

Compute

EC2

- Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud 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 them from common failure scenarios.



EC2 Benefits



Elastic Web-Scale Computing

Amazon EC2 enables you to increase or decrease capacity within minutes, not hours or days. You can commission one, hundreds, or even thousands of server instances simultaneously. Because this is all controlled with web service APIs, your application can automatically scale itself up and down depending on its needs.



Completely Controlled

You have complete control of your instances including root access and the ability to interact with them as you would any machine. You can stop any instance while retaining the data on the boot partition, and then subsequently restart the same instance using web service APIs. Instances can be rebooted remotely using web service APIs, and you also have access to their console output.



Flexible Cloud Hosting Services

You have the choice of multiple instance types, operating systems, and software packages. Amazon EC2 allows you to select a configuration of memory, CPU, instance storage, and the boot partition size that is optimal for your choice of operating system and application. For example, choice of operating systems includes numerous Linux distributions and [Microsoft Windows Server](#).



Integrated

Amazon EC2 is integrated with most AWS services such as Amazon Simple Storage Service (Amazon S3), Amazon Relational Database Service (Amazon RDS), and Amazon Virtual Private Cloud (Amazon VPC) to provide a complete, secure solution for computing, query processing, and cloud storage across a wide range of applications.



Reliable

Amazon EC2 offers a highly reliable environment where replacement instances can be rapidly and predictably commissioned. The service runs within Amazon's proven network infrastructure and data centers. The Amazon EC2 Service Level Agreement commitment is 99.95% availability for each Amazon EC2 Region.



Secure

Cloud security at AWS is the highest priority. As an AWS customer, you will benefit from a data center and network architecture built to meet the requirements of the most security-sensitive organizations. Amazon EC2 works in conjunction with Amazon VPC to provide security and robust networking functionality for your compute resources.



Inexpensive

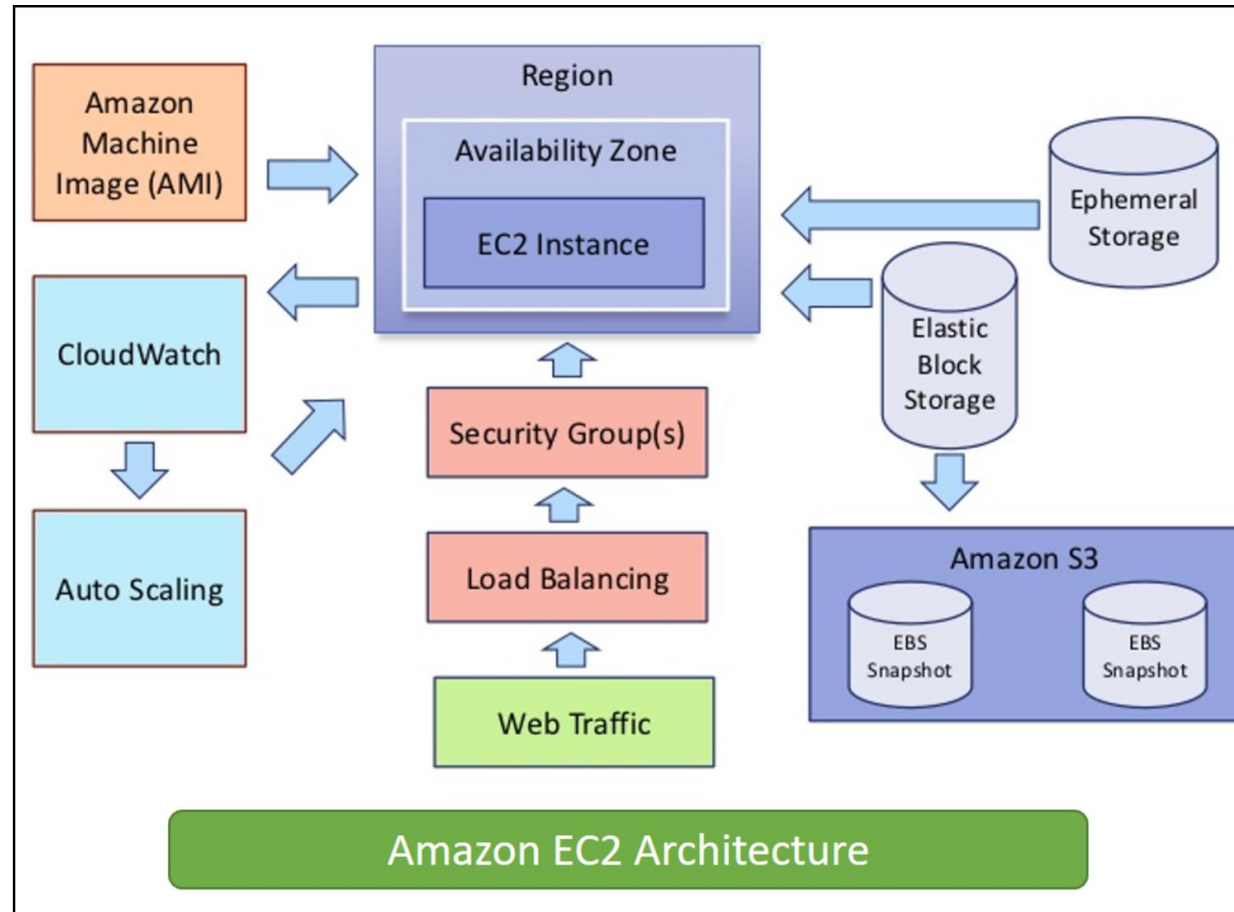
Amazon EC2 passes on to you the financial benefits of Amazon's scale. You pay a very low rate for the compute capacity you actually consume. See Amazon [EC2 Instance Purchasing Options](#) for more details.



