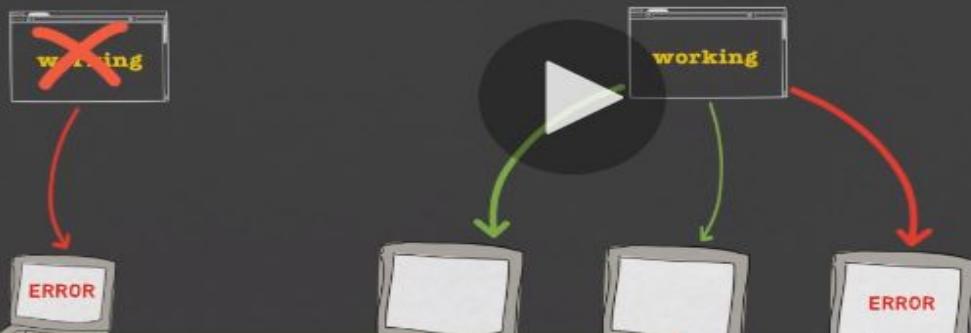


Course Breakdown

1. Module 1: AWS Basic Concepts.
2. Module 2: Identity & Access Management.
3. Module 3: Compute Services.
4. Module 4: Storage Services.
5. Module 5: Networking Services.
6. Module 6: Database Services.
7. Module 7: Monitoring and Logging Services.
8. Module 8: Messaging Services.
9. Module 9: AWS Security.
10. Module 10: AWS Billing and Support.

Fault Tolerance

The ability of a system to continue operating without interruption when one or more of its components fail.



Module1: AWS Basic Cloud Concepts

The Four Pillars of Cloud Computing: (Fault Tolerance, High Availability, Scalability and Elasticity).

Fault Tolerance:

The ability to tolerate a certain amount of failure and still remain functional, with the ability to cope and return to full capacity.

High Availability:

The Ability to access something whenever you wish to do so and be available for you.

Scalability:

The ability to expand in capacity and size when required.

Elasticity:

The ability to expand or reduce in capacity and size when required.

Benefits of the Cloud (AWS)

- 1) Act as a perfect back up for users.
- 2) Very easy to share files across people and devices.



These two benefits are concepts of Fault Tolerance and high Availability.

AWS Global Infrastructure:

AWS Regions:

AWS is made of various regions located all over the world. Those regions consists of Availability Zones that are located at specific AWS Data centers.



AWS Availability Zones:

AWS resources are made up of Availability zones (within a region) that work together. Those availability zones have low latency connections between them and each one is separated from the other zone to ensure fault tolerance.



aws English ▾

Create an AWS account

AWS Accounts Include
12 Months of Free Tier Access

Including use of Amazon EC2, Amazon S3, and Amazon DynamoDB
Visit www.amazon.com/free for full offer terms

Email address

Password

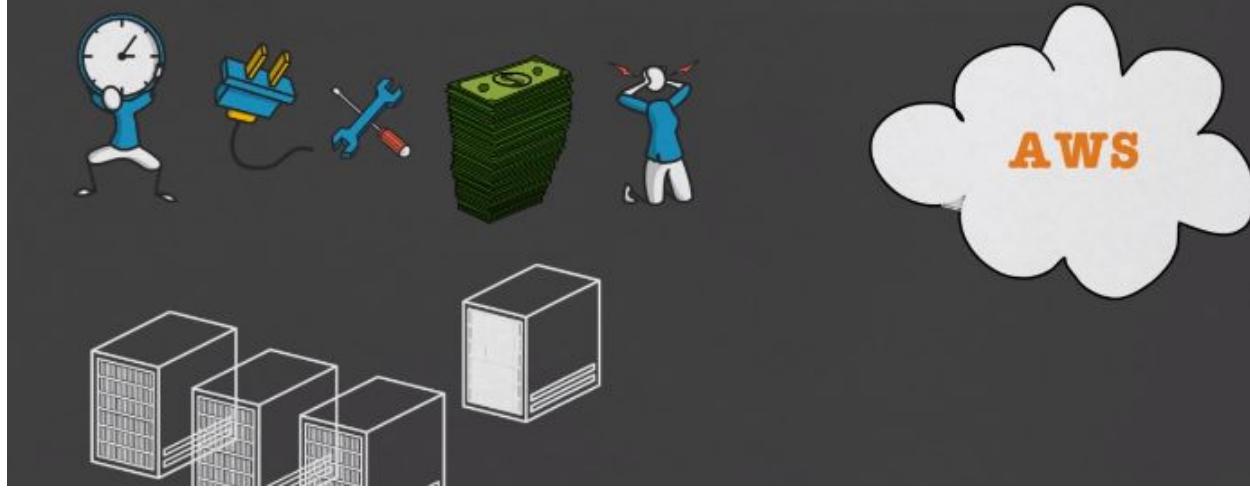
Confirm password * The passwords are not equal.

AWS account name

[Sign in to an existing AWS account](#)

© 2020 Amazon Web Services, Inc. or its affiliates.
All rights reserved.
[Privacy Policy](#) | [Terms of Use](#)

Why move to the Cloud?



IAM

Module 2: Identity & Access Management

What is IAM ?

It is the service where you can manage AWS accounts and their access to other AWS services.



IAM

IAM is used to manage:

- 1) User credentials.
- 2) User password policies.
- 3) Multi-factor authentication (MFA).
- 4) Roles.
- 5) Users.
- 6) Groups
- 7) Access policies.
- 8) API keys for programmatic (command line interface) access.



IAM Users:

Users who are granted access to an AWS account.

Each IAM user should have:

- 1) User name.
- 2) Password.
- 3) Permission to access AWS services.



Definitions of **IAM** features:

User Credentials:

The user name and password used to log into AWS account.

User password policies:

Password format requirements for an **aws** account.

Multi factor authentication:

A form of verification that requires additional code number.

Roles:

The permission granted to AWS services to share data.

API keys for programmatic (command line interface) access:

The use of command line interface to access AWS resources.

Identity and Access Management (IAM)

- Dashboard
- Access management
 - Groups
 - Users
 - Roles
 - Policies
 - Identity providers
 - Account settings
- Access reports
 - Access analyzer
 - Archive rules
 - Analyzer details
- Credential report
- Organization activity
- Service control policies (SCPs)

Welcome to Identity and Access Management

IAM users sign-in link:
<https://907987177566.signin.aws.amazon.com/console> | Customize

IAM Resources

Users: 0	Roles: 7	Identity Providers: 0
Groups: 0		
Customer Managed Policies: 0		

Security Status

1 out of 5 complete.

- ✓ Delete your root access keys**
- ⚠ Activate MFA on your root account**
- ⚠ Create individual IAM users**
- ⚠ Use groups to assign permissions**
- ⚠ Apply an IAM password policy**

Feature Spotlight

Introduction to AWS IAM



Additional Information

- IAM best practices
- IAM documentation
- Web Identity Federation Playground
- Policy Simulator
- Videos, IAM release history and additional resources

Create New Group Wizard

Step 1 : Group Name

Step 2 : Attach Policy

Step 3 : Review

Attach Policy

Select one or more policies to attach. Each group can have up to 10 policies attached.

Filter: Policy Type	Search	Showing 516 results		
	Policy Name	Attached Entities	Creation Time	
<input type="checkbox"/>	AdministratorAccess	1	2015-02-06 12:39 CST	
<input type="checkbox"/>	AlexaForBusiness_AdministratorAccess	1	2017-11-30 10:47 CST	
<input type="checkbox"/>	AmazonRDSEnhancedMonitoringRole	1	2015-11-11 13:58 CST	
<input type="checkbox"/>	AlexaForBusinessFullAccess	0	2017-11-30 10:47 CST	
<input type="checkbox"/>	AlexaForBusinessGatewayExecution	0	2017-11-30 10:47 CST	
<input type="checkbox"/>	AlexaForBusinessPolicyDelegatedAccessPolicy	0	2019-10-16 13:48 CST	
<input type="checkbox"/>	AlexaForBusinessReadOnlyAccess	0	2017-11-30 10:47 CST	

Create user with some group name and apply some policy and add some user to that group

User	Password	Email login instructions
elliot	***** Show	Send email

■ us-east-1.signin.www.amazon.com/iam?SignatureVersion=4&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAJMDATPLHVSJS83XQ&X-Amz-Date=2020-03-29T19%...



The image shows two side-by-side screenshots. On the left is the AWS IAM sign-in page, featuring the AWS logo at the top, followed by a form with fields for 'Account ID (12 digits) or account alias' containing '967897177566', 'IAM user name' (empty), 'Password' (empty), and a blue 'Sign In' button. Below the form are links for 'Sign in using root user email' and 'Forgot password?'. On the right is a promotional banner for 'RE:INFORCE', dated June 30 – July 1, 2020, in Houston, TX. The banner has a dark background with gold and orange curved lines, and text that reads: 'Two days and hundreds of sessions focused on cloud security, identity, and compliance.' At the bottom is a white 'Register Now' button.

AWS account: 907997177566

IAM user name: eliot

Old password:

New password:

Retype new password:

Confirm password change

[Forgot your root user password?](#)

User,Group,Roles , policies

Managing Roles

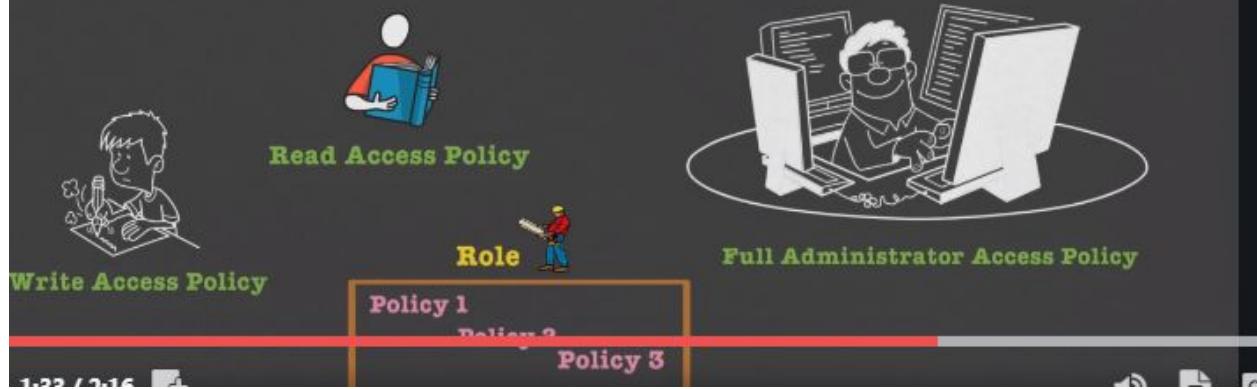
An IAM role is very similar to a user, in that it is an identity with permission policies that determine what the identity can and cannot do in AWS.

**Create Roles
Delete Roles**



Managing Policies

You manage access in AWS by creating policies and attaching them to IAM identities (users, groups, or roles) or AWS resources.



Managing Credentials

User Passwords



Configuring MFA



Access Keys



EC2 - COMPUTE SERVICES

Day 4: Compute Services

Elastic compute Cloud:

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.

You can use EC2 to launch virtual servers, configure security and networking, and manage storage.



EC2 components:

- 1) AMIs (Windows or Linux)
- 2) EBS (Local Storage)
- 3) Security groups (Security)
- 4) RAM
- 5) Instance type (Processing type)
- 6) IP addressing (Internet Access)

Amazon Machine Images (AMI):

AMI Categories:

- 1) Community AMIs: free to use and you can select the OS you prefer.
- 2) AWS Marketplace AMIs: you have to pay to use it and it comes with a licensed software.
- 3) My AMIs: all AMIs created by you.

Instance type options:

- 1) General purpose
- 2) Compute optimized
- 3) Memory optimized
- 4) Gpu optimized
- 5) Storage optimized

EC2 Basics (Buying Options):

On Demand:

This option allows you to choose any instance type you want and terminate it at any time based on your demand.

This option is considered the most expensive and flexible purchasing option.

You only charged when the instance is running. (You will be billed by second)



Reserved:

- 1) This option allows you to purchase an instance for a set period of time which is one or three years.
- 2) You get a significant price discount compared to using on demand option.
- 3) You have the choice to pay upfront, partial upfront or no upfront.
- 4) Once you purchased a reserved instance. You will be the owner for the selected period of time and responsible for paying the whole price (regardless of your usage).



Spot:

You have to bid on an instance type and only pay for and use that instance as long as the spot price is equal or lower than you bid.

This option allows amazon to sell the use of unused instances for a short period of time.

Prices are dependent on the supply and demand.

The instance will be only reserved to you if you have an active bid.

The reserved instance will be terminated if the spot price is greater than your bid.



AWS LAMBDA- SERVERLESS COMPUTE SERVICE

Run code for virtually any type of code which lambda supports like java and python.no need like ec2 to launch instance.it scales automatically.not charge when your code is not running.no need to worry about additional charges.

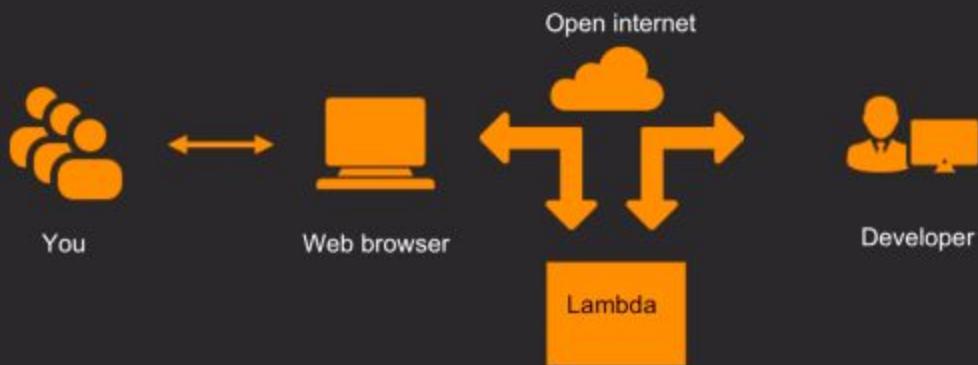
AWS Lambda

AWS Lambda lets you run code without provisioning or managing servers. You pay only for the compute time you consume.

With Lambda, you can run code for virtually any type of application or backend service - all with zero administration. Just upload your code and Lambda takes care of everything required to run and scale your code with high availability. You can set up your code to automatically trigger from other AWS services or call it directly from any web or mobile app.



Lambda working explanation:



Benefits & Use Cases of Lambda:

Benefits:

- 1) No servers to manage
- 2) Subsecond metering
- 3) Continuous scaling
- 4) Integrates with almost all other AWS courses.



Primary user cases:

- 1) Data processing
- 2) Real time file processing
- 3) Real time stream processing
- 4) Build serverless backends for web, mobile and 3rd party API request.



ECS - AMAZON ELASTIC CONTAINER SERVICES FULLY MANAGED CONTAINER SERVICE

Amazon Elastic Container Service:

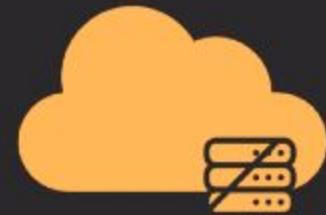
Amazon Elastic Container Service (ECS) is a highly scalable, high performance container management service that supports Docker containers and allows you to easily run applications on a managed cluster of Amazon EC2 instances.



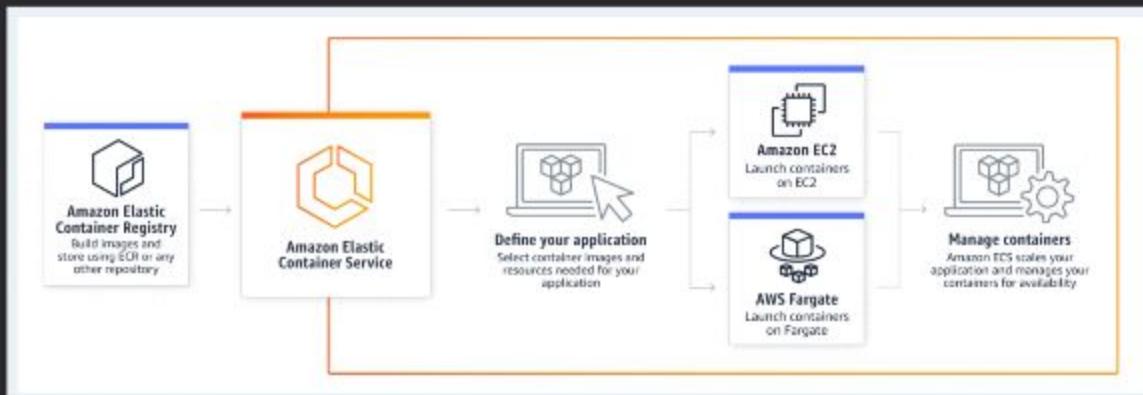
Amazon elastic Container

Benefits:

- 1) Serverless option
- 2) Application first with Capacity Providers
- 3) Performance at scale
- 4) Secure
- 5) Reliable
- 6) Optimized for cost



How it works?



Use Cases:

- 1) Hybrid Deployment
- 2) Machine Learning
- 3) Batch Processing
- 4) Web Applications



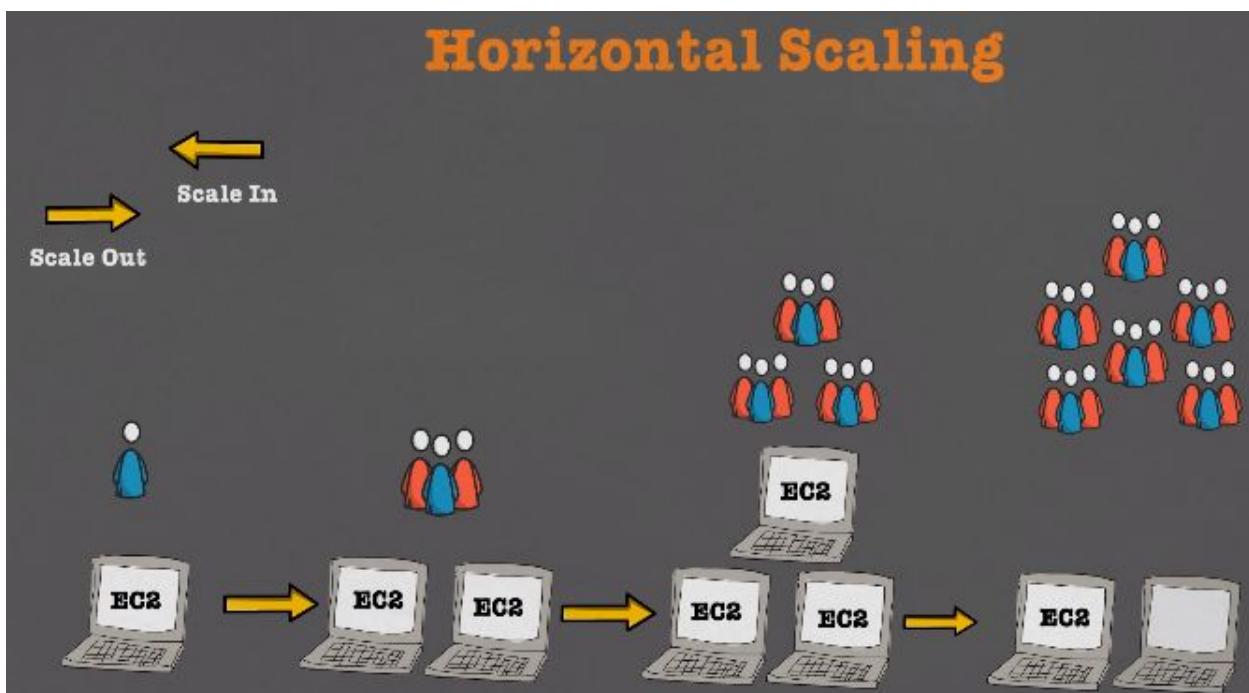
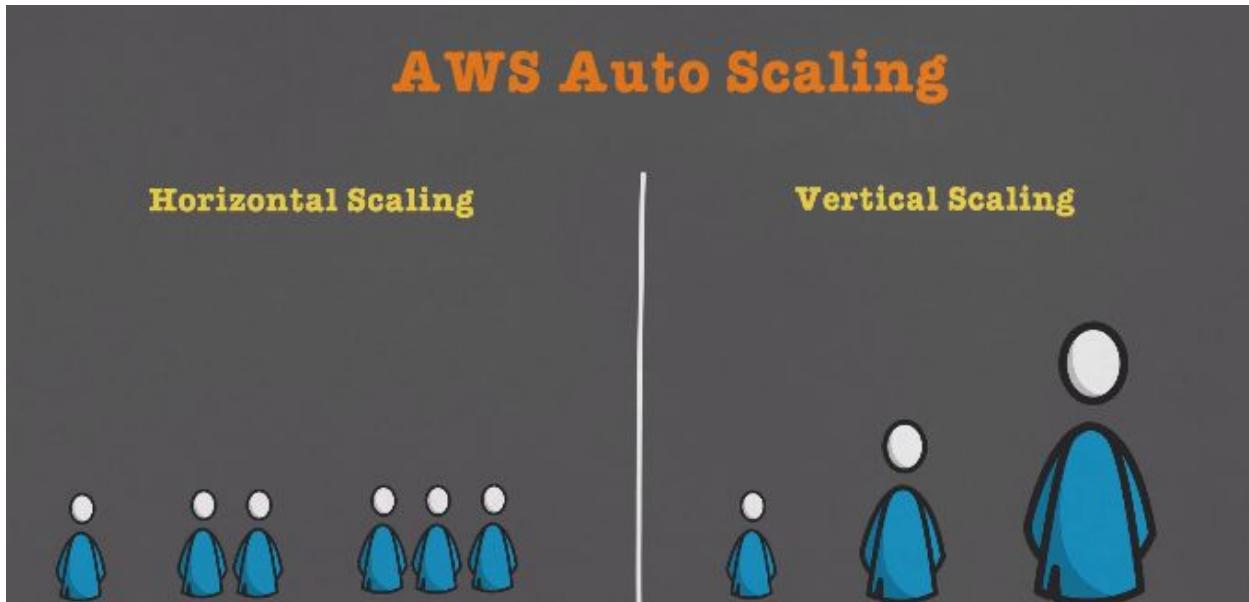
Low latency for on premises system, aws infra, aws apis, aws octopus, manage container on premises as well as container in cloud.

