



Amazon Web Service

AWS-Compute

Prakash Kumar - DevOps Trainer - Ethans



Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. 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.

Elastic Cloud Compute

- Elastic Cloud Compute – EC2
- Provides scalable computing capacity
- 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.
- Secure and reliable.



EC2-Types

- On Demand - allow you to pay a fixed rate by the hour (or by the second) with no commitment.
- Reserved - provide you with a capacity reservation, and offer a significant discount on the hourly charge for an instance. 1 Year or 3 Year Terms
- Spot - enable you to bid whatever price you want for instance capacity, providing for even greater savings if your applications have flexible start and end times.
- Dedicated Hosts - Physical EC2 server dedicated for your use. Dedicated Hosts can help you reduce costs by allowing you to use your existing server-bound software licenses.

On-Demand

- Users that want the low cost and flexibility of Amazon EC2 without any up-front payment or long-term commitment
- Applications with short term, spiky, or unpredictable workloads that cannot be interrupted

Reserved Instance

- Applications with steady state or predictable usage
- Applications that require reserved capacity
- Users able to make upfront payments to reduce their total computing costs even further
 - Standard RI's (Up to 75% off on demand)
 - Convertible RI's (Up to 54% off on demand) capability to change the attributes of the RI as long as the exchange results in the creation of Reserved Instances of equal or greater value.
 - Scheduled RI's available to launch within the time windows you reserve. This option allows you to match your capacity reservation to a predictable recurring schedule that only requires a fraction of a day, a week, or a month.

SPOT Instance

- Applications that have flexible start and end times
- Applications that are only feasible at very low compute prices
- Users with urgent computing needs for large amounts of additional capacity

Dedicated Instance

- Useful for regulatory requirements that may not support multi-tenant virtualization.
- Great for licensing which does not support multi-tenancy or cloud deployments.
- Can be purchased On-Demand (hourly.)
- Can be purchased as a Reservation for up to 70% off the On-Demand price.

On-Demand	Reserved	Spot	Dedicated
Pay for compute capacity by the hour with no long-term commitments	Make a low, one-time payment and receive a significant discount on the hourly charge	Bid for unused capacity, charged at a Spot Price which fluctuates based on supply and demand	Launch instances within Amazon VPC that run on hardware dedicated to a single customer
For spiky workloads, or to define needs	For committed utilization	For time-insensitive or transient workloads	For BYOL and highly sensitive/regulated workloads
			

Family	Speciality	Use case
D2	Dense Storage	Fileservers/Data Warehousing/Hadoop
R4	Memory Optimized	Memory Intensive Apps/DBs
M4	General Purpose	Application Servers
C4	Compute Optimized	CPU Intensive Apps/DBs
G2	Graphics Intensive	Video Encoding/ 3D Application Streaming
I2	High Speed Storage	NoSQL DBs, Data Warehousing etc
F1	Field Programmable Gate Array	Hardware acceleration for your code.
T2	Lowest Cost, General Purpose	Web Servers/Small DBs
P2	Graphics/General Purpose GPU	Machine Learning, Bit Coin Mining etc
X1	Memory Optimized	SAP HANA/Apache Spark etc

- How I remember them now;
 - **D** for Density
 - **R** for RAM
 - **M** - main choice for general purpose apps
 - **C** for Compute
 - **G** - Graphics
 - **I** for IOPS
 - **F** for FPGA
 - **T** cheap general purpose (think T2 Micro)
 - **P** - Graphics (think Pics)
 - **X** - Extreme Memory



What is EBS

Amazon EBS allows you to create storage volumes and attach them to Amazon EC2 instances. Once attached, you can create a file system on top of these volumes, run a database, or use them in any other way you would use a block device. Amazon EBS volumes are placed in a specific Availability Zone, where they are automatically replicated to protect you from the failure of a single component.