Easy to Start

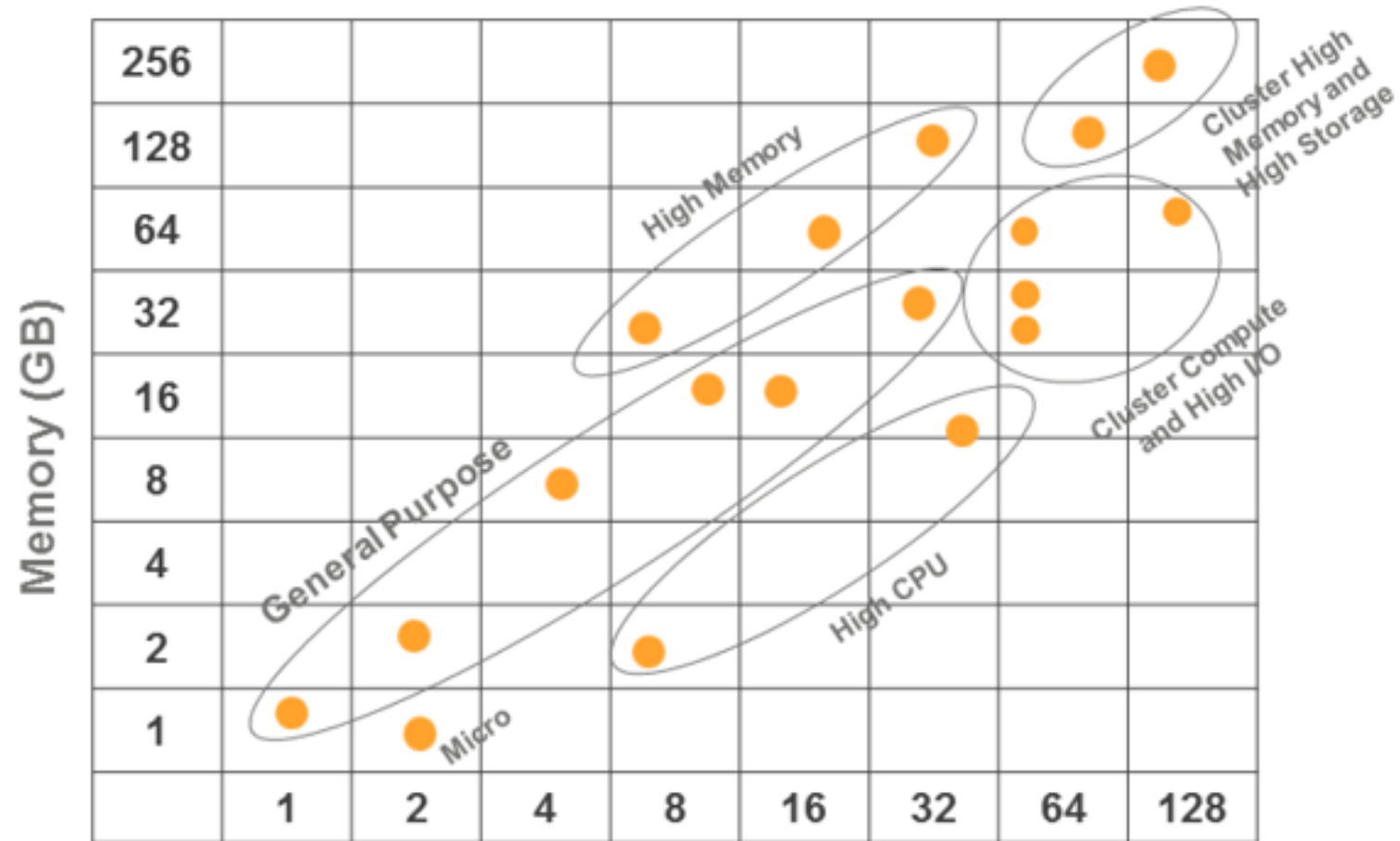
There are several ways to get started with Amazon EC2. You can use the [AWS Management Console](#), the AWS Command Line Tools (CLI), or [AWS SDKs](#). AWS is [free](#) to get started. To learn more, please visit our [tutorials](#).

EC2 Architecture



Source: Raja Malleswara Rao Pattamsetti, 2017

EC2 Instances



Choosing the right instance

- General Purpose Instances
 - T2 instances are a low-cost, burstable performance instance type that provide a baseline level of CPU performance with the ability to burst above the baseline. They offer a balance of compute, memory, and network resources for workloads that occasionally need to burst, such as web servers, build servers, and development environments.
 - M3 and M4 instances provide a balance of compute, memory, and network resources. These instances are ideal for applications that require high CPU and memory performance, such as encoding applications, high traffic content management systems, and memcached applications.
- Compute-Optimized Instances
 - C3 and C4 instances are optimized for compute-intensive workloads. These instances have proportionally more CPU than memory (RAM). They are well suited to applications such as high performance web servers, batch processing, and high- performance scientific and engineering applications.
- Memory-Optimized Instances
 - R3 instances are optimized for memory-intensive workloads. These instances offer large memory sizes for high throughput applications such as high performance databases, distributed memory caches, in-memory analytics, and large enterprise deployments of software such as SAP.
- GPU Instances
 - G2 instances are optimized for graphics and graphic processing unit (GPU) compute applications, such as machine learning, video encoding, and interactive streaming applications.
- Storage-Optimized Instances
 - I2 instances are optimized for storage and high random I/O performance, such as NoSQL databases, scale-out transactional databases, data warehousing, Hadoop, and cluster file systems.
-

Instance Types

Type		Feature Highlight	Use Case
General purpose	T2	<ul style="list-style-type: none">• Lowest-cost general-purpose instance type• Balance of compute, memory, and network resources	<ul style="list-style-type: none">• Development environments• Build servers• Code repositories• Low-traffic web applications• Early product experiments• Small databases
	M3	<ul style="list-style-type: none">• SSD-based instance storage for fast I/O performance• Balance of compute, memory, and network resources	<ul style="list-style-type: none">• Small and mid-size databases• Data processing tasks that require additional memory• Caching fleets and for running backend servers for SAP• Microsoft SharePoint and other enterprise applications