Use aws deep learning for training and services module

Run batch computing jobs, specific resource for batch

Web app like auto scale up and down, benefit for performance, networking, load balancer and vpc

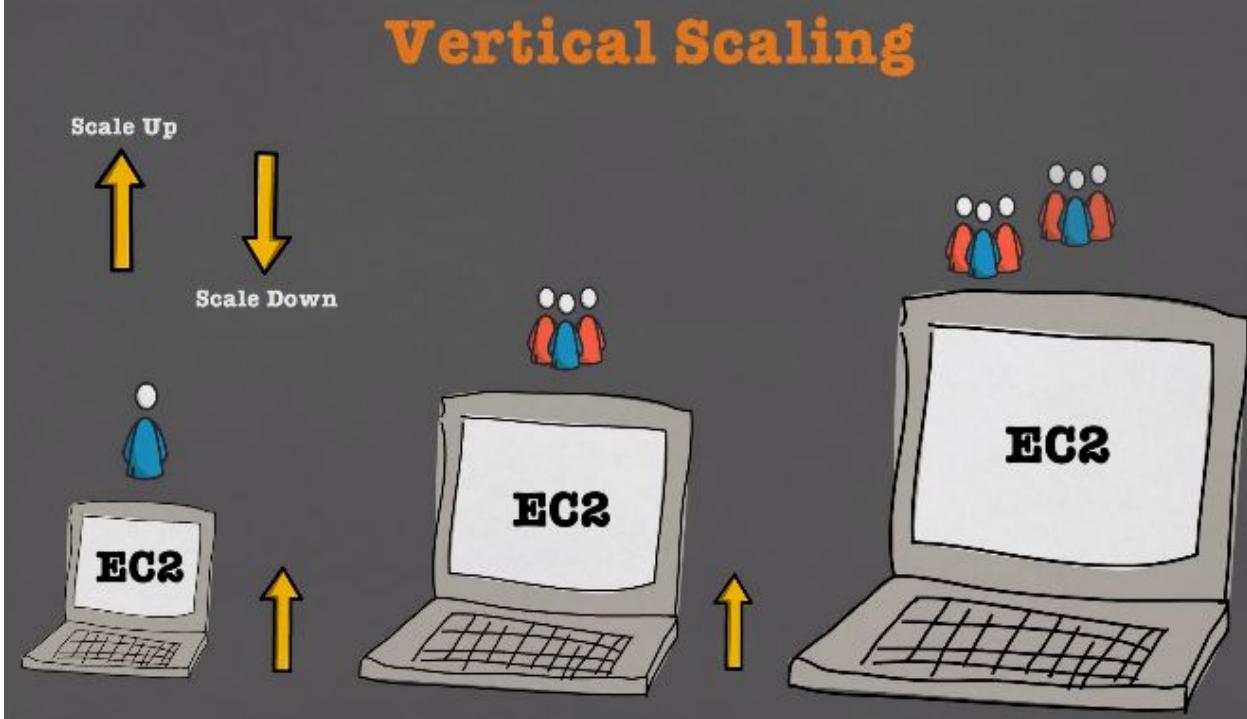
AWS AUTOSCALING



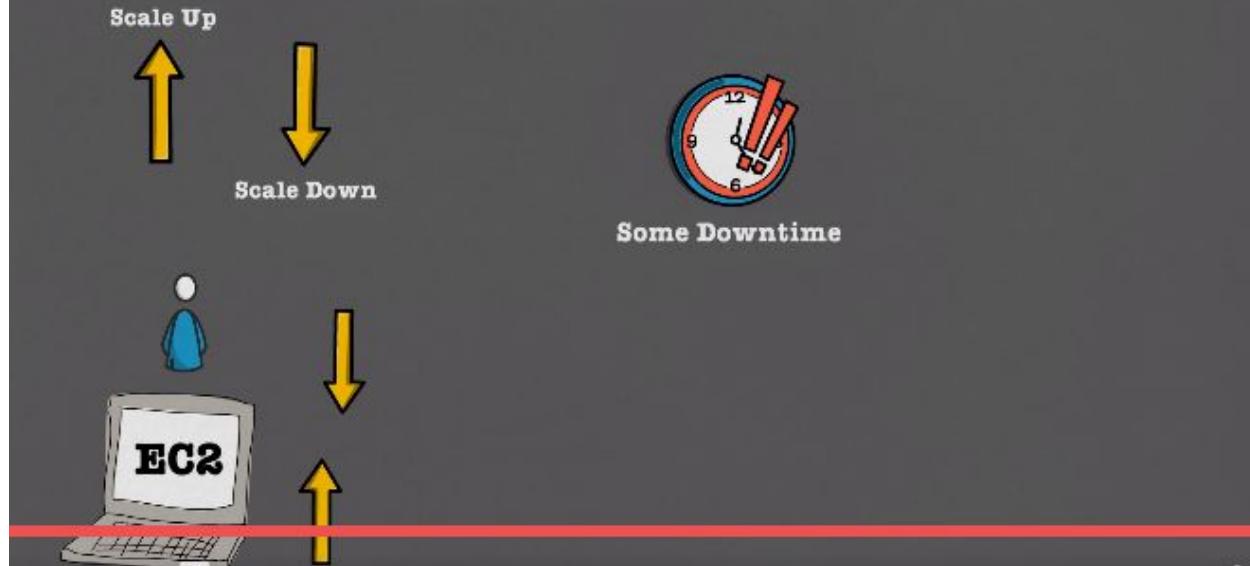
Horizontal Scaling



Vertical Scaling



Vertical Scaling



Screenshot of the AWS CloudFormation console showing the "Step 1: Choose an Amazon Machine Image (AMI)" step.

The page title is "Step 1: Choose an Amazon Machine Image (AMI)". The navigation bar includes "Services", "Resource Groups", "1. Choose AMI", "2. Choose Instance Type", "3. Configure Instance", "4. Add Storage", "5. Add Tags", "6. Configure Security Group", "7. Review", "Cancel and Exit", and "Kabary", "Ohio", "Support".

The "Quick Start" section shows the following AMIs:

- Amazon Linux 2 AMI (HVM), SSD Volume Type** - ami-0e01ce4ee18447327 (64-bit x86) / ami-032011374ab66a26e (64-bit Arm)
Free tier eligible
Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras.
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
 64-bit (x86)
 64-bit (Arm)
Select
- Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type** - ami-0998bf58313ab53da
Free tier eligible
Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
Select
64-bit (x86)
- Red Hat Enterprise Linux 8 (HVM), SSD Volume Type** - ami-0520e698dd500b1d1 (64-bit x86) / ami-0099847d600887c9f (64-bit Arm)
Free tier eligible
Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
 64-bit (x86)
 64-bit (Arm)
Select
- SUSE Linux Enterprise Server 15 SP1 (HVM), SSD Volume Type** - ami-04c5bab51cc148025 (64-bit x86) / ami-02e73902018018171 (64-bit Arm)
Free tier eligible
SUSE Linux Enterprise Server 15 Service Pack 1 (HVM), EBS General Purpose (SSD) Volume Type. Public Cloud, Advanced Systems Management, Web and Scripting, and Legacy modules enabled.
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
 64-bit (x86)
 64-bit (Arm)
Select

```
[ec2-user@ip-172-31-24-75 ~]$ cat /etc/redhat-release
Red Hat Enterprise Linux release 8.0 (Ootpa)
[ec2-user@ip-172-31-24-75 ~]$ uptime
22:24:35 up 1 min, 1 user, load average: 0.20, 0.15, 0.06
[ec2-user@ip-172-31-24-75 ~]$ sudo dmidecode | head -n 20
# dmidecode 3.2
Getting SMBIOS data from sysfs.
SMBIOS 2.7 present.
11 structures occupying 359 bytes.
Table at 0x0000E801F.

Handle 0x0000, DMI type 0, 24 bytes
BIOS Information
    Vendor: Xen
    Version: 4.2.amazon
    Release Date: 08/24/2006
    Address: 0xE8000
    Runtime Size: 96 kB
    ROM Size: 64 kB
    Characteristics:
        PCI is supported
        EDD is supported
        Targeted content distribution is supported
    BIOS Revision: 4.2
```

[ec2-user@ip-172-31-24-75 ~]\$ █

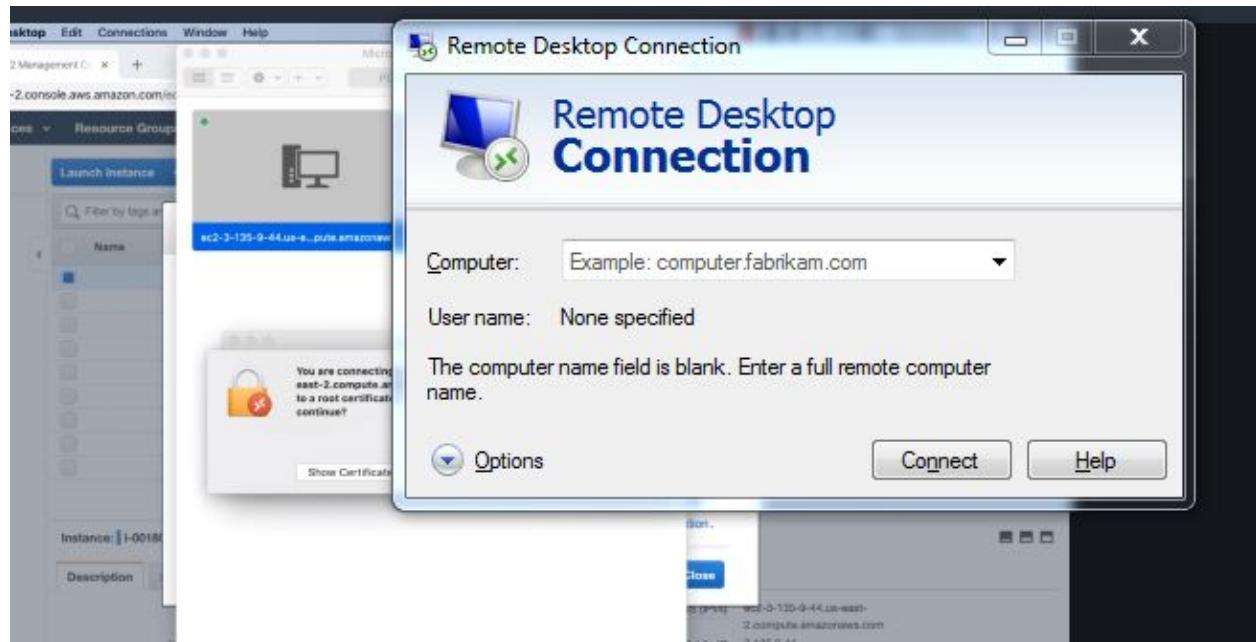
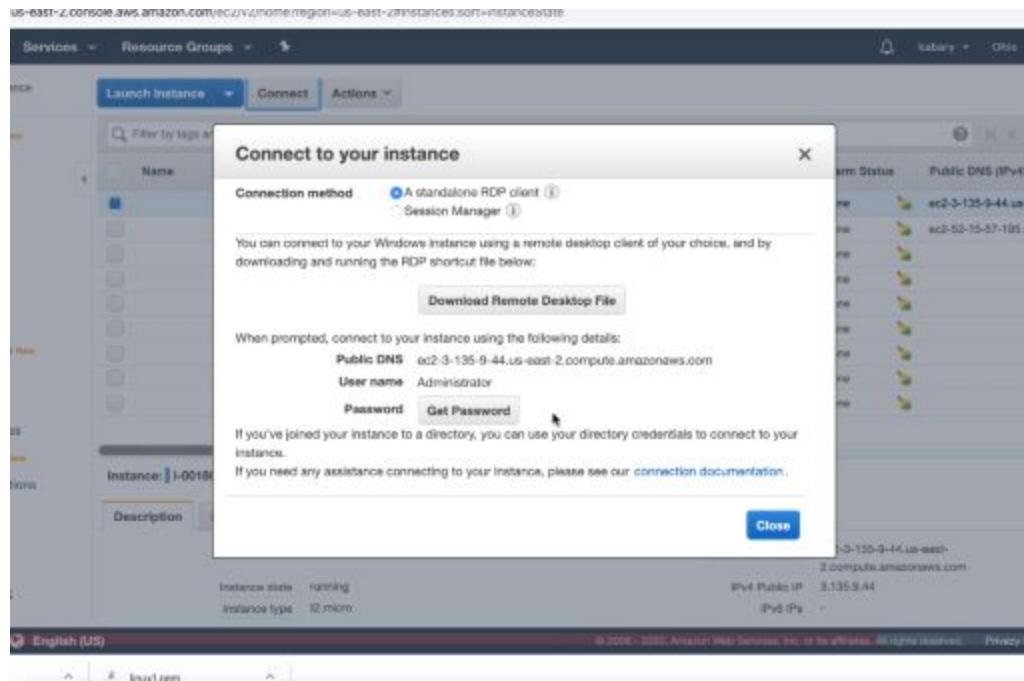
Step 1: Choose an Amazon Machine Image (AMI)

[Cancel and Exit](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace, or you can select one of your own AMIs.

The screenshot shows the 'Quick Start (19)' section of the AWS Launch Wizard. A search bar at the top left contains the text 'Windows'. Below it, a message states: 'AWS Launch Wizard for SQL Server offers an easy way to size, configure, and deploy Microsoft SQL Server Always On availability groups. Use AWS Launch Wizard for this launch?'. The main area displays three AMI options under the 'Windows' category:

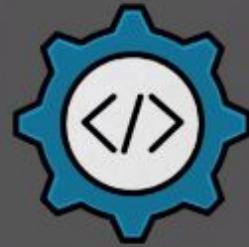
- Microsoft Windows Server 2019 Base - ami-0ce84304ef5b08a22**
Windows
Free tier eligible
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
[Select](#) 64-bit (x86)
- Microsoft Windows Server 2019 Base with Containers - ami-0a0509a48a35a41cf**
Windows
Free tier eligible
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
[Select](#) 64-bit (x86)
- Microsoft Windows Server 2019 with SQL Server 2017 Standard - ami-065d3a8bc01c7d154**
Windows
Free tier eligible
Root device type: ebs Virtualization type: hvm ENA Enabled: Yes
[Select](#) 64-bit (x86)



AWS Compute Services



ECS



Lambda

Amazon EC2

Elastic Compute Cloud



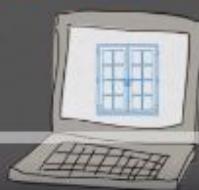
Virtual Machines

Linux



0:33 / 2:43

Windows



EC2 Purchase Options

On Demand

Pay for compute capacity by the hour.



Reserved

1 or 3 year commitment to get discounts.



Spot Pricing

Only run the instance if the market price is lower than your bid price.

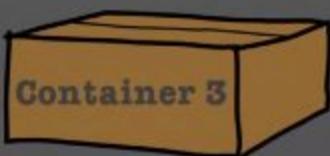
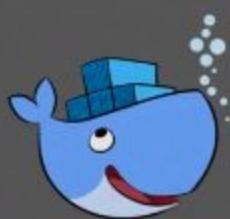
Market Price < Bid Price



Market Price > Bid Price

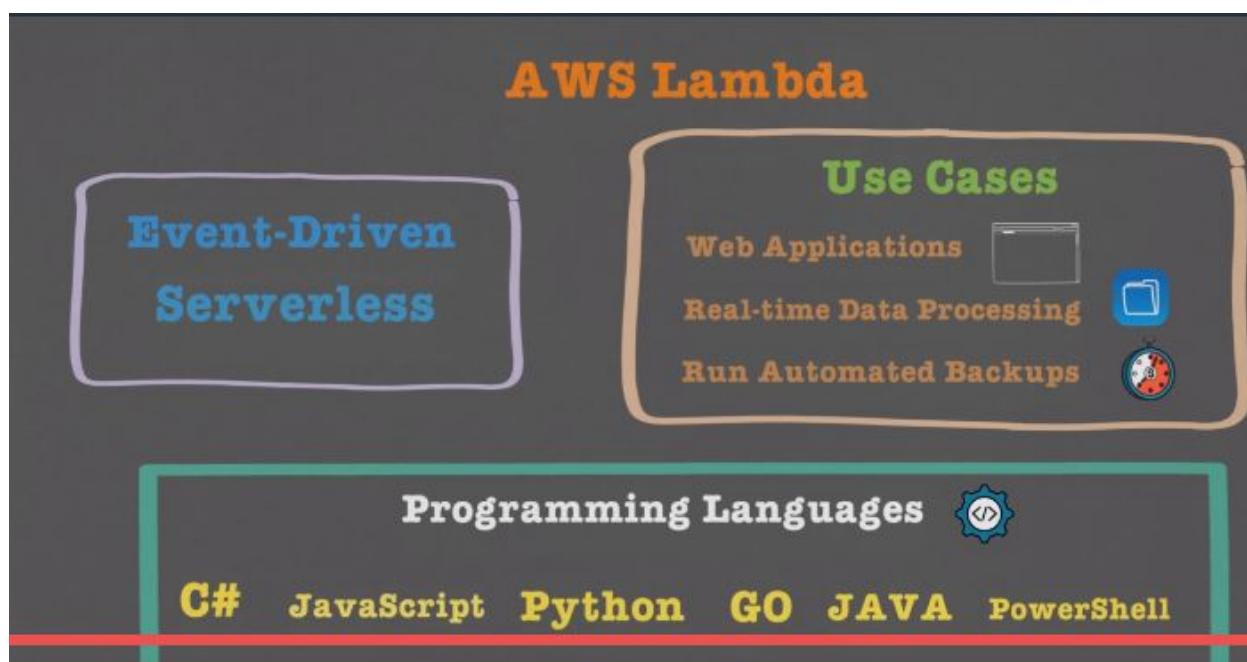
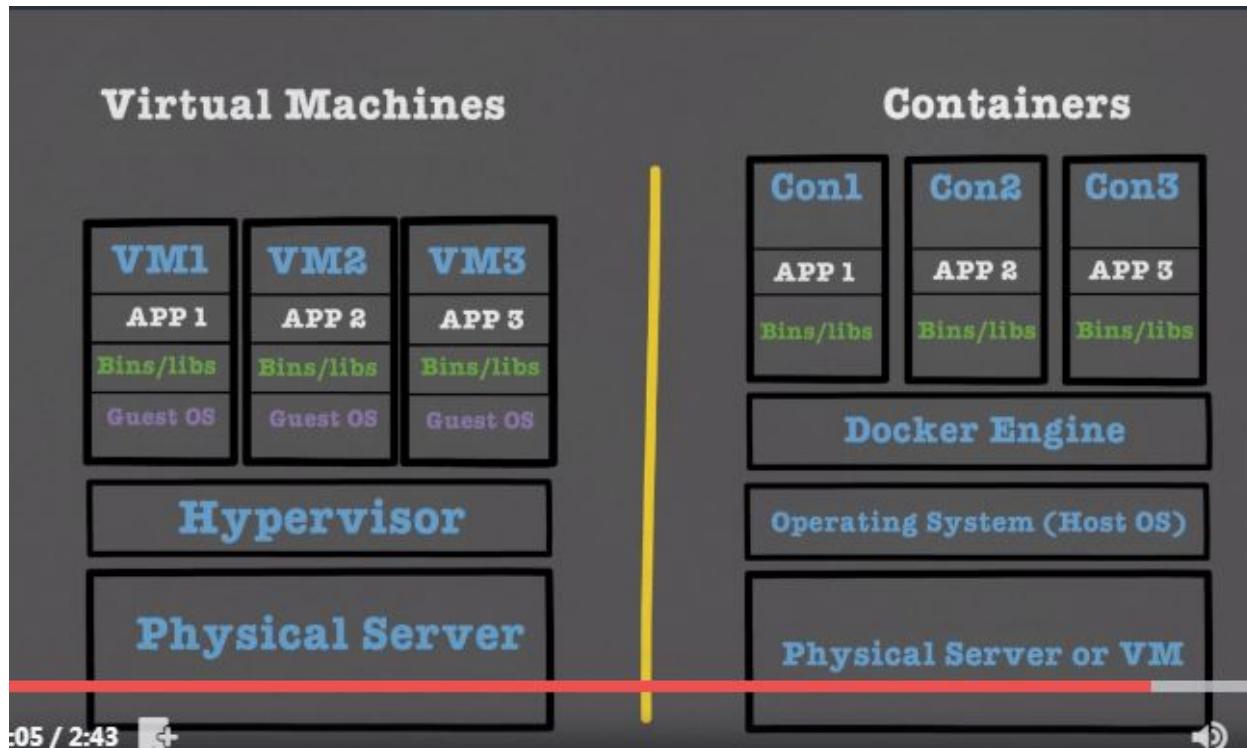
Amazon ECS

Manage and Run Docker Containers



Orchestration





S3- STORAGE SERVICES

Module (4): Storage Services

- Amazon S3
- Amazon Elastic Block Store
- Amazon Elastic File System

Storage Services

Storage Service:

What is the storage service (S3) ?

