

Cloud Computing – Assignment 1

Title: Case study on Amazon EC2

Objective: To understand the concept of Amazon EC2.

Problem Statement: Case study on Amazon EC2 to learn about Amazon EC2, Amazon Elastic Compute Cloud is a central part of Amazon.com's cloud computing platform, Amazon Web Services. How EC2 allows users to torrent virtual computers on which to run their own computer applications

Software Requirements: Browser (Chrome/Firefox), Amazon EC2

Hardware Requirements: Laptop/Desktop, internet connection

Theory:

Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) cloud. Using Amazon EC2 eliminates your need to invest in hardware up front, so you can develop and deploy applications faster. You can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage. Amazon EC2 enables you to scale up or down to handle changes in requirements or spikes in popularity, reducing your need to forecast traffic.



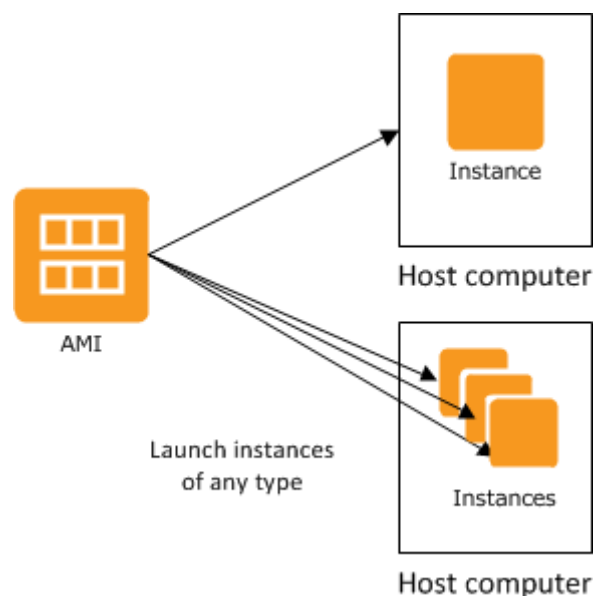
Features of Amazon EC2

- Amazon EC2 provides the following features:
- Virtual computing environments, known as instances
- Preconfigured templates for your instances, known as Amazon Machine Images (AMIs), that package the bits you need for your server (including the operating system and additional software)
- Various configurations of CPU, memory, storage, and networking capacity for your instances, known as instance types
- Secure login information for your instances using key pairs (AWS stores the public key, and you store the private key in a secure place)
- Storage volumes for temporary data that's deleted when you stop or terminate your instance, known as instance store volumes

- Persistent storage volumes for your data using Amazon Elastic Block Store (Amazon EBS), known as Amazon EBS volumes
- Multiple physical locations for your resources, such as instances and Amazon EBS volumes, known as Regions and Availability Zones
- A firewall that enables you to specify the protocols, ports, and source IP ranges that can reach your instances using security groups
- Static IPv4 addresses for dynamic cloud computing, known as Elastic IP addresses
- Metadata, known as tags, that you can create and assign to your Amazon EC2 resources
- Virtual networks you can create that are logically isolated from the rest of the AWS cloud, and that you can optionally connect to your own network, known as virtual private clouds (VPCs)

Instances and AMIs:

An Amazon Machine Image (AMI) is a template that contains a software configuration (for example, an operating system, an application server, and applications). From an AMI, you launch an instance, which is a copy of the AMI running as a virtual server in the cloud. You can launch multiple instances of an AMI, as shown in the following figure



