AWS Certified Cloud Practitioner

Getting Started


Amazon S3


EC2


Amazon EBS


ELB

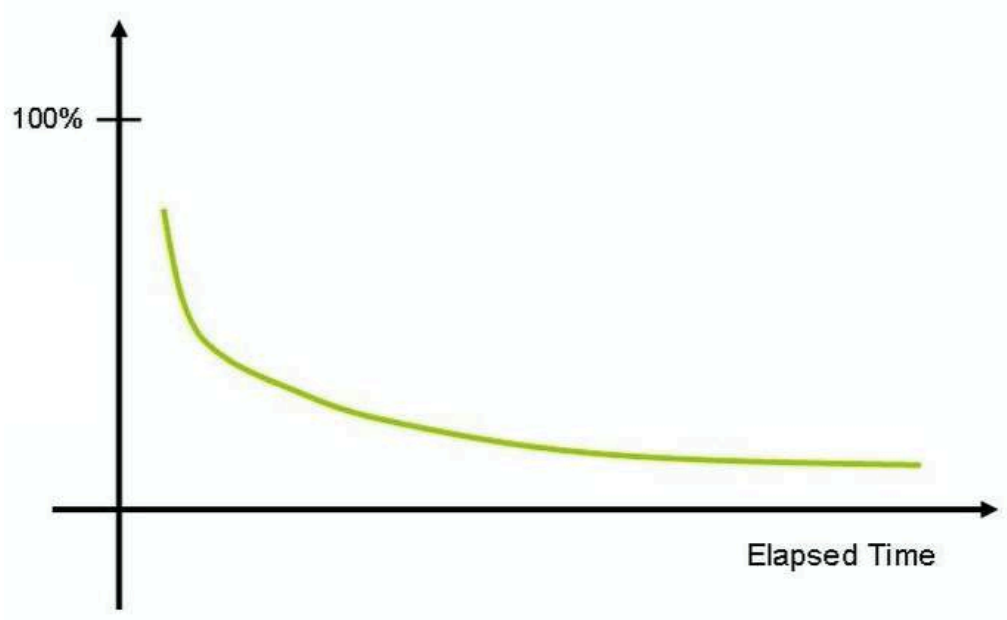

ECS

- AWS has *200+ services*. Exam expects you to understand *40+ services*.
- Exam *tests* your **decision making abilities**:
  - Which service do you choose in which situation?
- This course is **designed** to help you *make these choices*
- **Our Goal** : Help you get certified and start your cloud journey with AWS

How do you put your best foot forward?

In28 Minutes

- **Challenging certification** - Expects you to understand and **REMEMBER** a number of services
- As time passes, humans forget things.
- How do you improve your chances of remembering things?
  - Active learning - think and make notes
  - Review the presentation every once in a while



Our Approach

- Three-pronged approach to reinforce concepts:
  - Presentations (Video)
  - Demos (Video)
  - **Quizzes**
- (Recommended) Take your time. Do not hesitate to replay videos!
- (Recommended) Have Fun!

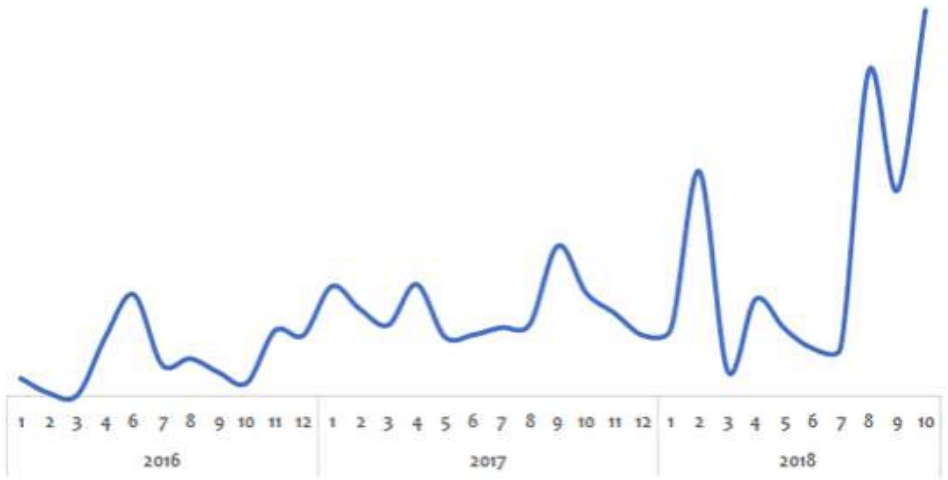
FASTEST ROADMAPS

in28minutes.com

In28 Minutes

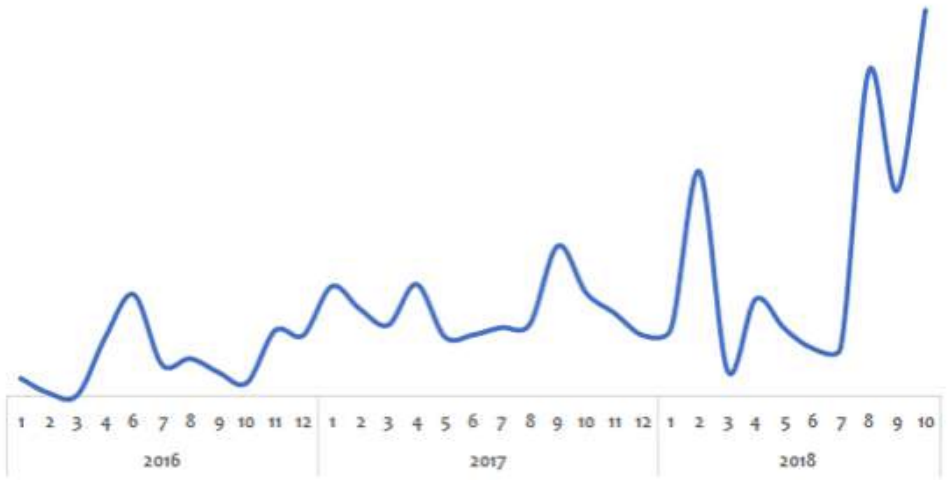
- Google Cloud Certifications
- Azure Certifications
- AWS Certifications
- DevOps
- Java Full Stack
- Java Microservices

Before the Cloud - Example 1 - Online Shopping App



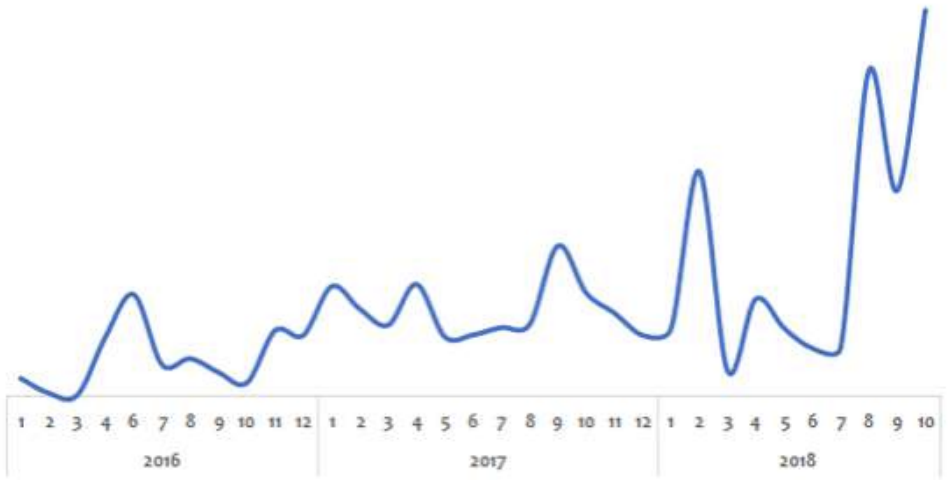
- Challenge:
  - Peak usage during holidays and weekends
  - Less load during rest of the time
- Solution (before the Cloud):
  - **Procure** (Buy) infrastructure **for peak load**
    - What would the infrastructure be doing during periods of low loads?

Before the Cloud - Example 2 - Startup



- Challenge:
  - It suddenly becomes popular.
  - How to handle the **sudden increase** in load?
- Solution (before the Cloud):
  - **Procure** (Buy) infrastructure assuming they would be successful
    - What if they are not successful?

Before the Cloud - Challenges



- High costs of procuring infrastructure
- Needs ahead of time planning (**Can you guess the future?**)
- Low infrastructure utilization
- Dedicated infrastructure maintenance team (**Can a startup afford it?**)

Silver Lining in the Cloud

In28 Minutes

- How about **provisioning (renting) resources** when you want them and releasing them back when you do not need them?
  - **On-demand resource provisioning**
  - Also called **Elasticity**

Cloud - Advantages

- Trade **"capital expense"** for **"variable expense"**
- Benefit from massive **economies of scale**
- Stop **guessing** capacity
- Increase speed and **agility**
- Stop spending money running and maintaining data centers
- **"Go global"** in minutes

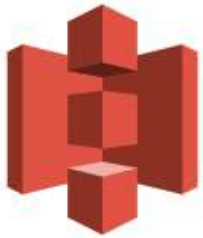
Amazon Web Services (AWS)

In 28 Minutes

- Leading cloud service provider
- Provides most (200+) services
- Reliable, secure and cost-effective
- The entire course is all about AWS. You will learn it as we go further.

Best path to learn AWS!

In28 Minutes


Amazon S3


EC2


Amazon EBS


ELB


ECS

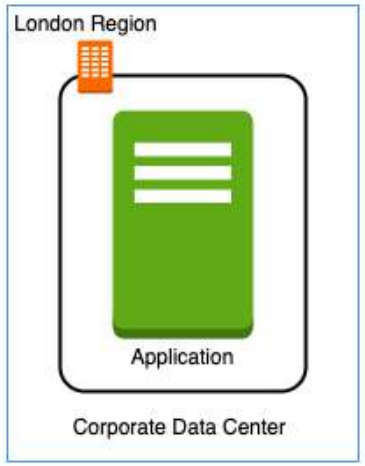
- Cloud applications make use of multiple AWS services.
- There is **no single path** to learn these services independently.
- HOWEVER, we've worked out a simple path!

Setting up AWS Account

- Create AWS Account
- Setup an IAM user

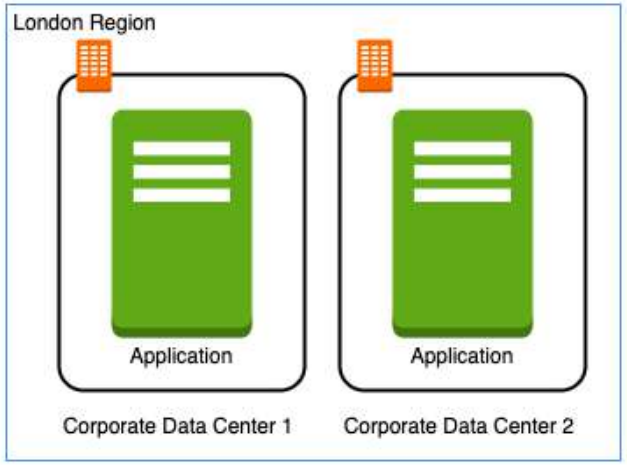
Regions and Zones

Regions and Zones



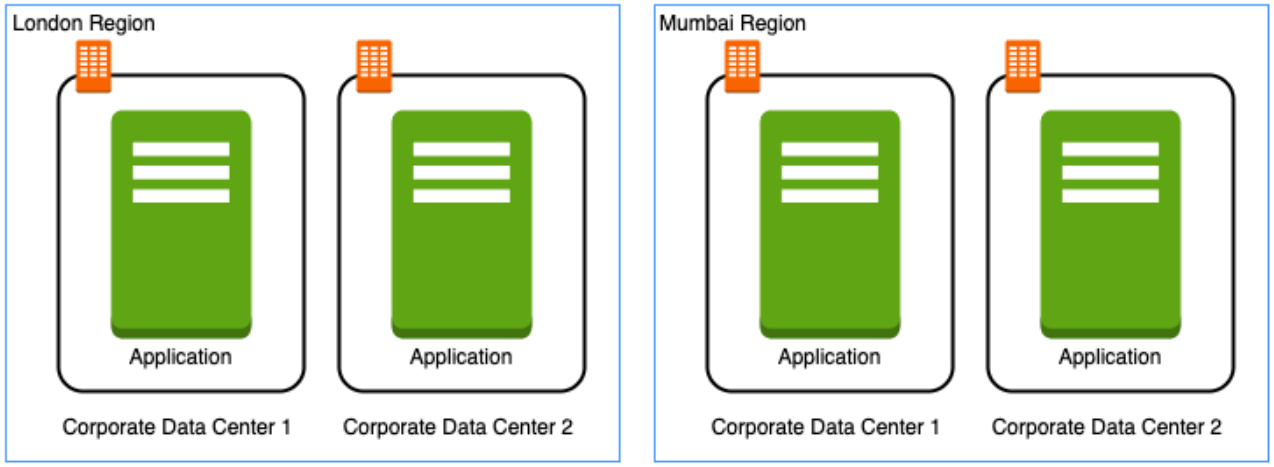
- Imagine that your application is deployed in a data center in London
- What would be the challenges?
  - Challenge 1 : Slow access for users from other parts of the world (**high latency**)
  - Challenge 2 : What if the data center crashes?
    - Your application goes down (**low availability**)

Multiple data centers



- Let's **add in one more data center** in London
- What would be the challenges?
  - Challenge 1 : Slow access for users from other parts of the world
  - Challenge 2 (**SOLVED**) : What if one data center crashes?
    - Your application is **still available** from the other data center
  - Challenge 3 : What if **entire region** of London is unavailable?
    - Your application goes down

Multiple regions



- Let's add a new region : Mumbai
- What would be the challenges?
  - Challenge 1 (**PARTLY SOLVED**) : Slow access for users from other parts of the world
    - You can solve this by adding deployments for your applications in other regions
  - Challenge 2 (SOLVED) : What if one data center crashes?
    - Your application is still live from the other data centers
  - Challenge 3 (**SOLVED**) : What if entire region of London is unavailable?
    - Your application is served from Mumbai

Regions

- Imagine setting up your own data centers in different regions around the world
  - **Would that be easy?**
- (Solution) AWS provides **20+ regions** around the world (expanding every year)

Regions - Advantages

- Low Latency
- Global Footprint
- Adhere to government **regulations**
- High Availability

Availability Zones

- Each AWS Region consists of multiple, isolated, and physically separate AZ's
- Availability Zones in a Region are connected through **low-latency** links
- Each Availability Zone:
  - Can have **One or more discrete data centers**
  - has **redundant** power, networking, and connectivity
- (Advantage) **Increase availability and fault tolerance** of applications in the same region
- (Advantage) Achieve high availability and greater fault-tolerance

Regions and Availability Zones examples

In 28 Minutes

New Regions and AZs are constantly added

Region Code	Region	Availability Zones	Availability Zones List
us-east-1	US East (N. Virginia)	6	us-east-1a us-east-1b us-east-1c us-east-1d us-east-1e us-east-1f
eu-west-2	Europe (London)	3	eu-west-2a eu-west-2b eu-west-2c
ap-south-1	Asia Pacific(Mumbai)	3	ap-south-1a ap-south-1b ap-south-1c

In28 Minutes

EC2 Fundamentals

EC2 (Elastic Compute Cloud)

EC2

EC2 Instances

- In corporate data centers, applications are deployed to physical servers
- Where do you deploy applications in the cloud?
  - Rent virtual servers
  - **EC2 instances** - Virtual servers in AWS
  - **EC2 service** - Provision EC2 instances or virtual servers

EC2 Features

- Create and manage lifecycle of EC2 instances
- **Attach storage** (& network storage) to your EC2 instances
- Manage **network connectivity** for an EC2 instance
- **Load balancing** and **auto scaling** for multiple EC2 instances

EC2 Hands-on

In28 Minutes

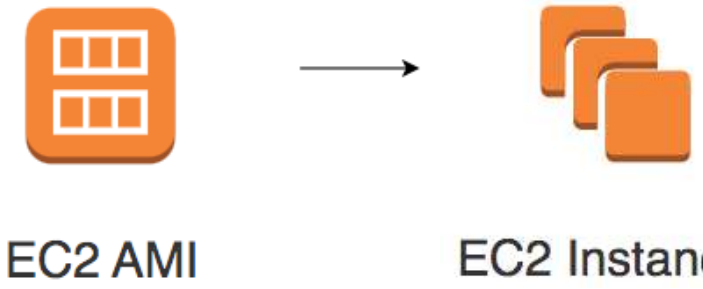
- Let's create a few EC2 instances and play with them
- Let's use EC2 Instance Connect to SSH into EC2 instances

Useful Commands

In 28 Minutes

```
sudo su
yum update -y
yum install httpd
systemctl start httpd
systemctl enable httpd
echo "Hello World 2" > /var/www/html/index.html
```

EC2 Concepts - AMI - Amazon Machine Image



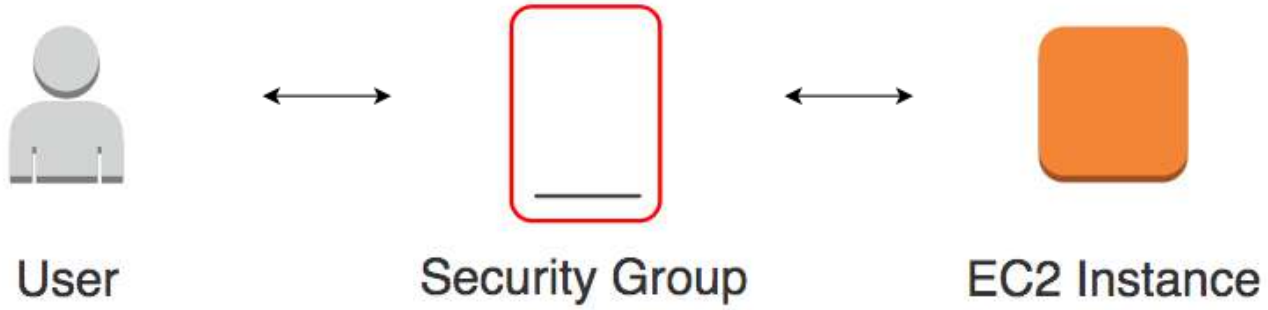
- What operating system and what software do you want on the instance?
- Three AMI sources:
  - Provided by AWS
  - **AWS Market Place**: Online store for customized AMIs. Per hour billing
  - **Customized AMIs**: Created by you.