More information: <https://aws.amazon.com/ec2/instance-types/>

Instance Types

Type		Feature Highlight	Use Case	
Compute-Optimized	C3	<ul style="list-style-type: none">• Support for enhanced networking• Support for clustering• SSD-backed instance storage	<ul style="list-style-type: none">• High-performance front-end fleets• Web servers• On-demand batch processing• High-performance science and engineering applications	<ul style="list-style-type: none">• Distributed analytics• Ad serving• Batch processing• MMO gaming• Video encoding• Distributed analytics

More information: <https://aws.amazon.com/ec2/instance-types/>

Instance Types

Type		Feature Highlight	Use Case	
Compute-Optimized	C4	<ul style="list-style-type: none">• Uses Intel Xeon E5-2666 v3 (code name Haswell) processor• Support for enhanced networking• EBS-optimized• Support for clustering• SSD-backed instance storage	<ul style="list-style-type: none">• High-performance front-end fleets• Top-end web servers• Risk analysis and rendering• High-performance science and engineering applications	<ul style="list-style-type: none">• Distributed analytics• Ad serving• Batch processing• Online gaming• Simulation• Distributed analytics

More information: <https://aws.amazon.com/ec2/instance-types/>

Instance Types

Instance Size	vCPU Count	RAM	Network Performance	Dedicated EBS Throughput
c4.large	2	3.75 GiB	Moderate	500 Mbps
c4.xlarge	4	7.5 GiB	Moderate	750 Mbps
c4.2xlarge	8	15 GiB	High	1,000 Mbps
c4.4xlarge	16	30 GiB	High	2,000 Mbps
c4.8xlarge	36	60 GiB	10 Gbps	4,000 Mbps

More information: <https://aws.amazon.com/ec2/instance-types/>

Instance Types

Type		Feature Highlight	Use Case
Memory-Optimized	R3	<ul style="list-style-type: none">• Lowest price point per GB of RAM• SSD Storage• Support for enhanced networking	<ul style="list-style-type: none">• High-performance databases• Distributed memory caches• In-memory analytics• Genome assembly and analysis• Larger deployments of SAP, Microsoft SharePoint, and other enterprise applications

More information: <https://aws.amazon.com/ec2/instance-types/>

Instance Types

Type		Feature Highlight	Use Case
GPU	G2	<ul style="list-style-type: none">• High-performance NVIDIA GPU with 1,536 CUDA cores and 4 GB of video memory• On-board hardware video encoder designed to support up to eight real-time HD video streams	<ul style="list-style-type: none">• Game streaming• Video encoding• 3D application streaming• Other server-side graphics workloads

More information: <https://aws.amazon.com/ec2/instance-types/>

Instance Types

Type		Feature Highlight	Use Case
Storage-Optimized	I2	<ul style="list-style-type: none">• SSD Storage• Support for TRIM• Support for enhanced networking• High random I/O performance	<ul style="list-style-type: none">• NoSQL databases (e.g., Cassandra and MongoDB)• Scale out transactional databases• Data warehousing• Hadoop• Cluster file systems
	D2	<ul style="list-style-type: none">• Up to 48 TB of HDD-based storage• High disk throughput• Lowest price per disk throughput performance on Amazon EC2• Amazon EC2 Enhanced Networking support	<ul style="list-style-type: none">• Massively Parallel Processing (MPP) data warehousing• MapReduce and Hadoop distributed computing• Distributed file systems• Network file systems• Log or data-processing applications