Instances

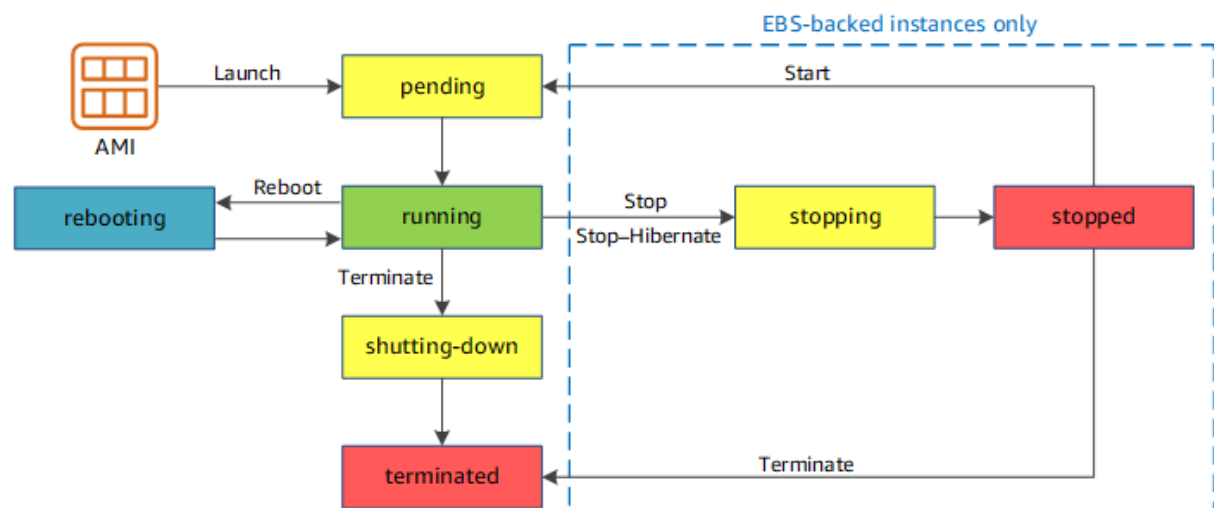
An instance is a virtual server in the cloud. Its configuration at launch is a copy of the AMI that you specified when you launched the instance.

You can launch different types of instances from a single AMI. An instance type essentially determines the hardware of the host computer used for your instance. Each instance type offers different compute and memory capabilities. Select an instance type based on the amount of memory and computing power that you need for the application or software that you plan to run on the instance.

After you launch an instance, it looks like a traditional host, and you can interact with it as you would any computer. You have complete control of your instances; you can use `sudo` to run commands that require root privileges.

Instance Lifecycle

The following illustration represents the transitions between instance states



Instance state	Description
pending	The instance is preparing to enter the running state. An instance enters the pending state when it launches for the first time, or when it is started after being in the stopped state.
running	The instance is running and ready for use.
stopping	The instance is preparing to be stopped or stop-hibernated.
stopped	The instance is shut down and cannot

	be used. The instance can be started at any time.
shutting-down	The instance is preparing to be terminated.
terminated	The instance has been permanently deleted and cannot be started.

AMIs

Amazon Web Services (AWS) publishes many Amazon Machine Images (AMIs) that contain common software configurations for public use. In addition, members of the AWS developer community have published their own custom AMIs. You can also create your own custom AMI or AMIs; doing so enables you to quickly and easily start new instances that have everything you need. For example, if your application is a website or a web service, your AMI could include a web server, the associated static content, and the code for the dynamic pages. As a result, after you launch an instance from this AMI, your web server starts, and your application is ready to accept requests.

All AMIs are categorized as either backed by Amazon EBS, which means that the root device for an instance launched from the AMI is an Amazon EBS volume, or backed by instance store, which means that the root device for an instance launched from the AMI is an instance store volume created from a template stored in Amazon S3.

Pricing -

Generally, Amazon EC2 is priced on a per instance / per hour basis. However, any instance can be rented on a per month basis as well. In such cases, Reserved and Spot Instances pricing can be applied resulting in significant discount. Instances are priced depending on their "size", namely how much CPU and RAM included.

Amazon EC2 price varies from \$2.5 per month for "nano" instances with 1 vCPU and 0.5 GB RAM on board to "xlarge" type of instances with 32 vCPU and 488 GB RAM billed up to \$3997.19 per month.

Amazon EC2 compared to similar cloud computing services [hide]					
	Amazon EC2	Microsoft Azure	Google Cloud Platform	Kamatera	Vultr
1vCPU 0.5GB RAM	\$3.29				\$2.5
1vCPU 0.75GB RAM		\$14.88			
1vCPU 1GB RAM	\$6.83			\$11	\$5
1vCPU 1.75GB RAM		\$44.64			
1vCPU 2GB RAM	\$13.14			\$17	\$10
1vCPU 3.75GB RAM			\$24.27		
2vCPU 3.5GB RAM		\$89.88			
2vCPU 4GB RAM				\$41	\$20
2vCPU 7.5GB RAM			\$48.55		
2vCPU 8GB RAM	\$52.56			\$61	
4vCPU 7GB RAM		\$178.56			
4vCPU 8GB RAM				\$86	\$40
4vCPU 15GB RAM			\$97.09		
4vCPU 15GB RAM	\$134			\$134	
6vCPU 16GB RAM				\$159	\$80
8vCPU 14GB RAM		\$357.12			
8vCPU 16GB RAM				\$184	
8vCPU 30GB RAM			\$194.18		
8vCPU 32GB RAM	\$219.64			\$280	\$160
8vCPU 49GB RAM				\$328	
8vCPU 56GB RAM		\$744			
16vCPU 32GB RAM	\$412.53				
8vCPU 65GB RAM				\$408	
12vCPU 65GB RAM				\$626	