EC2 Concepts - Instance Families

In 28 Minutes

- Optimized combination of **compute(CPU, GPU), memory, disk (storage) and networking** for specific workloads
- 270+ instances across 40+ types for different workloads
  - m (m4, m5, m6) - General Purpose
  - c (c4, c5, c5n) - Compute Optimized
  - r (r4, r5, r5a, r5n) - Memory (RAM) optimized
  - i (i3) - Storage (I/O) optimized
  - g (g3, g4) - GPU Optimize - Graphics Processing

EC2 Important Concepts - Security Groups



- **Virtual firewall** to control **incoming and outgoing** traffic to/from AWS resources (EC2 instances, databases etc)
- Provides additional layer of security - Defense in Depth

Security Groups Rules

- **Default deny** - If there are no rules configured, no outbound/inbound traffic is allowed
- Allows **allow rules ONLY**
- **Separate rules** for inbound and outbound traffic

EC2 Security - Key Pairs

- EC2 uses public key cryptography for protecting login credentials
- Key pair - public key and a private key
  - Public key is stored in EC2 instance
  - Private key is stored by customer

EC2 IP Addresses

In28 Minutes

- Public IP addresses are internet addressable.
- Private IP addresses are **internal** to a corporate network
- You CANNOT have two resources with same public IP address.
- HOWEVER, two different corporate networks CAN have resources with same private IP address
- **All EC2 instances** are assigned private IP addresses
- (Remember) When you stop an EC2 instance, public IP address is lost

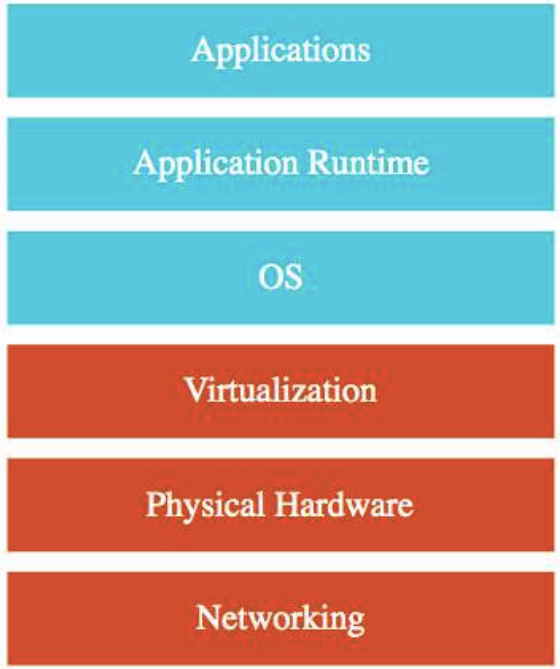
Elastic IP Addresses

- Scenario : How do you get a **constant public IP address** for a EC2 instance?
  - Quick and dirty way is to **use an Elastic IP**!
- Elastic IP can be switched to another EC2 instance **within the same region**
- Elastic IP **remains attached** even if you stop the instance. You have to manually detach it.
- Remember : You are charged for an Elastic IP when you are NOT using it! Make sure that you explicitly release an Elastic IP when you are not using it

IAAS (Infrastructure as a Service)

In 28 Minutes

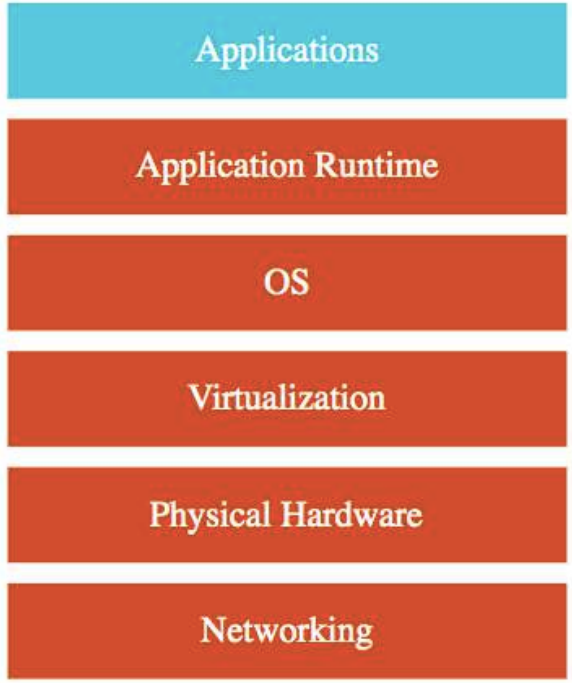
- Use **only infrastructure** from cloud provider
  - Computers (virtual or on dedicated hardware), data storage space and Networking features
- Also called "**Lift and Shift**"
- **Example**: Using EC2 to deploy your applications
- **Example**: Using EC2 to create your database
- **Cloud Provider** is responsible for:
  - Physical Infrastructure (Hardware, Networking)
  - Virtualization Layer (Hypervisor, Host OS)
- **Customer** is responsible for:
  - Guest OS upgrades and patches
  - Application Code and Runtime
  - Availability, Fault Tolerance, Scalability etc.



PAAS (Platform as a Service)

In28 Minutes

- Use a platform provided by cloud
- **Cloud provider** is responsible for:
  - OS (incl. upgrades and patches)
  - Application Runtime
  - Auto scaling, Availability & Load balancing etc..
- **Customer** is responsible for:
  - Application code and/or
  - Configuration

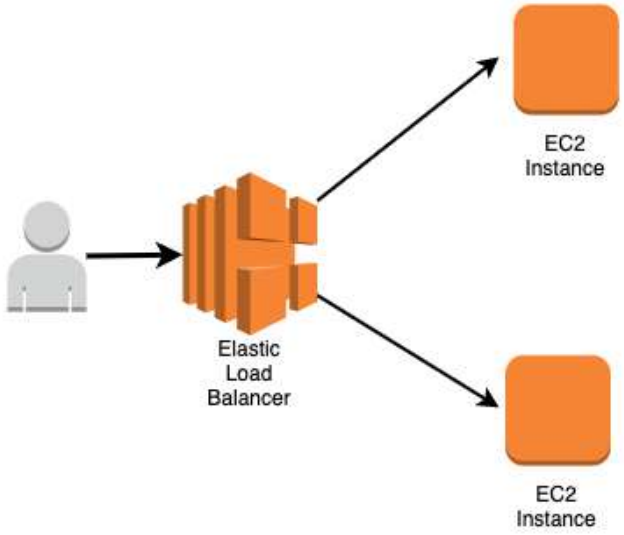


AWS Managed Service Offerings

- **Elastic Load Balancing** - Distribute incoming traffic across multiple targets
- **AWS Elastic Beanstalk** - Run and Manage Web Apps
- **Amazon RDS** - Relational Databases - MySQL, Oracle, SQL Server etc
- And a lot more...

Elastic Load Balancer

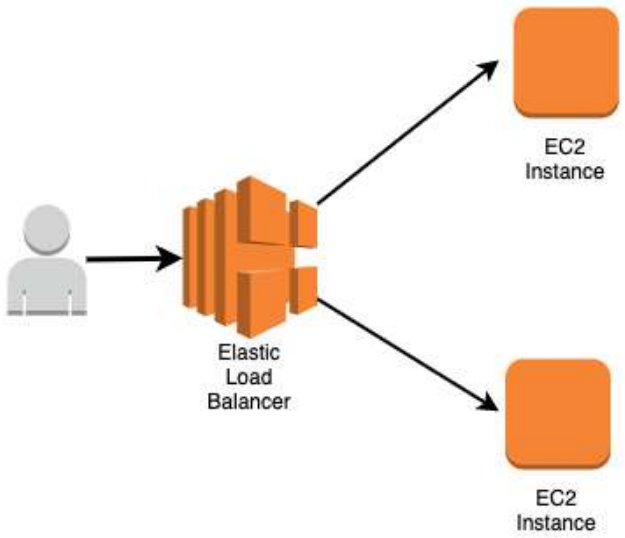
- Distribute traffic across EC2 instances in one or more AZs in a single region
- **Managed service** - AWS ensures that it is highly available
- Auto scales to handle huge loads
- Load Balancers can be **public or private**
- **Health checks** - route traffic to healthy instances



Three Types of Elastic Load Balancers

In 28 Minutes

- **Classic** Load Balancer (Layer 4 and Layer 7)
  - Old generation supporting Layer 4(TCP/TLS) and Layer 7(HTTP/HTTPS) protocols
  - Not Recommended by AWS
- **Application** Load Balancer (Layer 7)
  - **Most popular** and frequently used ELB in AWS
  - New generation supporting HTTP/HTTPS
  - Supports advanced routing approaches (Headers, Query Params, Path and Host Based)
- **Network** Load Balancer (Layer 4)
  - New generation supporting TCP/TLS and UDP
  - Very high performance usecases



Availability

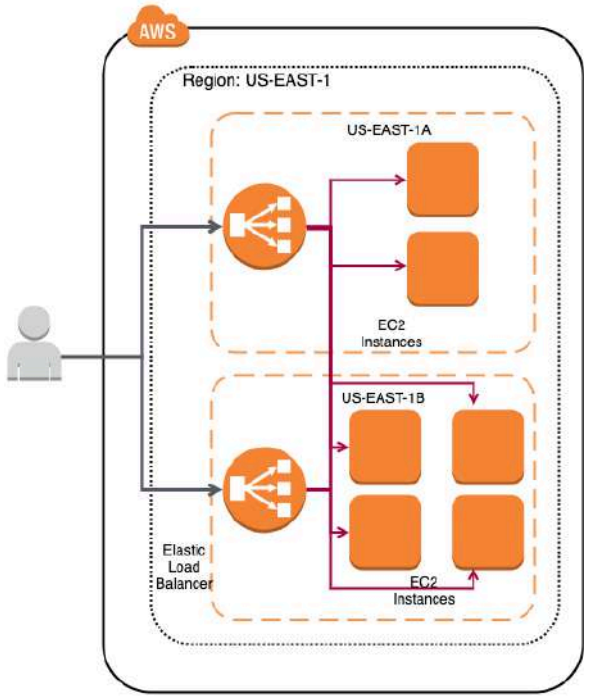
- Are the applications available **when the users need them**?
- **Percentage of time** an application provides the operations expected of it
- **Example**: 99.99% availability. Also called four 9's availability

Availability Table

Availability	Downtime (in a month)	Comment
99.95%	22 minutes	
99.99% (four 9's)	4 and 1/2 minutes	Most online apps aim for 99.99% (four 9's)
99.999% (five 9's)	26 seconds	Achieving 5 9's availability is tough

Availability Basics - EC2 and ELB

- Deploy to multiple AZs
- Deploy to multiple regions



Scalability

- A system is handling 1000 transactions per second. Load is expected to increase 10 times in the next month
  - Can we handle a **growth in users, traffic, or data size** without any drop in performance?
  - Does ability to serve more growth increase **proportionally** with resources?
- Ability to **adapt** to changes in demand (users, data)
- What are the options that can be considered?
  - Deploy to a bigger instance with bigger CPU and more memory
  - Increase the number of application instances and setup a load balancer
  - And a lot more.

Vertical Scaling

- Deploying application/database to **bigger instance**:
  - A larger hard drive
  - A faster CPU
  - More RAM, CPU, I/O, or networking capabilities
- There are limits to vertical scaling

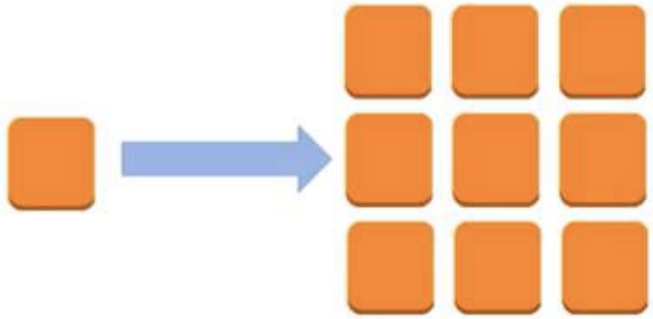
Vertical Scaling for EC2

In28 Minutes

Instance	vCPU*	CPU Credits / hour	Mem (GiB)	Storage	Network Performance
t2.nano	1	3	0.5	EBS-Only	Low
t2.micro	1	6	1	EBS-Only	Low to Moderate
t2.small	1	12	2	EBS-Only	Low to Moderate
t2.medium	2	24	4	EBS-Only	Low to Moderate
t2.large	2	36	8	EBS-Only	Low to Moderate
t2.xlarge	4	54	16	EBS-Only	Moderate
t2.2xlarge	8	81	32	EBS-Only	Moderate

- Increasing **EC2 instance size**:
  - *t2.micro* to *t2.small* or
  - *t2.small* to *t2.2xlarge* or
  - ...

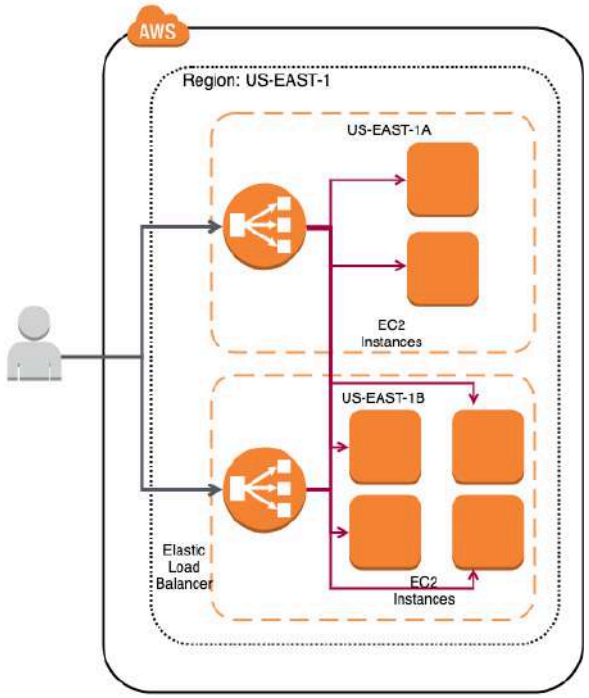
Horizontal Scaling



- Deploying multiple instances of application/database
- (Typically but not always) Horizontal Scaling is preferred to Vertical Scaling:
  - Vertical scaling has limits
  - Vertical scaling can be expensive
  - Horizontal scaling increases availability
- (BUT) Horizontal Scaling needs additional infrastructure:
  - Load Balancers etc.