Amazon S3 is a web service that allows the user to store and reclaim any data from the web at any time.



S3

store and retrieve any amount of data anytime from anywhere,no limit for data storage
5TB AND COMMAND LINE 5GB .also scale up data.

storage service (S3) structure:

Bucket:

Any folder created in S3.

Folder:

Any sub folder created in a bucket.

Objects:

Files that are stored in a bucket.

Regions:

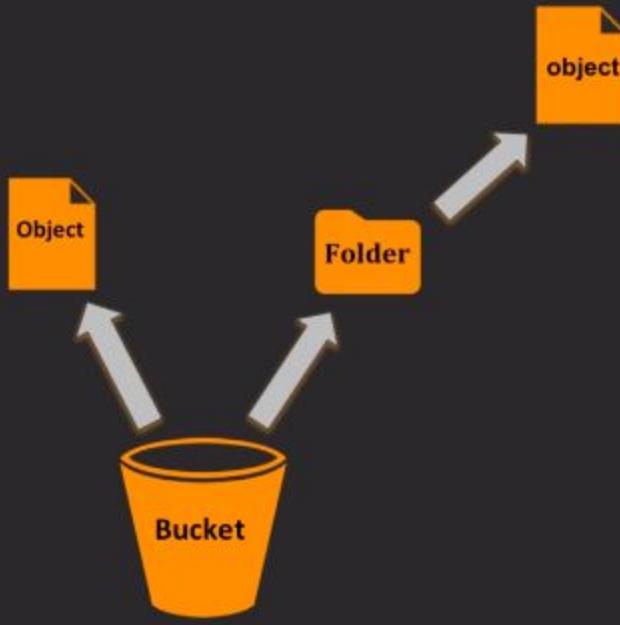
When a user create a bucket, you must select a specific region for it to exist in.

Any data the user upload to the S3 bucket will be physically located in the data center in that region.

It is prefered to select a region that is physically close to you.



The structure of S3:



Buckets & Folders:

Creating an S3 Bucket:

Choose a bucket name (must follow some certain rules) and select the region.

Upload an object to a Bucket.

Create a folder in a Bucket.

Storage (S3) classes:

What is the storage class ?

- 1) It is the classification assigned to each object in S3.
- 2) Each storage class has its own characteristics.
- 3) Each object must be assigned a storage class.
- 4) The storage class is changeable and can be edited at any time.



Storage classes types:

- 1) Standard
- 2) Standard-IA
- 3) One zone-IA
- 4) Intelligent-Tiering
- 5) Glacier

Each storage class type has its own characteristics like:

- 1) Storage cost
- 2) Object durability
- 3) Object availability
- 4) Frequency of access

Object Durability:

Is the percent over a one year time period that a file stored in S3 storage will not be lost.

Object availability:

Is the percent over a one year time period that a file stored in S3 will be accessible.

Changing storage class:

- 1) All new objects uploaded to S3 is set to standard storage class by default.
- 2) You need to set the storage class prior or during the upload process of a file
- 3) You can change between all storage classes instantly except the Glacier class.



S3 storage features:

- 1) Object sharing
- 2) Object life cycles
- 3) Object versioning



Some other features:

- 1) Security
- 2) Durable and scalable.
- 3) Reliable



EBS: AMAZON ELASTIC BLOCK STORE- HARD DRIVE

Amazon **Elastic Block Store:**

Amazon Elastic Block Store (EBS) is an easy to use, high performance block storage service designed for use with Amazon Elastic Compute Cloud (EC2) for both throughput and transaction intensive workloads at any scale.

A broad range of workloads, such as relational and non-relational databases, enterprise applications, containerized applications, big data analytics engines, file systems, and media workflows are widely deployed on Amazon EBS.



Block Storage

Benefits of Amazon **Elastic Block Store:**

- 1) Performance for any workload
- 2) Easy to Use
- 3) Highly available and durable
- 4) Virtually unlimited scale
- 5) Secure
- 6) Cost-effective



EFS - AMAZON ELASTIC FILE SYSTEM

Amazon Elastic File System:

Amazon Elastic File System (Amazon EFS) provides a simple, scalable, fully managed elastic NFS file system for use with AWS Cloud services and on-premises resources.

It is built to scale on demand to petabytes without disrupting applications, growing and shrinking automatically as you add and remove files, eliminating the need to provision and manage capacity to accommodate growth.



Elastic File System

Benefits of Amazon Elastic File System:

- 1) Scalable Performance
- 2) Dynamic elasticity
- 3) Fully Managed
- 4) Cost-effective
- 5) Security and compliance



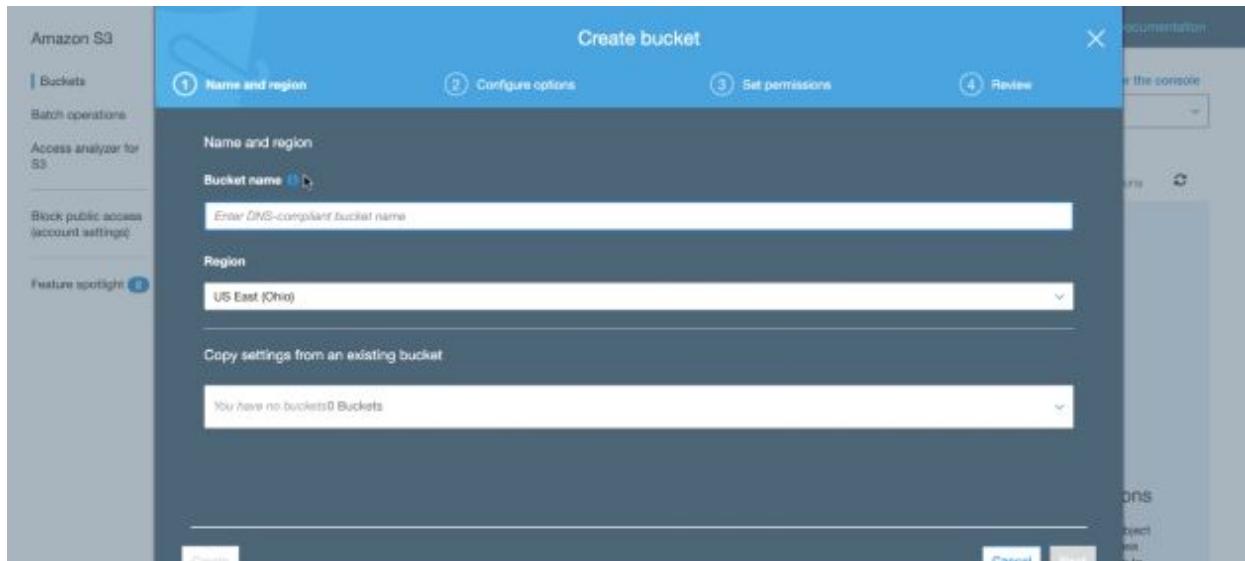
Files stored across multiple availability zone with in a region

Storage capacity is automatically add or down as you use and move files

Pay for what you use

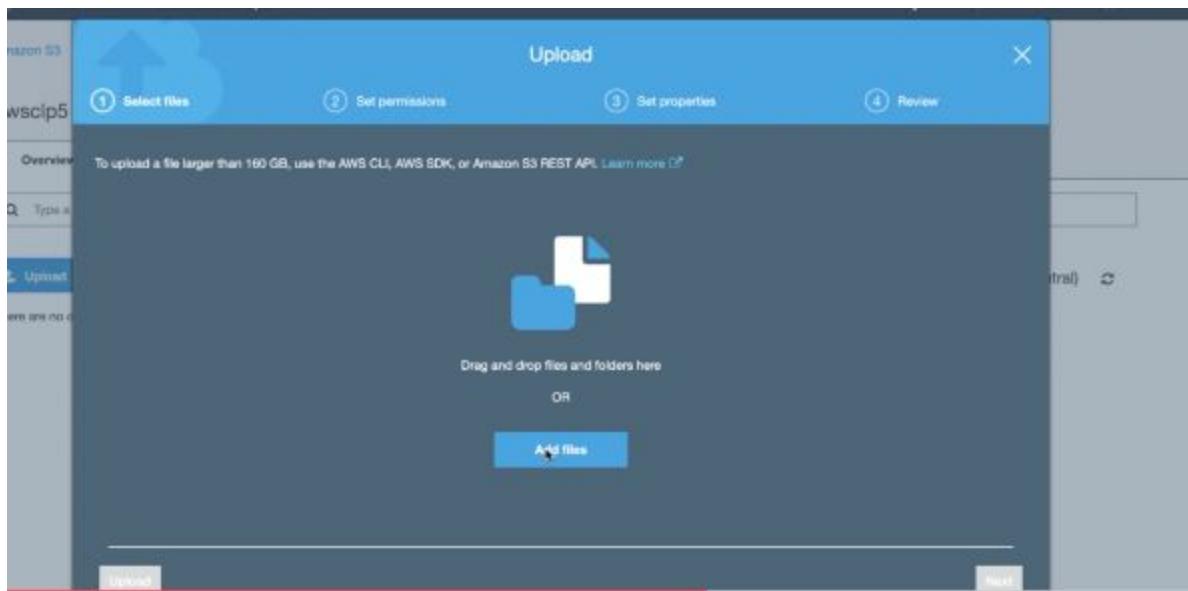
EFS you can share file between between regions or multiple efs instances

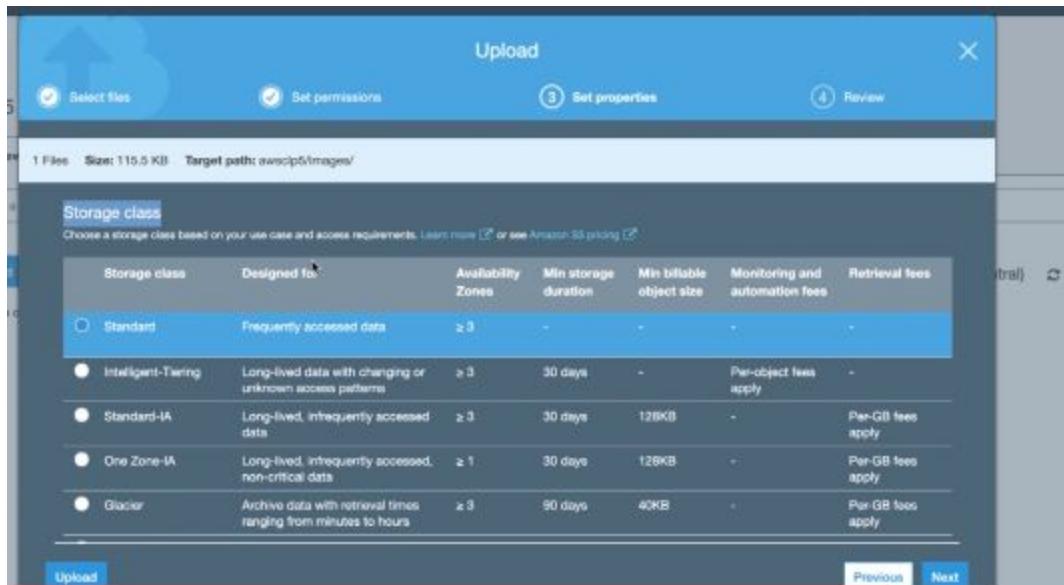
EBS you can share file available in particular region and is not shared



Bucket name is globally unique and access can be public or private

The screenshot shows the 'S3 buckets' list page. It displays one bucket named 'awsclp5'. The bucket details are: Access level 'Objects can be public', Region 'Canada (Central)', and Date created 'Mar 14, 2020 9:03:43 PM GMT-0600'. The top navigation bar includes 'Discover the console' and search/filter options for 'Search for buckets' and 'All access types'.





The screenshot shows the AWS S3 object details for 'penguin.png'. The object was uploaded on Mar 14, 2020, at 9:04:47 PM GMT-0600. It has an Etag of f050c48ba3504414279a860eda1f54 and is stored in the Standard storage class. The object key is 'images/penguin.png' and its URL is <http://awsclp5.s3.ca-central-1.amazonaws.com/images/penguin.png>.

Owner: d8468a972a82d7309f72c02006299b07620b60dc938169eb1107c28130273b91

Last modified: Mar 14, 2020 9:04:47 PM GMT-0600

Etag: f050c48ba3504414279a860eda1f54

Storage class: Standard

Server-side encryption: None

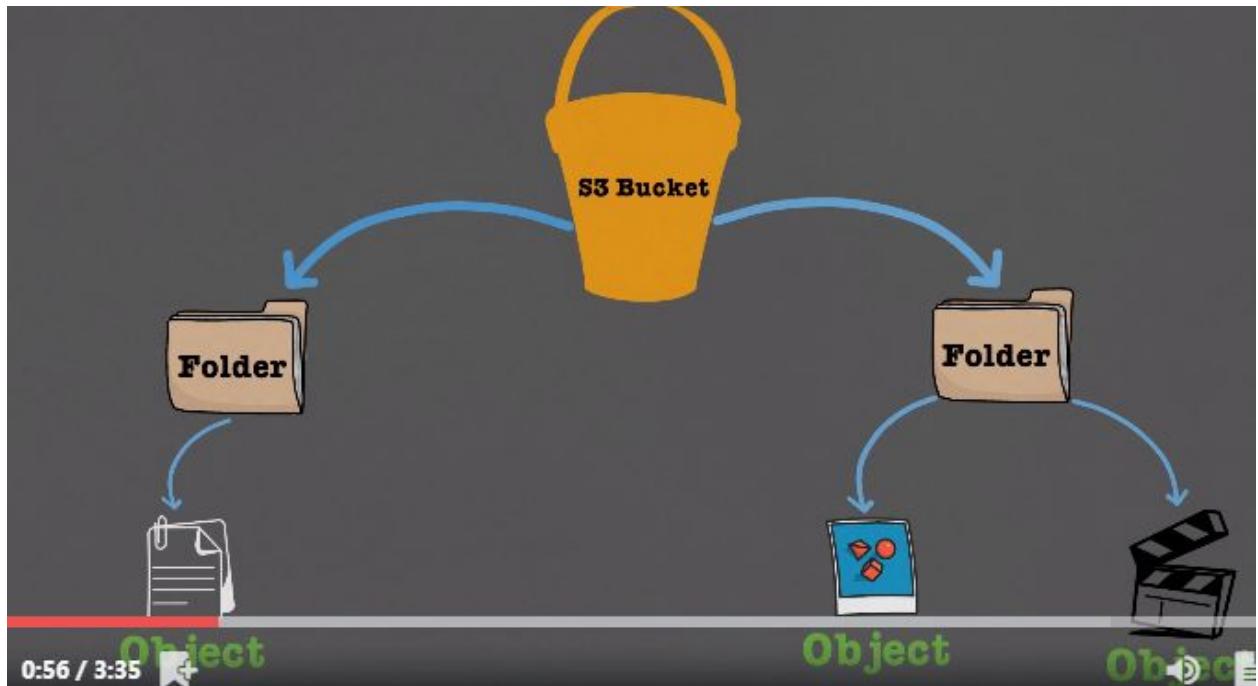
Size: 115.5 KB

Key: images/penguin.png

Object URL: <http://awsclp5.s3.ca-central-1.amazonaws.com/images/penguin.png>

AWS Storage Services





S3 Storage Classes

Standard

Frequent Accessed Data
Multiple Availability Zones

Standard-IA

Infrequently Accessed Data
Multiple Availability Zones

One Zone-IA

Infrequently Accessed Data
Only One Availability Zone

Glacier

Rarely Accessed Data
Archive
Low Cost

Intelligent-Tiering

Unknown
Changing Access
Automatic Assignment

Elastic Block Store

EBS



EBS



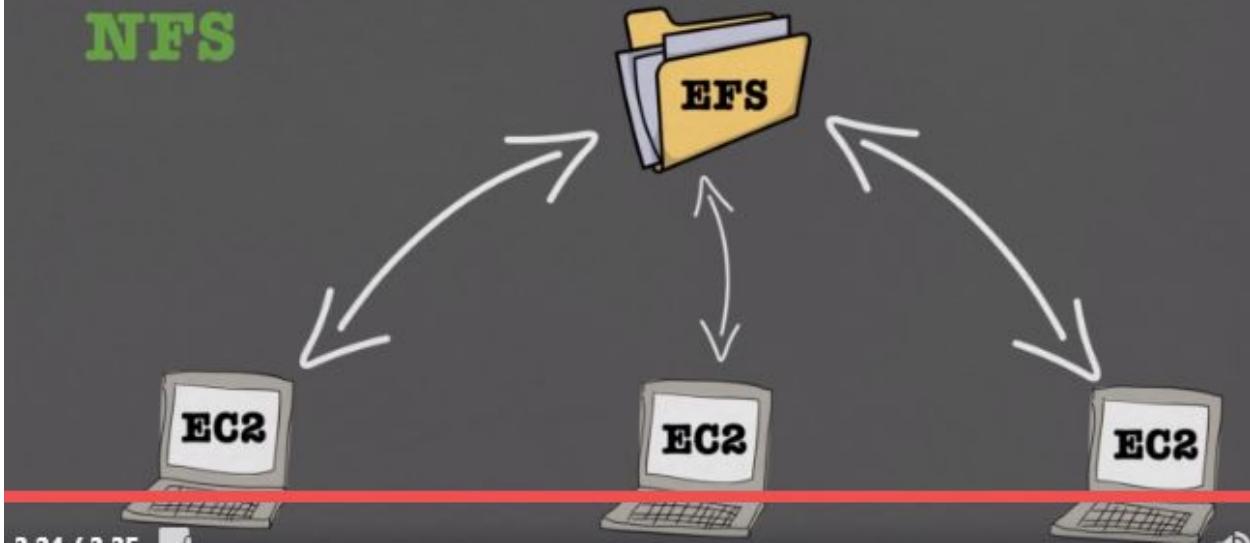
EBS



Solid state drive and Hard disk drive

Elastic File System

NFS



Share file between many devices

Module (5): DNS Services

- Route 53
- Amazon CloudFront
- Network Services VPC

ROUTE 53 - DOMAIN NAME SYSTEM DNS

Day 5: DNS Services

What is Route 53 ?

Amazon Route 53 is a highly available and scalable cloud Domain Name System (DNS) web service. It is designed to give developers and businesses an extremely reliable and cost effective way to route end users to Internet applications by translating names like www.example.com into the numeric IP addresses like 192.0.2.1 that computers use to connect to each other.



[Register a domain or search domain](#)

Amazon Route 53 effectively connects user requests to infrastructure running in AWS – such as Amazon EC2 instances, Elastic Load Balancing load balancers, or Amazon S3 buckets – and can also be used to route users to infrastructure outside of AWS.

You can use Amazon Route 53 to configure DNS health checks to route traffic to healthy endpoints or to independently monitor the health of your application and its endpoints.

Amazon Route 53 Traffic Flow makes it easy for you to manage traffic globally through a variety of routing types, including Latency Based Routing, Geo DNS, Geo Proximity, and Weighted Round Robin—all of which can be combined with DNS Failover in order to enable a variety of low-latency, fault-tolerant architectures.

Using Amazon Route 53 Traffic Flow's simple visual editor, you can easily manage how your end-users are routed to your application's endpoints—whether in a single AWS region or distributed around the globe.