Setting Up with Amazon EC2:

Sign Up for AWS

When you sign up for Amazon Web Services (AWS), your AWS account is automatically signed up for all services in AWS, including Amazon EC2.

With Amazon EC2, you pay only for what you use. If you are a new AWS customer, you can get started with Amazon EC2 for free.

If you have an AWS account already, skip to the next task. If you don't have an AWS account, use the following procedure to create one.

To create an AWS account

- Open <https://portal.aws.amazon.com/billing/signup>
- Follow the online instructions.

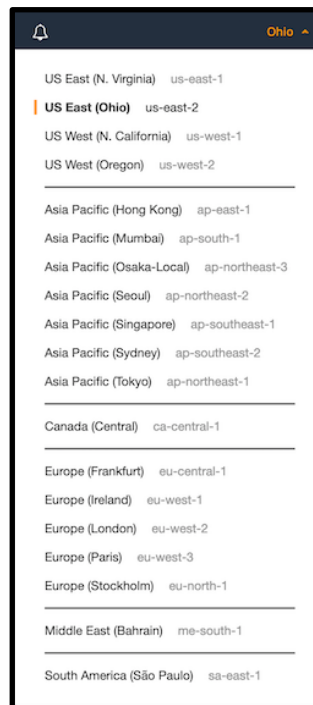
Create a Key Pair

AWS uses public-key cryptography to secure the login information for your instance. A Linux instance has no password; you use a key pair to log in to your instance securely. You specify the name of the key pair when you launch your instance, then provide the private key when you log in using SSH.

If you haven't created a key pair already, you can create one using the Amazon EC2 console. Note that if you plan to launch instances in multiple Regions, you'll need to create a key pair in each Region.

To create a key pair

- Sign in to the AWS Management Console and open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>
- From the navigation bar, select a Region for the key pair. You can select any Region that's available to you, regardless of your location. However, key pairs are specific to a region; for example, if you plan to launch an instance in the US East (Ohio) Region, you must create a key pair for the instance in the US East (Ohio) Region.



- In the navigation pane, under NETWORK & SECURITY, choose Key Pairs.
- Choose Create key pair.
- Do the following:
 - ⚙ For Name, enter a descriptive name for the new key pair, such as your name, followed by -key-pair, plus the Region name. For example, me-key-pair-useast2.
 - ⚙ For File format, choose the format in which to save the private key.
 - Choose pem to save the private key in a format that is used with OpenSSH.
 - Choose ppk to save the private key in a format that is used with PuTTY, a tool that enables you to connect to a Linux instance from Windows.
 - ⚙ Choose Create.
- The private key file is automatically downloaded by your browser. The base file name is the name you specified as the name of your key pair, and the file name extension is .pem. Save the private key file in a safe place.
- If you will use an SSH client on a Mac or Linux computer to connect to your Linux instance, use the following command to set the permissions of your private key file so that only you can read it.

```
chmod 400 your_user_name-key-pair-region_name.pem
```

If you do not set these permissions, then you cannot connect to your instance using this key pair.

Create a Security Group:

Security groups act as a firewall for associated instances, controlling both inbound and outbound traffic at the instance level. You must add rules to a security group that enable you to connect to your instance from your IP address using SSH. You can also add rules that allow inbound and outbound HTTP and HTTPS access from anywhere.

Prerequisites -

You'll need the public IPv4 address of your local computer. The security group editor in the Amazon EC2 console can automatically detect the public IPv4 address for you. Alternatively, you can use the search phrase "what is my IP address" in an Internet browser.