Horizontal Scaling for EC2

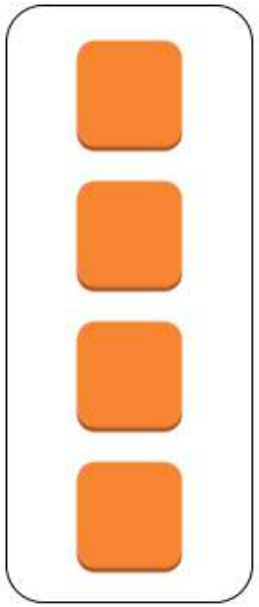
- Distribute EC2 instances
  - in a single AZ
  - in multiple AZs in single region
  - in multiple AZs in multiple regions
- **Auto scale**: Auto Scaling Group
- **Distribute load** : Elastic Load Balancer



EC2 Tenancy - Shared vs Dedicated

In28 Minutes

- **Shared Tenancy** (Default)
  - Single host machine can have instances from multiple customers
- **EC2 Dedicated Instances**
  - Virtualized instances on hardware dedicated to one customer
  - You do NOT have visibility into the hardware of underlying host
- **EC2 Dedicated Hosts**
  - Physical servers dedicated to one customer
  - You have visibility into the hardware of underlying host (sockets and physical cores)
  - (Use cases) Regulatory needs or server-bound software licenses like Windows Server, SQL Server



Host

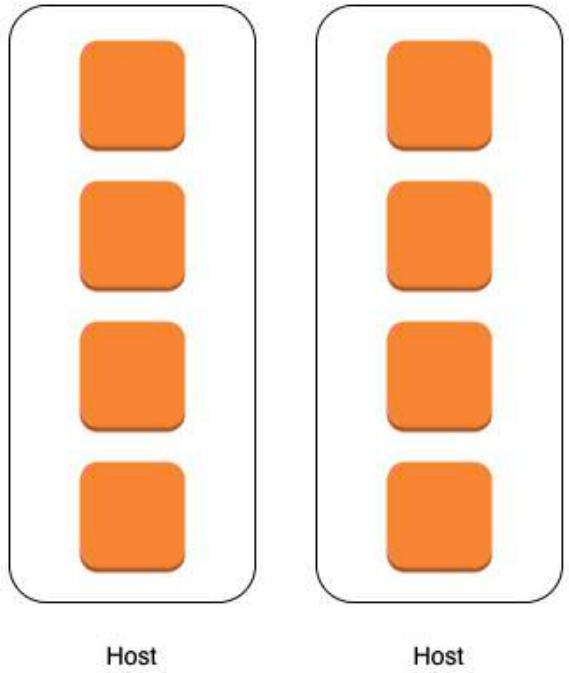
EC2 Pricing Models Overview

In28 Minutes

Pricing Model	Description	Details
On Demand	Request when you want it	Flexible and Most Expensive
Spot	Quote the maximum price	Cheapest (upto 90% off) BUT NO Guarantees
Reserved	Reserve ahead of time	Upto 75% off. 1 or 3 years reservation.
Savings Plans	Commit spending $X per hour on (EC2 or AWS Fargate or Lambda)	Upto 66% off. No restrictions. 1 or 3 years reservation.

EC2 On-Demand

- On demand resource provisioning - **Use and Throw**!
- Highest cost and highest flexibility
- This is what we have been using until now in this course
- **Ideal for**:
  - A web application which receives spiky traffic
  - A batch program which has unpredictable runtime and cannot be interrupted
  - A batch program being moved from on-premises to cloud for the first time



EC2 Spot instances

In 28 Minutes

- **(Old Model)** Bid a price. Highest bidder wins
- **(New Model)** Quote your maximum price. Prices decided by long term trends.
- Up to 90% off (compared to On-Demand)
- Can be terminated with a **2 minute** notice
- Ideal for **Non time-critical workloads** that can **tolerate interruptions** (fault-tolerant)
  - A batch program that does not have a strict deadline AND can be stopped at short notice and re-started

EC2 Reserved Instances

In 28 Minutes

- Reserve EC2 instances ahead of time!
- Get upto 75% OFF!
- **Payment models**:
  - No Upfront - $0 upfront. Pay monthly installment.
  - Partial Upfront - $XYZ upfront. Pay monthly installment
  - All Upfront - Full amount upfront. $0 monthly installment.
  - **Cost wise** : Earlier you pay, more the discount. All Upfront < Partial Upfront < No Upfront
  - A difference upto 5%

EC2 Savings Plans

In28 Minutes

- EC2 Compute Savings Plans
  - **Commitment** : I would spend X dollars per hour on AWS compute resources (Amazon EC2 instances, AWS Fargate and/or AWS Lambda) for a 1 or 3 year period
  - Up to 66% off (compared to on demand instances)
  - Provides **complete flexibility**:
    - You can change instance family, size, OS, tenancy or AWS Region of your Amazon EC2 instances
    - You can switch between Amazon EC2, AWS Fargate and/or AWS Lambda
- EC2 Instance Savings Plans
  - **Commitment** : I would spend X dollars per hour on Amazon EC2 instances of a specific instance family (General Purpose, for example) within a specific region (us-east-1, for example)
  - Up to 72% off (compared to on demand instances)
  - You can switch operating systems (Windows to Linux, for example)

EC2 Pricing Models Overview

In28 Minutes

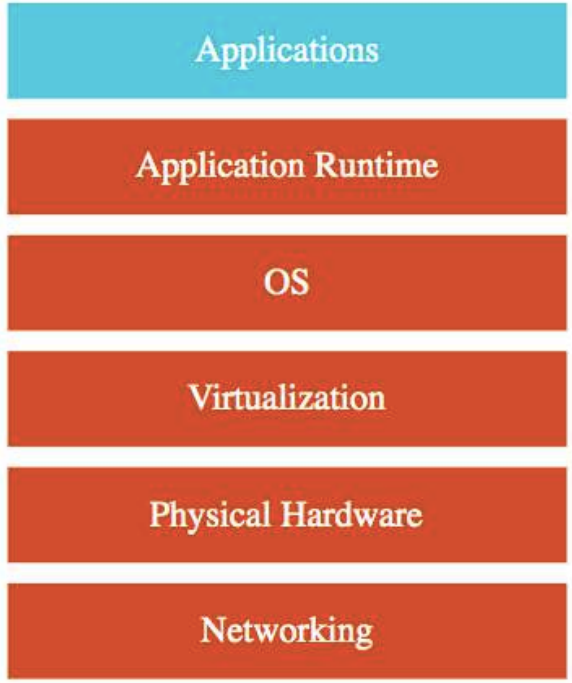
https://www.ec2instances.info/

Pricing Model	Usecases
On Demand	Spiky workloads.
Spot	Cost sensitive, Fault tolerant, Non immediate workloads.
Reserved	Constant workloads that run all the time.
Savings Plans	Constant workloads that run all the time and you want more flexibility.

AWS Elastic BeanStalk

In 28 Minutes

- Next level of **Platform as a Service**!
- **Simplest way** to deploy and scale your web applications in AWS
  - Provides end-to-end web application management
- Supports Java, .NET, Node.js, PHP, Ruby, Python, Go, and Docker applications
- **No usage charges** - Pay for AWS resources provisioned
- **Features**:
  - Automatic load balancing
  - Auto scaling
  - Managed platform updates
  - Application health monitoring



AWS Elastic BeanStalk Demo

In28 Minutes

- Deploy an application to cloud using AWS Elastic Beanstalk

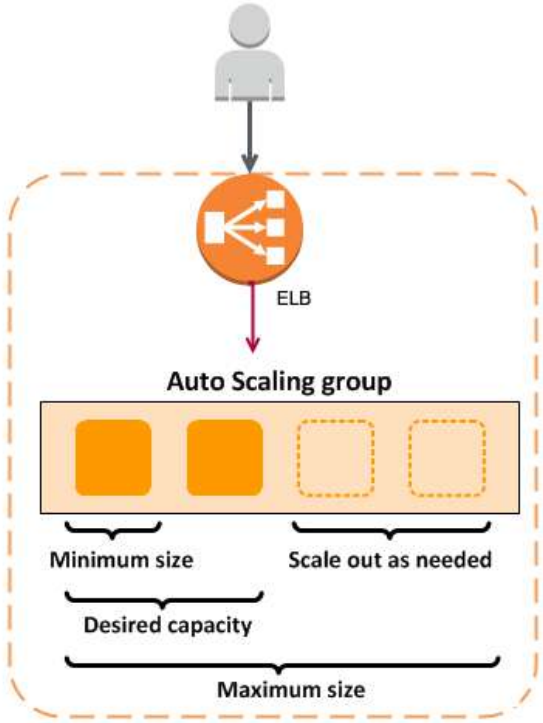
AWS Elastic Beanstalk Concepts

In28 Minutes

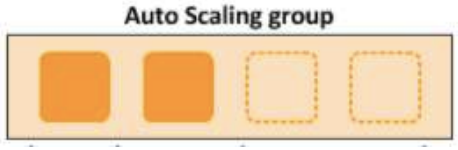
- **Application** - A container for environments, versions and configuration
- **Application Version** - A specific version of deployable code (stored in S3)
- **Environment** - An application version deployed to AWS resources. You can have multiple environments running different application versions for the same application.

Auto Scaling Components

- **Launch Configuration/Template** (What?)
  - EC2 instance size and AMI
- **Auto Scaling Group** (Where?)
  - Min, max and desired size of ASG
  - Health checks
- **Auto Scaling Policies** (When?)
  - When and How to execute scaling?



Dynamic Scaling Policy Types



Scaling Policy	Example(s)	Description
Target tracking scaling	Maintain CPU Utilization at 70%.	Modify current capacity based on a target value for a specific metric.
Simple scaling	+5 if CPU utilization > 80% -3 if CPU utilization < 60%	Waits for cooldown period before triggering additional actions.
Step scaling	+1 if CPU utilization between 70% and 80% +3 if CPU utilization between 80% and 100% Similar settings for scale down	Warm up time can be configured for each instance

Scaling Policies - Background

- Two parts:
  - CloudWatch alarm (Is CPU utilization >80%? or < 60%).
  - Scaling action (+5 EC2 instances or -3 EC2 instances)

Serverless

- What are the things we think about when we develop an application?
  - Where do we deploy the application?
  - What kind of server? What OS?
  - How do we take care of scaling the application?
  - How do we ensure that it is always available?
- **What if we do not need to worry about servers and focus on building our application?**
- Enter **Serverless**

Serverless

In28 Minutes

- Remember: **Serverless does NOT mean "No Servers"**
- **Serverless for me**:
  - You don't worry about infrastructure
  - Flexible scaling
  - Automated high availability
  - Pay for use:
    - You don't have to provision servers or capacity!
- **You focus on code** and the cloud managed service takes care of all that is needed to scale your code to serve millions of requests!

AWS Lambda

- World before Lambda - ELB with EC2 Servers!
- You don't worry about servers or scaling or availability
- You only worry about your code
- You pay for what you use
  - Number of requests
  - Duration of requests
  - Memory consumed

AWS Lambda - Supported Languages

In28 Minutes

- Java
- Go
- PowerShell
- Node.js
- C#
- Python,
- Ruby
- and a lot more...

AWS Lambda Event Sources

In 28 Minutes

- Amazon API Gateway
- AWS Cognito
- Amazon DynamoDB (event)
- Amazon CloudFront (Lambda@Edge)
- AWS Step Functions
- Amazon Kinesis (event)
- Amazon Simple Storage Service
- Amazon Simple Queue Service (event)
- Amazon Simple Notification Service
- The list is endless...

Other Compute Services

- Amazon Lightsail
  - Use case 1 : Pre-configured development stacks like LAMP, Nginx, MEAN, and Node.js.
  - Use case 2 : Run websites on WordPress, Magento, Plesk, and Joomla
  - Low, predictable monthly price.
- AWS Batch
  - Use Case: Run batch computing workloads on AWS
  - Use Amazon EC2 and Amazon EC2 Spot Instances

Making Your Development Simpler with AWS

In 28 Minutes

Service	Example Use Case	Explanation
AWS Cloud9	A developer wanting an IDE that can be accessed from any location using a web browser	A cloud-based IDE that lets you write, run, and debug your code with just a browser. Includes all the essential tools for popular programming languages.
AWS CloudShell	An AWS user needing quick access to AWS services via a command-line without any setup on their local machine	A browser-based shell that provides a secure way to manage AWS resources Pre-authenticated with your AWS credentials. Includes a set of pre-installed tools.
AWS Amplify	A startup aiming to quickly build and deploy a responsive web & mobile application with authentication and API capabilities	Enable developers to build scalable and secure cloud-powered web and mobile applications. Build and deploy full-stack web and mobile apps in hours.

Making Your Development Simpler with AWS - 2

In 28 Minutes

Service	Example Use Case	Explanation
Amazon CodeCatalyst	A team looking to quickly get started with an application on AWS. Spark a faster planning, development, and delivery lifecycle on AWS.	Simplifies the process of setting up and running an app. Offers blueprints (.NET serverless application, Node.js API with AWS Fargate, Java API with AWS Fargate, ..) for various programming languages and frameworks. Features: CI/CD, deployable code, issue tracking, and automation of best practices.
AWS CodeArtifact	A software development team looking for a centralized repository to store and manage reusable code artifacts and dependencies	Fully managed artifact repository service. Securely store, publish, and share software packages. Integrates with popular package managers and build tools like Maven, Gradle, npm, pip,... Provides fine-grained access controls.

Storage

Amazon S3 (Simple Storage Service)

In 28 Minutes

- **Most popular, very flexible & inexpensive** storage service
- Store large objects using a **key-value** approach
- Also called **Object Storage**
- Provides REST API to access and modify objects
- Provides **unlimited storage**:
  - (S3 storage class) **99.99% availability** & **(11 9's - 99.999999999) durability**
  - Objects are **replicated in a single region (across multiple AZs)**
- **Store all file types** - text, binary, backup & archives:
  - Media files and archives
  - Application packages and logs
  - Backups of your databases or storage devices
  - Staging data during on-premise to cloud database migration

Amazon S3

Amazon S3 - Objects and Buckets

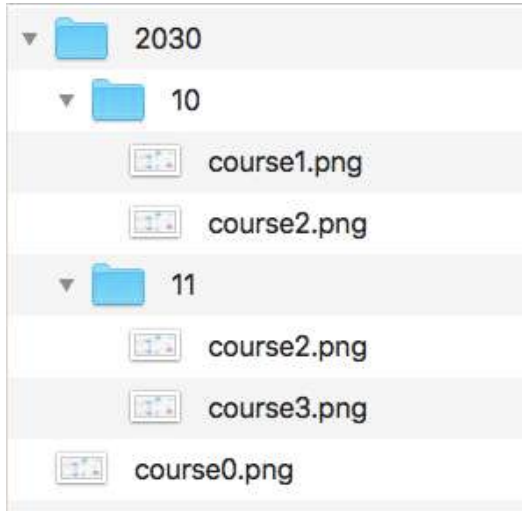
In28 Minutes

- Amazon S3 is a **global service**. NOT associated with a region.
  - HOWEVER a bucket is created in a specific AWS region
- Objects are stored in buckets
  - Bucket names are **globally unique**
  - Bucket names are used as part of object URLs => Can contain ONLY lower case letters, numbers, hyphens and periods.
  - Unlimited objects in a bucket
- Each object is identified by a **key value pair**
  - **Key is unique** in a bucket
  - Max object size is **5 TB**

Amazon S3

Amazon S3 Key Value Example

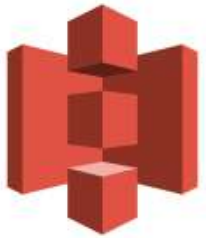
Key	Value
2030/course0.png	image-binary-content
2030/10/course1.png	image-binary-content
2030/10/course2.png	image-binary-content
2030/11/course2.png	image-binary-content
2030/11/course3.png	image-binary-content



Amazon S3 Storage Classes - Introduction

In28 Minutes

- **Different kinds of data** can be stored in Amazon S3
  - Media files and archives
  - Application packages and logs
  - Backups of your databases or storage devices
  - Long term archives
- Huge variations in **access patterns**
- **Trade-off** between access time and cost
- **S3 storage classes** help to optimize your costs while meeting access time needs



Amazon S3

Amazon Glacier

Amazon S3 Storage Classes

In28 Minutes

Storage Class	Scenario	AZs
Standard	Frequently accessed data	>=3
Standard-IA	Long-lived, infrequently accessed data (backups for disaster recovery)	>=3
One Zone-IA	Long-lived, infrequently accessed, non-critical data (Easily re-creatable data - thumbnails for images)	1
Intelligent-Tiering	Long-lived data with changing or unknown access patterns	>=3
Glacier	Archive data with retrieval times ranging from minutes to hours	>=3
Glacier Deep Archive	Archive data that rarely, if ever, needs to be accessed with retrieval times in hours	>=3
Reduced Redundancy (Not recommended)	Frequently accessed, non-critical data	>=3

Amazon S3 Storage Classes - Comparison

In28 Minutes

Feature	Standard	Intelligent Tiering	Standard IA	One Zone IA	Glacier	Glacier Deep Archive
Availability (Designed)	99.99%	99.9%	99.9%	99.5%	99.99%	99.99%
Availability (SLA)	99.9%	99%	99%	99%	99.9%	99.9%
Replication AZs	>=3	>=3	>=3	1	>=3	>=3
First byte: ms (milliseconds)	ms	ms	ms	ms	minutes or hours	few hours
Per GB Cost (varies)	$0.025	varies	$0.018	$0.0144	$0.005	$0.002
Encryption	Optional	Optional	Optional	Optional	Mandatory	Mandatory

Amazon S3 Cost

In 28 Minutes

- Important **pricing elements**:
  - Cost of Storage (per GB)
  - (If Applicable) Retrieval Charge (per GB)
  - Monthly tiering fee (Only for Intelligent Tiering)
  - Data transfer fee
- **FREE of cost**:
  - Data transfer into Amazon S3
  - Data transfer from Amazon S3 to Amazon CloudFront
  - Data transfer from Amazon S3 to services in the same region

Amazon S3 Glacier

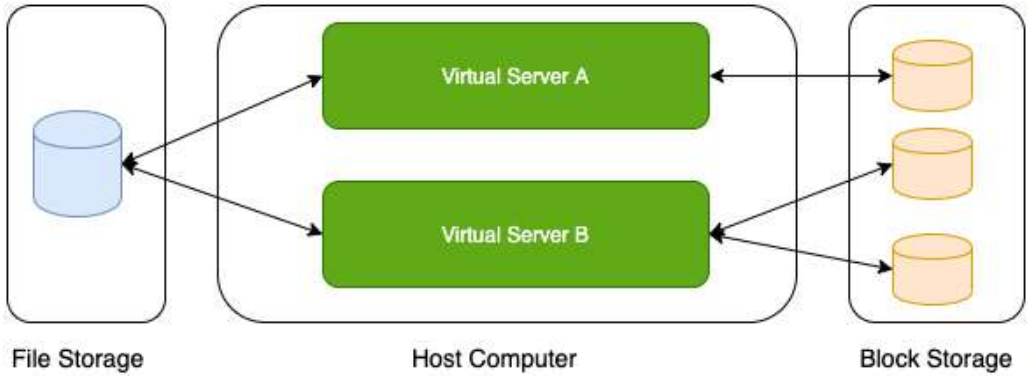
- In addition to existing as a S3 Storage Class, S3 Glacier is a separate AWS Service on it own!
- **Extremely low cost storage** for archives and long-term backups:
  - Old media content
  - Archives to meet regulatory requirements (old patient records etc)
  - As a replacement for magnetic tapes
- High durability (11 9s - 99.999999999%)
- High scalability (unlimited storage)
- High security (**encrypted** at rest and in transfer)

Amazon S3 vs S3 Glacier

In 28 Minutes

Feature	Amazon S3	S3 Glacier
Terminology	Objects (files) are stored in Buckets (containers)	Archives (files) are stored in Vaults (containers)
Keys	Objects keys are user defined	Archive keys are system generated identifiers
Mutability	(Default) Allows uploading new content to object	After an archive is created, it cannot be updated (Perfect for regulatory compliance)
Max size	Each object can be upto 5TB	Each archive can be upto 40TB
Encryption	Optional	Mandatory using AWS managed keys and AES-256. You can use client side encryption on top of this.