Amazon Route 53 also offers Domain Name Registration – you can purchase and manage domain names such as example.com and Amazon Route 53 will automatically configure DNS settings for your domains.



Main Benefits:

- 1) Domain Registration: which is registering domain names
- 2) Domain Name System (DNS): which is translating domain names into IP Addresses .
- 3) Health checking: which is sending automated requests over the internet to your application to make sure if it is reachable, available and functional or not.



CLOUDFRONT - CONTENT DELIVERY NETWORK- CDN

Amazon CloudFront

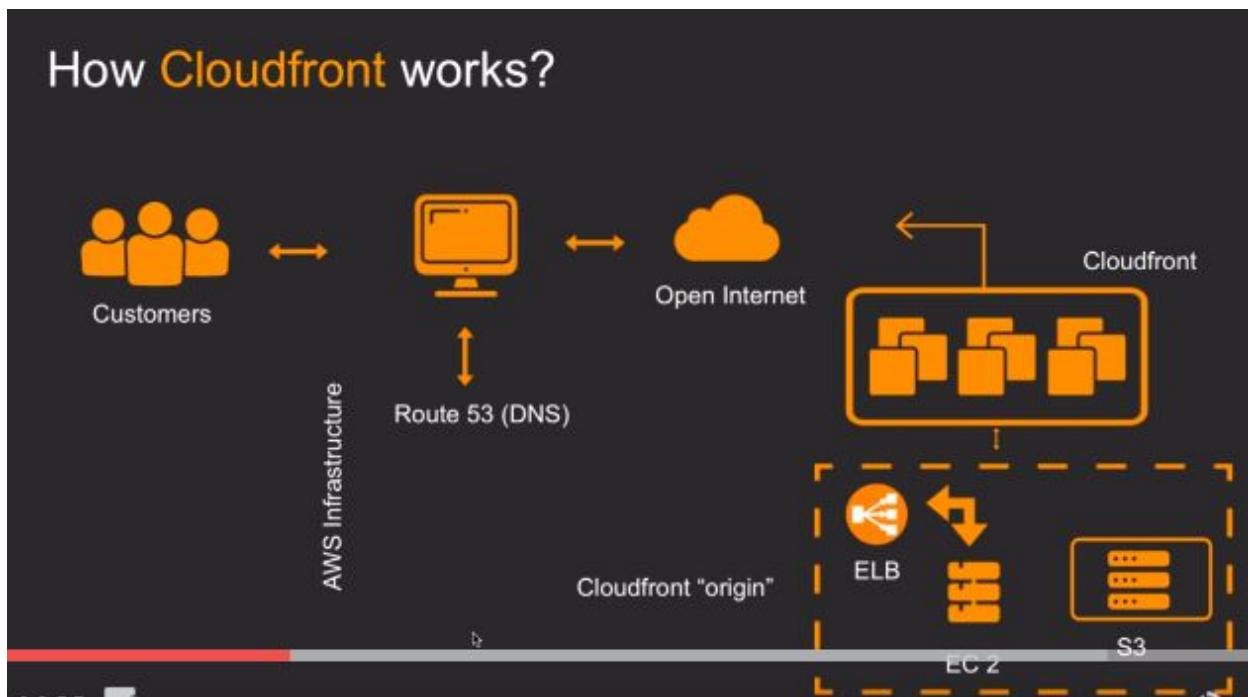
Amazon CloudFront is a fast content delivery network (CDN) service that securely delivers data, videos, applications, and APIs to customers globally with low latency, high transfer speeds, all within a developer-friendly environment.

CloudFront is integrated with AWS – both physical locations that are directly connected to the AWS global infrastructure, as well as other AWS services.

CloudFront works seamlessly with services including AWS Shield for DDoS mitigation, Amazon S3, Elastic Load Balancing or Amazon EC2 as origins for your applications, and Lambda@Edge to run custom code closer to customers' users and to customize the user experience. Lastly, if you use AWS origins such as Amazon S3, Amazon EC2 or Elastic Load Balancing, you don't pay for any data transferred between these services and CloudFront.

Allow you store or cache the content at edge location over the world,its allow customer to access the content quickly and give you extra security

How Cloudfront works?



Edge location may be any place in the world, increase performance and reduce latency. also cloud front will protect you from crash. request of customer come to edge location not to root server. edge location will protect from crash

Summary of Cloudfront benefits:

- 1) Cache content at Edge locations for fast distribution to customers.
- 2) Built-in distributed denial of service (DDoS) attack protection.
- 3) Integrates with many AWS services (S3, EC2, ELB, Route 53 and Lambda)



ELB- ELASTIC LOAD BALANCING AND AUTO SCALING

Distribute traffic among the ec2 instances associated to it.

3 type Application ELB, Network ELB, Classic ELB

Elastic load balancing & auto scaling:

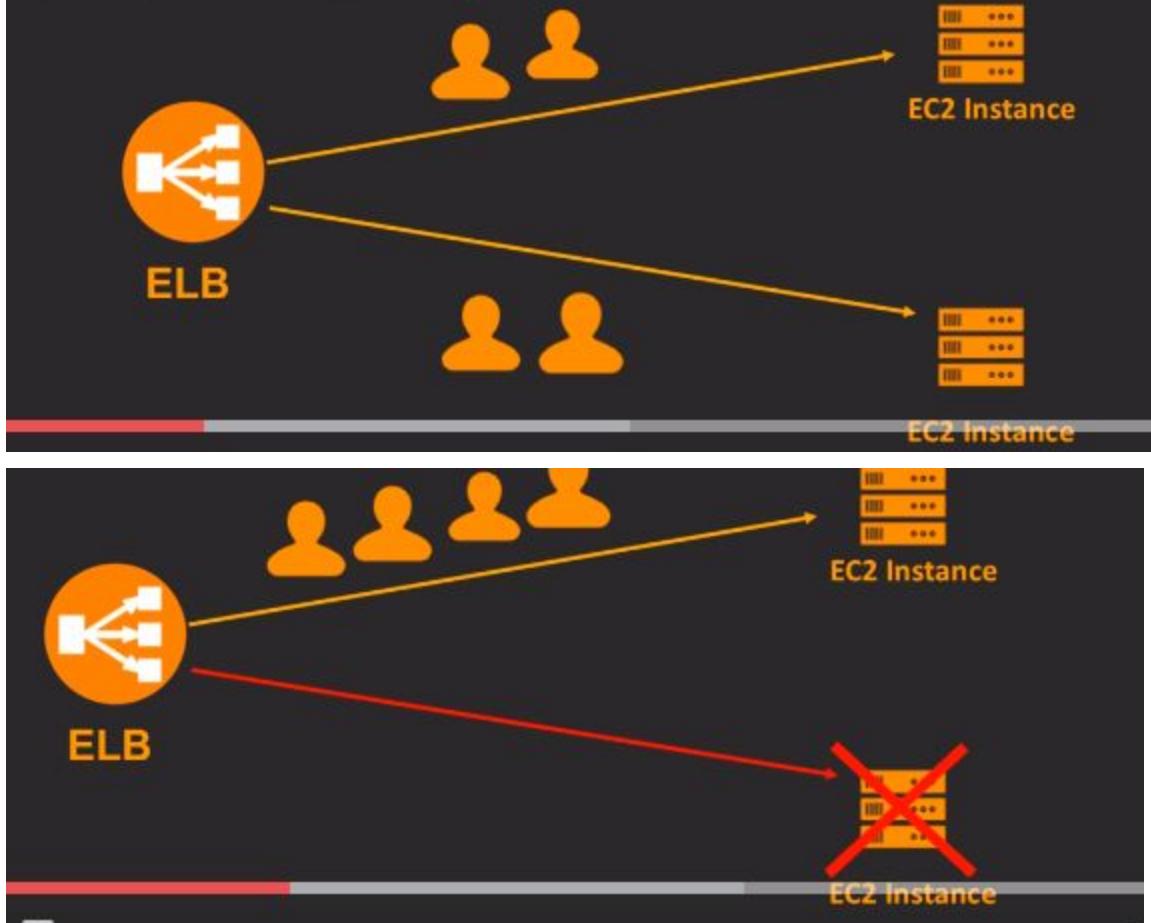
Elastic Load Balancing:

Elastic Load Balancing automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, IP addresses, and Lambda functions. 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.



How Elastic load balancer works

In case a server crashes, the Elastic load balancer (**ELB**) will reroute all traffic (users) to the working server(s).



Auto Scaling

AWS Auto Scaling monitors your applications and automatically adjusts capacity to maintain steady, predictable performance at the lowest possible cost. Using AWS Auto Scaling, it's easy to setup application scaling for multiple resources across multiple services in minutes.

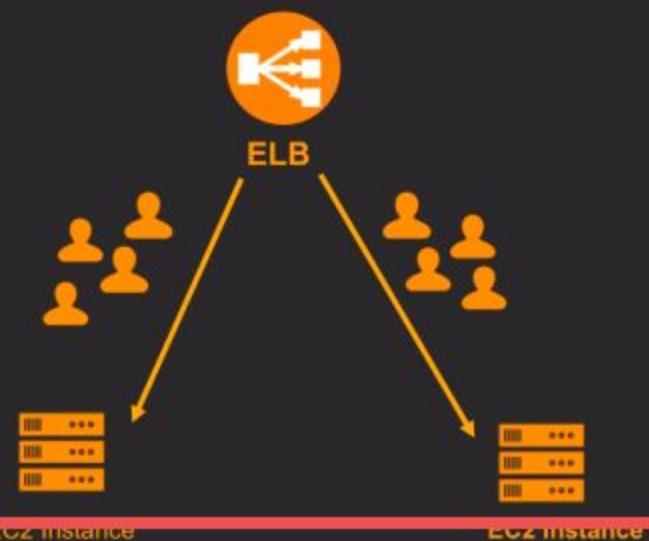
The service provides a simple, powerful user interface that lets you build scaling plans for resources including Amazon EC2 instances and Spot Fleets, Amazon ECS tasks, Amazon DynamoDB tables and indexes, and Amazon Aurora Replicas.

AWS Auto Scaling makes scaling simple with recommendations that allow you to optimize performance, costs, or balance between them.

If you're already using Amazon EC2 Auto Scaling to dynamically scale your Amazon EC2 instances, you can now combine it with AWS Auto Scaling to scale additional resources for other AWS services.

With AWS Auto Scaling, your applications always have the right resources at the right time.

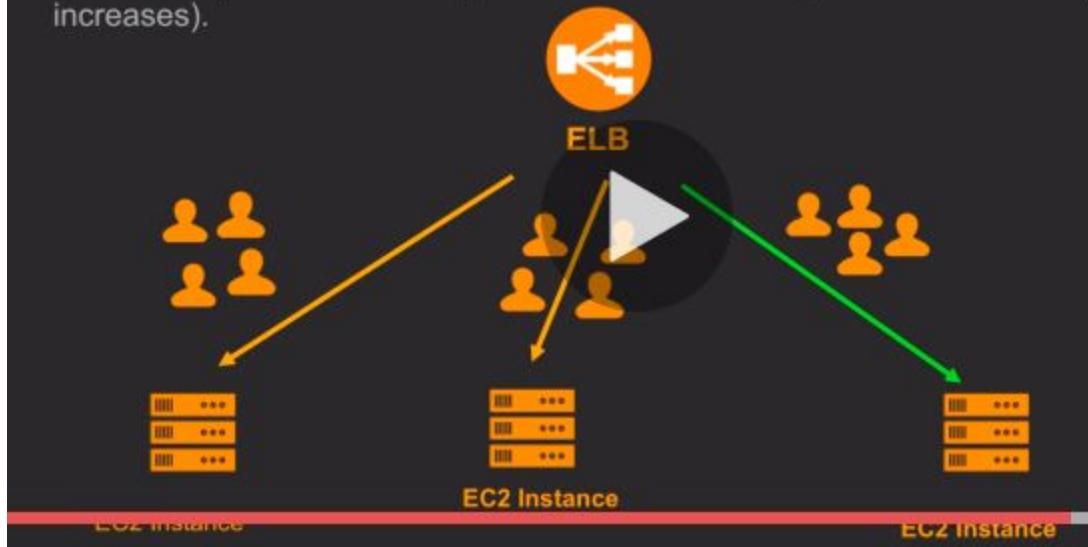
How Auto Scaling works?



Add additional server based on demand its increase and decrease

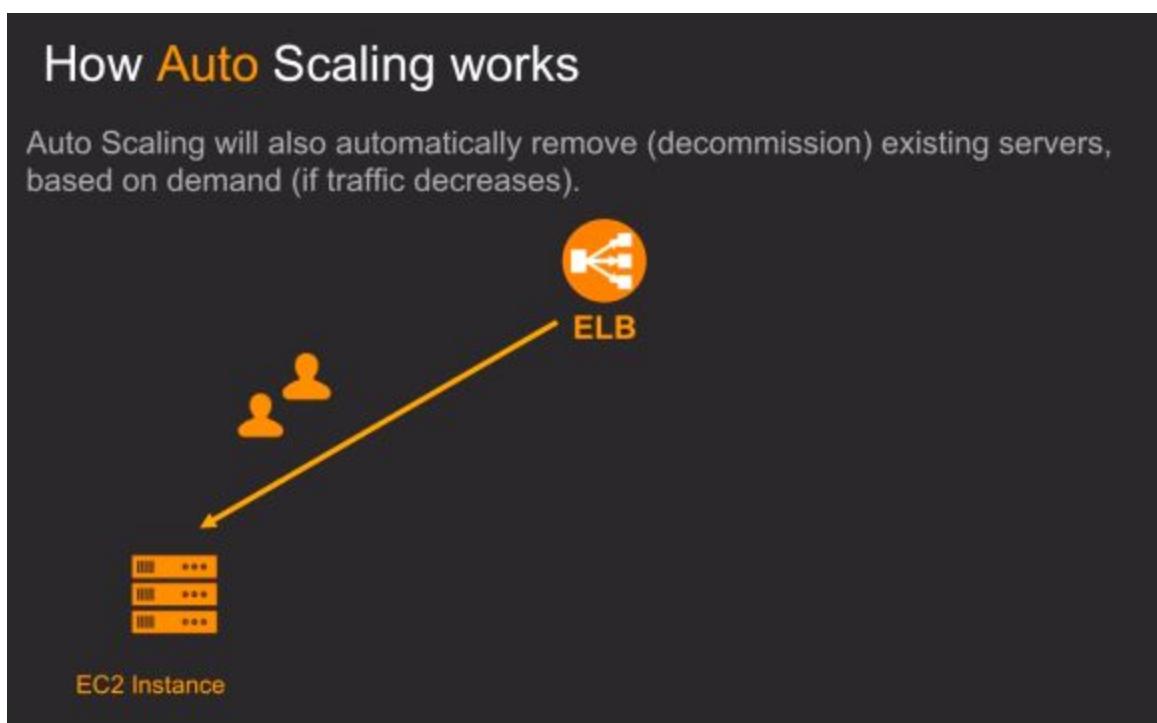
How Auto Scaling works

Auto Scaling will automatically add additional servers, based on demand (increases).



How Auto Scaling works

Auto Scaling will also automatically remove (decommission) existing servers, based on demand (if traffic decreases).



Horizontal vs Vertical Scaling

Horizontal Scaling

adding more machines into your pool of resources. For example, adding more EC2 instances.



Vertical Scaling

adding more power (CPU, RAM) to an existing machine. For example, adding more memory to an existing EC2 instance.



VPC - VIRTUAL PRIVATE CLOUD -NETWORK SERVICES

Network Services:

What is the Virtual Private Cloud (VPC) ?

An elastic network populated by infrastructure, platform, and application services that share common security and interconnection.

Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.

A default VPC is built for every AWS account created.

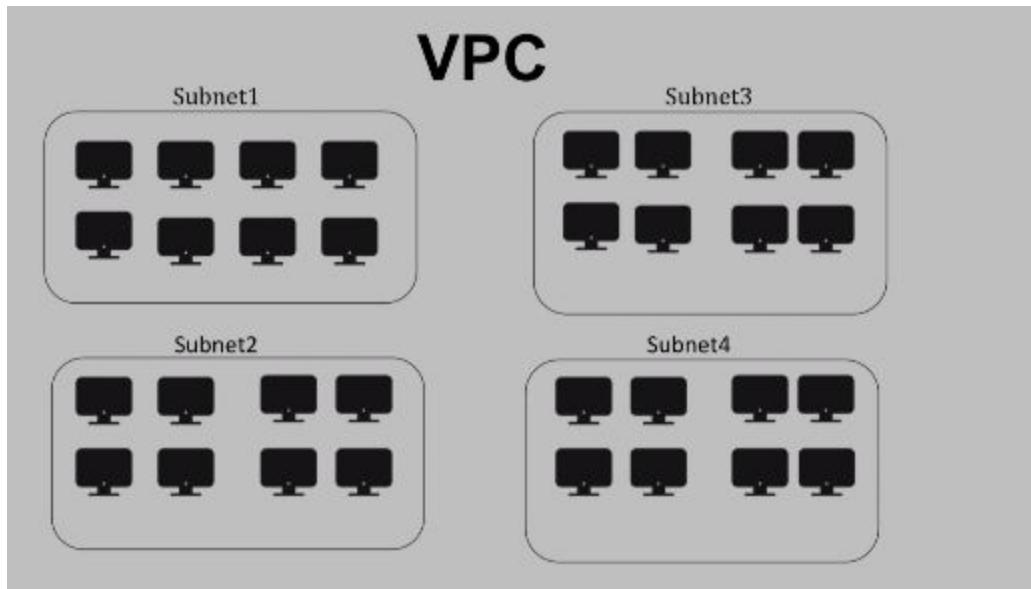


Subnets:

A subnet is a subsection of a network within a VPC.

An Analogy: a University might have a large dedicated network (VPC), and each faculty in the university will have its own subnet. So there will be a subnet for the engineering faculty, a different subnet for nursing faculty, ... etc.





Networking Security:

What is a Firewall ?

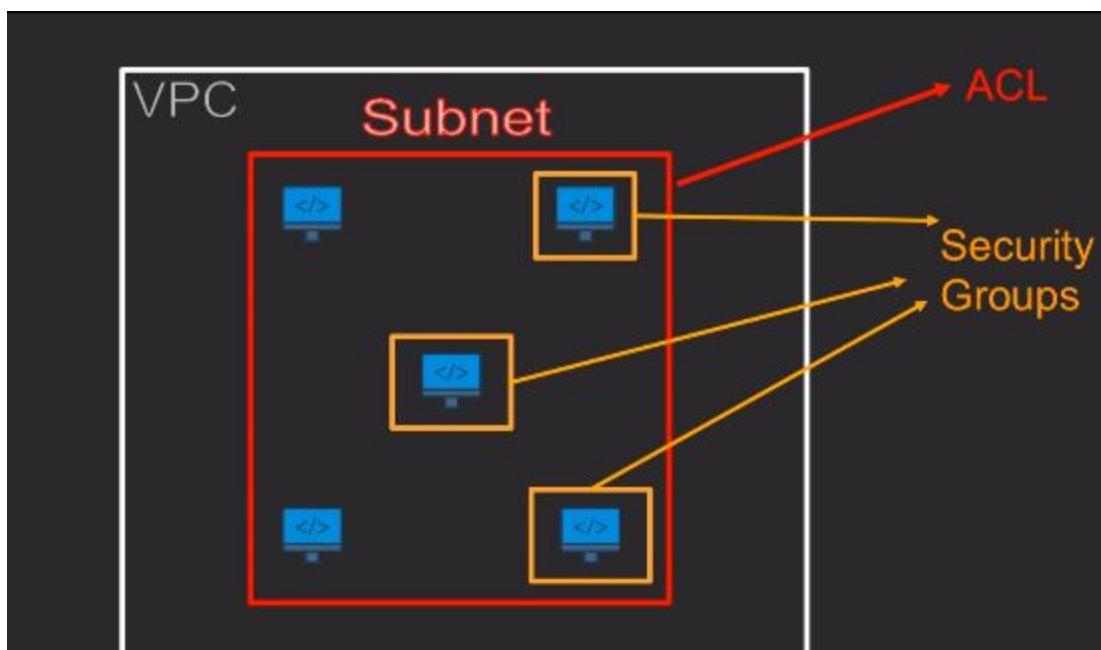
A firewall is a software that either allows or blocks certain kinds of internet traffic to pass through it.

There are two types of firewall in AWS:

Network Access Control List (ACL) is a firewall that operates on a **subnet level**.

Security Group is a firewall that operates on a **server-instance level**.

Example of Networking Security:



Each subnet have its own security group firewall layer at server level
AMAZON VPN

Amazon **VPN Gateway**:

AWS Virtual Private Network (AWS VPN) lets you establish a secure and private encrypted tunnel from your network or device to the AWS global network.