EBS TYPE

- General Purpose SSD (GP2)
 - General purpose, balances both price and performance.
 - Ratio of 3 IOPS per GB with up to 10,000 IOPS and the ability to burst up to 3000 IOPS for extended periods of time for volumes at 3334 GiB and above.
- Provisioned IOPS SSD (IO1)
 - Designed for I/O intensive applications such as large relational or NoSQL databases.
 - Use if you need more than 10,000 IOPS.

EBS TYPE

- Throughput Optimized HDD (ST1)
 - Big data
 - Data warehouses
 - Log processing
 - Cannot be a boot volume
- Cold HDD (SC1)
 - Lowest Cost Storage for infrequently accessed workloads
 - File Server
 - Cannot be a boot volume.

EBS TYPE

- Magnetic (Standard)
 - Lowest cost per gigabyte of all EBS volume types that is bootable. Magnetic volumes are ideal for workloads where data is accessed infrequently, and applications where the lowest storage cost is important.

EBS Demo – Mount EBS Volume

Step 1: Now, login to your ec2 instance and list the available disks using the following command.

```
lsblk
```

The above command will list the disk you attached to your instance.

Step 2: Check if the volume has any data using the following command.

```
sudo file -s /dev/xvdf
```

If the above command output shows “/dev/xvdf: data”, it means your volume is empty.

Step 3: Format the volume to ext4 filesystem using the following command.

```
sudo mkfs -t ext4 /dev/xvdf
```

Step 4: Create a directory of your choice to mount our new ext4 volume. I am using the name “newvolume”

```
sudo mkdir /newvolume
```

Step 4: Mount the volume to “newvolume” directory using the following command.

```
sudo mount /dev/xvdf /newvolume/
```

Step 5: cd into newvolume directory and check the disk space for confirming the volume mount.

```
cd /newvolume
```

```
df -h
```

The above command would show the free space in the newvolume directory.

To unmount the volume, you have to use the following command.

```
umount /dev/xvdf
```