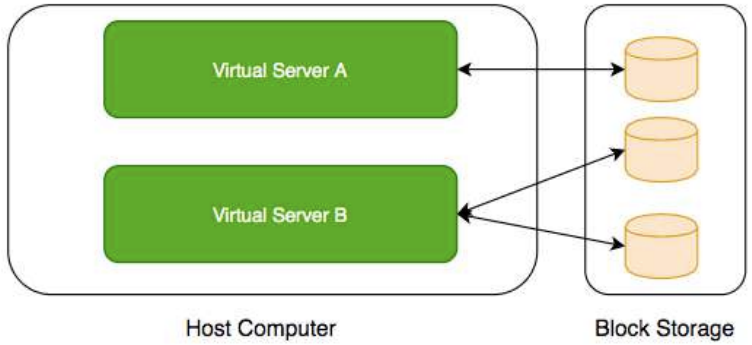
Storage Types - Block Storage and File Storage



- What is the type of storage of your hard disk?
  - **Block Storage**
- You've created a file share to share a set of files with your colleagues in a enterprise. What type of storage are you using?
  - **File Storage**

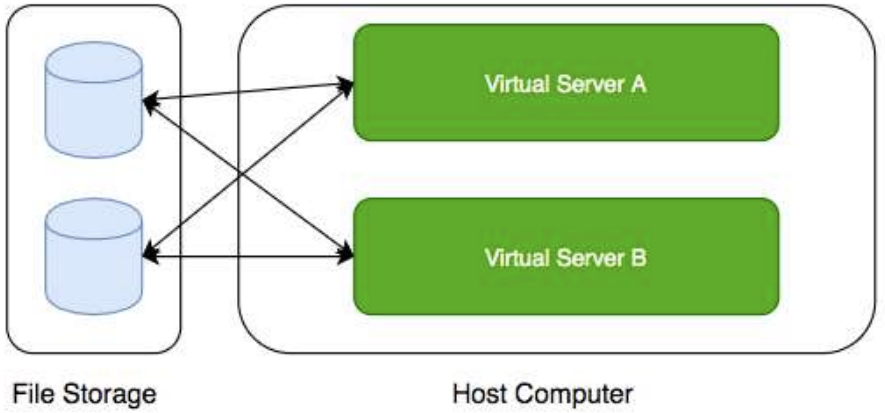
Block Storage

- Use case: Hard-disks attached to your computers
- Typically, ONE Block Storage device can be connected to ONE virtual server
- HOWEVER, you can connect multiple different block storage devices to one virtual server

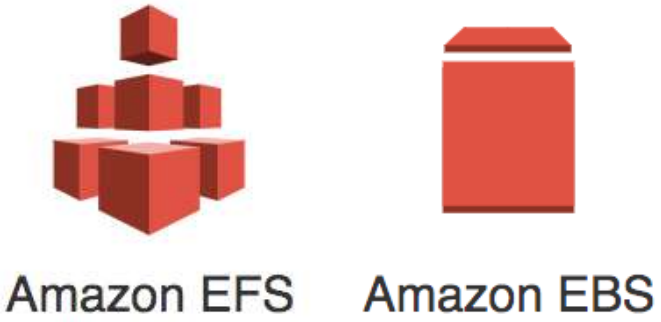


File Storage

- Media workflows need huge shared storage for supporting processes like video editing
- Enterprise users need a quick way to share files in a secure and organized way
- These file shares are shared by several virtual servers



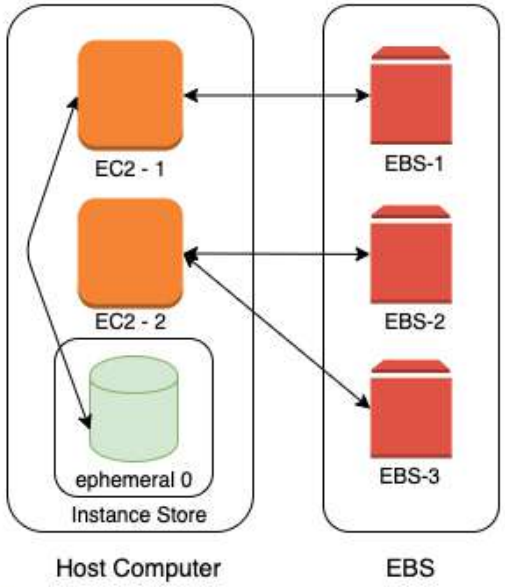
AWS - Block Storage and File Storage



- **Block Storage:**
  - Amazon Elastic Block Store (EBS)
  - Instance store
- **File Storage:**
  - Amazon EFS (for Linux instances)
  - Amazon FSx Windows File Servers
  - Amazon FSx for Lustre (high performance use cases)

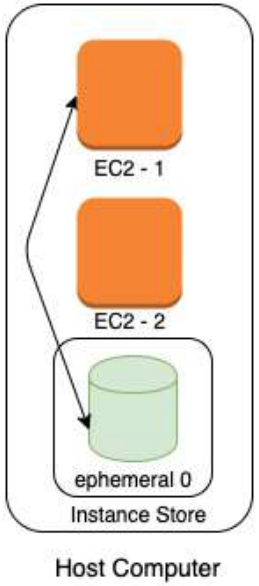
EC2 - Block Storage

- Two popular types of block storage can be attached to EC2 instances:
  - **Elastic Block Store (EBS)**
  - **Instance Store**
- **Instance Stores** are physically attached to the EC2 instance
  - Temporary data
  - Lifecycle tied to EC2 instance
- **Elastic Block Store (EBS)** is network storage
  - More durable
  - Lifecycle NOT tied to EC2 instance



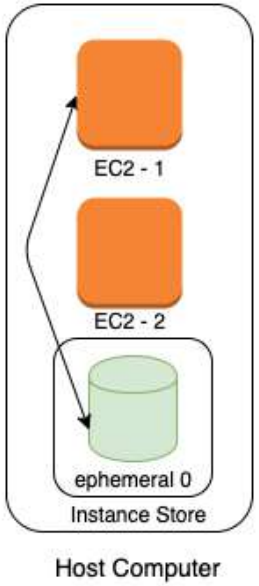
Instance Store

- **Physically attached** to your EC2 instance
- **Ephemeral storage**
  - Temporary data.
  - Data is lost when hardware fails or an instance is terminated.
  - Use case: cache or scratch files
- **Lifecycle is tied** to EC2 instance
- Only some of the EC2 instance types support **Instance Store**



Instance Store - Advantages and Disadvantages

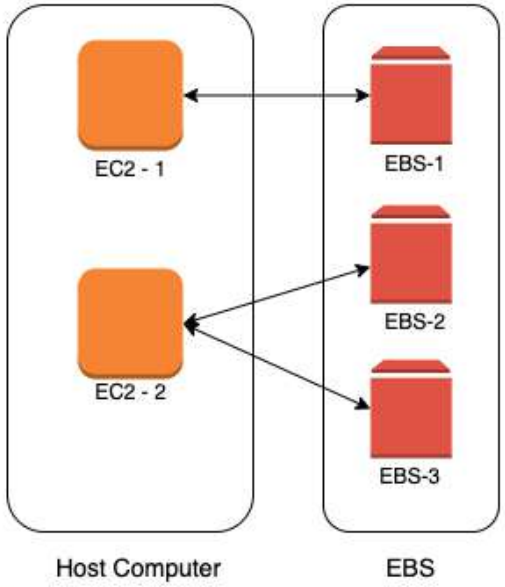
- **Advantages**
  - Very Fast I/O (2-100X of EBS)
  - (Cost Effective) **No extra cost**. Cost is included in the cost of EC2 instance
  - Ideal for storing **temporary information** - cache, scratch files etc
- **Disadvantages**
  - **Slow boot up** (up to 5 minutes)
  - **Ephemeral storage** (data is lost when hardware fails or instance is terminated)
  - **CANNOT take a snapshot** or restore from snapshot
  - Fixed size based on instance type
  - You cannot detach and attach it to another EC2 instance



Amazon Elastic Block Store (EBS)

In 28 Minutes

- **Network block storage** attached to your EC2 instance
- **Provisioned capacity**
- Very flexible.
  - **Increase size when you need it** - when attached to EC2 instance
- Independent lifecycle from EC2 instance
  - Attach/Detach from one EC2 instance to another
- *99.999% Availability* & replicated within the same AZ
- **Use case** : Run your custom database



Amazon EBS vs Instance Store

In 28 Minutes

Feature	Elastic Block Store (EBS)	Instance Store
Attachment to EC2 instance	As a network drive	Physically attached
Lifecycle	Separate from EC2 instance	Tied with EC2 instance
Cost	Depends on provisioned size	Zero (Included in EC2 instance cost)
Flexibility	Increase size	Fixed size
I/O Speed	Lower (network latency)	2-100X of EBS
Snapshots	Supported	Not Supported
Use case	Permanent storage	Ephemeral storage
Boot up time	Low	High

Hard Disk Drive vs Solid State Drive

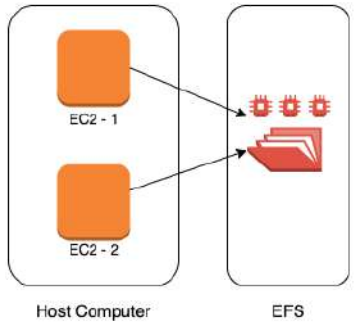
In 28 Minutes

Amazon EBS offers HDD and SSD options!
How do you choose between them?

Feature	HDD(Hard Disk Drive)	SSD(Solid State Drive)
Performance - IOPS	Low	High
Throughput	High	High
Great at	Large sequential I/O operations	Small, Random I/O operations & Sequential I/O
Recommended for	Large streaming or big data workloads	Transactional workloads
Cost	Low	Expensive
Boot Volumes	Not Recommended	Recommended

Amazon EFS

- **Petabyte scale, Auto scaling, Pay for use** shared file storage
- Compatible with Amazon EC2 Linux-based instances
- **(Usecases)** Home directories, file share, content management
- (Alternative) Amazon FSx for Lustre
  - File system **optimized for performance**
  - High performance computing (HPC) and media processing use cases
  - Automatic encryption at-rest and in-transit
- (Alternative) Amazon FSx Windows File Servers
  - Fully managed Windows file servers
  - Accessible from Windows, Linux and MacOS instances
  - Integrates with Microsoft Active Directory (AD) to support Windows-based environments and enterprises.
  - Automatic encryption at-rest and in-transit

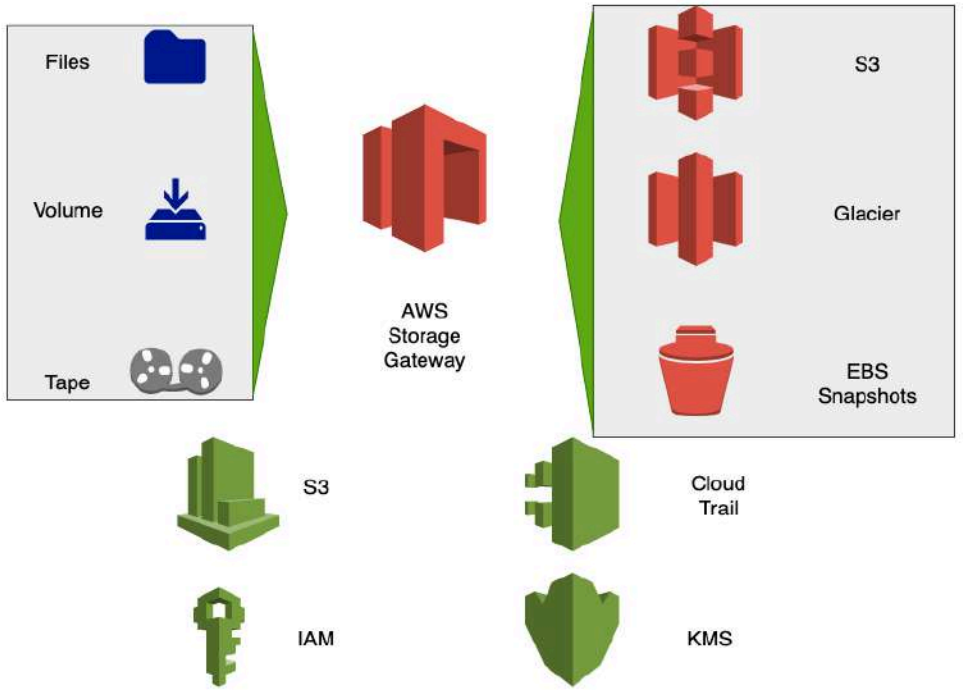


Review of storage options

Type	Examples	Latency	Throughput	Shareable
Block	EBS, Instance Store	Lowest	Single	Attached to one instance at a time. Take snapshots to share.
File	EFS, FSx Windows, FSx for Lustre	Low	Multiple	Yes
Object	S3	Low	Web Scale	Yes
Archival	Glacier	Minutes to hours	High	No

AWS Storage Gateway

- **Hybrid storage** (cloud + on premise)
- Unlimited cloud storage for on-premise software applications and users with good performance
- (Remember) Storage Gateway and S3 Glacier **encrypt data** by default
- **Three Options**
  - AWS Storage File Gateway
  - AWS Storage Tape Gateway
  - AWS Storage Volume Gateway



AWS Storage File Gateway

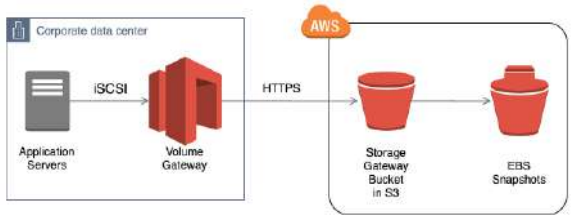
- **Problem Statement**: Large on-premise file share with terabytes of data
  - Users put files into file share and applications use the files
  - Managing it is becoming expensive
  - Move the file share to cloud without performance impact
- AWS Storage File Gateway provides cloud storage for your file shares
  - Files **stored** in **Amazon S3 & Glacier**

AWS Storage Tape Gateway

- Tape backups used in enterprises (archives)
  - Stored off-site - expensive, physical wear and tear
- **AWS Storage Tape Gateway** - Avoid physical tape backups
- **No change needed** for tape backup infrastructure
- Backup data to virtual tapes (actually, Amazon S3 & Glacier)

