file like test.ppk. When PuTTYgen prompts you to save the key without a passphrase, click **Yes**.

The file can be used with PuTTY to connect to your Amazon EC2 host as described in the next section.

Step 3: SSH with PuTTY

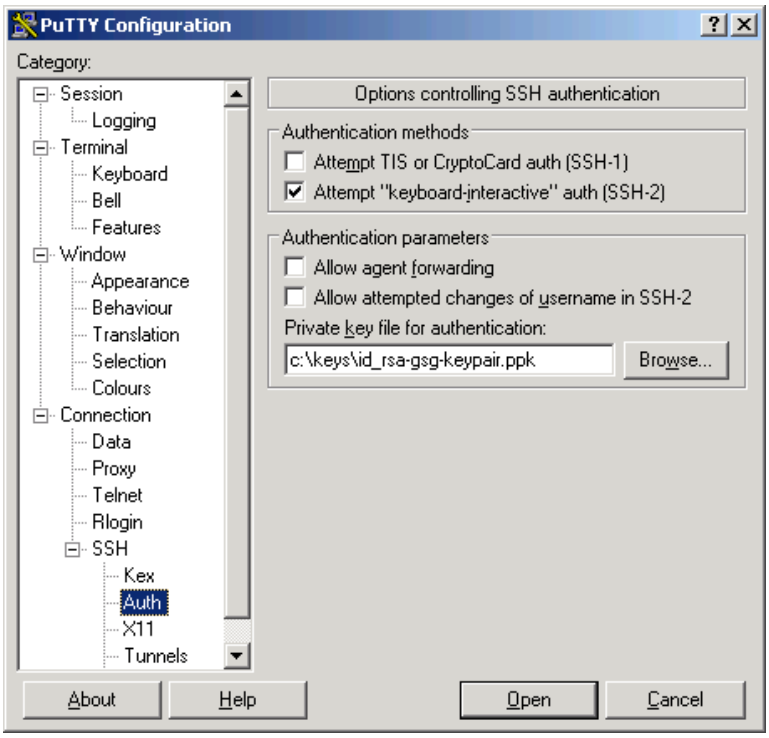
This section assumes that you have converted an Amazon EC2 generated private key file to a PuTTY private key file and have successfully launched an instance.

To use SSH with PuTTY

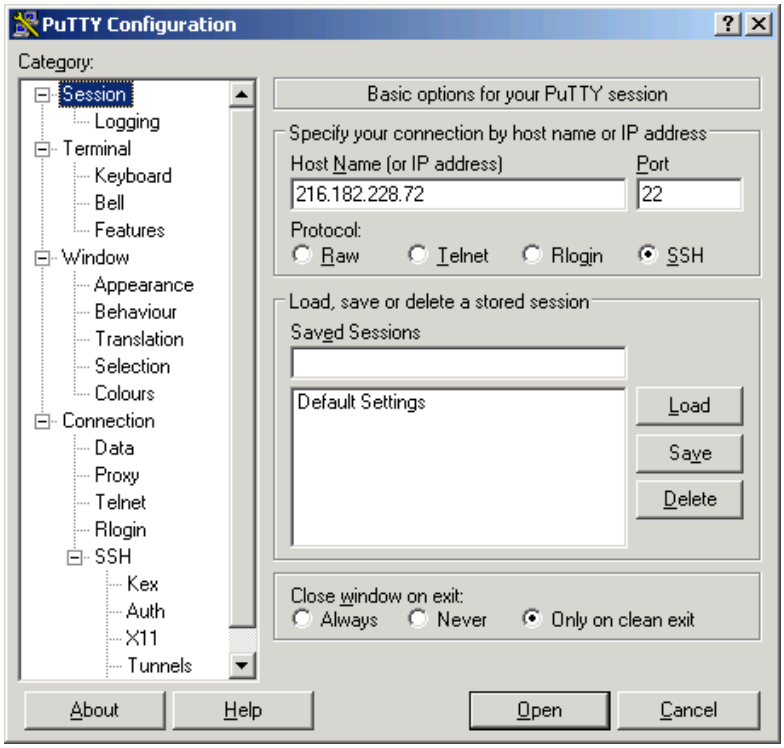
1. Start PuTTY. A graphical configuration utility opens.

   Click **Connection**, point to **SSH**, and select **Authentication**. The PuTTY Configuration dialog box appears.

   Click **Browse**, and select the PuTTY private key file you generated earlier it will be named something like test.ppk.



2. Under **Session**, enter root@*hostname* or root@*ip_address*. Click Open to connect to your Amazon EC2 instance.



Additional References

The following table lists related resources that you'll find useful as you work with this service.

Resource	Description
Amazon EC2 Developer Guide	A comprehensive look at all of the features associated with Amazon EC2.
Amazon EC2 Getting Started Guide	A quick tutorial on how to use the command line tools for Amazon EC2. This guide also includes how to bundle an AMI on Linux/UNIX.
Amazon EC2 Feature Guide	Detailed information about each of the new features released by Amazon EC2.
Amazon EC2 Release Notes	The Release Notes give a high-level overview of the current release. They specifically note any new features, corrections, and known issues.
AWS Developer Resource Center	A central starting point to find documentation, code samples, release notes, and other information to help you build innovative applications with AWS.
Discussion Forums	A community-based forum for developers to discuss technical questions related to Amazon Web Services.
AWS Support Center	The home page for AWS Technical Support, including access to our Developer Forums, Technical FAQs, Service Status page, and AWS Premium Support (if you are subscribed to this program).
AWS Premium Support Information	The primary web page for information about AWS Premium Support, a one-on-one, fast-response support channel to help you build and run applications on AWS Infrastructure Services.
E-mail address for questions related to your AWS account: <webservices@amazon.com>	This e-mail address is *only* for account questions. For technical questions, use the Discussion Forums.