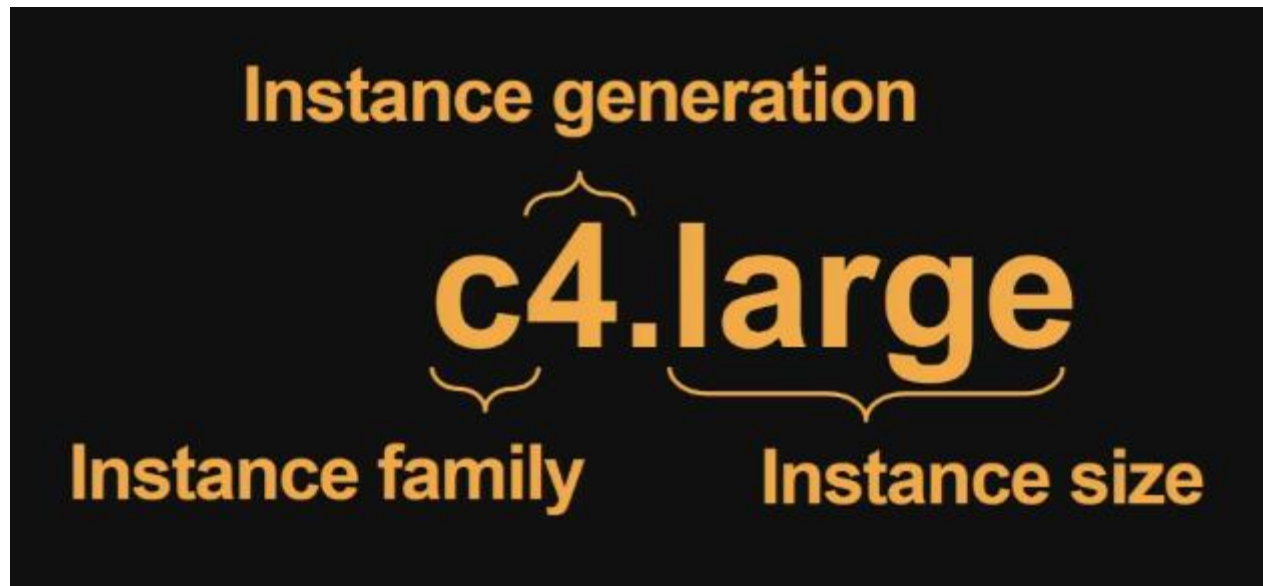
IMP Points

- Know the differences between;
 - On Demand
 - Spot
 - Reserved
 - Dedicated Hosts
- Remember with spot instances;
 - If you terminate the instance, you pay for the hour
 - If AWS terminates the spot instance, you get the hour it was terminated in for free.

IMP Points

- EBS Consists of;
 - SSD, General Purpose - GP2 - (Up to 10,000 IOPS)
 - SSD, Provisioned IOPS - IO1 - (More than 10,000 IOPS)
 - HDD, Throughput Optimized - ST1 - frequently accessed workloads
 - HDD, Cold - SC1 - less frequently accessed data.
 - HDD, Magnetic - Standard - cheap, infrequently accessed storage


Instance Type



AWS Cloud Security tools



Putty and PuttyGen

 root@ip-172-31-16-147:/home/ec2-user

login as: ec2-user

Authenticating with public key "imported-openssh-key"

```
  _ | _ | _ )  
  _ | ( _ /   Amazon Linux AMI  
  _ | \ _ | _ |
```

<https://aws.amazon.com/amazon-linux-ami/2017.09-release-notes/>

[ec2-user@ip-172-31-16-147 ~]\$ sudo su

[root@ip-172-31-16-147 ec2-user]#

Security Groups

A security group acts as a virtual firewall that controls the traffic for one or more instances. When you launch an instance, you associate one or more security groups with the instance. You add rules to each security group that allow traffic to or from its associated instances. You can modify the rules for a security group at any time; the new rules are automatically applied to all instances that are associated with the security group. When we decide whether to allow traffic to reach an instance, we evaluate all the rules from all the security groups that are associated with the instance.

Security Groups are state full:

There is no deny traffic rule, by default everything is allow

Upgrade EBS Volume

1. Choose AMI 2. Choose Instance Type 3. Configure Instance **4. Add Storage** 5. Tag Instance 6. Configure Security Group 7. Review

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Type ⓘ	Device ⓘ	Snapshot ⓘ	Size (GiB) ⓘ	Volume Type ⓘ	IOPS ⓘ	Delete on Termination ⓘ	Encrypted ⓘ
Root	/dev/xvda	snap-6ee3234e	8	General Purpose (SSD) ⌵	24 / 3000	<input checked="" type="checkbox"/>	Not Encrypted
EBS ⌵	/dev/sdb ⌵	Search (case-insensitive)	8	Magnetic ⌵	N/A	<input type="checkbox"/>	Not Encrypted ⓧ

Add New Volume



Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

LAB-EBS

tributes or search by keyword



1 to 3 of 3

	Availability Zone	State	Alarm Status	Attachment Information	Monitoring	Volume Status	Encryption
at ...	us-west-2b	in-use	None	i-05cea825d27b0dbfe (demo elb):/dev/sdb (attached)		✓ Okay	Not Encrypted
at ...	us-west-2b	in-use	None	i-05cea825d27b0dbfe (demo elb):/dev/xvda (attached)		✓ Okay	Not Encrypted
at ...	us-west-2b	in-use	None	i-072a8a40af2f911ac (linux aws old):/dev/xvda (attached)		✓ Okay	Not Encrypted

...



```
-h, --help          display this help text and exit

For more information see mkfs(8).
root@ip-172-31-28-117 ec2-user]# mkfs -t ext4 /dev/xvdb
mkfs 1.42.12 (29 Aug 2011)
creating filesystem with 2097152 4k blocks and 524288 inodes
filesystem UUID: 588dcf4e-5ab6-4ffc-abbb-105e5e4cf992
superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done

root@ip-172-31-28-117 ec2-user]#
```

Encrypt Root Device Volume and Create an AMI - Lab

AMIs that are backed by Amazon EBS snapshots can take advantage of Amazon EBS encryption. Snapshots of both data and root volumes can be encrypted and attached to an AMI.