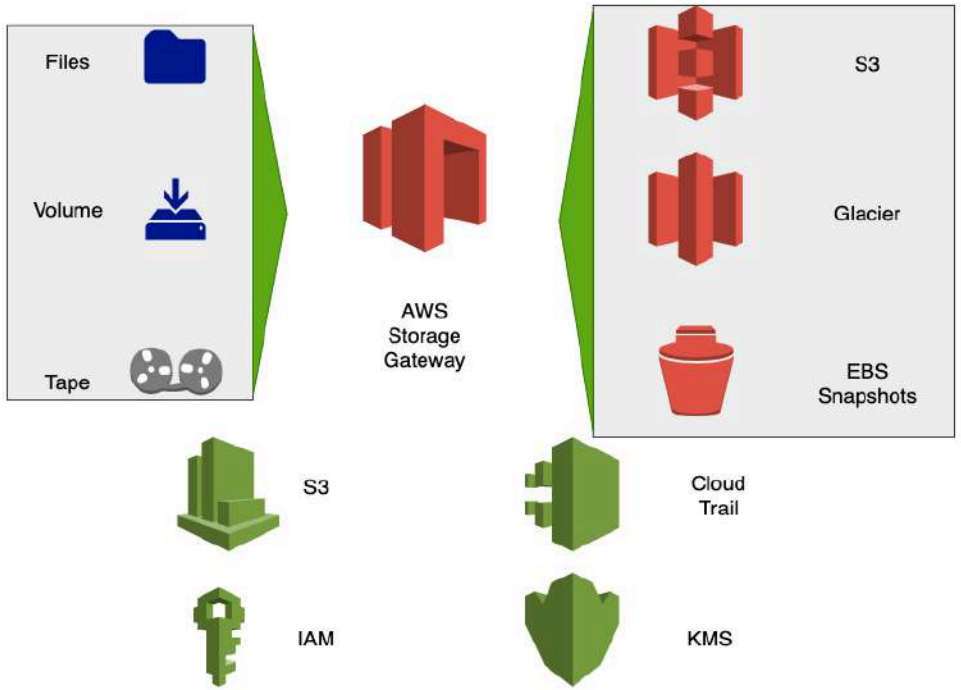
AWS Storage Volume Gateway

- **Volume Gateway** : Move block storage to cloud
- Automate backup and disaster recovery
- Use cases: Backup and disaster recovery, Migration of application data
- (Option 1) **Cached** (Gateway Cached Volumes):
  - Primary Data Store - **AWS - Amazon S3**
  - **On-premise cache** stores frequently accessed data
- (Option 2) **Stored** (Gateway Stored Volumes):
  - Primary Data Store - **On-Premises**
  - Asynchronous copy to AWS
  - Stored as EBS snapshots



AWS Storage Gateway - Summary

- Key to look for : **Hybrid storage** (cloud + on premise)
- File share moved to cloud => **AWS Storage File Gateway**
- Tape Backups on cloud => **AWS Storage Tape Gateway**
- Volume Backups on cloud (Block Storage) => **AWS Storage Volume Gateway**
  - High performance => **Stored**
  - Otherwise => **Cached**



Databases

Databases Primer

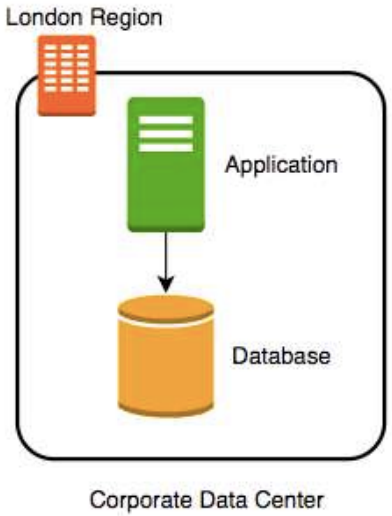
In 28 Minutes

- Databases provide **organized** and **persistent** storage for your data
- To **choose between different database types**, we would need to understand:
  - Availability
  - Durability
  - Consistency
  - Transactions etc
- Let's get started on a **simple journey** to understand these

Database

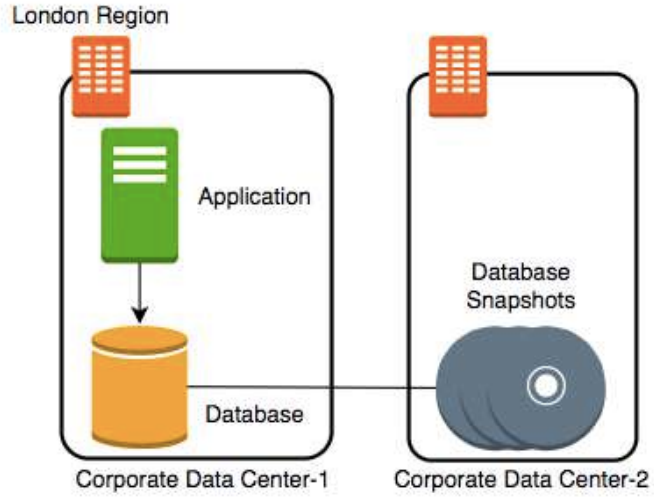
Database - Getting Started

- Imagine a database deployed **in a data center in London**
- Let's consider some challenges:
  - **Challenge 1**: Your database will go down if the data center crashes or the server storage fails
  - **Challenge 2**: You will lose data if the database crashes



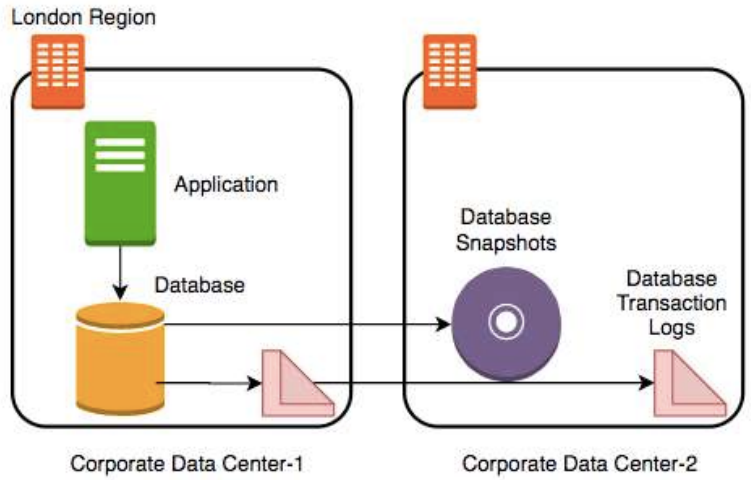
Database - Snapshots

- Let's automate taking copy of the database (**take a snapshot**) every hour to another data center in London
- Let's consider some challenges:
  - **Challenge 1**: Your database will go down if the data center crashes
  - **Challenge 2** (PARTIALLY SOLVED): You will lose data if the database crashes
    - You can setup database from latest snapshot. But depending on when failure occurs you can lose up to an hour of data
  - **Challenge 3**(NEW): Database will be slow when you take snapshots



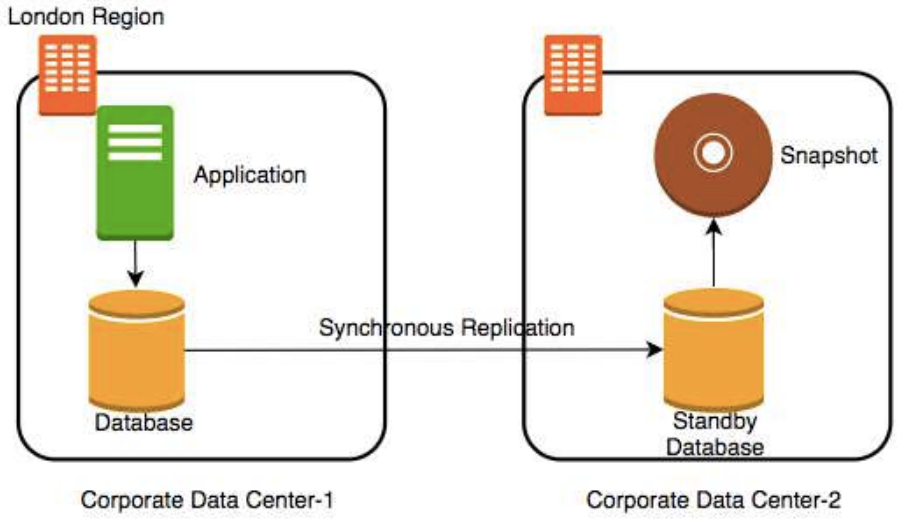
Database - Transaction Logs

- Let's add **transaction logs** to database and create a **process to copy it over** to the second data center
- Let's consider some challenges:
  - **Challenge 1**: Your database will go down if the data center crashes
  - **Challenge 2** (SOLVED): You will lose data if the database crashes
    - You can setup database from latest snapshot and apply transaction logs
  - **Challenge 3**: Database will be slow when you take snapshots



Database - Add a Standby

- Let's add a **standby database** in the second data center with replication
- Let's consider some challenges:
  - **Challenge 1** (SOLVED): Your database will go down if the data center crashes
    - You can switch to the standby database
  - **Challenge 2** (SOLVED): You will lose data if the database crashes
  - **Challenge 3** (SOLVED): Database will be slow when you take snapshots
    - Take snapshots from standby.
    - Applications connecting to master will get good performance always



Availability and Durability

In 28 Minutes

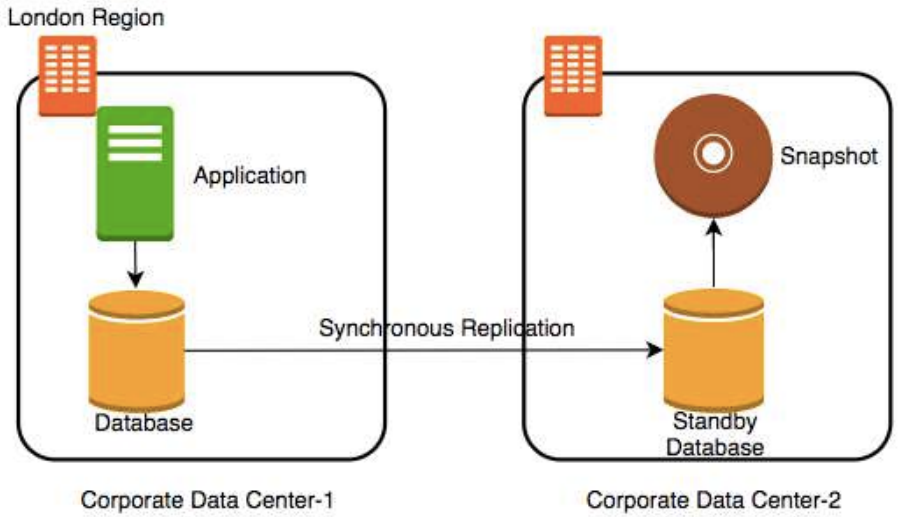
- **Availability**
  - Will I be able to access my data now and when I need it?
  - Percentage of time an application provides the operations expected of it
- **Durability**
  - Will my data be available after 10 or 100 or 1000 years?
- Examples of measuring availability and durability:
  - 4 9's - 99.99
  - 11 9's - 99.999999999
- Typically, an **availability of four 9's** is considered very good
- Typically, a **durability of eleven 9's** is considered very good

Availability

Availability	Downtime (in a month)	Comment
99.95%	22 minutes	
99.99% (4 9's)	4 and 1/2 minutes	Typically online apps aim for 99.99% (4 9's) availability
99.999% (5 9's)	26 seconds	Achieving 5 9's availability is tough

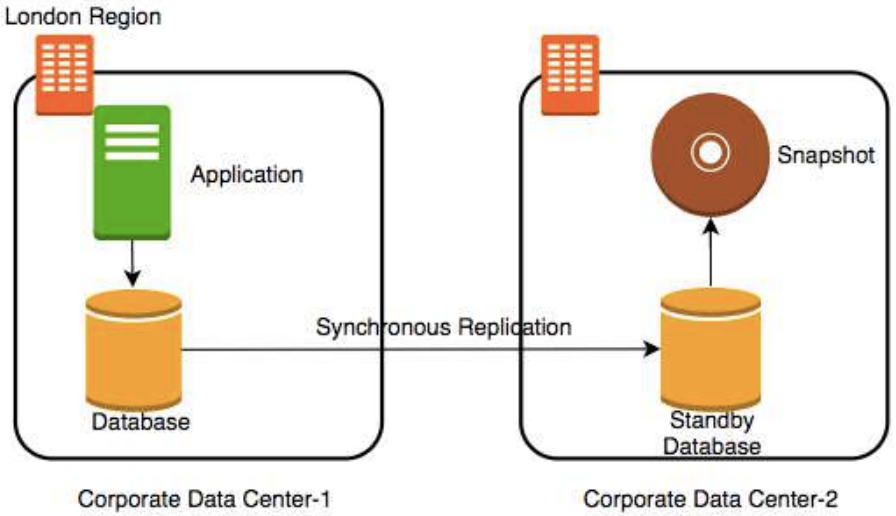
Durability

- What does a **durability of 11 9's** mean?
  - If you **store one million files for ten million years**, you would expect to **lose one file**
- Why should durability be high?
  - Because **we hate losing data**
  - Once we lose data, it is gone



Increasing Availability and Durability of Databases

- **Increasing Availability**:
  - Have multiple standbys available
    - in multiple AZs
    - in multiple Regions
- **Increasing Durability**:
  - Multiple copies of data (standbys, snapshots, transaction logs and replicas)
    - in multiple AZs
    - in multiple Regions
- **Replicating data** comes with its own challenges!
  - We will talk about them a little later



Database Terminology : RTO and RPO

In 28 Minutes

- Imagine a **financial transaction being lost**
- Imagine a **trade being lost**
- Imagine a **stock exchange going down** for an hour
- **Typically** businesses are fine with some downtime but they hate losing data
- Availability and Durability are technical measures
- How do we measure **how quickly we can recover from failure**?
  - **RPO (Recovery Point Objective)**: Maximum acceptable period of data loss
  - **RTO (Recovery Time Objective)**: Maximum acceptable downtime
- Achieving **minimum RTO and RPO is expensive**
- **Trade-off** based on the criticality of the data

Database

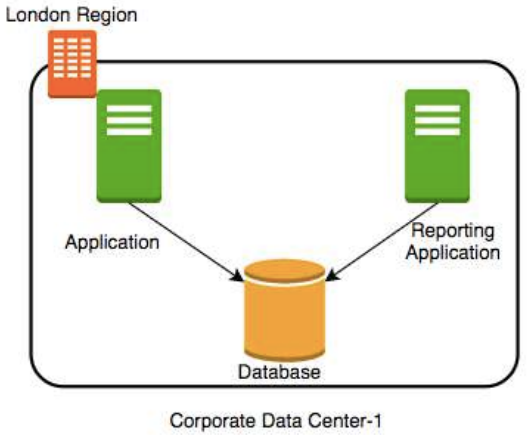
Achieving RTO and RPO - Failover Examples

In28 Minutes

Scenario	Solution
Very small data loss (RPO - 1 minute) **Very small downtime (RTO - 5 minutes)**	**Hot standby** - Automatically synchronize data Have a standby ready to pick up load Use automatic failover from master to standby
Very small data loss (RPO - 1 minute) **BUT I can tolerate some downtimes (RTO - 15 minutes)**	**Warm standby** - Automatically synchronize data Have a standby with minimum infrastructure Scale it up when a failure happens
Data is critical (RPO - 1 minute) but I can tolerate downtime of a few hours (RTO - few hours)	Create regular data **snapshots and transaction logs** Create database from snapshots and transactions logs when a failure happens
Data can be lost without a problem (for example: cached data)	Failover to a completely new server

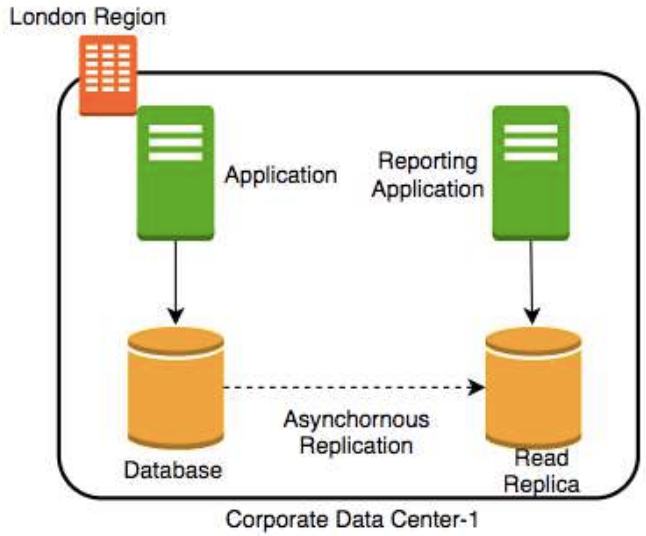
(New Scenario) Reporting and Analytics Applications

- New reporting and analytics applications are being launched using the same database
  - These applications will ONLY read data
- Within a few days you see that the database performance is impacted
- How can we fix the problem?
  - **Vertically scale the database** - increase CPU and memory
  - **Create a database cluster** - typically database clusters are expensive to setup
  - **Create read replicas** - Run read only applications against read replicas