More information: <https://aws.amazon.com/ec2/instance-types/>

Instance Types

	Local Storage	EBS-Backed Storage
Storage Volume	Storage volume is defined by the instance type that you chose	User defined
Over the network	No; Local direct attach storage	Yes
Persistence	Non-persisted storage <ul style="list-style-type: none">• Temporal in nature• When the Amazon EC2 instance is stopped or terminated, the resources allocated for the Amazon EC2 instance are released back to the fleet	Persistent storage <ul style="list-style-type: none">• User can create snapshots• If "<i>DeleteOnTermination</i>" attribute is set to "<i>false</i>", the EBS volume will persist after termination of the EC2 instance to which it was attached

More information: <https://aws.amazon.com/ec2/instance-types/>


Intel Technologies

	Burstable	Balanced	Compute	Memory	GPU	I/O	Storage
AWS Instance Type	T2	M4	C4	R3	G2	I2	D2
Intel® processor	Intel® Xeon® family	Intel® Xeon® E5-2676 v3	Intel® Xeon® E5-2666 v3	Intel® Xeon® E5-2670 v2	Intel® Xeon® E5-2670	Intel® Xeon® E5-2670 v2	Intel® Xeon® E5-2676 v3
Intel® process technology	●	22nm Haswell	22nm Haswell	22nm Ivy Bridge	32nm Sandy Bridge	22nm Ivy Bridge	22nm Haswell
Intel® AVX	●	●	●	●	●	●	●
Intel® AVX2		●	●				●
Intel® Turbo Boost	●	●	●	●	●	●	●
Storage	EBS only	EBS only	EBS only	SSD	SSD	SSD	HDD

EC2 Storage Types

	Local Storage	EBS-Backed Storage
Storage Volume	Storage volume is defined by the instance type that you chose	User defined
Over the network	No; Local direct attach storage	Yes
Persistence	Non-persisted storage <ul style="list-style-type: none">• Temporal in nature• When the Amazon EC2 instance is stopped or terminated, the resources allocated for the Amazon EC2 instance are released back to the fleet	Persistent storage <ul style="list-style-type: none">• User can create snapshots• If "<i>DeleteOnTermination</i>" attribute is set to "<i>false</i>", the EBS volume will persist after termination of the EC2 instance to which it was attached

Purchase Options

On-Demand Instances	Reserved Instances	Spot Instances
<ul style="list-style-type: none">✓ Pay as you go for compute power✓ Pay only for what you use, no up-front commitments or long-term contracts 	<ul style="list-style-type: none">✓ 1- or 3-year terms✓ Reserve computing capacity at a significantly discounted hourly rate compared to On-Demand Instance pricing✓ Compute capacity is available when needed	<ul style="list-style-type: none">✓ Bid on unused Amazon EC2 capacity✓ Spot Price based on supply/demand, determined automatically✓ Spot Price below bid, instances start✓ Spot Price above bid, instances terminate

Amazon EC2 Quick Start

adapted from

http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html

Amazon Elastic Compute Cloud (EC2)

- Amazon Machine Images (**AMIs**) are the basic building blocks of Amazon EC2
- An AMI is a template that contains a software configuration (operating system, application server and applications) that can run on Amazon's computing environment
- AMIs can be used to launch an ***instance***, which is a copy of the AMI running as a virtual server in the cloud.