To create a security group with least privilege

- Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>
- From the navigation bar, select a Region for the security group. Security groups are specific to a Region, so you should select the same Region in which you created your key pair.
- Choose Security Groups in the navigation pane.
- Choose Create Security Group.
- Enter a name for the new security group and a description. Use a name that is easy for you to remember, such as your user name, followed by _SG_, plus the Region name. For example, me_SG_uswest2.
- In the VPC list, select your default VPC for the Region.
- On the Inbound tab, create the following rules (choose Add Rule for each new rule), and then choose Create:



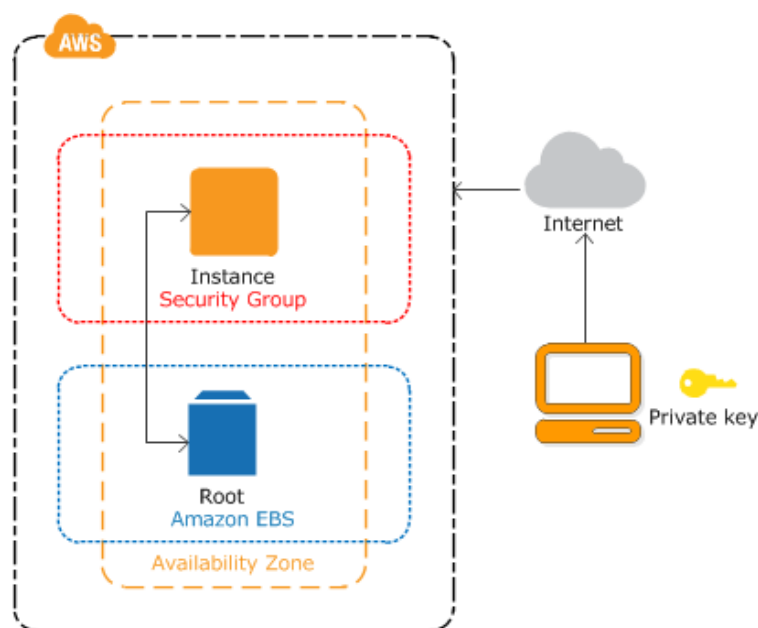
Choose HTTP from the Type list, and make sure that Source is set to Anywhere (0.0.0.0/0).

- ⚙️ Choose HTTPS from the Type list, and make sure that Source is set to Anywhere (0.0.0.0/0).
- ⚙️ Choose SSH from the Type list. In the Source box, choose My IP to automatically populate the field with the public IPv4 address of your local computer. Alternatively, choose Custom and specify the public IPv4 address of your computer or network in CIDR notation. To specify an individual IP address in CIDR notation, add the routing suffix /32, for example, 203.0.113.25/32. If your company allocates addresses from a range, specify the entire range, such as 203.0.113.0/24.

Getting Started with Amazon EC2 Linux Instances:




An instance is a virtual server in the AWS cloud. With Amazon EC2, you can set up and configure the operating system and applications that run on your instance.

The instance is an Amazon EBS-backed instance (meaning that the root volume is an EBS volume). You can either specify the Availability Zone in which your instance runs, or let Amazon EC2 select an Availability Zone for you. When you launch your instance, you secure it by specifying a key pair and security group. When you connect to your instance, you must specify the private key of the key pair that you specified when launching your instance.



Step 1: Launch an Instance

To launch an instance:

- Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>.
- From the console dashboard, choose Launch Instance.
- The Choose an Amazon Machine Image (AMI) page displays a list of basic configurations, called Amazon Machine Images (AMIs), that serve as templates for your instance. Select an HVM version of Amazon Linux 2. Notice that these AMIs are marked "Free tier eligible."
- On the Choose an Instance Type page, you can select the hardware configuration of your instance. Select the t2.micro type, which is selected by default. Notice that this instance type is eligible for the free tier.
- Choose Review and Launch to let the wizard complete the other configuration settings for you.
- On the Review Instance Launch page, under Security Groups, you'll see that the wizard created and selected a security group for you. You can use this security group, or alternatively you can select the security group that you created when getting set up using the following steps:
 -  Choose Edit security groups.
 -  On the Configure Security Group page, ensure that Select an existing security group is selected.
 -  Select your security group from the list of existing security groups, and then choose Review and Launch.
- On the Review Instance Launch page, choose Launch.
- When prompted for a key pair, select Choose an existing key pair, then select the key pair that you created when getting set up.