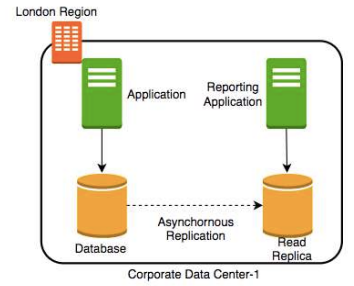
Database - Read Replicas

- Add **read replica**
- Connect reporting and analytics applications to **read replica**
- Reduces load on the master databases
- Upgrade read replica to master database (supported by some databases)
- Create read replicas **in multiple regions**
- **Take snapshots** from read replicas



Consistency

- How do you ensure that data in multiple database instances (standbys and replicas) is updated simultaneously?
- **Strong consistency** - Synchronous replication to all replicas
  - Will be slow if you have multiple replicas or standbys
- **Eventual consistency** - Asynchronous replication. A little lag - few seconds - before the change is available in all replicas
  - In the intermediate period, different replicas might return different values
  - Used when scalability is more important than data integrity
  - Examples : Social Media Posts - Facebook status messages, Twitter tweets, Linked in posts etc
- **Read-after-Write consistency** - Inserts are immediately available. Updates and deletes are eventually consistent
  - Amazon S3 provides read-after-write consistency



Database Categories

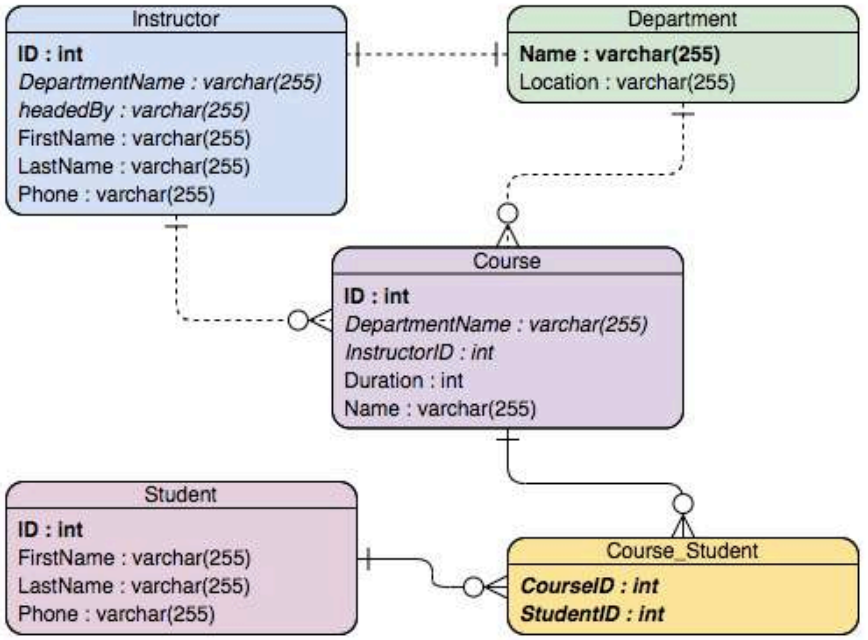
In 28 Minutes

- There are **several categories** of databases:
  - Relational (OLTP and OLAP), Document, Key Value, Graph etc
- **Choosing type of database** for your use case is not easy. A few factors:
  - Do you want a **fixed schema**?
    - Do you want flexibility in defining and changing your schema? (schemaless)
  - What level of **transaction properties** do you need? (example: consistency)
  - What kind of **latency** do you want? (seconds, milliseconds or microseconds)
  - **How many transactions** do you expect? (hundreds or thousands or millions of transactions per second)
  - **How much data** will be stored? (MBs or GBs or TBs or PBs)
  - and a lot more...

Database

Relational Databases

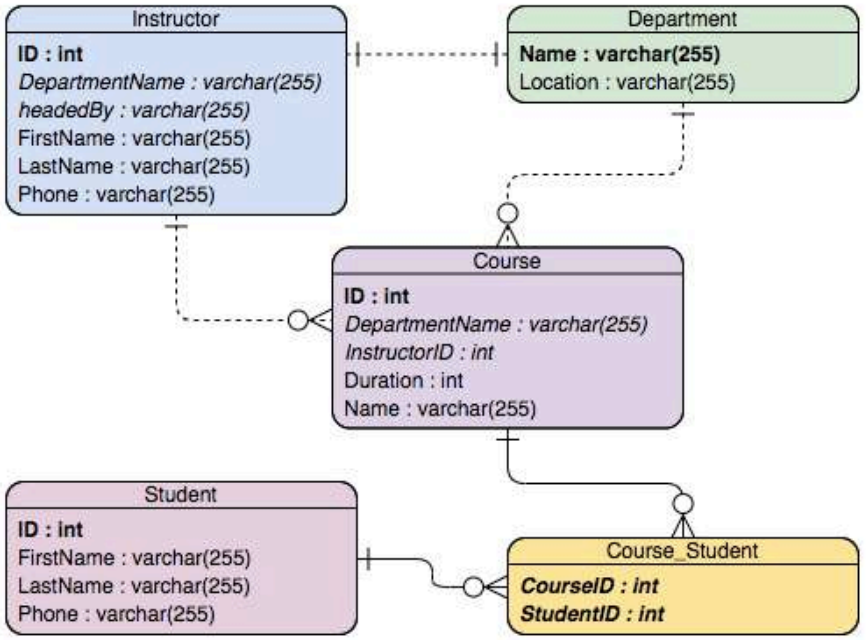
- **Only option** until a decade back!
- Most **popular (or unpopular)** type of databases
- **Predefined schema** - tables & relationships
- Supports **Complex SQL Queries**
- Very **strong transactional** capabilities
- Used for
  - OLTP (Online Transaction Processing) use cases and
  - OLAP (Online Analytics Processing) use cases



Relational Database - OLTP (Online Transaction Processing)

In28 Minutes

- Applications where **large number of users make large number of small transactions**
  - small data reads, updates and deletes
- **Use cases**: Most traditional applications, ERP, CRM, e-commerce, banking applications
- **Popular databases**: MySQL, Oracle, SQL Server etc



Amazon RDS (Relational Database Service)

In 28 Minutes

- Do you want to manage the setup, backup, scaling, replication and patching of your relational databases?
  - Or do you want to use a managed service?
- Amazon RDS is a managed relational database service for OLTP use cases
- Supports:
  - Amazon Aurora
  - PostgreSQL
  - MySQL (InnoDB storage engine full supported)
  - MariaDB (Enhanced MySQL)
  - Oracle Database
  - Microsoft SQL Server

Amazon RDS

Amazon RDS - Features

- Multi-AZ deployment (standby in another AZ)
- Read replicas:
  - Same AZ
  - Multi AZ (Availability+)
  - Cross Region(Availability++)
- Storage auto scaling (up to a configured limit)
- Automated backups (restore to point in time)
- Manual snapshots

Amazon RDS

Amazon RDS - You vs AWS

In 28 Minutes

- AWS is responsible for
  - Availability (according to your configuration)
  - Durability
  - Scaling (according to your configuration)
  - Maintenance (patches)
  - Backups
- You are responsible for
  - Managing database users
  - App optimization (tables, indexes etc)
- You CANNOT
  - SSH into database EC2 instances or setup custom software (NOT ALLOWED)
  - Install OS or DB patches. RDS takes care of them (NOT ALLOWED)

Amazon RDS

Amazon Aurora

- MySQL and PostgreSQL-compatible
- 2 copies of data each in a minimum of 3 AZ
- Provides "Global Database" option
  - Up to five read-only, secondary AWS Regions
    - Low latency for global reads
    - Safe from region-wide outages
  - Minimal lag time, typically less than 1 second

Amazon RDS - When to use?

In28 Minutes

- Use Amazon RDS for transactional applications needing
  - Pre-defined schema
  - Strong transactional capabilities
  - Complex queries
- Amazon RDS is **NOT recommended** when
  - You need highly scalable massive read/write operations - for example millions of writes/second
    - Go for DynamoDB
  - When you want to upload files using simple GET/PUT REST API
    - Go for Amazon S3
  - When you need heavy customizations for your database or need access to underlying EC2 instances
    - Go for a custom database installation

Amazon RDS

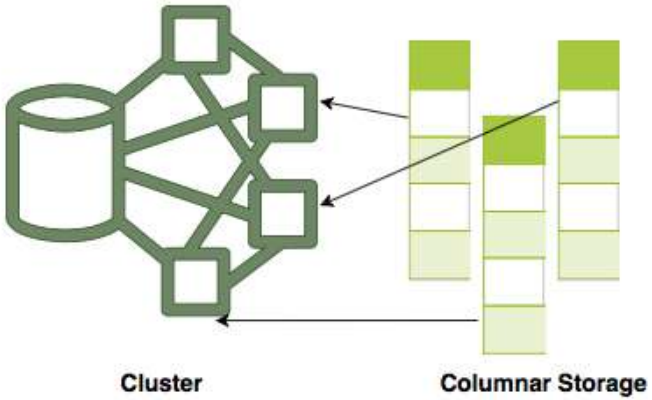
RDS - Scenarios

Scenario	Solution
You want full control of OS or need elevated permissions	Consider going for a custom installation (EC2 + EBS)
You want to migrate data from an on-premise database to cloud database of the same type	Consider using AWS Database Migration Service
I will need RDS for at least one year. How can I reduce costs?	Use Amazon RDS reserved instances.

Relational Database - OLAP (Online Analytics Processing)

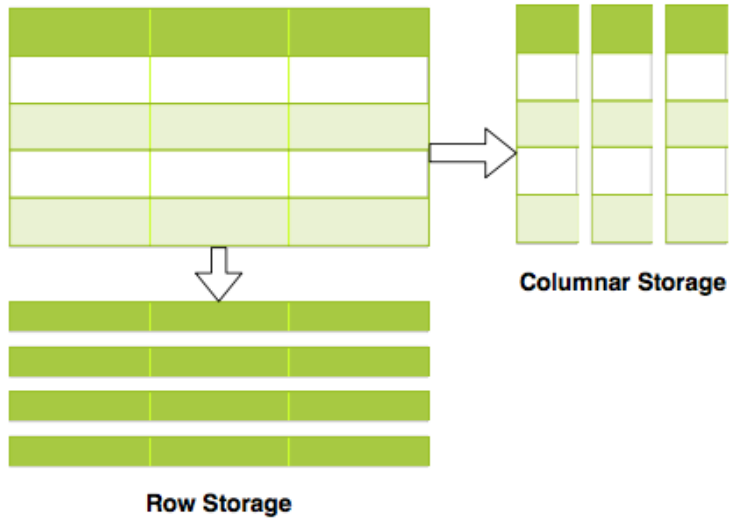
In 28 Minutes

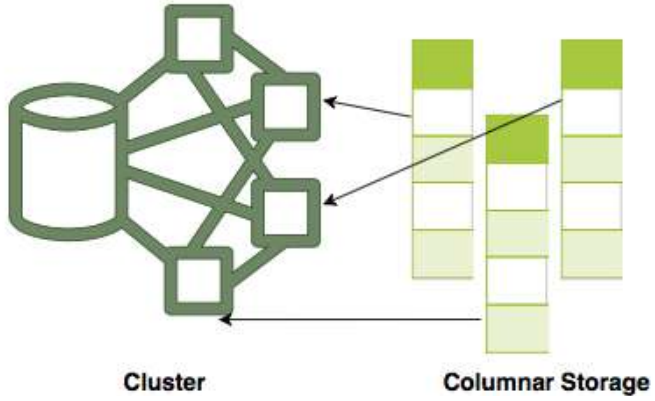
- Applications allowing users to **analyze petabytes of data**
- **Examples** : Reporting applications, Data ware houses, Business intelligence applications, Analytics systems
- **Sample application** : Decide insurance premiums analyzing data from last hundred years
- Data is consolidated from multiple (transactional) databases



Relational Databases - OLAP vs OLTP

- OLAP and OLTP use **similar data structures**
- BUT **very different approach in how data is stored**
- **OLTP databases** use row storage
  - Each table row is stored together
  - Efficient for processing small transactions
- **OLAP databases** use columnar storage
  - Each table column is stored together
  - **High compression** - store petabytes of data efficiently
  - **Distribute data** - one table in multiple cluster nodes
  - **Execute single query across multiple nodes** - Complex queries can be executed efficiently





Amazon Redshift

- Redshift is a relational database (tables and relationships)
- What is the need for another relational database?
  - RDS is optimized for online transaction processing
  - RDS is optimized to provide a balance between both reads and write operations
- (However) OLAP workloads have exponentially larger reads on the databases compared to writes:
  - Can we use a different approach to design the database?
  - How about creating a cluster and splitting the execution of the same query across several nodes?
- Redshift is a **petabyte-scale distributed data ware house** based on PostgreSQL

Redshift

Amazon Redshift

In 28 Minutes

- Three important characteristics of Redshift:
  - Massively parallel processing (MPP) - storage and processing can be split across multiple nodes
  - Columnar data storage
  - High data compression
- As a result
  - A single row of data might be stored across multiple nodes
  - A query to Redshift leader node is distributed to multiple compute nodes for execution
- Start with a single node configuration and scale to multi node configuration
- You can dynamically add and remove nodes

Amazon Redshift

In 28 Minutes

- Used for traditional ETL(Extract, Transform, Load), OLAP and Business Intelligence (BI) use cases
  - Optimized for high-performance analysis and reporting of very large datasets
- Supports standard SQL
- Integration with data loading, reporting, mining and analytics tools

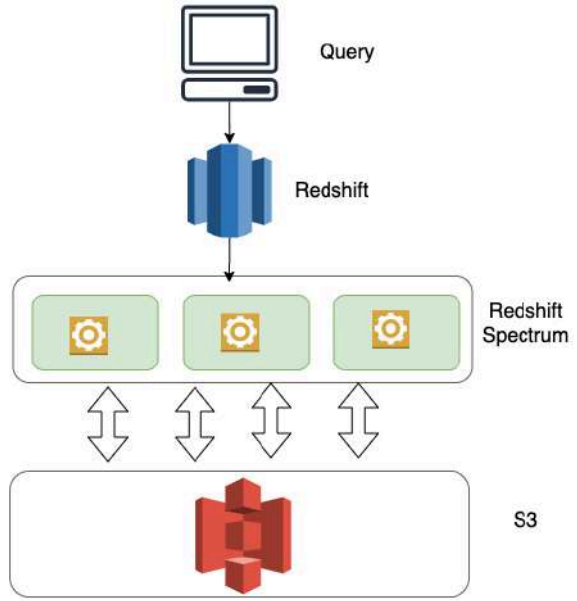
Redshift

Amazon EMR - Elastic MapReduce

- Managed Hadoop service with high availability and durability
- EMR gives access to underlying OS => You can SSH into it
- Important tools in Hadoop eco system are natively supported:
  - Examples: Pig, Hive, Spark or Presto
- Install others using bootstrap actions
- Use cases
  - Log processing for insights
  - Click stream analysis for advertisers
  - Genomic and life science dataset processing

Amazon Redshift Spectrum

- Run SQL queries against datasets in Amazon S3
  - Does need for any intermediate data stores
- Auto scales based on your queries
- Scale storage and compute independently
- Eliminate expensive data transfers from S3 to data warehousing solutions (Cost effective)
- Query against Amazon EMR (as well)



Amazon Redshift and EMR Alternatives

In28 Minutes

Alternative	Scenario
Amazon EMR	For big data frameworks like Apache Spark, Hadoop, Presto, or Hbase to do large scale data processing that needs high customization For example: machine learning, graph analytics etc
Amazon Redshift	Run complex queries against data warehouse - housing structured and unstructured data pulled in from a variety of sources
Amazon Redshift Spectrum	Run queries directly against S3 without worrying about loading entire data from S3 into a data warehouse
Amazon Athena	Quick ad-hoc queries without worrying about provisioning a compute cluster (serverless) Amazon Redshift Spectrum is recommended if you are executing queries frequently against structured data

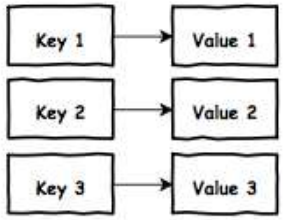
Document Database

- Structure data **the way your application needs it**
- Create **one table instead of dozens**!
- **Quickly evolving** semi structured data (**schema-less**)
- Easily **distributable**
- **Advantages** : (Horizontally) Scalable to **terabytes of data** with **millisecond responses** upto **millions of transactions per second**
- **Use cases** : Content management, catalogs, user profiles

Document Database


```
{
    "id": 1,
    "name": "Jane Doe",
    "username": "abcdefgh",
    "email": "someone@gmail.com",
    "address": {
      "street": "Some Street",
      "suite": "Apt. 556",
      "city": "Hyderabad",
      "zipcode": "500018",
      "geo": {
        "lat": "-3.31",
        "lng": "8.14"
      }
    },
    "phone": "9-999-999-9999",
    "website": "in28minutes.com",
    "company": {
```

Key-value



Key Value Database

userId	session
user1	{ "name": "Jane", "previousAction" : "someAction1" }
user2	{ "name": "Doe", "previousAction" : "someAction2" }
user3	{ "name": "Doe", "previousAction" : "someAction3" }