Getting Started with Amazon EC2

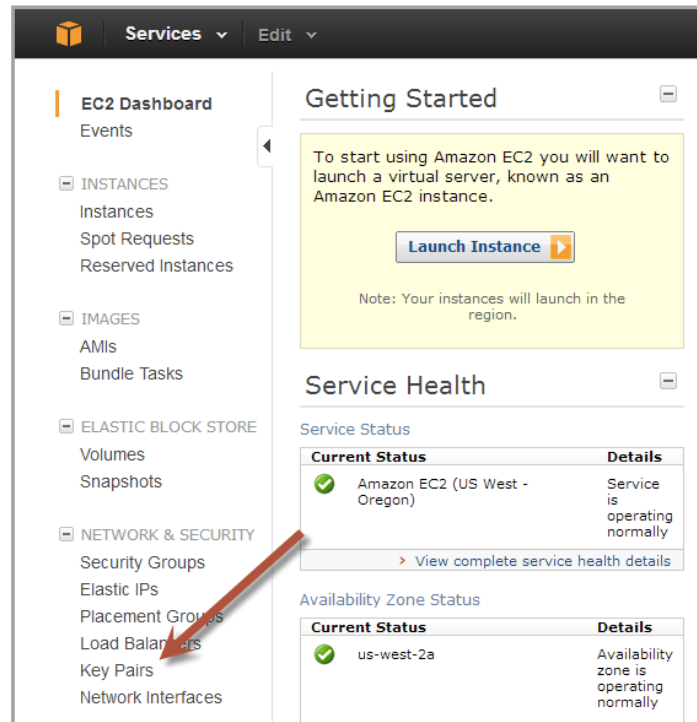
- Step 1: Sign up for Amazon EC2
- Step 2: Create a key pair
- Step 3: Launch an Amazon EC2 instance
- Step 4: Connect to the instance
- Step 5: Customize the instance
- Step 6: Terminate instance and delete the volume created

Creating a key pair

- AWS uses public-key cryptography to encrypt and decrypt login information.
- AWS only stores the public key, and the user stores the private key.
- There are two options for creating a key pair:
 - Have Amazon EC2 generate it for you
 - Generate it yourself using a third-party tool such as OpenSSH, then import the public key to Amazon EC2

Generating a key pair with Amazon EC2

1. Open the Amazon EC2 console at <http://console.aws.amazon.com/ec2/>
2. On the navigation bar select region for the key pair
3. Click **Key Pairs** in the navigation pane to display the list of key pairs associated with the account

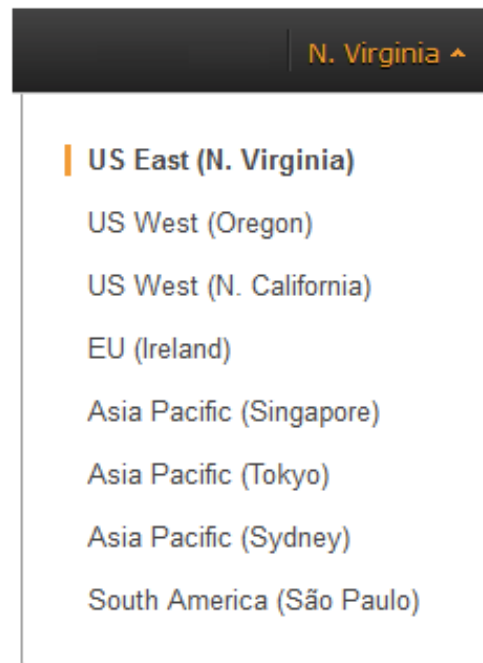


Generating a key pair with EC2 (cont.)

4. Click **Create Key Pair**
5. Enter a name for the key pair in the **Key Pair Name** field of the dialog box and click **Create**
6. The private key file, with .pem extension, will automatically be downloaded by the browser.

Launching an Amazon EC2 instance

1. Sign in to AWS Management Console and open the Amazon EC2 console at <http://console.aws.amazon.com/ec2/>
2. From the navigation bar select the region for the instance



Launching an Amazon EC2 instance (cont.)

3. From the Amazon EC2 console dashboard, click **Launch Instance**

Create a New Instance

Cancel

Select an option below:

☐ **Classic Wizard**

Launch an On-Demand or Spot instance using the classic wizard with fine-grained control over how it is launched.

☒ **Quick Launch Wizard**

Launch an On-Demand instance using an editable, default configuration so that you can get started in the cloud as quickly as possible.

☐ **AWS Marketplace**

AWS Marketplace is an online store where you can find and buy software that runs on AWS. Launch with 1-Click and pay by the hour.

Name Your Instance: Pick a meaningful name, e.g. Web Server.

Choose a Key Pair:

Public/private key pairs allow you to securely connect to your instance after it launches.