AWS **VPN** is comprised of two services:

- 1) AWS Site-to-Site VPN
- 2) AWS Client VPN.



AWS Site to Site VPN:

AWS Site-to-Site VPN enables you to securely connect your on-premises network or branch office site to your Amazon Virtual Private Cloud (Amazon VPC).

Benefits:

- 1) Highly Available
- 2) Secure
- 3) Robust Monitoring
- 4) Accelerate Applications



How it works?



AWS Client VPN:

AWS Client VPN enables you to securely connect users to AWS or on-premises networks. AWS Client VPN includes a free client-side application, which provides access to AWS services from remote networks.

Benefits:

- 1) Fully Managed
- 2) Elastic
- 3) Access any resource from any Location



How It works?



AMAZON DIRECT CONNECT

Amazon Direct Connect:

AWS Direct Connect is a cloud service solution that makes it easy to establish a dedicated network connection from your premises to AWS.



Amazon Direct Connect

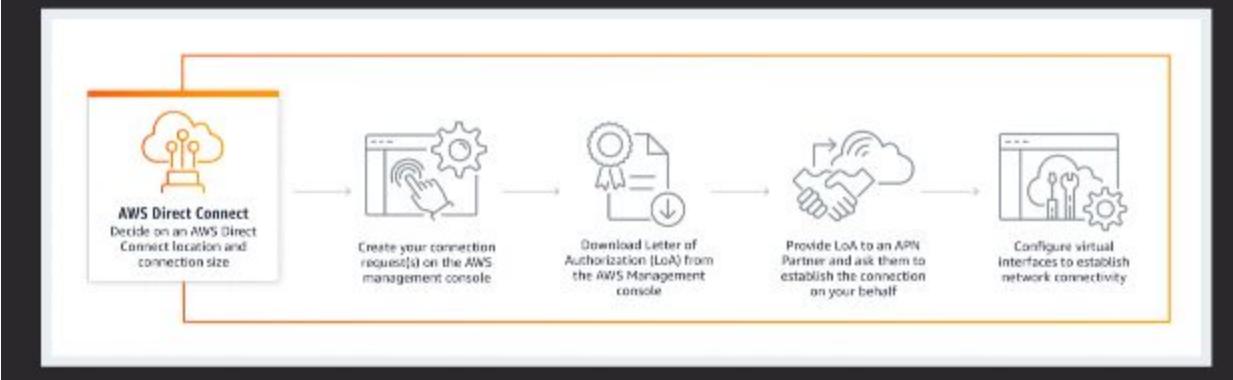
private connectivity between aws and your office data center,reduce network cost,increase bandwidth,provide more constant network,speed.

Benefits:

- 1) PRIVATE CONNECTIVITY TO YOUR AMAZON VPC
- 2) ELASTIC
- 3) SIMPLE
- 4) COMPATIBLE WITH ALL AWS SERVICES
- 5) CONSISTENT NETWORK PERFORMANCE
- 6) REDUCES YOUR BANDWIDTH COSTS



How it works?



```

[ec2-user@ip-172-31-24-75 ~]$ cat /etc/redhat-release
Red Hat Enterprise Linux release 8.0 (Ootpa)
[ec2-user@ip-172-31-24-75 ~]$ sudo su -
Last login: Sun Mar 15 23:43:02 UTC 2020 on pts/0
[root@ip-172-31-24-75 ~]# yum install httpd
Last metadata expiration check: 1:14:30 ago on Sun 15 Mar 2020 10:32:43 PM UTC.
Dependencies resolved.

Package           Arch      Version            Repository      Size
=====
Installing:
httpd             x86_64   2.4.37-16.module+el8.1.0+4134+e6bad0ed  rhui-rhel-8-appstream-rhui-rpms  1.4 M
Installing dependencies:
apr-util          x86_64   1.6.1-6.el8          rhui-rhel-8-appstream-rhui-rpms  105 k
mod_http2         x86_64   1.11.3-3.module+el8.1.0+4134+e6bad0ed  rhui-rhel-8-appstream-rhui-rpms  158 k
apr               x86_64   1.6.3-9.el8          rhui-rhel-8-appstream-rhui-rpms  125 k
httpd-tools        x86_64   2.4.37-16.module+el8.1.0+4134+e6bad0ed  rhui-rhel-8-appstream-rhui-rpms  103 k
httpd-filesystem  noarch   2.4.37-16.module+el8.1.0+4134+e6bad0ed  rhui-rhel-8-appstream-rhui-rpms  35 k
mailcap           noarch   2.1.48-3.el8          rhui-rhel-8-baseos-rhui-rpms    39 k
redhat-logos-httpd noarch   81.1-1.el8          rhui-rhel-8-baseos-rhui-rpms    26 k
Installing weak dependencies:
apr-util-bdb      x86_64   1.6.1-6.el8          rhui-rhel-8-appstream-rhui-rpms  25 k
apr-util-openssl  x86_64   1.6.1-6.el8          rhui-rhel-8-appstream-rhui-rpms  27 k

Transaction Summary

Install  10 Packages

Total download size: 2.0 M
Installed size: 6.1 M
Is this ok? [y/N]: y

[root@ip-172-31-24-75 ~]# systemctl start httpd
[root@ip-172-31-24-75 ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
  Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
  Active: active (running) since Sun 2020-03-15 23:47:24 UTC; 3s ago
    Docs: man:httpd.service(8)
Main PID: 26195 (httpd)
  Status: "Started, listening on: port 80"
   Tasks: 213 (limit: 4998)
  Memory: 24.3M
 CGroup: /system.slice/httpd.service
         ├─26195 /usr/sbin/httpd -DFOREGROUND
         ├─26196 /usr/sbin/httpd -DFOREGROUND
         ├─26197 /usr/sbin/httpd -DFOREGROUND
         ├─26198 /usr/sbin/httpd -DFOREGROUND
         ├─26199 /usr/sbin/httpd -DFOREGROUND

Mar 15 23:47:24 ip-172-31-24-75.us-east-2.compute.internal systemd[1]: Starting The Apache HTTP Server...
Mar 15 23:47:24 ip-172-31-24-75.us-east-2.compute.internal systemd[1]: Started The Apache HTTP Server.
Mar 15 23:47:24 ip-172-31-24-75.us-east-2.compute.internal httpd[26195]: Server configured, listening on: port 80
[root@ip-172-31-24-75 ~]# netstat -tulpen | grep -i listen
tcp        0      0 0.0.0.0:22              0.0.0.0:*                LISTEN      0          25946      1147/sshd
tcp6       0      0 ::1:22                 ::*:*                  LISTEN      0          25948      1147/sshd
tcp6       0      0 ::1:80                 ::*:*                  LISTEN      0          78603      26195/httpd
[root@ip-172-31-24-75 ~]#

```

Allow port 80 in security group

```
-- telnet 52.15.57.195 80
mymacbook:~ aabdalmas$ telnet 52.15.57.195 80
Trying 52.15.57.195...
^C
mymacbook:~ aabdalmas$ telnet 52.15.57.195 80
Trying 52.15.57.195...
Connected to ec2-52-15-57-195.us-east-2.compute.amazonaws.com.
Escape character is '^]'.

```

Inbound rules [Info](#)

Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info
SSH	TCP	22	Custom Q 0.0.0.0/0 X	Delete
SSH	TCP	22	Custom Q ::/0 X	Delete
Custom TCP	TCP	80	Anywhere Q 0.0.0.0/0 X ::/0 X	Delete

NACL

VPC Dashboard [Create network ACL](#) Actions [View](#)

Filter by VPC: [Select a VPC](#)

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

Endpoint Services

NAT Gateways

Peering Connections

Security

Network ACLs [View](#) All rules

Rule #	Type	Protocol	Port Range	Source	Allow / Deny
100	All Traffic	All	All	0.0.0.0/0	ALLOW

Edit inbound rules

Network ACL: acl-0b0be8cbdd81c8bd1

Rule #	Type	Protocol	Port Range	Source	Allow / Deny
101	All Traffic	All	All	0.0.0.0/0	ALLOW
100	Custom TCP Rule	TCP [8]	80	0.0.0.0/0	ALLOW

[Add Rule](#) [Cancel](#)

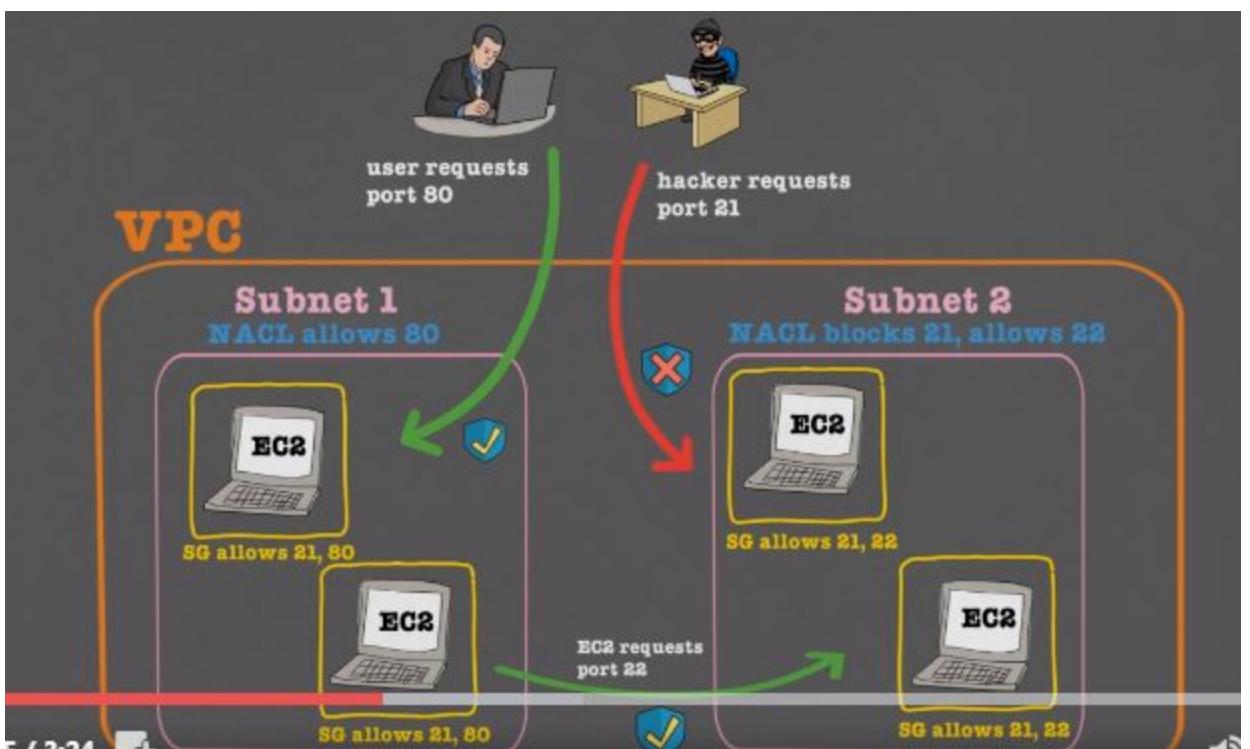
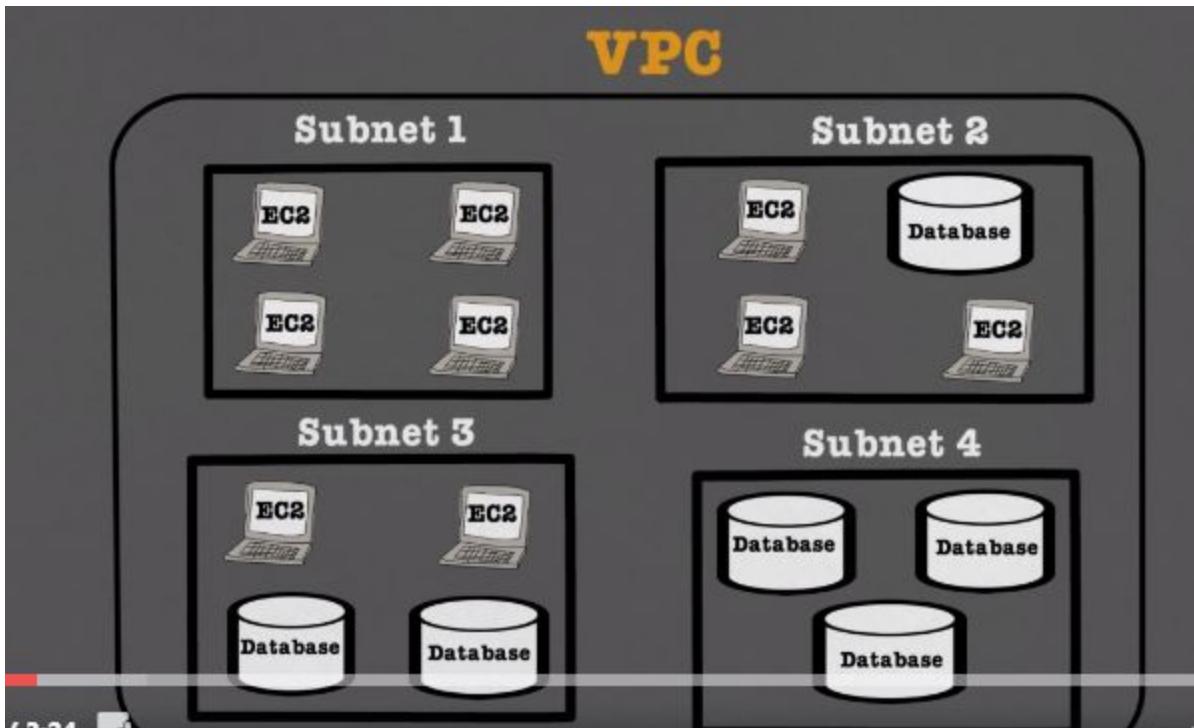
If we deny port 80 in nacl subnet level

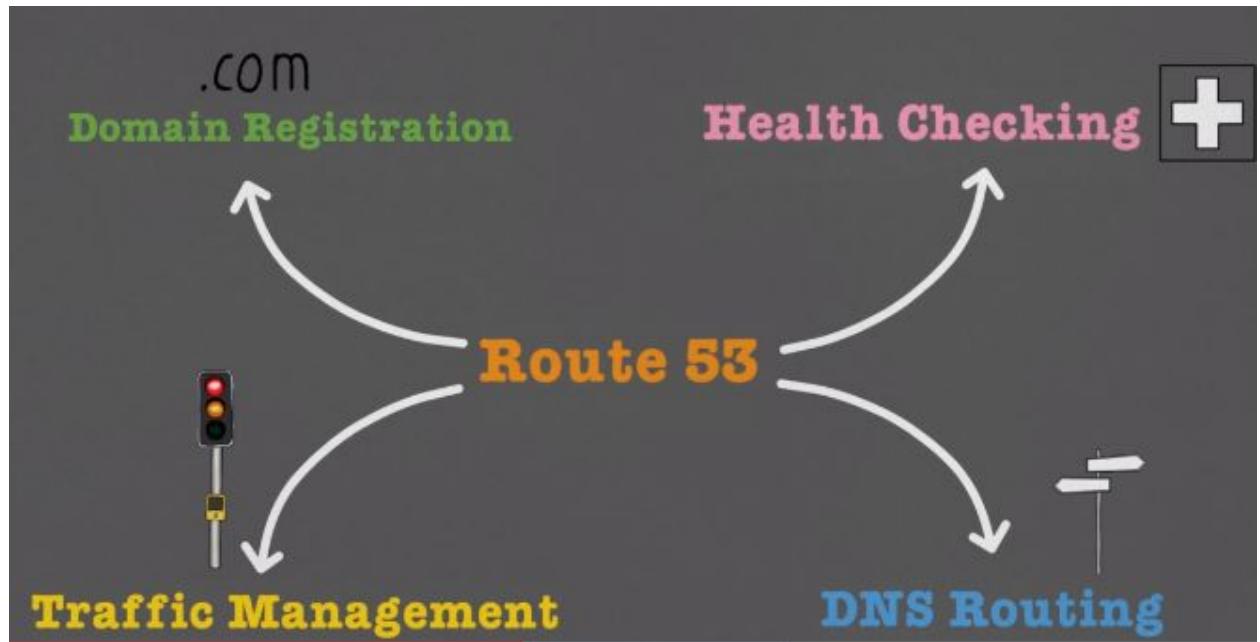
```
mymacbook:~ aabdalma$ telnet 52.15.57.195 80
Trying 52.15.57.195...
^C
mymacbook:~ aabdalma$ telnet 52.15.57.195 80
Trying 52.15.57.195...
Connected to ec2-52-15-57-195.us-east-2.compute.amazonaws.com.
Escape character is '^]'.
Connection closed by foreign host.
mymacbook:~ aabdalma$ telnet 52.15.57.195 80
Trying 52.15.57.195...
```

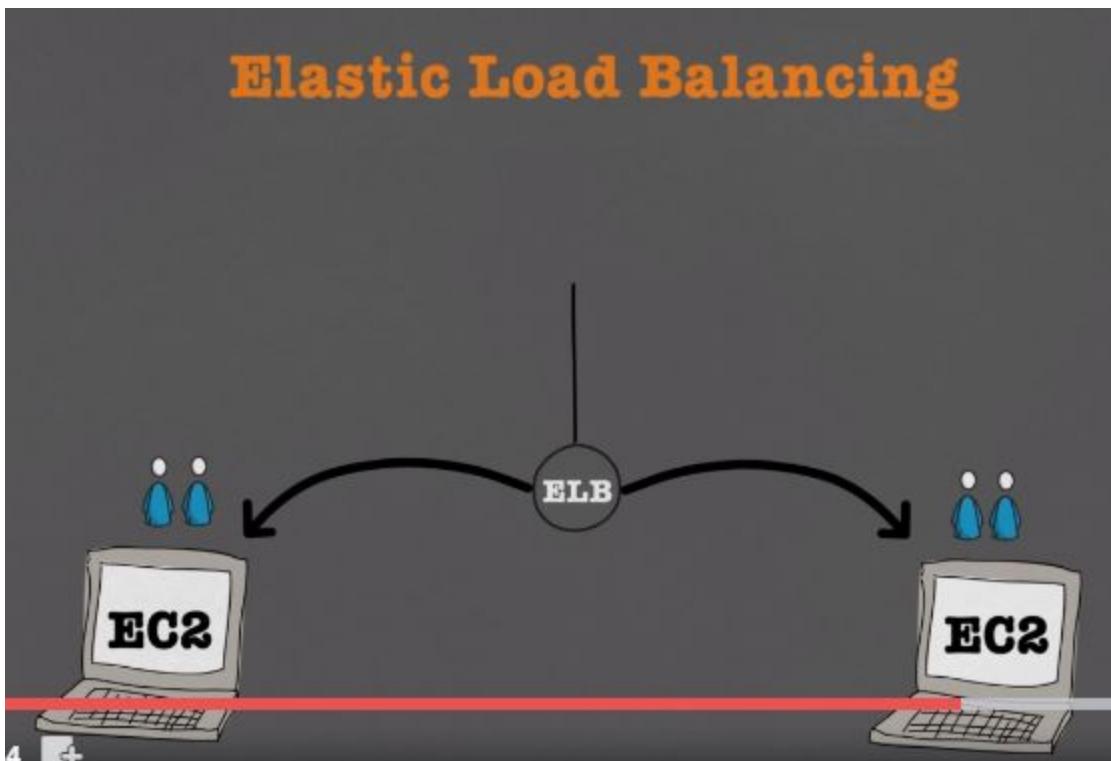
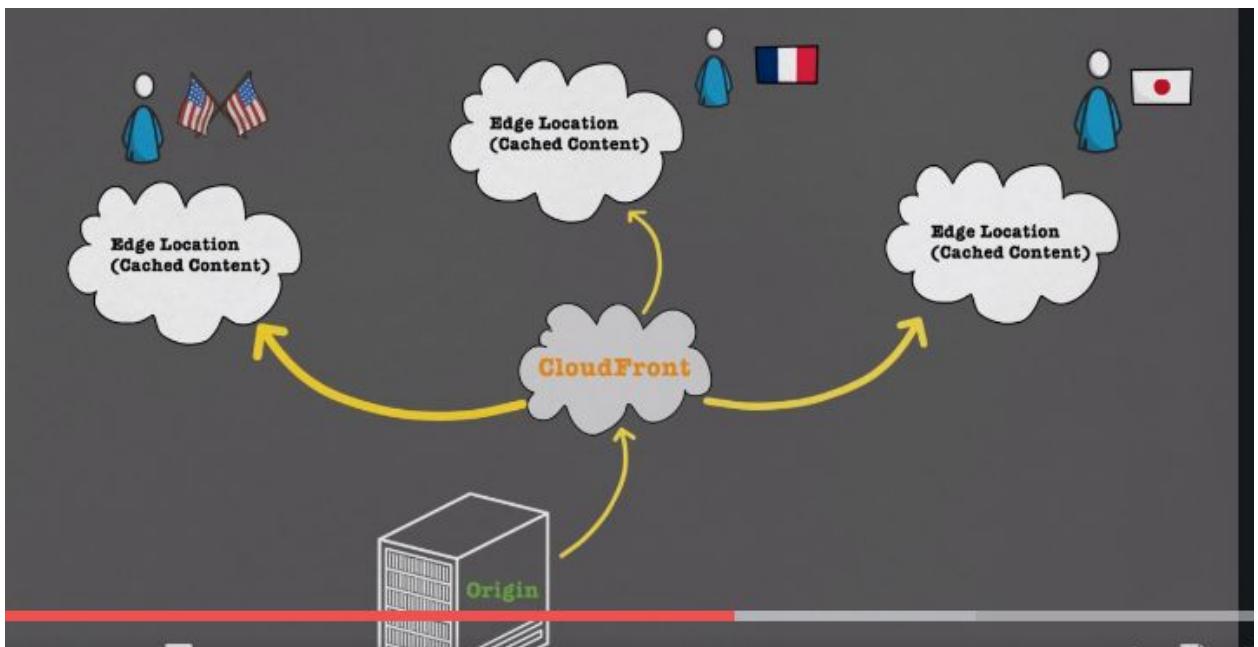
**Even though security group allow port also nacl can block the port traffic
If you allow port 80 in nacl for subnet level then telnet will work**

```
mymacbook:~ aabdalma$ telnet 52.15.57.195 80
Trying 52.15.57.195...
^C
mymacbook:~ aabdalma$ telnet 52.15.57.195 80
Trying 52.15.57.195...
Connected to ec2-52-15-57-195.us-east-2.compute.amazonaws.com.
Escape character is '^]'.
Connection closed by foreign host.
mymacbook:~ aabdalma$ telnet 52.15.57.195 80
Trying 52.15.57.195...
telnet: connect to address 52.15.57.195: Operation timed out
telnet: Unable to connect to remote host
mymacbook:~ aabdalma$ telnet 52.15.57.195 80
Trying 52.15.57.195...
Connected to ec2-52-15-57-195.us-east-2.compute.amazonaws.com.
Escape character is '^]'.
```