EC2 instances with encrypted volumes are launched from AMIs in the same way as other instances.

Load Balancers and Health Checks

Elastic Load Balancing automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, and IP addresses. It can handle the varying load of your application traffic in a single Availability Zone or across multiple Availability Zones. Elastic Load Balancing offers three types of load balancers that all feature the high availability, automatic scaling, and robust security necessary to make your applications fault tolerant.

Type of ELBs:

1. Application Load Balancer: Application Load Balancer is best suited for load balancing of HTTP and HTTPS traffic and provides advanced request routing targeted at the delivery of modern application architectures, including microservices and containers. Operating at the individual request level (Layer 7), Application Load Balancer routes traffic to targets within Amazon Virtual Private Cloud (Amazon VPC) based on the content of the request.

Load Balancers and Health Checks

Network Load Balancer : Network Load Balancer is best suited for load balancing of TCP traffic where extreme performance is required. Operating at the connection level (Layer 4), Network Load Balancer routes traffic to targets within Amazon Virtual Private Cloud (Amazon VPC) and is capable of handling millions of requests per second while maintaining ultra-low latencies. Network Load Balancer is also optimized to handle sudden and volatile traffic patterns.

Classic Load Balancer: Classic Load Balancer provides basic load balancing across multiple Amazon EC2 instances and operates at both the request level and connection level. Classic Load Balancer is intended for applications that were built within the EC2-Classic network

ELB-LAB

```
[root@ip-172-31-16-147 ec2-user]# cd /var/www/html
```

```
[root@ip-172-31-16-147 html]# ls
```

```
index.html
```

```
[root@ip-172-31-16-147 html]# nano index.html
```

```
[root@ip-172-31-16-147 html]# nano healthcheck.html
```

```
[root@ip-172-31-16-147 html]# ^C
```

```
[root@ip-172-31-16-147 html]#
```

Cloud Watch EC2

Amazon CloudWatch is a monitoring service for AWS cloud resources and the applications you run on AWS. You can use Amazon CloudWatch to collect and track metrics, collect and monitor log files, set alarms, and automatically react to changes in your AWS resources. Amazon CloudWatch can monitor AWS resources such as Amazon EC2 instances, Amazon DynamoDB tables, and Amazon RDS DB instances, as well as custom metrics generated by your applications and services, and any log files your applications generate

Cloud Watch EC2- LAB



Services ▾

Resource Groups ▾



Prakash ▾

C

CloudWatch

Dashboards

* demoaws

Alarms

ALARM

0

INSUFFICIENT

0

OK

0

Billing

Events

Rules

Event Buses ^{NEW}

Logs

Metrics

Welcome to CloudWatch Events

CloudWatch Events helps you to respond to state changes in your AWS resources. When your resources change state they automatically send events into a stream. You can create rules that match selected events in the stream and route them to targets to take action. You can also use rules to take action on a pre-determined schedule. For example, you can configure rules to:

- Automatically invoke an AWS Lambda function to update DNS entries when an event notifies you that Amazon EC2 instance enters the Running state
- Direct specific API records from CloudTrail to a Kinesis stream for detailed analysis of potential security or availability risks
- Take a snapshot of an Amazon EBS volume on a schedule

Create rule

Start Responding to CloudWatch Events





EXAM TIPS

- Standard Monitoring = 5 Minutes
- Detailed Monitoring = 1 Minute

IMP Points

- Dashboards - Creates awesome dashboards to see what is happening with your AWS environment.
- Alarms - Allows you to set Alarms that notify you when particular thresholds are hit.
- Events - CloudWatch Events helps you to respond to state changes in your AWS resources.
- Logs - CloudWatch Logs helps you to aggregate, monitor, and store logs.