☐ Select Existing ☒ Create New ☐ None

Name: Download

Please note that you need to download the key pair before you can continue.

Choose a Launch Configuration:

More Amazon Machine Images NEW!

Search through public and AWS Marketplace AMIs or choose from your own custom AMIs.

	Amazon Linux AMI 2012.03	<small>The Amazon Linux AMI 2012.03 is an EBS-backed, PV-GRUB image. It includes Linux 3.2, AWS tools, and repository access to multiple versions of MySQL, PostgreSQL, Python, Ruby, and Tomcat.</small>	64 bit <input checked="" type="radio"/> 32 bit <input type="radio"/>	<small>Free tier eligible</small>
	Red Hat Enterprise Linux 6.3	<small>Red Hat Enterprise Linux version 6.3, EBS-boot.</small>	64 bit <input checked="" type="radio"/> 32 bit <input type="radio"/>	
	SUSE Linux Enterprise Server 11	<small>SUSE Linux Enterprise Server 11 Service Pack 2 basic install, EBS boot with Amazon EC2 AMI Tools preinstalled; Apache 2.2, MySQL 5.0, PHP 5.3, and Ruby 1.8.7</small>	64 bit <input checked="" type="radio"/> 32 bit <input type="radio"/>	
	Ubuntu Server 12.04 LTS	<small>Ubuntu Server 12.04 LTS with support available from Canonical (http://www.ubuntu.com/cloud/services).</small>	64 bit <input checked="" type="radio"/> 32 bit <input type="radio"/>	<small>Free tier eligible</small>

Note: You can customize your settings in the next step.

Continue


[Submit Feedback](#) [Getting Started Guide](#)

Launching an Amazon EC2 instance (cont.)

4. On the **Create a New Instance** page, click **Quick Launch Wizard**
5. In **Name Your Instance**, enter a name for the instance
6. In **Choose a Key Pair**, choose an existing key pair, or create a new one
7. In **Choose a Launch Configuration**, a list of basic machine configurations are displayed, from which an instance can be launched
8. Click **Continue** to view and customize the settings for the instance

Launching an Amazon EC2 instance (cont.)

9. Select a security group for the instance. A **Security Group** defines the firewall rules specifying the incoming network traffic delivered to the instance. Security groups can be defined on the Amazon EC2 console, in **Security Groups** under **Network and Security**


 **Security Group:** quicklaunch-1

Details **Inbound** Outbound

Create a new rule: Custom TCP rule ▼

Port range:
(e.g., 80 or 49152-65535)

Source:
(e.g., 192.168.2.0/24, sg-47ad482e, or 1234567890/default)

 Add Rule





Apply Rule Changes




TCP Port (Service)	Source	Action
22 (SSH)	0.0.0.0/0	Delete

Launching an Amazon EC2 instance (cont.)

10. Review settings and click **Launch** to launch the instance
11. Close the confirmation page to return to EC2 console
12. Click **Instances** in the navigation pane to view the status of the instance. The status is **pending** while the instance is launching

After the instance is launched, its status changes to **running**

	Name 	Instance	AMI ID	Root Device	Type	State	Public DNS
	GSG Tutorial	 i-e1ab569a	ami-aecd60c7	ebs	t1.micro	 pending	

	Name 	Instance	AMI ID	Root Device	Type	State	Public DNS
<input type="checkbox"/>	GSG Tutorial	 i-e1ab569a	ami-aecd60c7	ebs	t1.micro	 running	ec2-50-19-54-72.compute-1.amazonaws.com

Connecting to an Amazon EC2 instance

- There are several ways to connect to an EC2 instance once it's launched.
- **Remote Desktop Connection** is the standard way to connect to Windows instances.
- An **SSH client** (standalone or web-based) is used to connect to Linux instances.