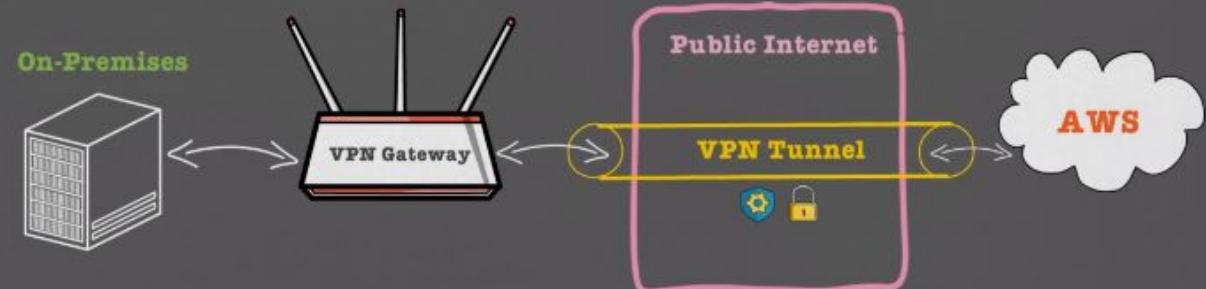
So security group rule only apply to one ec2 virtual machine but nacl has much wider scope because it associate multiple subnet.



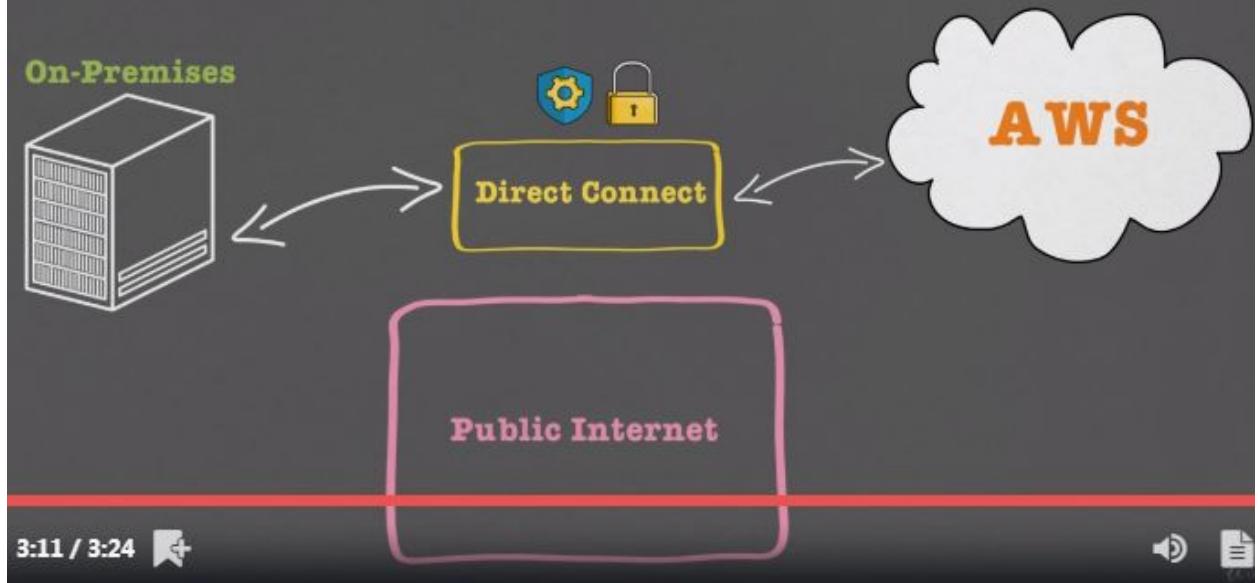




VPN Gateway



AWS Direct Connect



Dedicated network connection
DATABASE

Module (6): Database Services

- Amazon RDS
- Amazon Dynamo DB
- Amazon ElastiCache
- Amazon Redshift
- Amazon Athena

RDS AND DYNAMO DB

DataBase Services:

There are **two** main **categories** of Databases:

- 1) Relational Databases which is "**SQL**"
- 2) Non-relational Databases which is "**NoSQL**"

Those two types of **Databases** are offered by **Amazon**:



RDS



DynamoDB

What is **Relational data** service:

"Amazon Relational Database Service (Amazon RDS) makes it easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while automating time-consuming administration tasks such as hardware provisioning, database setup, patching and backups. It frees you to focus on your applications so you can give them the fast performance, high availability, security and compatibility they need."

Some of **SQL** options:

- 1) Amazon Aurora
- 2) PostgreSQL
- 3) MySQL
- 4) MariaDB
- 5) Oracle Database
- 6) SQL Server

Note: You can use the AWS Database Migration Service to easily migrate or replicate your existing databases to Amazon RDS.

Benefits of **RDS**:

- 1) Easy to administer
- 2) Highly scalable
- 3) Available and durable
- 4) Fast
- 5) Secure
- 6) Inexpensive



What is **DynamoDB** ?

"Amazon DynamoDB is a key-value and document database that delivers single-digit millisecond performance at any scale. It's a fully managed, multiregion, multimaster, durable database with built-in security, backup and restore, and in-memory caching for internet-scale applications. DynamoDB can handle more than 10 trillion requests per day and can support peaks of more than 20 million requests per second."

Benefits of DynamoDB ?

1) Performance at scale

DynamoDB supports some of the world's largest scale applications by providing consistent, single-digit millisecond response times at any scale. You can build applications with virtually unlimited throughput and storage.

DynamoDB global tables replicate your data across multiple AWS Regions to give you fast, local access to data for your globally distributed applications.



2) No servers to manage:

DynamoDB is serverless with no servers to provision, patch, or manage and no software to install, maintain, or operate.

DynamoDB automatically scales tables up and down to adjust for capacity and maintain performance. Availability and fault tolerance are built in, eliminating the need to architect your applications for these capabilities.

DynamoDB provides both provisioned and on-demand capacity modes so that you can optimize costs by specifying capacity per workload, or paying for only the resources you consume.



3) Enterprise ready:

DynamoDB supports ACID transactions to enable you to build business-critical applications at scale.

DynamoDB encrypts all data by default and provides fine-grained identity and access control on all your tables. You can create full backups of hundreds of terabytes of data instantly with no performance impact to your tables, and recover to any point in time in the preceding 35 days with no downtime.

DynamoDB is also backed by a service level agreement for guaranteed availability.



Comparison between RDS and DynamoDB:



RDS

Store data in tables and is used for well structured data.



Example: Table



DynamoDB

Store data in json-like, name-value documents and is used for unstructured data



Example: Json

ELASTIC CACHE

What is the **Elasticache** ?

Amazon ElastiCache offers fully managed Redis and Memcached. Seamlessly deploy, run, and scale popular open source compatible in-memory data stores. Build data-intensive apps or improve the performance of your existing apps by retrieving data from high throughput and low latency in-memory data stores. Amazon ElastiCache is a popular choice for Gaming, Ad-Tech, Financial Services, Healthcare, and IoT apps.

Elasticache Benefits:

- 1) EXTREME PERFORMANCE
- 2) FULLY MANAGED
- 3) SCALABLE



ElastiCache

AMAZON REDSHIFT

What is the **RedShift** ?

Amazon Redshift is an Internet hosting service and data warehouse product which forms part of the larger cloud-computing platform Amazon Web Services.



RedShift

Use for petabyte data analysis using existing business intelligence

RedShift Benefits:

1) Business intelligence

Redshift makes it simple and cost effective to run high performance queries on petabytes of structured data so that you can build powerful reports and dashboards using your existing business intelligence tools.



2) Operational analytics on business events

Bring together structured data from your data warehouse and semi-structured data such as application logs from your S3 data lake to get real-time operational insights on your applications and systems.



AMAZON ATHENA

Amazon Athena

Amazon Athena is an interactive query service that makes it easy to analyze data in Amazon S3 using standard SQL. Athena is serverless, so there is no infrastructure to manage, and you pay only for the queries that you run.

Athena is easy to use. Simply point to your data in Amazon S3, define the schema, and start querying using standard SQL. Most results are delivered within seconds. With Athena, there's no need for complex ETL jobs to prepare your data for analysis. This makes it easy for anyone with SQL skills to quickly analyze large-scale datasets.

Benefits of Amazon Athena:

1) Start Querying Instantly

Serverless, no ETL

Athena is serverless. You can quickly query your data without having to setup and manage any servers or data warehouses. Just point to your data in Amazon S3, define the schema, and start querying using the built-in query editor. Amazon Athena allows you to tap into all your data in S3 without the need to set up complex processes to extract, transform, and load the data (ETL).



2) Pay Per Query

Only pay for **data scanned**



With Amazon Athena, you pay only for the queries that you run. You are charged \$5 per terabyte scanned by your queries. You can save from 30% to 90% on your per-query costs and get better performance by compressing, partitioning, and converting your data into columnar formats. Athena queries data directly in Amazon S3. There are no additional storage charges beyond S3.

Fast, Really Fast



Interactive performance even for large datasets

With Amazon Athena, you don't have to worry about having enough compute resources to get fast, interactive query performance. Amazon Athena automatically executes queries in parallel, so most results come back within seconds.

AMAZON NEPTUNE

High performance database engine for storing billions of query millisecond latency, use for graph models easily build query.graph use cases like fraud detection, drug discovery, network security, knowledge graph. continues backup to amazon s3 and replication across availability zones ,fully managed so need to worry about db hardware, software patch, backup.

Amazon Neptune:

Amazon Neptune is a fast, reliable, fully managed graph database service that makes it easy to build and run applications that work with highly connected datasets.



Amazon Neptune

Benefits:

- 1) Supports open graph APIs
- 2) High performance and scalability
- 3) High availability and durability
- 4) Highly secure
- 5) Fully managed



Use Cases:

- 1) Social Networking
- 2) Fraud Detection
- 3) Knowledge Graphs
- 4) Life Sciences
- 5) Network / IT Operations



SNOWBALL- PETABYTE SCALE DATA TRANSFER

Amazon Snowball:

Snowball is a petabyte-scale data transport solution that uses devices designed to be secure to transfer large amounts of data into and out of the AWS Cloud.



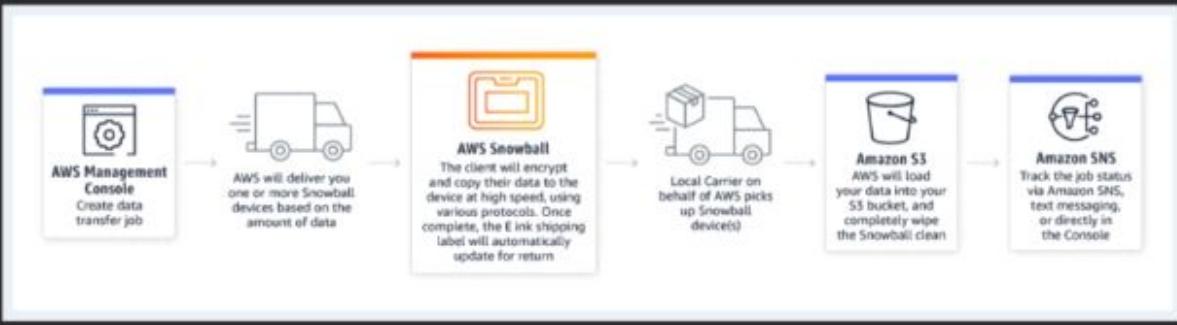
Amazon Snowball

Benefits:

- 1) High speed
- 2) Extremely scalable
- 3) Tamper resistant and secure
- 4) Simple and compatible
- 5) Low cost
- 6) Easy data retrieval



How it Works?



Use Cases:

- 1) Cloud migration
- 2) Disaster recovery
- 3) Datacenter decommission
- 4) Content distribution



Lab

Amazon RDS

Dashboard Databases Query Editor Performance insights Snapshots Automated backups Reserved instances Proxies Subnet groups Parameter groups Option groups Events Event subscriptions Recommendations Certificate update

Amazon Aurora
Amazon Aurora is a MySQL- and PostgreSQL-compatible enterprise-class database, starting at <\$1/day. Aurora supports up to 64TB of auto-scaling storage capacity, 6-way replication across three availability zones, and 15 low-latency read replicas. [Learn more](#)

Create database Or, Restore Aurora DB cluster from S3

Resources Refresh

You are using the following Amazon RDS resources in the US East (Ohio) region (used/quota)

DB Instances (0/40)	Parameter groups (1)
Allocated storage (0 TB/100 TB)	Default (1)
Click here to increase DB instances limit.	Custom (0/100)
DB Clusters (0/40)	Option groups (1)
Reserved instances (0/40)	Default (1)
Snapshots (0)	Custom (0/20)
Manual (0/100)	Subnet groups (1/50)
Automated (0)	Supported platforms VPC
Recent events (35)	Default network vpc-0dc57ee719426bb43
Event subscriptions (0/20)	

Create database

Recommended for you

RDS Read Replicas
Elastically scale out beyond the capacity constraints of a single DB instance for read-heavy database workloads. [Learn more](#)

RDS Multi-AZ Deployments
Enable Multi-AZ configurations to build a more resilient DR strategy. [Learn more](#)

RDS Multi-AZ Configurations
Enhance your database's availability with RDS Multi-AZ configurations for a more resilient DR strategy. [Learn more](#)

RDS Proxy - Now in Preview
Pool and share database connections for

Create database

Choose a database creation method [Info](#)

Standard Create You set all of the configuration options, including ones for availability, security, backups, and maintenance.

Easy Create Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type [Info](#)

<input checked="" type="radio"/> Amazon Aurora 	<input type="radio"/> MySQL 	<input type="radio"/> MariaDB 
<input type="radio"/> PostgreSQL 	<input type="radio"/> Oracle 	<input type="radio"/> Microsoft SQL Server 

Settings

DB instance identifier: [Info](#)
 Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.
 The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens (1 to 15 for SQL Server). First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

Credentials Settings

Master username: [Info](#)
 Type a login ID for the master user of your DB instance.
 1 to 16 alphanumeric characters. First character must be a letter.

Auto generate a password
 Amazon RDS can generate a password for you, or you can specify your own password.

Master password: [Info](#)
 Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), " (double quote) and @ (at sign).

Confirm password: [Info](#)

Yes
 Amazon EC2 instances and devices outside the VPC can connect to your database. Choose one or more VPC security groups that specify which EC2 instances and devices inside the VPC can connect to the database.

No
 RDS will not assign a public IP address to the database. Only Amazon EC2 instances and devices inside the VPC can connect to your database.

VPC security group
 Choose one or more RDS security groups to allow access to your database. Ensure that the security group rules allow incoming traffic from EC2 instances and devices outside your VPC. (Security groups are required for publicly accessible databases.)

Choose existing
 Create new
 Choose existing VPC security group: ▾

Existing VPC security groups
 X

Availability zone: [Info](#)
 ▾

Database port: [Info](#)
 TCP/UDP port the database will use for application connections.

Additional configuration
 Database options, backup enabled, backtrack disabled, Enhanced Monitoring disabled, maintenance, CloudWatch Logs, delete protection enabled

Yes
Amazon EC2 instances and devices outside the VPC can connect to your database. Choose one or more VPC security groups that specify which EC2 instances and devices inside the VPC can connect to the database.

No
RDS will not assign a public IP address to the database. Only Amazon EC2 instances and devices inside the VPC can connect to your database.

VPC security group
Choose one or more RDS security groups to allow access to your database. Ensure that the security group rules allow incoming traffic from EC2 instances and devices outside your VPC. (Security groups are required for publicly accessible databases.)

Choose existing
Choose existing VPC security group

Create new
Create new VPC security group

Existing VPC security groups
Choose VPC security groups ▾
default X

Availability zone [Info](#)
No preference ▾

Database port [Info](#)
(TCP/UDP port the database will use for application connections.)
3306

Additional configuration
Database options, backup enabled, backtrack disabled, Enhanced Monitoring disabled, maintenance, CloudWatch Logs, delete, endpoint details

[View](#) [Overview](#)

Databases Group resources [Edit](#) [Modify](#) Actions ▾ [Restore from S3](#) [Create database](#)

DB identifier	Role	Engine	Region & AZ	Size	Status	CPU
mariadb1	Instance	MariaDB	us-east-2c	db.t2.micro	Available	

Connectivity & security | Monitoring | Logs & events | Configuration | Maintenance & backups | Tags

Connectivity & security

Endpoint & port	Networking	Security
Endpoint mariadb1.ca6eujskiko.us-east-2.rds.amazonaws.com	Availability zone us-east-2c	VPC security groups default (sg-02790fb19c19e3cb) (active)
Port 3306	VPC vpc-0dc57ee7194268843	Public accessibility Yes
	Subnet group default-vpc-0dc57ee7194268843	Certificate authority rds-ca-2019
	Subnets subnet-08498ca9f9e46ce3f	Expiration authority date Aug 22nd, 2024

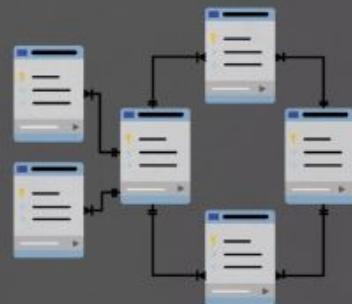
```
mymacbook:~ aabdalmash mysql -h mariadb1.ca6eujbsklko.us-east-2.rds.amazonaws.com -P 3306  
-u admin -p  
Enter password:  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 25  
Server version: 5.5.5-10.2.21-MariaDB-log Source distribution  
  
Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql> create database school;  
Query OK, 1 row affected (0.07 sec)  
  
mysql> use school;  
Database changed  
mysql> create table students (n
```

```
mysql> create database school;  
Query OK, 1 row affected (0.07 sec)  
  
mysql> use school;  
Database changed  
mysql> create table students (name varchar(50), id INT);  
Query OK, 0 rows affected (0.07 sec)  
  
mysql> select * from students;  
Empty set (0.06 sec)  
  
mysql> insert into students (name,id) values ('Bill Gates', 1234);  
Query OK, 1 row affected (0.06 sec)  
  
mysql> select * from students;  
+-----+-----+  
| name   | id   |  
+-----+-----+  
| Bill Gates | 1234 |  
+-----+-----+  
1 row in set (0.06 sec)
```

Amazon RDS

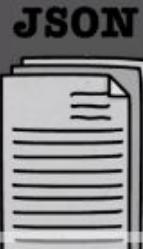
Relational Database Service
Structured Data
SQL

Amazon Aurora
PostgreSQL
Oracle Database **SQL Server**
MariaDB **MySQL**



Amazon DynamoDB

Non-Relational Database
Serverless
Unstructured Data
NoSQL
Key-Value



Amazon Redshift

Data warehouse

Petabytes of Data

Data Analytics



1 Petabytes = 1000 Terabytes

Amazon ElastiCache

In-Memory Data Store

Caching Service

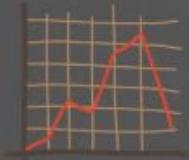
Improves Performance of Web Applications

Access Data Fast



Amazon Neptune

Graph Database



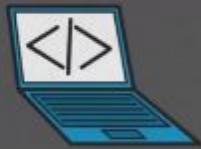
Fraud Detection

Social Networking



Recommendation Engines

Amazon Athena



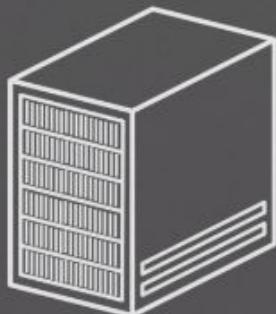
Run SQL Queries

Serverless



Amazon
Snowball

Migrate Petabytes of Data
to and from AWS Cloud.