Alternatively, you can create a new key pair. Select Create a new key pair, enter a name for the key pair, and then choose Download Key Pair. This is the only chance for you to save the private key file, so be sure to download it. Save the private key file in a safe place. You'll need to provide the name of your key pair when you launch an instance and the corresponding private key each time you connect to the instance.

When you are ready, select the acknowledgement check box, and then choose Launch Instances.

- A confirmation page lets you know that your instance is launching. Choose View Instances to close the confirmation page and return to the console.
- On the Instances screen, you can view the status of the launch. It takes a short time for an instance to launch. When you launch an instance, its initial state is pending. After the instance starts, its state changes to running and it receives a public DNS name. (If the Public DNS (IPv4) column is hidden, choose Show/Hide Columns (the gear-shaped icon) in the top right corner of the page and then select Public DNS (IPv4).)
- It can take a few minutes for the instance to be ready so that you can connect to it. Check that your instance has passed its status checks; you can view this information in the Status Checks column.

Step 2: Connect to Your Instance

Connect to the Linux instances that you launched and transfer files between your local computer and your instance. There are several ways to connect to your Linux instance -

Options for Linux and macOS X

- SSH client
- EC2 Instance Connect
- AWS Systems Manager Session Manager

Options for Windows

- PuTTY
- SSH client
- AWS Systems Manager Session Manager
- Windows Subsystem for Linux

SSH client option (Linux) -

Install an SSH client on your local computer as needed

Your local computer might have an SSH client installed by default. You can verify this by typing ssh at the command line. If your computer doesn't recognize the command, install an SSH client.

Connect to Your Linux Instance using an SSH Client

Use the following procedure to connect to your Linux instance using an SSH client.

To connect to your instance using SSH:

- In a terminal window, use the ssh command to connect to the instance. You specify the path and file name of the private key (.pem), the user name for your AMI, and the public DNS name or IPv6 address for your instance. To connect to your instance, do one of the following:

(Public DNS) To connect using your instance's public DNS, enter the following command.

```
ssh -i /path/my-key-pair.pem ec2-user@ec2-198-51-100-1.compute-1.amazonaws.com
```

(IPv6) Alternatively, if your instance has an IPv6 address, to connect using your instance's IPv6 address, enter the following command.

```
ssh -i /path/my-key-pair.pem ec2-user@2001:db8:1234:1a00:9691:9503:25ad:1761
```

You see a response like the following:

```
The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com
(10.254.142.33)'
can't be established.
RSA key fingerprint is
1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f.
Are you sure you want to continue connecting (yes/no)?
```

- (Optional) Verify that the fingerprint in the security alert matches the fingerprint that you previously obtained. If these fingerprints don't match, someone might be attempting a "man-in-the-middle" attack. If they match, continue to the next step.
- Enter yes. You see a response like the following:

```
Warning: Permanently added 'ec2-198-51-100-1.compute-
```

```
1.amazonaws.com' (RSA)
to the list of known hosts.
```

To use SCP to transfer a file:

- Transfer a file to your instance using the instance's public DNS name. For example, if the name of the private key file is my-key-pair, the file to transfer is SampleFile.txt, the user name is ec2-user, and the public DNS name of the instance is ec2-198-51-100-1.compute-1.amazonaws.com, use the following command to copy the file to the ec2-user home directory.

```
scp -i /path/my-key-pair.pem /path/SampleFile.txt ec2-user@ec2-198-
51-100-1.compute-1.amazonaws.com:~
```

You see a response like the following:

```
The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com
(10.254.142.33)'
can't be established.
RSA key fingerprint is
1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f.
Are you sure you want to continue connecting (yes/no)?
```

- (IPv6 only) Alternatively, you can transfer a file using the IPv6 address for the instance. The IPv6 address must be enclosed in square brackets ([]), which must be escaped (\).
- (Optional) Verify that the fingerprint in the security alert matches the fingerprint that you previously obtained. If these fingerprints don't match, someone might be attempting a "man-in-the-middle" attack. If they match, continue to the next step.
- Enter yes.