Using IAM Roles with EC2



Services ▾

Resource Groups ▾



Prakash ▾

Global ▾

Support ▾

Search IAM

Dashboard

Groups

Users

Roles

Policies

Identity providers

Account settings

Credential report

Encryption keys

- [IAM Roles Documentation](#)
- [Best practices for setting up cross-account access](#)

Create role

Delete role



Q Search

Showing 1 result

Role name ▾

Description

Trusted entities



s3-admin-access

Role for EC2 instance

AWS service: ec2

LAB

* configure

```
[ec2-user@ip-172-31-23-43 ~]$ aws configure
```

```
AWS Access Key ID [None]:
```

```
AWS Secret Access Key [None]:
```

```
Default region name [None]: Oregon
```

```
Default output format [None]:
```

```
[ec2-user@ip-172-31-23-43 ~]$ cd .aws
```

```
[ec2-user@ip-172-31-23-43 .aws]$ ls
```

```
config
```

```
[ec2-user@ip-172-31-23-43 .aws]$ nano config
```

```
[ec2-user@ip-172-31-23-43 .aws]$
```

EC2 Instance Metadata

Instance metadata is data about your instance that you can use to configure or manage the running instance.

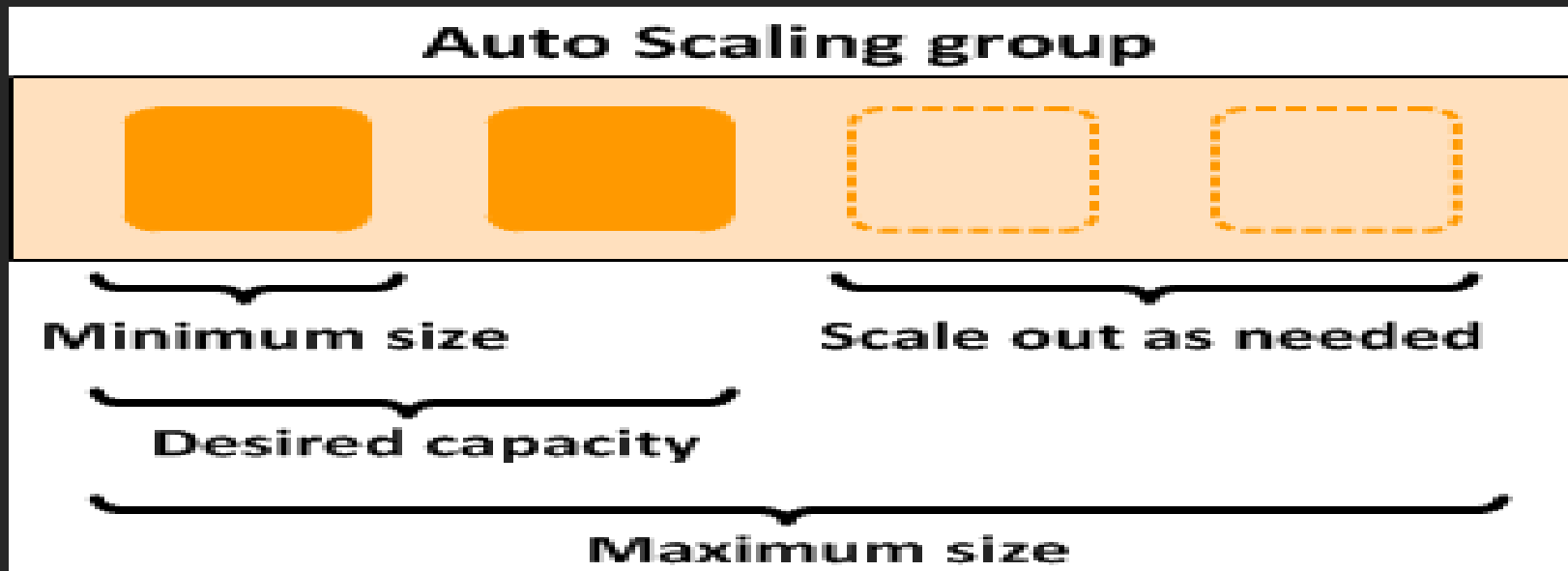
```
ec2-user@ip-172-31-23-43:~$ curl http://169.254.169.254/latest/meta-data/
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
hostname
iam/
instance-action
instance-id
instance-type
local-hostname
local-ipv4
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
ec2-user@ip-172-31-23-43 ~]$ curl http://169.254.169.254/latest/meta-data/public-ipv4
52.88.124.142
ec2-user@ip-172-31-23-43 ~]$ curl http://169.254.169.254/latest/meta-data/public-ipv4 > awsdemo.html
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           % Done    0     0     0    0         0      0      0      0
100    13  100    13    0     0    4126      0  --:--:-- --:--:-- --:--:--   4333
ec2-user@ip-172-31-23-43 ~]$ ls
awsdemo.html
ec2-user@ip-172-31-23-43 ~]$
```