On-Premises



MONITORING AND LOGGING SERVICES

A Udemy

Module (7): Monitoring and logging services

- Amazon CloudWatch
- Amazon CloudTrail

CLOUDWATCH - MONITORING TOOL FOR APPLICATION

Amazon CloudWatch :

Amazon CloudWatch is a monitoring and observability service built for DevOps engineers, developers, site reliability engineers (SREs), and IT managers. CloudWatch provides you with data and actionable insights to monitor your applications, respond to system-wide performance changes, optimize resource utilization, and get a unified view of operational health. CloudWatch collects monitoring and operational data in the form of logs, metrics, and events, providing you with a unified view of AWS resources, applications, and services that run on AWS and on-premises servers. You can use CloudWatch to detect anomalous behavior in your environments, set alarms, visualize logs and metrics side by side, take automated actions, troubleshoot issues, and discover insights to keep your applications running smoothly.

How Cloudwatch monitoring works?

Cloud watch dashboard is used to view metrics that the user selects.



CloudWatch



EC2

CPU Utilization.
Status check.
Disk read/write.



S3

Check number
of objects and
bucket size



Billing

Check Monthly
Bill

How CloudWatch Alarms works?

User can view alarms in CloudWatch or have the alarm trigger an action like an SNS message.



CloudWatch



EC2

Threshold Set:
CPU Utilization > 70%



S3

Threshold Set:
Number of
Objects > 100



Billing

Threshold Set:
Current monthly
bill > 1000\$

CloudWatch main Features:

- 1) Easily collect and store logs.
- 2) Monitor Metrics for most of AWS resources.
- 3) Create custom dashboards for viewing metrics.
- 4) Set alarms and events



CLOUD TRAIL

What is a CloudTrail ?

AWS CloudTrail is a service that enables governance, compliance, operational auditing, and risk auditing of your AWS account. With CloudTrail, you can log, continuously monitor, and retain account activity related to actions across your AWS infrastructure.

CloudTrail provides event history of your AWS account activity, including actions taken through the AWS Management Console, AWS SDKs, command line tools, and other AWS services. This event history simplifies security analysis, resource change tracking, and troubleshooting. In addition, you can use CloudTrail to detect unusual activity in your AWS accounts. These capabilities help simplify operational analysis and troubleshooting.

Lab

The screenshot shows the AWS CloudWatch Metrics console. A modal dialog titled "Create Alarm" is open, allowing users to set up notifications for metric data. The dialog includes fields for "Send a notification to:" (SNS topic), "Take the action:" (options: Recover this instance, Stop this instance, Terminate this instance, Reboot this instance), "Whenever:" (Average CPU Utilization >= 8 Percent), "For at least:" (1 consecutive period(s) of 5 Minutes), and a "Name of alarm:" field containing "awsEc2-i-0c4273eacc2801e0d-CPU-Utilization". Below the dialog, a line chart titled "CPU Utilization Percent" shows a sharp spike reaching approximately 8% utilization around 16:00 on March 17, 2017. At the bottom, there are tabs for "CPU Utilization (Percent)", "Disk Reads (Bytes)", "Disk Read Operations (Operations)", and "Disk Writes (Bytes)".

[ec2-user@ip-172-31-24-75 ~]\$ yes > /dev/null &

[1] 17979

[ec2-user@ip-172-31-24-75 ~]\$ htop

CPU | ||| 100.0% Tasks: 34, 21 thr; 1 running
 Mem | ||| 157M/820M Load average: 0.02 0.01
 Swp 0K/0K Uptime: 15:58:31

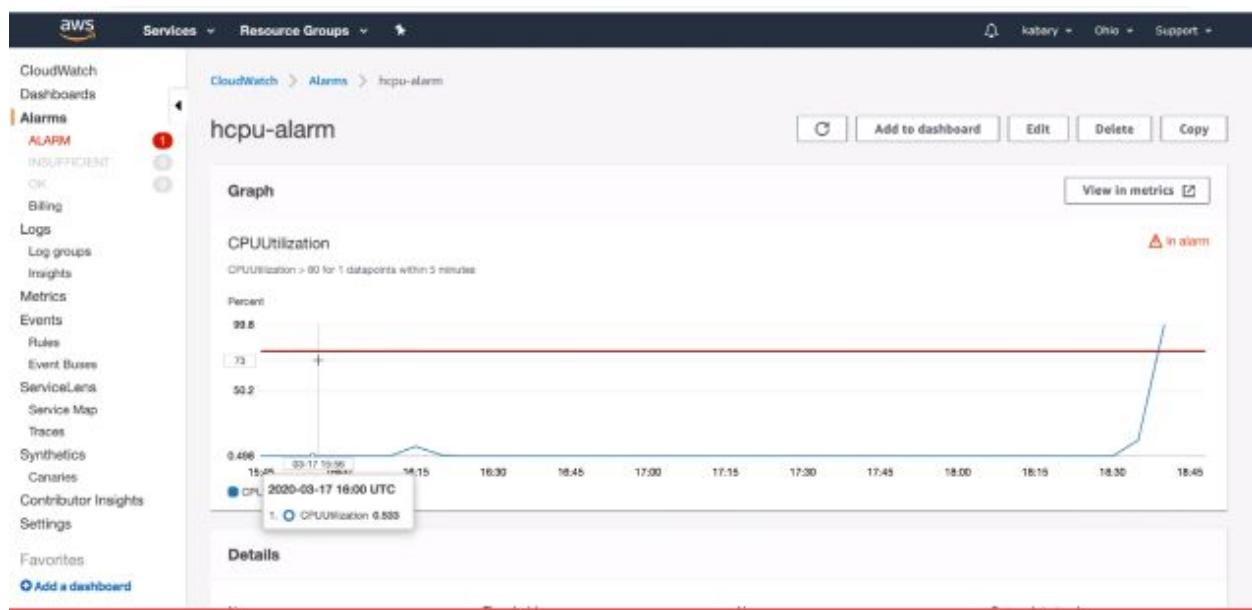
PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
1	root	20	0	174M	13412	9088	S	0.0	1.6	0:09.84	/usr/lib/systemd/syst
509	root	20	0	101M	10964	9856	S	0.0	1.3	0:01.22	/usr/lib/systemd/syst
534	root	20	0	105M	10580	7980	S	0.0	1.3	0:00.19	/usr/lib/systemd/syst
576	root	16	-4	75588	1928	1320	S	0.0	0.2	0:00.00	/sbin/auditd
575	root	16	-4	75588	1928	1320	S	0.0	0.2	0:00.25	/sbin/auditd
626	dbus	20	0	81700	5260	4612	S	0.0	0.6	0:00.00	/usr/bin/dbus-daemon
617	dbus	20	0	81700	5260	4612	S	0.0	0.6	0:00.48	/usr/bin/dbus-daemon
633	root	20	0	156M	6452	5652	S	0.0	0.8	0:03.33	/sbin/rngd -f
619	root	20	0	156M	6452	5652	S	0.0	0.8	0:03.63	/sbin/rngd -f
621	root	20	0	180M	10960	9372	S	0.0	1.3	0:00.07	/usr/sbin/sssd -i --l
638	polkitd	20	0	1729M	22504	15916	S	0.0	2.7	0:00.00	/usr/lib/polkit-1/pol
639	polkitd	20	0	1729M	22504	15916	S	0.0	2.7	0:00.10	/usr/lib/polkit-1/pol
640	polkitd	20	0	1729M	22504	15916	S	0.0	2.7	0:00.00	/usr/lib/polkit-1/pol
641	polkitd	20	0	1729M	22504	15916	S	0.0	2.7	0:00.00	/usr/lib/polkit-1/pol
642	polkitd	20	0	1729M	22504	15916	S	0.0	2.7	0:00.00	/usr/lib/polkit-1/pol
643	polkitd	20	0	1729M	22504	15916	S	0.0	2.7	0:00.00	/usr/lib/polkit-1/pol

```
[ec2-user@ip-172-31-24-75 ~]$ yes > /dev/null &
[1] 17979
[ec2-user@ip-172-31-24-75 ~]$ htop
[ec2-user@ip-172-31-24-75 ~]$ ls
counter.sh
[ec2-user@ip-172-31-24-75 ~]$ cat counter.sh
#!/bin/bash

number=1

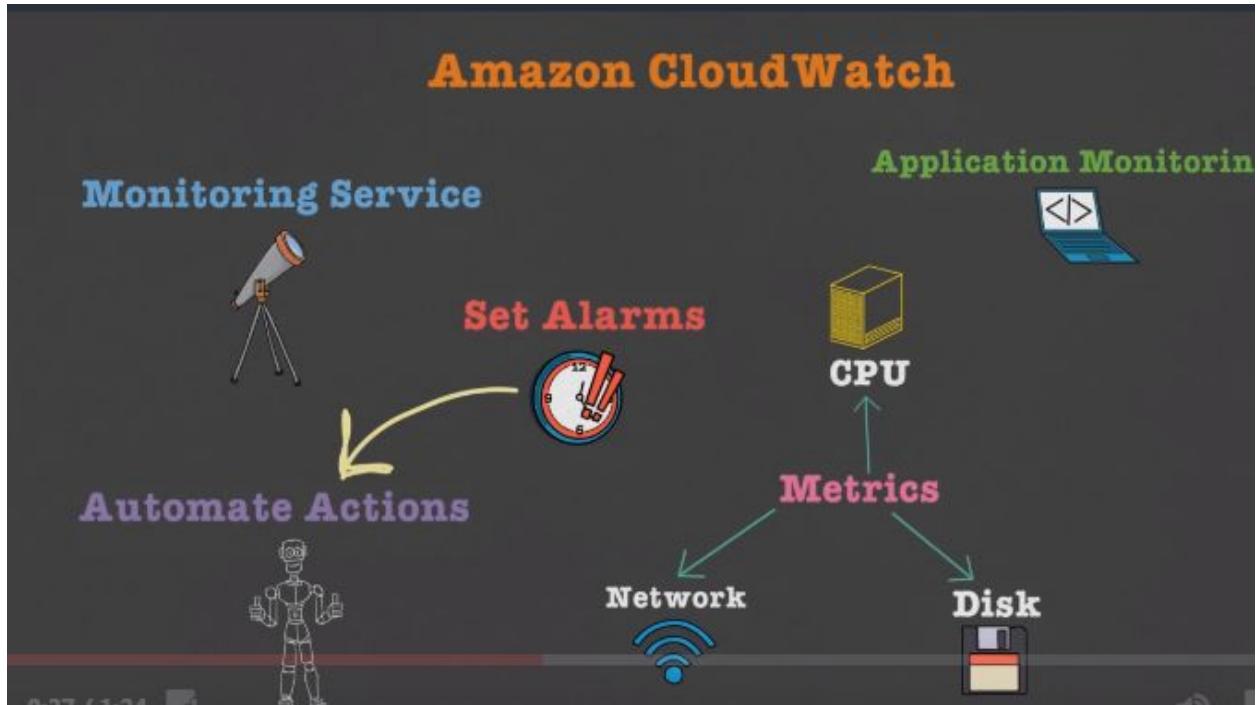
while [ $number -le 360 ]; do
    echo "Counter: $number"
    sleep 1
    number=$((number+1))
done
[ec2-user@ip-172-31-24-75 ~]$
```

```
[ec2-user@ip-172-31-24-75 ~]$ ./counter.sh
Counter: 1
Counter: 2
Counter: 3
Counter: 4
Counter: 5
Counter: 6
Counter: 7
Counter: 8
Counter: 9
Counter: 10
Counter: 11
Counter: 12
Counter: 13
Counter: 14
```



The screenshot shows an email from AWS Notifications (no-reply@aws.amazon.com) titled "ALARM: "hcpu-alarm" in US East (Ohio)". The email indicates that the CloudWatch Alarm "hcpu-alarm" has entered the ALARM state because "Threshold Crossed: 1 datapoint [99.83333333333334 (17/03/20 18:42:00)] was greater than the threshold (80.0)." at "Tuesday 17 March, 2020 18:47:43 UTC". It provides a link to view the alarm in the AWS Management Console: <https://us-east-2.console.aws.amazon.com/cloudwatch/home?region=us-east-2#alarms:alarm=hcpu-alarm>. The email also details the alarm's configuration, including its name, description, state change, history, timestamp, AWS account, and alarm arn. It specifies the threshold as 80.0 and notes that the alarm is in the ALARM state when the metric is GreaterThanThreshold 80.0 for 300 seconds. The monitored metric is CPUUtilization, with a period of 300 seconds. The email concludes with a screenshot of a terminal session showing the uptime command being run multiple times, with the last output showing a load average of 0.21, 0.08, 0.03.

```
[root@ip-172-31-24-75 ~]# uptime
18:49:09 up 0 min, 1 user, load average: 0.25, 0.09, 0.03
[root@ip-172-31-24-75 ~]# uptime
18:49:12 up 0 min, 1 user, load average: 0.23, 0.08, 0.03
[root@ip-172-31-24-75 ~]# uptime
18:49:17 up 1 min, 1 user, load average: 0.21, 0.08, 0.03
[root@ip-172-31-24-75 ~]#
```



MESSAGING SERVICES

Module (8): **Messaging Services**

- Amazon Simple Notification Service
- Amazon Simple Email Service
- Amazon Simple Queue Service

SNS- SIMPLE NOTIFICATION SERVICE

What is simple notifications Service (SNS)?

Amazon Simple Notification Service (SNS) is a highly available, durable, secure, fully managed pub/sub messaging service that enables you to decouple microservices, distributed systems, and serverless applications. Amazon SNS provides topics for high-throughput, push-based, many-to-many messaging. Using Amazon SNS topics, your publisher systems can fan out messages to a large number of subscriber endpoints for parallel processing, including Amazon SQS queues, AWS Lambda functions, and HTTP/S webhooks. Additionally, SNS can be used to fan out notifications to end users using mobile push, SMS, and email.

You can get started with Amazon SNS in minutes by using the AWS Management Console, AWS Command Line Interface (CLI), or AWS Software Development Kit (SDK).

How SNS Works ?

First the CloudWatch detects the crash, then triggers an alarm. The CloudWatch alarm triggers a text message to the SYS. Admin. about the crashed instance. Then SYS. Admin. Fixes the problem.



SNS Components:

Topics: How you label and group different endpoints that you send messages to.

Subscriptions: The endpoints that a topic sends messages to. Example: SYS. Admin. e-mail or Mobile number.

Publishers: The human/alarm/event that gives SNS the message that needs to be sent.



SNS Benefits:

- 1) Send automated or manual notifications.
- 2) Send notification to mobile phone, e-mail and HTTP endpoints.
- 3) Integrated with other AWS services so that alarms, events and actions in your AWS can trigger notifications.



SES -AMAZON SIMPLE EMAIL SERVICE

What is Amazon Simple Email Service?

“Amazon Simple Email Service (Amazon SES) is a cloud-based email sending service designed to help digital marketers and application developers send marketing, notification, and transactional emails. It is a reliable, cost-effective service for businesses of all sizes that use email to keep in contact with their customers.”



Benefits of Amazon Simple Email Service (SES):

1) PROTECT YOUR REPUTATION

Amazon SES includes features such as content filtering technologies, dedicated IP addresses, and a reputation dashboard help protect and enhance your sender reputation. Maintaining a good reputation ensures that your messages reach your customers' inboxes.

2) PAY ONLY FOR WHAT YOU USE

Pay as you go, and pay only for what you use. There are no upfront fees, no time-consuming pricing negotiations, no fixed expenses, and no minimum charges. And, if you send from an application hosted in Amazon EC2, the first 62,000 emails you send every month are free.

3) CREATE RULES TO MEET YOUR SPECIFIC NEEDS

Use configuration sets to create rules that are applied to the emails you send using Amazon SES. Send email metrics to Amazon CloudWatch for further analysis, or receive notifications through Amazon SNS.

4) RECEIVE INCOMING EMAIL



Close the loop on your email program by using Amazon SES to receive email. Incoming email can be delivered automatically to an Amazon S3 bucket. You can use AWS Lambda to execute custom code when messages are received, or use Amazon SNS to deliver notifications when you receive incoming messages that contain certain keywords.

SQS - SIMPLE QUEUE SERVICE

What is Amazon Simple Queue Service ?

Amazon Simple Queue Service (SQS) is a fully managed message queuing service that enables you to decouple and scale microservices, distributed systems, and serverless applications. SQS eliminates the complexity and overhead associated with managing and operating message oriented middleware, and empowers developers to focus on differentiating work. Using SQS, you can send, store, and receive messages between software components at any volume, without losing messages or requiring other services to be available.

SQS offers two types of message queues:

1) Standard queues

offer maximum throughput, best-effort ordering, and at-least-once delivery.

2) SQS FIFO queues

are designed to guarantee that messages are processed exactly once, in the exact order that they are sent.

Benefits of SQS:

1) Eliminate **administrative** overhead

AWS manages all ongoing operations and underlying infrastructure needed to provide a highly available and scalable message queuing service. With SQS, there is no upfront cost, no need to acquire, install, and configure messaging software, and no time-consuming build-out and maintenance of supporting infrastructure.

SQS queues are dynamically created and scale automatically so you can build and grow applications quickly and efficiently.



2) Keep **sensitive data** secure

You can use Amazon SQS to exchange sensitive data between applications using server-side encryption (SSE) to encrypt each message body.

Amazon SQS SSE integration with AWS Key Management Service (KMS) allows you to centrally manage the keys that protect SQS messages along with keys that protect your other AWS resources.

AWS KMS logs every use of your encryption keys to AWS CloudTrail to help meet your regulatory and compliance needs.



3) Reliably **deliver** messages

Use Amazon SQS to transmit any volume of data, at any level of throughput, without losing messages or requiring other services to be available.

SQS lets you decouple application components so that they run and fail independently, increasing the overall fault tolerance of the system.

Multiple copies of every message are stored redundantly across multiple availability zones so that they are available whenever needed.



4) Scale **elastically** and cost-effectively

Amazon SQS leverages the AWS cloud to dynamically scale based on demand. SQS scales elastically with your application so you don't have to worry about capacity planning and pre-provisioning.

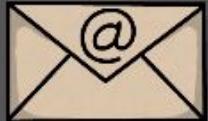
There is no limit to the number of messages per queue, and standard queues provide nearly unlimited throughput.

Costs are based on usage which provides significant cost saving versus the "always-on" model of self-managed messaging middleware.