You see a response like the following:

```
Warning: Permanently added 'ec2-198-51-100-1.compute-
1.amazonaws.com' (RSA)
to the list of known hosts.
```

```
Sending file modes: C0644 20 SampleFile.txt
Sink: C0644 20 SampleFile.txt
SampleFile.txt
```

- To transfer files in the other direction (from your Amazon EC2 instance to your local computer), reverse the order of the host parameters. For example, to transfer the SampleFile.txt file from your EC2 instance back to the home directory on your local computer as SampleFile2.txt, use the following command on your local computer:

```
scp -i /path/my-key-pair.pem ec2-user@ec2-198-51-100-1.compute-1.amazonaws.com:~/SampleFile.txt ~/SampleFile2.txt
```

- (IPv6 only) Alternatively, you can transfer files in the other direction using the instance's IPv6 address:

```
scp -i /path/my-key-pair.pem ec2-user@[2001:db8:1234:1a00:9691:9503:25ad:1761\]:~/SampleFile.txt ~/SampleFile2.txt
```


Step 3: Clean Up Your Instance

After you've finished with the instance, you should clean up by terminating the instance.

To terminate your instance

- In the navigation pane, choose Instances. In the list of instances, select the instance.
- Choose Actions, Instance State, Terminate.
- Choose Yes, Terminate when prompted for confirmation.
- Amazon EC2 shuts down and terminates your instance. After your instance is terminated, it remains visible on the console for a short while, and then the entry is deleted.

Output:


Ubuntu

Free tier eligible

Ubuntu Server 14.04 LTS (HVM), SSD Volume Type - ami-d732f0b7

Select

64-bit

Ubuntu Server 14.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Root device type: ebs Virtualization type: hvm

Step 2: Choose an Instance Type

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)
<input type="radio"/>	General purpose	t2.nano	1	0.5	EBS only
<input checked="" type="radio"/>	General purpose	t2.micro Free tier eligible	1	1	EBS only

Step 3: Configure Instance Details

Number of instances [Launch into Auto Scaling Group](#)

Purchasing option ☐ Request Spot instances

Network [Create new VPC](#)

Subnet [Create new subnet](#)
245 IP Addresses available

Auto-assign Public IP

IAM role [Create new IAM role](#)

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

Step 4: Add Storage

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS
Root	/dev/sda1	snap-47713105	<input type="text" value="8"/>	General Purpose SSD (GP2)	100 / 3000

[Add New Volume](#) [Cancel](#) [Previous](#) [Review and Launch](#) [Next: Tag Instance](#)

Step 5: Tag Instance

Key (127 characters maximum)	Value (255 characters maximum)
Name	ec21-linux

[Create Tag](#) (Up to 50 tags maximum) [Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Security Group](#)

Step 6: Configure Security Group

Assign a security group: ☒ Create a **new** security group
☐ Select an **existing** security group

Security group name:


Description:

Type ⓘ	Protocol ⓘ	Port Range ⓘ
SSH	TCP	22

[Add Rule](#) [Cancel](#) [Previous](#) [Review and Launch](#)

Step 7: Review Instance Launch

▼ AMI Details [Edit AMI](#)

 **Ubuntu Server 14.04 LTS (HVM), SSD Volume Type - ami-d732f0b7**

Free tier eligible Ubuntu Server 14.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical
Root Device Type: ebs Virtualization type: hvm

▼ Instance Type [Edit instance type](#)

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)
t2.micro	Variable	1	1	EBS only

[Cancel](#) [Previous](#) [Launch](#)

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. [Learn more about removing existing key pairs from a public AMI.](#)

Create a new key pair

Key pair name

Download Key Pair

You have to download the **private key file** (*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

Cancel

Launch Instances

Launch Instance

Connect

Actions

Filter by tags and attributes or search by keyword

1 to 17 of 17

	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks
<input type="checkbox"/>	WPS Instance	i-89bf6251	m4.large	us-west-2a	running	2/2 checks
<input type="checkbox"/>	Ec21_linux	i-8d263195	t2.micro	us-west-2a	running	2/2 checks
<input type="checkbox"/>	DoNotTouch	i-43fccc	m1.large	us-west-2b	stopped	

Instances: i-8226319a (Ec21_linux), i-8326319b (Ec21_linux)

Conclusion:

Amazon Elastic Compute Cloud (EC2) forms a central part of Amazon.com's cloud-computing platform, Amazon Web Services (AWS), by allowing users to rent virtual computers on which to run their own computer applications. EC2 encourages scalable deployment of applications by providing a web service to configure a virtual machine.