Auto Scaling Groups Lab

Auto Scaling helps you ensure that you have the correct number of Amazon EC2 instances available to handle the load for your application. You create collections of EC2 instances, called Auto Scaling groups. You can specify the minimum number of instances in each Auto Scaling group, and Auto Scaling ensures that your group never goes below this size. You can specify the maximum number of instances in each Auto Scaling group, and Auto Scaling ensures that your group never goes above this size. If you specify the desired capacity, either when you create the group or at any time thereafter, Auto Scaling ensures that your group has this many instances. If you specify scaling policies, then Auto Scaling can launch or terminate instances as demand on your application increases or decreases.


Auto Scaling Groups Lab

Example: Auto Scaling group has a minimum size of 1 instance, a desired capacity of 2 instances, and a maximum size of 4 instances. The scaling policies that you define adjust the number of instances, within your minimum and maximum number of instances



EFS LAB- Elastic file System

Amazon Elastic File System (Amazon EFS) is a file storage service for Amazon Elastic Compute Cloud (Amazon EC2) instances. Amazon EFS is easy to use and provides a simple interface that allows you to create and configure file systems quickly and easily. With Amazon EFS, storage capacity is elastic, growing and shrinking automatically as you add and remove files, so your applications have the storage they need, when they need it.

- 
- Supports the Network File System version 4 (NFSv4) protocol
 - You only pay for the storage you use (no pre-provisioning required)
 - Can scale up to the petabytes
 - Can support thousands of concurrent NFS connections
 - Data is stored across multiple AZ's within a region
 - Read After Write Consistency

LAB-EFS

File systems

Create file system

Actions ▾

Name

Other details

Owner ID 83

Life cycle state Av

Performance mode Ge

Encrypted No

File system access

DNS name fs-

Amazon EC2 mount instructions

AWS Direct Connect mount instruct

Mount targets

Amazon EC2 mount instructions

2. Open an SSH client and connect to your EC2 instance. (find out how to [connect](#))
3. Install the nfs client on your EC2 instance.

- On an Amazon Linux, Red Hat Enterprise Linux, or SuSE Linux instance:

```
sudo yum install -y nfs-utils
```

- On an Ubuntu instance:

```
sudo apt-get install nfs-common
```

Mounting your file system

1. Open an SSH client and connect to your EC2 instance. (find out how to [connect](#))
2. Create a new directory on your EC2 instance, such as "efs".

- ```
sudo mkdir efs
```

3. Mount your file system using the DNS name. [Mounting considerations](#)

- ```
sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2 fs-3e75dc97.efs.us-west-2.amazonaws.com:/ efs
```

If you are unable to connect, please see our [troubleshooting documentation](#).

Close