- Use a **simple key-value pair** to store data. Key is a unique identifier.
- Values can be objects, compound objects or simple data values
- **Advantages** : (Horizontally) Scalable to **terabytes of data** with **millisecond responses** upto **millions of transactions per second**
- **Use cases** : shopping carts, session stores, gaming applications and very high traffic web apps

Amazon DynamoDB

In 28 Minutes

- Fast, scalable, distributed for any scale
- Flexible NoSQL Key-value & document database (schemaless)
- Single-digit millisecond responses for million of TPS
- Do not worry about scaling, availability or durability
  - Automatically partitions data as it grows
  - Maintains 3 replicas within the same region
- No need to provision a database
  - Create a table and configure read and write capacity (RCU and WCU)
  - Automatically scales to meet your RCU and WCU
- Provides an expensive serverless mode
- Use cases: User profiles, shopping carts, high volume read write applications

DynamoDB

DynamoDB Tables

In28 Minutes

- Hierarchy : Table > item(s) > attribute (key value pair)
- Mandatory primary key
- Other than the primary key, tables are schemaless
  - No need to define the other attributes or types
  - Each item in a table can have distinct attributes
- Max 400 KB per item in table
  - Use S3 for large objects and DynamoDB for smaller objects


```
{
    "id": 1,
    "name": "Jane Doe",
    "username": "abcdefgh",
    "email": "someone@gmail.com",
    "address": {
      "street": "Some Street",
      "suite": "Apt. 556",
      "city": "Hyderabad",
      "zipcode": "500018",
      "geo": {
        "lat": "-3.31",
        "lng": "8.14"
      }
    },
    "phone": "9-999-999-9999",
    "website": "in28minutes.com",
    "company": {
      "name": "in28minutes"
    }
}
```

DynamoDB vs RDS

Feature	DynamoDB	RDS
Scenario	Millisecond latency with millions of TPS	Stronger consistency (schema) and transactional capabilities
Schema	Schemaless (needs only a primary key - Great for use cases where your schema is evolving)	Well-defined schema with relationships
Data Access	Using REST API provided by AWS using AWS SDKs or AWS Management Console or AWS CLI	SQL queries
Complex Data Queries Involving Multiple Tables	Difficult to run	Run complex relational queries with multiple entities
Scaling	No upper limits	64 TB
Consistency	Typically lower consistency	Typically higher consistency

In-memory Databases (or Caches)

In28 Minutes

- **Retrieving data from memory is much faster from retrieving data from disk**
- You can speed up dynamic database-driven websites by caching data and objects in memory. Ex: Memcached.
- You can deliver microsecond latency by storing **persistent data in memory**. Ex: Redis
- **Use cases** : Caching, session management, gaming leader boards, geospatial applications

Amazon ElastiCache

In28 Minutes

- Highly scalable & low latency in-memory data store
- Used for distributed caching
- (Option 1) ElastiCache Memcached:
  - Low maintenance simple caching solution
  - Easy horizontal scaling with auto discovery
  - Use case: Speed up database-driven websites by caching data
- (Option 2) ElastiCache Redis:
  - Persistence
  - Advanced Features:
    - Publish subscribe messaging
    - Read replicas and failover
    - Encryption
  - Usecases: gaming leader boards, queues, real-time analytics

ElastiCache

Databases - Summary

In 28 Minutes

Database Type	AWS Service	Description
Relational OLTP databases	Amazon RDS	Row storage Transactional usecases needing **predefined schema** and very **strong transactional** capabilities
Relational OLAP databases	Amazon Redshift	Columnar storage Reporting, analytics & intelligence apps needing **predefined schema**
Document & Key Databases	Amazon DynamoDB	Apps needing **quickly evolving** semi structured data (**schema-less**) Scale to **terabytes of data** with **millisecond responses** upto **millions of TPS** Content management, catalogs, user profiles, shopping carts, session stores and gaming applications

Databases - Summary

Database Type	AWS Service	Description
Graph Databases	Amazon Neptune	Store and navigate data with **complex relationships** Social Networking Data (Twitter, Facebook), Fraud Detection
In memory databases/caches	Amazon ElastiCache	Applications needing **microsecond** responses **Redis** - persistent data **Memcached** - simple caches

Databases - Questions

Scenario	Solution
A start up with quickly evolving tables	DynamoDB
Transaction application needing to process million transactions per second	DynamoDB
Very high consistency of data is needed while processing thousands of transactions per second	RDS
Cache data from database for a web application	Amazon ElastiCache
Relational database for analytics processing of petabytes of data	Amazon Redshift

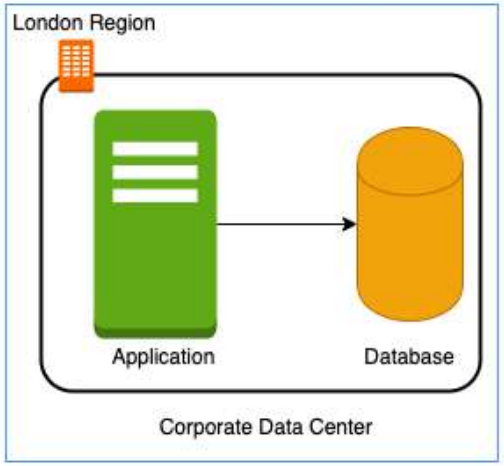
Other Storage Services

- Amazon DocumentDB
  - Managed document database service
  - Compatible with MongoDB
- Amazon Keyspaces
  - Managed service for Apache Cassandra
  - Serverless (Pay for use)
- AWS Backup
  - Centrally manage and automate backups across AWS services
  - Automate backup compliance and monitoring

Networking

Need for Amazon VPC

- In a corporate network or an on-premises data center:
  - Can anyone on the internet **see the data exchange** between the application and the database?
    - **No**
  - Can anyone from internet **directly connect to your database**?
    - Typically **NO**.
    - You need to connect to your corporate network and then access your applications or databases.
- Corporate network provides a **secure internal network** protecting your resources, data and communication from external users
- How do you do create **your own private network** in the cloud?
  - Enter **Virtual Private Cloud (VPC)**



Amazon VPC (Virtual Private Cloud)

In28 Minutes

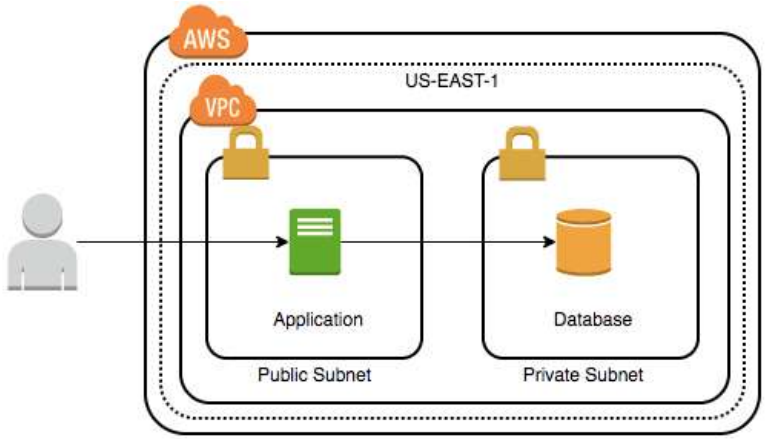
- Your **own isolated network** in AWS cloud
  - Network traffic within a VPC is isolated (not visible) from all other Amazon VPCs
- You **control all the traffic** coming in and going outside a VPC
- **(Best Practice)** Create all your AWS resources (compute, storage, databases etc) **within a VPC**
  - Secure resources from unauthorized access AND
  - Enable secure communication between your cloud resources

Need for VPC Subnets

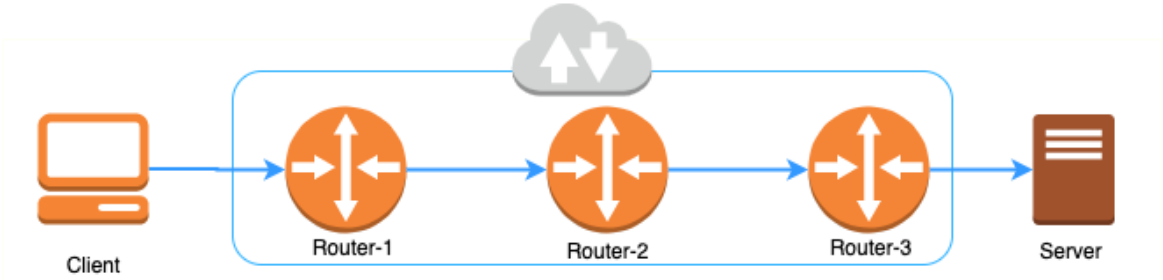
- Different resources are created on cloud - databases, compute (EC2) etc
- Each type of resource has **its own access needs**
- Public Elastic Load Balancers are accessible from internet (**public** resources)
- Databases or EC2 instances should NOT be accessible from internet
  - ONLY applications within your network (VPC) should be able to access them(**private** resources)
- How do you **separate public resources from private resources** inside a VPC?

VPC Subnets

- (Solution) **Create different subnets** for public and private resources
  - Resources in a public subnet **CAN** be accessed from internet
  - Resources in a private subnet **CANNOT** be accessed from internet
  - BUT resources in public subnet can talk to resources in private subnet
- Each VPC is created in a Region
- Each Subnet is created in an Availability Zone
- **Example** : VPC - us-east-1 => Subnets - AZs us-east-1a or us-east-1b or ..



Routing on the internet



- You have an IP address of a website you want to visit
- There is **no direct connection** from your computer to the website
- Internet is actually a **set of routers** routing traffic
- Each router has a set of rules that help it decide the path to the destination IP address

Routing inside AWS

In 28 Minutes

Destination	Target
172.31.0.0/16	Local
0.0.0.0/0	igw-1234567

- In AWS, **route tables** are used for routing
- Route tables can be associated with VPCs and subnets
- Each route table consists of a **set of rules** called routes
  - Each route or routing rule has a **destination and target**
  - What Range of Addresses should be routed to which target resource?
- **Rule 1** - Route requests to VPC CIDR 172.31.0.0/16 (172.31.0.0 to 172.31.255.255) to local resources within the VPC
- **Rule 2** - Route all other IP addresses (0.0.0.0/0) to internet (internet gateway)

Public Subnet vs Private Subnet

- Public Subnet
  - Communication allowed from subnet to internet
  - Communication allowed from internet to subnet
- Private Subnet
  - Communication NOT allowed from internet to subnet

Public Subnet vs Private Subnet

Name	Destination	Target	Explanation
RULE 1	172.31.0.0/16	Local	Local routing
RULE 2	0.0.0.0/0	igw-1234567	Internet routing

- An **Internet Gateway** enables internet communication for subnets
- Any subnet which has a route to an internet gateway is called a **public subnet**
- Any subnet which **DOES NOT** have route to an internet gateway is called a **private subnet**

Network Address Translation(NAT) Instance and Gateway

In 28 Minutes

- How do you **allow instances in a private subnet to download software updates** and security patches while denying inbound traffic from internet?
- How do you allow instances in a private subnet to **connect privately to other AWS Services** outside the VPC?
- **Three Options**:
  - **NAT Gateway**: Managed Service
  - **NAT Instance**: Install a EC2 instance with specific NAT AMI and configure as a gateway
  - **Egress-Only Internet Gateways**: For IPv6 subnets

NAT gateway vs NAT instance

Feature	NAT gateway	NAT instance
Managed by	AWS	You
Created in	Public Subnet	Public Subnet
Internet Gateway	Needed	Needed
High Availability	Yes (in an AZ) Multi AZ (higher availability)	You are responsible.

Network Access Control List

- Security groups control traffic to a specific resource in a subnet.
- How about stopping traffic from **even entering the subnet**?
- NACL provides **stateless firewall** at subnet level.
- Each subnet **must** be associated with a NACL.
- **Default NACL** allows all inbound and outbound traffic.
- **Custom created NACL** denies all inbound and outbound traffic by default.
- Rules have a priority number.
  - Lower number => Higher priority.

Security Group vs NACL

Feature	Security Group	NACL
Level	Assigned to a specific instance(s)/resource(s)	Configured for a subnet. Applies to traffic to all instances in a subnet.
Rules	Allow rules only	Both allow and deny rules
State	Stateful. Return traffic is automatically allowed.	Stateless. You should explicitly allow return traffic.
Evaluation	Traffic allowed if there is a matching rule	Rules are prioritized. Matching rule with highest priority wins.

VPC Flow Logs

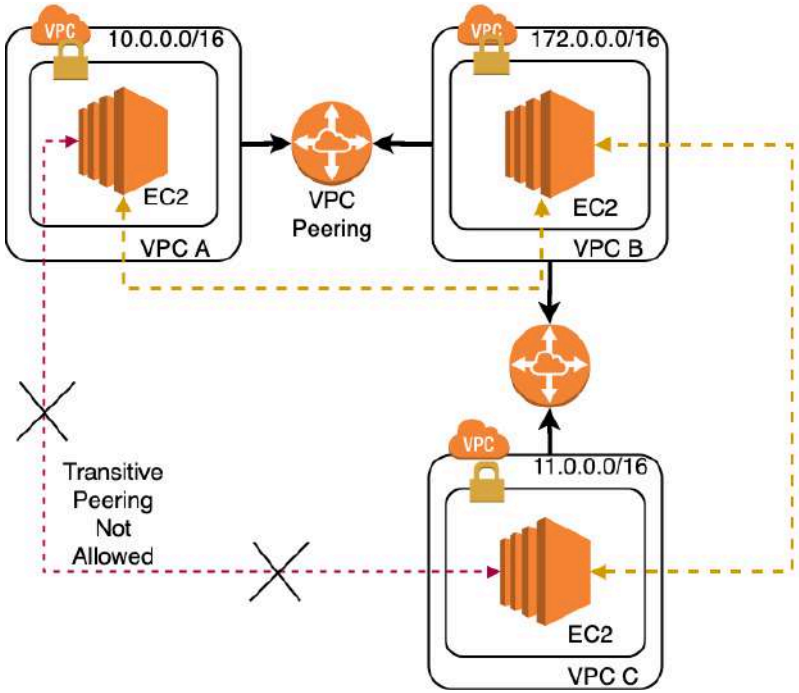
In 28 Minutes

- Monitor network traffic
- Troubleshoot connectivity issues (NACL and/or security groups misconfiguration)
- Capture traffic going in and out of your VPC (network interfaces)
- Can be created for
  - a VPC
  - a subnet
- Publish logs to Amazon CloudWatch Logs or Amazon S3
- Flow log records contain ACCEPT or REJECT
  - Is traffic is permitted by security groups or network ACLs?

VPC Flow Logs

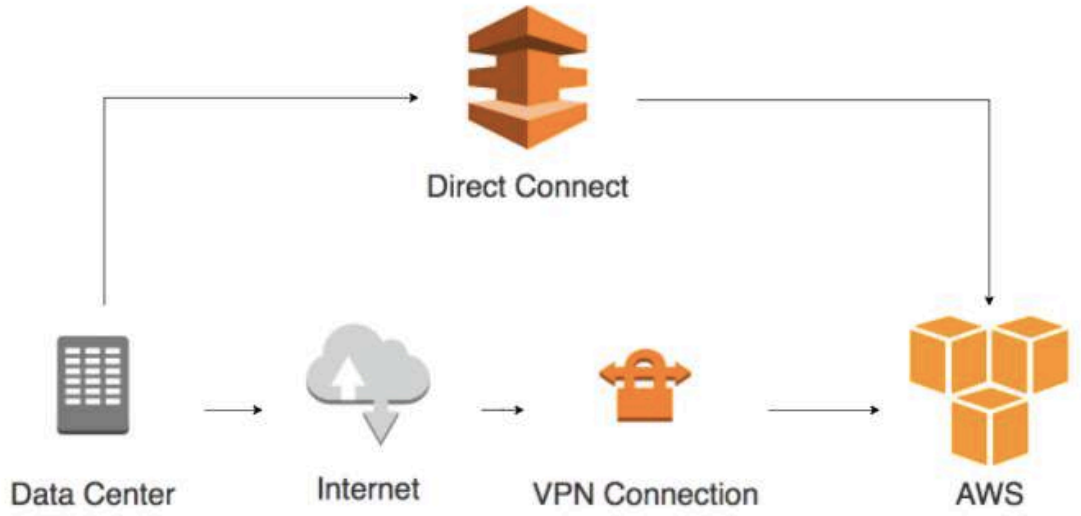
VPC Peering

- Connect VPCs belonging to same or different AWS accounts irrespective of the region of the VPCs
- Allows private communication between the connected VPCs
- Peering uses a request/accept protocol
  - Owner of requesting VPC sends a request
  - Owner of the peer VPC has one week to accept



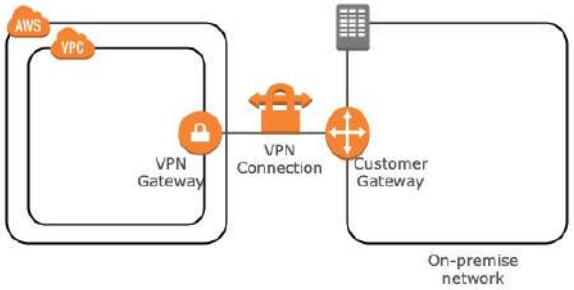
AWS and On-Premises - Overview

- AWS Managed VPN
  - IPsec VPN tunnels from VPC to customer network
- AWS Direct Connect (DX)
  - Private dedicated network connection from on-premises to AWS



AWS Managed VPN

- IPsec VPN tunnels from VPC to customer network
- Traffic over internet - encrypted using IPsec protocol
- VPN gateway to connect one VPC to customer network
- Customer gateway installed in customer network
  - You need a Internet-routable IP address of customer gateway



AWS Direct Connect (DC)

- Private dedicated network connection from on-premises to AWS
- Advantages:
  - Private network
  - Reduce your (ISP) bandwidth costs
  - Consistent Network performance because of private network
- (REMEMBER) Establishing DC connection can take more than a month

AWS Infrastructure Extension and Edge Services

In 28 Minutes

Service	Example Use Case	Explanation
AWS Local Zones	A gaming company providing seamless online multiplayer experience in a specific city	Extend AWS infrastructure to metro areas closer to end-users for single-digit millisecond latency. Ideal for real-time user engagement like online gaming.
AWS Outposts	A healthcare organization processing patient data with regulatory requirements to keep data on-premises	Bring native AWS services, infrastructure, and operating models to on-premises. Suitable for workloads with regulatory or data residency needs.
AWS Wavelength	Developing an augmented reality (AR) app for mobile devices requiring real-time edge data processing	Embed AWS infrastructure in mobile service providers data centers at the 5G network edge. Enables ultra-low latency & high-bandwidth applications for mobile & connected devices.