Amazon SES

Email Sending Service



Digital Marketers



Pay Only for What You Use



Application Developers

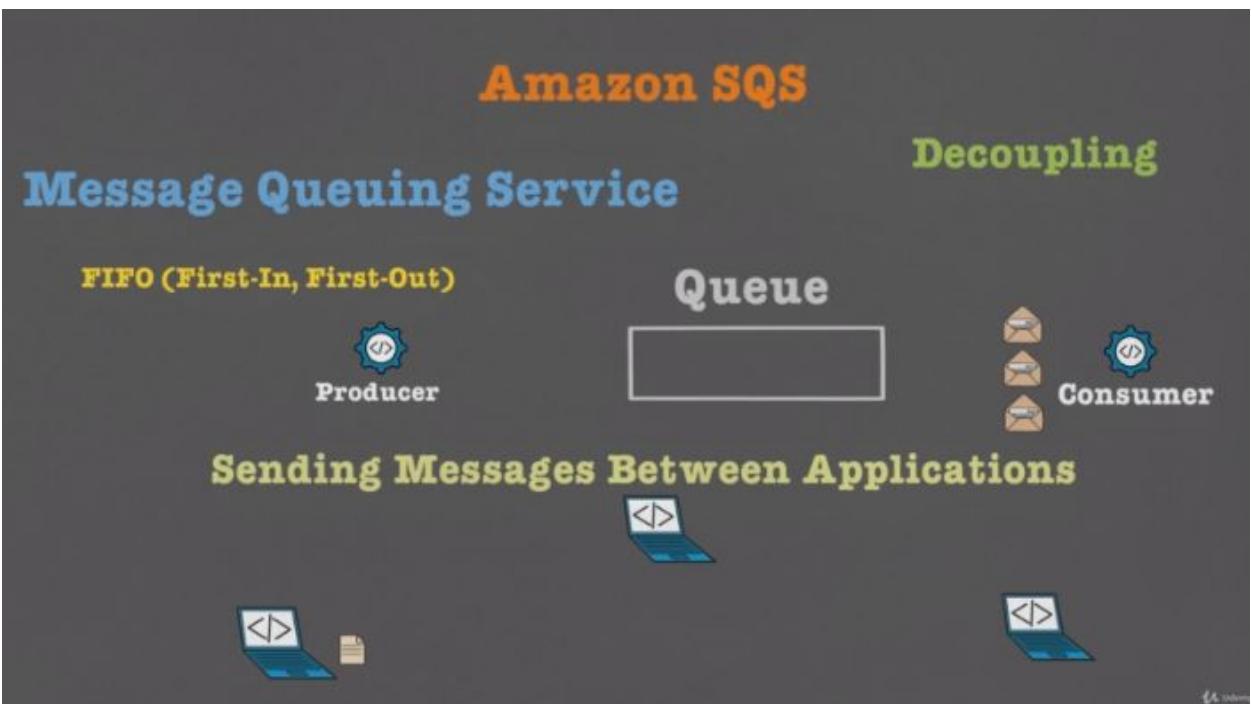


Customers



0:27 / 2:15





AWS SECURITY

Module 9: Breakdown

- DDoS and Penetration Testing
- Account Compromised
- Amazon Inspector
- AWS Shield
- AWS Web Application Firewall

DDOS AND PENETRATION TESTING

DDoS & Penetration Testing

AWS Customer Support Policy for Penetration Testing

AWS customers are welcome to carry out security assessments or penetration tests against their AWS infrastructure without prior approval for 8 services, listed in the next section under "Permitted Services."

Please ensure that these activities are aligned with the policy set out below. Note: Customers are not permitted to conduct any security assessments of AWS infrastructure, or the AWS services themselves.

Customer Service Policy for Penetration Testing

Permitted Services

- 1) Amazon EC2 instances, NAT Gateways, and Elastic Load Balancers
- 2) Amazon RDS
- 3) Amazon CloudFront
- 4) Amazon Aurora
- 5) Amazon API Gateways
- 6) AWS Lambda and Lambda Edge functions
- 7) Amazon Lightsail resources
- 8) Amazon Elastic Beanstalk environments

Prohibited Activities

- 1) DNS zone walking via Amazon Route 53 Hosted Zones
- 2) Denial of Service (DoS), Distributed Denial of Service (DDoS), Simulated DoS, Simulated DDoS
- 3) Port flooding
- 4) Protocol flooding
- 5) Request flooding (login request flooding, API request flooding)

ACCOUNT COMPROMISED STEPS

Account compromised ?

If you suspect that your account is compromised, do the following:

- 1) Change your AWS account root user password.
- 2) Rotate and delete all root and AWS Identity and Access Management (IAM) access keys.
- 3) Delete any potentially compromised IAM users, and change the password for all other IAM users.
- 4) Delete any resources on your account you didn't create, such as EC2 instances and AMIs, EBS volumes and snapshots, and IAM users.
- 5) Respond to any notifications you received from AWS Support through the AWS Support Center.

AMAZON INSPECTOR

Amazon Inspector:

Amazon Inspector is an automated security assessment service that helps improve the security and compliance of applications deployed on AWS.



Amazon Inspector

Benefits:

- 1) IDENTIFY APPLICATION SECURITY ISSUES
- 2) INTEGRATE SECURITY INTO DEVOPS
- 3) INCREASE DEVELOPMENT AGILITY
- 4) LEVERAGE AWS SECURITY EXPERTISE
- 5) STREAMLINE SECURITY COMPLIANCE
- 6) ENFORCE SECURITY STANDARDS



AWS SHIELD

AWS Shield

AWS Shield is a managed Distributed Denial of Service (DDoS) protection service that safeguards applications running on AWS.



AWS Shield

Benefits:

- 1) Seamless integration and deployment
- 2) Customizable protection
- 3) Managed Protection and Attack Visibility
- 4) Cost Efficient



Protection Use Cases:

- 1) Web applications and APIs
- 2) DNS
- 3) Other applications (like UDP-based applications)



5 / 10:52

AWS WEB APPLICATION FIREWALL

AWS Web Application Firewall (WAF)

AWS WAF is a web application firewall that helps protect your web applications or APIs against common web exploits that may affect availability, compromise security, or consume excessive resources.



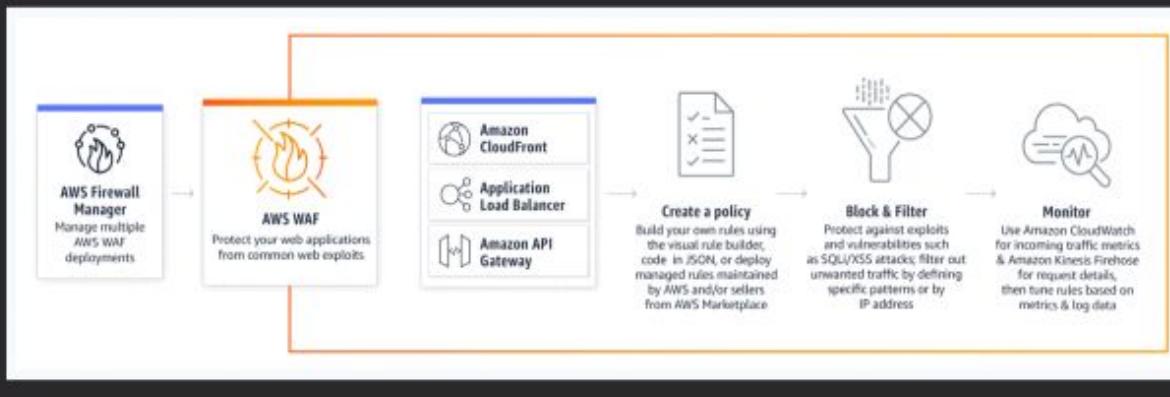
Control over how traffic reaches your application by enabling new security rules
Attack like Sql injection or cross site script rules specific traffic pattern defined, includes full feature api that you can automate create, deployment, maintenance of security rule, pay only for what we used the price based on rules and traffic web request received, cloudfront, elb . Filter traffic based on rule we define.

Benefits:

- 1) Agile protection against web attacks
- 2) Save time with managed rules
- 3) Improved web traffic visibility
- 4) Ease of deployment & maintenance
- 5) Cost effective web application protection
- 6) Security integrated with how you develop applications



How it works?



```

nmap                                         x86_64
Installing for dependencies:                 x86_64
nmap-ncat                                     x86_64                                         2:6.40-13.amzn2

Transaction Summary
=====
Install 1 Package (+1 Dependent package)

Total download size: 4.2 M
Installed size: 17 M
Is this ok [y/d/N]: y
Downloading packages:
(1/2): nmap-ncat-6.40-13.amzn2.x86_64.rpm
(2/2): nmap-6.40-13.amzn2.x86_64.rpm
-----
Total
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : 2:nmap-ncat-6.40-13.amzn2.x86_64
  Installing : 2:nmap-6.40-13.amzn2.x86_64
  Verifying   : 2:nmap-6.40-13.amzn2.x86_64
  Verifying   : 2:nmap-ncat-6.40-13.amzn2.x86_64

Installed:
  nmap.x86_64 2:6.40-13.amzn2

Dependency Installed:
  nmap-ncat.x86_64 2:6.40-13.amzn2

Complete!
[root@ip-172-31-28-125 ec2-user]# nmap -v 15.206.68.6

Starting Nmap 6.40 ( http://nmap.org ) at 2020-04-21 16:11 UTC
Initiating Ping Scan at 16:11
Scanning 15.206.68.6 [4 ports]
Completed Ping Scan at 16:11, 3.01s elapsed (1 total hosts)
Nmap scan report for 15.206.68.6 [host down]
Read data files from: /usr/bin/../share/nmap
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.03 seconds
  Raw packets sent: 8 (304B) | Rcvd: 2 (80B)

```

Nmap install and scan with ec2 public IP : nmap 15.206.68.6 -v

```
mymacbook:~ root# nmap 52.15.57.195 -v
Starting Nmap 7.80 ( https://nmap.org ) at 2020-03-16 14:36 CST
Initiating Ping Scan at 14:36
Scanning 52.15.57.195 [4 ports]
Completed Ping Scan at 14:36, 0.06s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 14:36
Completed Parallel DNS resolution of 1 host. at 14:36, 0.05s elapsed
Initiating SYN Stealth Scan at 14:36
Scanning ec2-52-15-57-195.us-east-2.compute.amazonaws.com (52.15.57.195) [1000 ports]
Discovered open port 80/tcp on 52.15.57.195
Discovered open port 22/tcp on 52.15.57.195
Completed SYN Stealth Scan at 14:36, 6.20s elapsed (1000 total ports)
Nmap scan report for ec2-52-15-57-195.us-east-2.compute.amazonaws.com (52.15.57.195)
Host is up (0.063s latency).
Not shown: 998 filtered ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http

Read data files from: /usr/local/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 6.39 seconds
Raw packets sent: 2006 (88.224KB) | Rcvd: 11 (464B)
mymacbook:~ root#
```

```
[root@ip-172-31-28-125 ec2-user]# nmap --script ssh-brute.nse -p 22 15.206.68.6
Starting Nmap 6.40 ( http://nmap.org ) at 2020-04-21 16:21 UTC
NSE: Failed to load ssh-brute.nse:
ssh-brute.nse:6: module 'libssh2-utility' not found:
    NSE failed to find nselib/libssh2-utility.lua in search paths.
    no field package.preload['libssh2-utility']
    no file '/usr/local/share/lua/5.2/libssh2-utility.lua'
    no file '/usr/local/share/lua/5.2/libssh2-utility/init.lua'
    no file '/usr/local/lib/lua/5.2/libssh2-utility.lua'
    no file '/usr/local/lib/lua/5.2/libssh2-utility/init.lua'
    no file './libssh2-utility.lua'
    no file '/usr/local/lib/lua/5.2/libssh2-utility.so'
    no file '/usr/local/lib/lua/5.2/loadall.so'
    no file './libssh2-utility.so'
stack traceback:
[C]: in function 'require'
ssh-brute.nse:6: in function <ssh-brute.nse:1>
NSE: failed to initialize the script engine:
/usr/bin/.../share/nmap/nse_main.lua:547: could not load script
stack traceback:
[C]: in function 'error'
/usr/bin/.../share/nmap/nse_main.lua:547: in function 'new'
/usr/bin/.../share/nmap/nse_main.lua:783: in function 'get_chosen_scripts'
/usr/bin/.../share/nmap/nse_main.lua:1271: in main chunk
[C]: in ?
```

QUITTING!

Amazon Inspector



Automated Security Assessment



Strengthen Security



Improves Compliance



Run Agent on EC2 Instances



AWS WAF

Web Application Firewall



SQL Injection



Web Attacks

Cross-Site Scripting



Data denial of service attack is malicious traffic incoming to server

Penetration Testing

Testing for Vulnerabilities



Nmap Metasploit

Permitted Services

Amazon EC2 instances and Elastic Load Balancers
Amazon RDS
Amazon CloudFront
Amazon Aurora
Amazon API Gateways
AWS Lambda and Lambda Edge functions
Amazon Lightsail resources
Amazon Elastic Beanstalk environments



Prohibited Activities

DNS zone walking via Amazon Route 53 Hosted Zones
Denial of Service (DoS), Distributed Denial of Service (DDoS)
Port flooding
Protocol flooding
Request flooding (login request flooding, API request flooding)



Account Compromised?



- ★ Change your AWS account root user password.
- ★ Rotate and delete all root and AWS Identity and Access Management (IAM) access keys.
- ★ Delete any potentially compromised IAM users, and change the password for all other IAM users.
- ★ Delete any resources on your account you didn't create, such as EC2 instances and AMIs, EBS volumes and snapshots, and IAM users.
- ★ Respond to any notifications you received from AWS Support through the AWS Support Center.

AWS BILLING AND SUPPORT

Module 10: Breakdown

- AWS Organizations
- Consolidated billing
- AWS pricing principles
- Services pricing and payment methods
- TCO calculator
- Cost Explorer
- Shared Responsibility Model
- AWS account support plan
- AWS Trusted Advisor
- White papers and Documentation

Day 7: AWS Billing & Pricing:

What is AWS Organizations ?

AWS Organizations helps you centrally govern your environment as you grow and scale your workloads on AWS. Whether you are a growing startup or a large enterprise, Organizations helps you to centrally manage billing; control access, compliance, and security; and share resources across your AWS accounts.



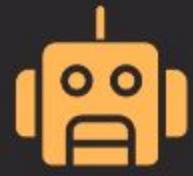
What can you do by using AWS Organizations ?

Using AWS Organizations:

- 1) You can automate account creation
- 2) Create groups of accounts to reflect your business needs
- 3) Apply policies for these groups for governance
- 4) You can also simplify billing by setting up a single payment method for all of your AWS accounts.
- 5) Through integrations with other AWS services, you can use Organizations to define central configurations and resource sharing across accounts in your organization.
- 6) AWS Organizations is available to all AWS customers at no additional charge.

Benefits and main features of AWS Organizations:

- 1) Control access to AWS services.
- 2) Manage access policies across multiple AWS accounts.
- 3) Automate AWS account creation and management.
- 4) Consolidate Billing across multiple AWS accounts.



CONSOLIDATED BILLING

What is **consolidated billing** ?

AWS Organizations provides consolidated billing so that you can track the combined costs of all the linked accounts in your organization.

You can use the consolidated billing feature in AWS Organizations to consolidate billing and payment for multiple AWS accounts or multiple Amazon Internet Services Pvt. Ltd (AISPL) accounts. Every organization in AWS Organizations has a master (payer) account that pays the charges of all the member (linked) accounts.

Consolidated billing has the following **benefits**:

- 1) **One bill** – You get one bill for multiple accounts.
- 2) **Easy tracking** – You can track the charges across multiple accounts and download the combined cost and usage data.
- 3) **Combined usage** – You can combine the usage across all accounts in the organization to share the volume pricing discounts, Reserved Instance discounts, and Savings Plans. This can result in a lower charge for your project, department, or company than with individual standalone accounts. For more information, see Volume Discounts.
- 4) **No extra fee** – Consolidated billing is offered at no additional cost.

PRICING PRINCIPLE

AWS pricing principles:

Pay-as-you-go

Pay-as-you-go allows you to easily adapt to changing business needs without overcommitting budgets and improving your responsiveness to changes.

With a pay-as-you-go model, you can adapt your business depending on need and not on forecasts, reducing the risk of over positioning or missing capacity.

This Means that you only pay for what you use. There are no upfront costs. Your payment automatically ends when you stop using an AWS feature or service.

Save when you reserve

For certain services like Amazon EC2 and Amazon RDS, you can invest in reserved capacity. With Reserved Instances, you can save up to 75% over equivalent on-demand capacity. When you buy Reserved Instances, the larger the upfront payment, the greater the discount.

Pay less by using more

With AWS, you can get volume based discounts and realize important savings as your usage increases. For services such as S3, pricing is tiered, meaning the more you use, the less you pay per GB. AWS also gives you options to acquire services that help you address your business needs.

No Long term contracts are required.

There are no termination fees.

AWS offers a Free Tier option offered to new AWS customers. Free Tier is an offer to new customers to use only limited AWS resources free of charge for 12 months.

SERVICE PRICING AND PAYMENTS

Services Pricing:

Amazon Simple Storage Service (S3):

Pay only for what you use. There is no minimum fee. There are four cost components to consider when deciding on which S3 storage class best fits your data profile – storage pricing, request and data retrieval pricing, data transfer and transfer acceleration pricing, and data management features pricing.

Amazon Elastic Cloud Compute (EC2):

Amazon EC2 is free to try. There are five ways to pay for Amazon EC2 instances: On-Demand, Savings Plans, Reserved Instances, and Spot Instances. You can also pay for Dedicated Hosts which provide you with EC2 instance capacity on physical servers dedicated for your use.

Different payment Methods:

On-Demand

With On-Demand instances, you pay for compute capacity by the hour or the second depending on which instances you run. No longer-term commitments or upfront payments are needed. You can increase or decrease your compute capacity depending on the demands of your application and only pay the specified per hourly rates for the instance you use.



On-Demand instances are recommended for:

Users that prefer 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

Applications being developed or tested on Amazon EC2 for the first time

Spot instances:

Amazon EC2 Spot instances allow you to request spare Amazon EC2 computing capacity for up to 90% off the On-Demand price.

Spot instances are recommended for:

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

Savings Plans

Savings Plans are a flexible pricing model that offer low prices on EC2 and Fargate usage, in exchange for a commitment to a consistent amount of usage (measured in \$/hour) for a 1 or 3 year term.

Dedicated Hosts

A Dedicated Host is a 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, including Windows Server, SQL Server, and SUSE Linux Enterprise Server (subject to your license terms), and can also help you meet compliance requirements.

Can be purchased On-Demand (hourly).

Can be purchased as a Reservation for up to 70% off the On-Demand price.

Reserved Instances

Reserved Instances provide you with a significant discount (up to 75%) compared to On-Demand instance pricing. In addition, when Reserved Instances are assigned to a specific Availability Zone, they provide a capacity reservation, giving you additional confidence in your ability to launch instances when you need them.

For applications that have steady state or predictable usage, Reserved Instances can provide significant savings compared to using On-Demand instances. See [How to Purchase Reserved Instances](#) for more information.



Reserved Instances are recommended for:

- 1) Applications with steady state usage
- 2) Applications that may require reserved capacity
- 3) Customers that can commit to using EC2 over a 1 or 3 year term to reduce their total computing costs



Per Second Billing

With per-second billing, you pay for only what you use. It takes cost of unused minutes and seconds in an hour off of the bill, so you can focus on improving your applications instead of maximizing usage to the hour. Especially, if you manage instances running for irregular periods of time, such as dev/testing, data processing, analytics, batch processing and gaming applications.

EC2 usage are billed on one second increments, with a minimum of 60 seconds. Similarly, provisioned storage for EBS volumes will be billed per-second increments, with a 60 second minimum.



Per-second billing is available for instances launched in:

On-Demand, Reserved and Spot forms

All regions and Availability Zones

Amazon Linux and Ubuntu



AWS TCO CALCULATOR

What is the TCO Calculator ?

TCO calculators allow you to estimate the cost savings when using AWS and provide a detailed set of reports that can be used in executive presentations. The calculators also give you the option to modify assumptions that best meet your business needs.

AWS helps you reduce Total Cost of Ownership (TCO) by reducing the need to invest in large capital expenditures and providing a pay-as-you-go model that empowers you to invest in the capacity you need and use it only when the business requires it.



AWS Total Cost of Ownership (TCO) Calculator:

Use this new calculator to compare the cost of your applications in an on-premises or traditional hosting environment to AWS. Describe your on-premises or hosting environment configuration to produce a detailed cost comparison with AWS.

1. Describe your existing or planned on-premises or hosting infrastructure in four steps, or enter detailed configurations.
2. Get an instant summary report which shows you the three year TCO comparison by cost categories.
3. Download a full report including detailed cost breakdowns, Methodology, Assumptions, and FAQ or store the report in Amazon S3 for sharing with others.

AWS COST EXPLORER

What is the Cost Explorer ?

AWS Cost Explorer has an easy-to-use interface that lets you visualize, understand, and manage your AWS costs and usage over time.

Get started quickly by creating custom reports that analyze cost and usage data. Analyze your data at a high level (for example, total costs and usage across all accounts) or dive deeper into your cost and usage data to identify trends and pinpoint cost drivers.



AWS Cost Explorer Features:

Get started quickly

A set of default reports are included to help you quickly gain insight into your cost drivers and usage trends.

Set time interval and granularity

Set a custom time period, and determine whether you would like to view your data at a monthly or daily level of granularity.

Filter/Group your data

Dig deeper into your data by taking advantage of filtering and grouping functionality, using a variety of available dimensions.

Forecast future costs and usage

Use forecasting to get a better idea of what your costs and usage may look like in the future, so that you can plan ahead.

Save your progress

Once you arrive at a helpful view, save your progress as a new report that you can refer back to in the future.

Build custom applications

Directly access the interactive, ad-hoc analytics engine that powers AWS Cost Explorer.

SHARED RESPONSIBILITY MODEL

AWS ACCOUNT SUPPORT PLAN

AWS TRUSTED ADVISOR

WHITEPAPER AND DOCUMENTATION

AWS certified cloud practitioner Guide Author@ Suraj Samantara

AWS TCO

<https://awstcocalculator.com/>

TYPES OF CLOUD COMPUTING

BASICS CLOUD CONCEPT

EXAM DETAILS