Connecting to Linux/UNIX Instances from Linux/UNIX with SSH

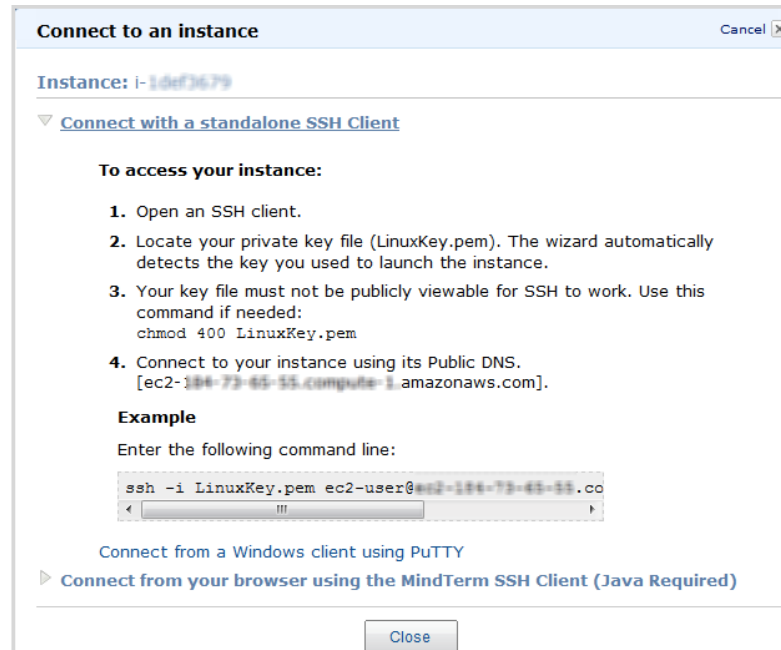
Prerequisites:

- Most Linux/UNIX computers include an SSH client by default, if not it can be downloaded from openssh.org
 - Enable SSH traffic on the instance (using security groups)
 - Get the path the private key used when launching the instance
1. In a command line shell, change directory to the path of the private key file
 2. Use the **chmod** command to make sure the private key file isn't publicly viewable

```
chmod 400 My_Keypair.pem
```

Connecting to Linux/UNIX Instances(cont.)

3. Right click on the instance to connect to on the AWS console, and click **Connect**.
4. Click **Connect using a standalone SSH client**.
5. Enter the example command provided in the Amazon EC2 console at the command line shell



```
ssh -i <your key a name>.pem ec2-user@ec2-184-72-204-112.compute-1.amazonaws.com
```


Transferring files to Linux/UNIX instances from Linux/UNIX with SCP

Prerequisites:

- Enable SSH traffic on the instance
- Install an SCP client (included by default mostly)
- Get the ID of the Amazon EC2 instance, public DNS of the instance, and the path to the private key

If the key file is `My_Keypair.pem`, the file to transfer is `samplefile.txt`, and the instance's DNS name is `ec2-184-72-204-112.compute-1.amazonaws.com`, the command below copies the file to the `ec2-user` home

```
scp -i My_Keypair.pem samplefile.txt ec2-user@ec2-184-72-204-112.compute-1.amazonaws.com:~
```

Terminating Instances

- If the instance launched is not in the free usage tier, as soon as the instance starts to boot, the user is billed for each hour the instance keeps running.
- A terminated instance cannot be restarted.
- To terminate an instance:
 1. Open the Amazon EC2 console
 2. In the navigation pane, click **Instances**
 3. Right-click the instance, then click **Terminate**
 4. Click **Yes, Terminate** when prompted for confirmation

Homework

1. Using the Vpc configuration you created previously
2. Simple EC2 with Python in the public subnet
 1. Create an EC2 (public subnet) with Python and use Python to print out “Hello World”. An example about how to do it, you can check the following youtube:
 - <https://www.youtube.com/watch?v=M2Wc8JIS-p8>
3. Word counting
 1. Use the EC2 instance created above and develop a Python program to count word frequencies. Use:
<http://programminghistorian.org/lessons/counting-frequencies>