VPC - Review

- **VPC**: Virtual Network to protect resources and communication from outside world.
- **Subnet**: Seperate private resources from public resources
- **Internet Gateway**: Allows Public Subnets to connect/accept traffic to/from internet
- **NAT Gateway**: Allow internet traffic from private subnets
- **VPC Peering**: Connect one VPC with other VPCs
- **VPC Flow Logs**: Enable logs to debug problems
- **AWS Direct Connect**: Private pipe from AWS to on-premises
- **AWS VPN**: Encrypted (IPsec) tunnel over internet to on-premises

IAM - Fundamentals

Typical identity management in the cloud

- You have **resources** in the cloud (examples - a virtual server, a database etc)
- You have **identities (human and non-human)** that need to access those resources and perform actions
  - For example: launch (stop, start or terminate) a virtual server
- How do you **identify users** in the cloud?
- How do you configure resources they can access?
- How can you configure what actions to allow?
- In AWS, *Identity and Access Management (IAM)* provides this service

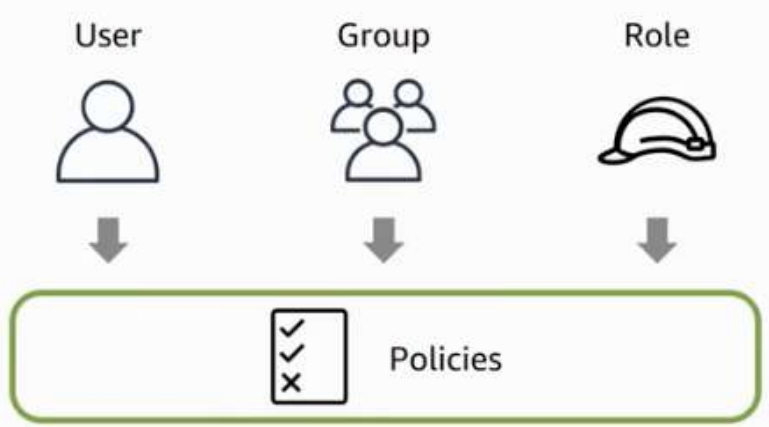
AWS Identity and Access Management (IAM)

In 28 Minutes

- **Authentication** (is it the right user?) and
- **Authorization** (do they have the right access?)
- **Identities** can be
  - AWS users or
  - Federated users (externally authenticated users)
- Provides very **granular** control
  - Limit a single user:
    - to perform single action
    - on a specific AWS resource
    - from a specific IP address
    - during a specific time window

Important IAM Concepts

- **IAM users:** Users created in an AWS account
  - Has credentials attached (name/password or access keys)
- **IAM groups:** Collection of IAM users
- **Roles:** Temporary identities
  - Does NOT have credentials attached
  - (Advantage) Expire after a set period of time
- **Policies:** Define permissions
  - **AWS managed policies** - Standalone policy predefined by AWS
  - **Customer managed policies** - Standalone policy created by you
  - **Inline policies** - Directly embedded into a user, group or role

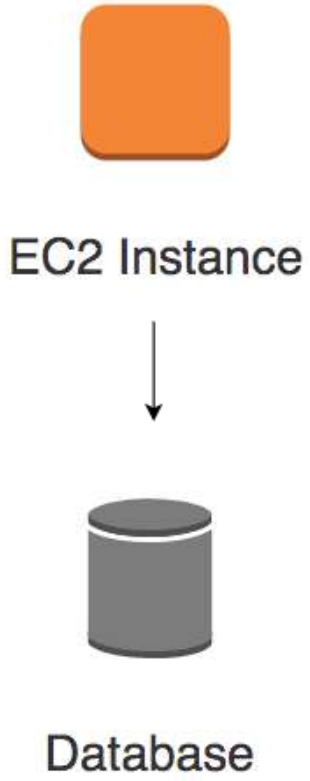


In28 Minutes

Data Encryption
KMS & Cloud HSM

Data States

- **Data at rest**: Stored on a device or a backup
  - Examples : data on a hard disk, in a database, backups and archives
- **Data in motion**: Being transferred across a network
  - Also called **Data in transit**
  - **Examples** :
    - Data copied from on-premise to cloud storage
    - An application in a VPC talking to a database
  - **Two Types**:
    - In and out of AWS
    - Within AWS
- **Data in use**: Active data processed in a non-persistent state
  - Example: Data in your RAM

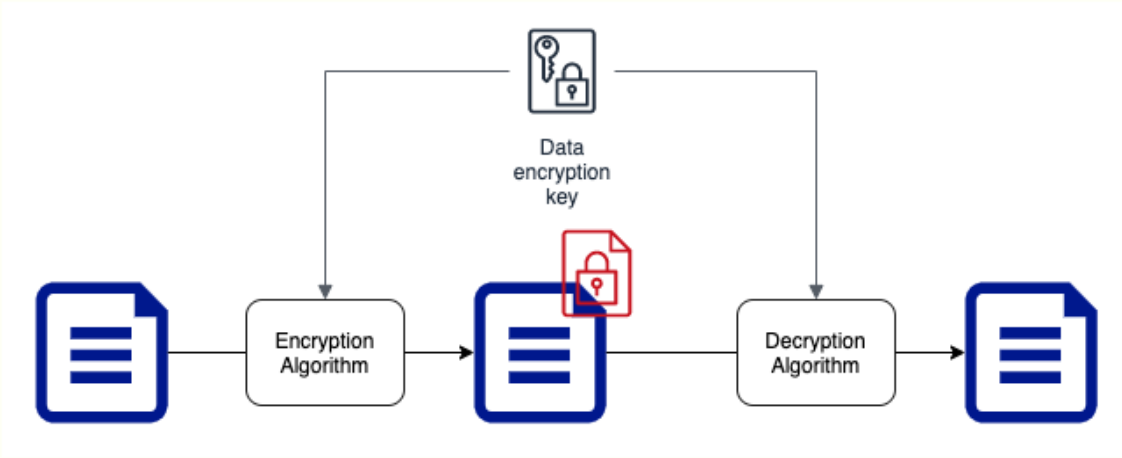


Encryption

In 28 Minutes

- If you store data as is, what would happen if an **unauthorized entity gets access** to it?
  - Imagine losing an unencrypted hard disk
- **First law of security** : Defense in Depth
- Typically, enterprises encrypt all data
  - Data on your hard disks
  - Data in your databases
  - Data on your file servers
- Is it sufficient if you encrypt data at rest?
  - **No**. **Encrypt data in transit** - between application to database as well.

Symmetric Key Encryption

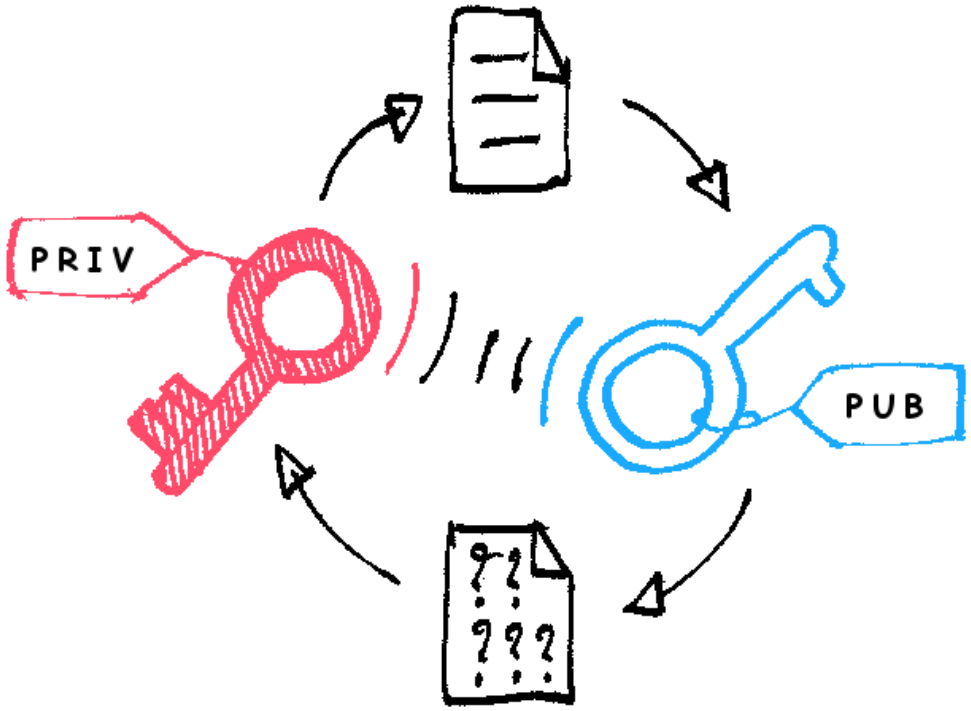


- Symmetric encryption algorithms use the **same key for encryption and decryption**
- Key Factor 1: Choose the **right encryption algorithm**
- Key Factor 2: How do we **secure the encryption key**?
- Key Factor 3: How do we **share the encryption key**?

Asymmetric Key Encryption

In 28 Minutes

- **Two Keys** : Public Key and Private Key
- Also called **Public Key Cyptography**
- Encrypt data with Public Key and decrypt with Private Key
- Share Public Key with everybody and keep the Private Key with you(YEAH, ITS PRIVATE!)
- No crazy questions:
  - Will somebody not figure out private key using the public key?
- How do you create Asymmetric Keys?



https://commons.wikimedia.org/wiki/File:Asymmetric_encryption_(colored).p

KMS and Cloud HSM

- How do you generate, store, use and replace your keys?
- AWS provides two important services - **KMS** and **Cloud HSM**
  - Manage your keys
  - Perform encryption and decryption

AWS KMS

Cloud HSM

AWS KMS

- Create and manage **cryptographic keys** (symmetric and asymmetric)
- **Control their use** in your applications and AWS Services
- Define key usage permissions (including **cross account** access)
- Track key usage in AWS CloudTrail (regulations & compliance)
- **Integrates with almost all AWS services** that need data encryption
- **Automatically rotate master keys** once a year
  - No need to re-encrypt previously encrypted data (versions of master key are maintained)
- **Schedule key deletion** to verify if the key is used
  - Mandatory minimum wait period of 7 days (max-30 days)

AWS KMS

AWS CloudHSM

- Managed (highly available & auto scaling) **dedicated single-tenant** Hardware Security Module(HSM) for regulatory compliance
  - (Remember) AWS KMS is a multi-tenant service
- FIPS 140-2 Level 3 compliant
- AWS **CANNOT access your encryption master keys** in CloudHSM
  - In KMS, AWS can access your master keys
  - Be ultra safe with your keys when you are using CloudHSM
  - **(Recommendation)** Use two or more HSMs in separate AZs in a production cluster

Cloud HSM

AWS CloudHSM

- AWS KMS can use CloudHSM cluster as **"custom key store"** to store the keys:
  - AWS Services can continue to talk to KMS for data encryption
  - (AND) KMS does the necessary integration with CloudHSM cluster
- (Best Practice) **CloudWatch** for monitoring and **CloudTrail** to track key usage
- **Use cases**
  - (Web servers) Offload SSL processing
  - Certificate Authority
  - Digital Rights Management
  - TDE for Oracle databases

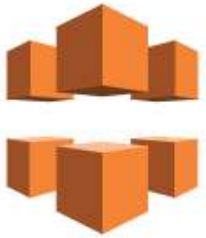
AWS Shield

- Shields from Distributed Denial of Service (DDoS) attacks
  - Disrupt normal traffic of a server by overwhelming it with a flood of Internet traffic
- Protect
  - Amazon Route 53
  - Amazon CloudFront
  - AWS Global Accelerator
  - Amazon Elastic Compute Cloud (EC2) instances
  - Elastic Load Balancers (ELB)

AWS Shield - Standard and Advanced

- AWS Shield Standard
  - Zero Cost. Automatically enabled.
  - Protection against common infrastructure (layer 3 and 4) DDoS attacks
- AWS Shield Advanced
  - Paid service
  - Enhanced protection for Amazon EC2, Elastic Load Balancing (ELB), Amazon CloudFront, AWS Global Accelerator, and Amazon Route 53
  - 24x7 access to the AWS DDoS Response Team (DRT)
  - Protects your AWS bill from usage spikes as a result of a DDoS attack
- Protect any web application (from Amazon S3 or external) from DDoS by putting Amazon CloudFront enabled with AWS Shield in front of it

AWS Shield



CloudFront

AWS WAF - Web Application Firewall

In28 Minutes

- AWS WAF protect your web applications from OWASP Top 10 exploits, CVE and a lot more!
  - OWASP (Open Web Application Security Project) Top 10
    - List of broadly agreed "**most critical security risks to web applications**"
    - Examples : SQL injection, cross-site scripting etc
  - Common Vulnerabilities and Exposures (CVE) is a list of information-security vulnerabilities and exposures
- Can be deployed on Amazon CloudFront, Application Load Balancer, Amazon API Gateway
- Customize rules & trigger realtime alerts (CloudWatch Alarms)
- Web traffic filtering : block attacks
  - Filter traffic based on IP addresses, geo locations, HTTP headers and body (block attacks from specific user-agents, bad bots, or content scrapers)

Amazon Macie

- Fully managed data security and data privacy service
- Automatically discover, classify, and protect sensitive data in Amazon S3 buckets
- When migrating data to AWS use S3 for staging
  - Run Macie to discover secure data
- Uses machine learning
- Recognizes sensitive data
  - Example: personally identifiable information (PII) or intellectual property
- Provides you with dashboards and alerts
  - Gives visibility into how data is being accessed or moved

AWS Inspector: Enhanced Security Scanning

- **AWS Inspector**: Automated Security Scanning
- **Discover AWS workloads**: Scans Amazon EC2 instances, containers, and Lambda functions for vulnerabilities
- **Security**: Identifies software vulnerabilities and checks for unintended network exposures
- **Compliance**: Helps ensure your AWS workloads comply with security standards and best practices
- **Continuous monitoring**: Automatically assesses new and existing workloads to improve your security posture over time

AWS Systems Manager Parameter Store

- Manage application environment configuration and secrets
  - database connections, password etc
- Supports hierarchical structure
- Store configuration at one place
  - multiple applications
  - multiple environments
- Maintains history of configuration over a period of time
- Integrates with KMS, IAM, CloudWatch and SNS

AWS Secrets Manager

In 28 Minutes

- Rotate, Manage and retrieve database credentials, API keys, and other secrets for your applications
- Integrates with KMS(encryption), Amazon RDS, Amazon Redshift, and Amazon DocumentDB
- (KEY FEATURE) Rotate secrets automatically without impacting applications
- (KEY FEATURE) Service dedicated to secrets management
- Recommended for workloads needing HIPAA, PCI-DSS compliance

AWS Single Sign On

- Cloud-based single sign-on (SSO) service
- Centrally manage SSO access to all of your AWS accounts
- Integrates with Microsoft AD (Supports using your existing corporate accounts)
- Supports Security Assertion Markup Language (SAML) 2.0
- Deep integration with AWS Organizations (Centrally manage access to multiple AWS accounts)
- One place auditing in AWS CloudTrail

Other Important Security Services

In28 Minutes

- **Amazon GuardDuty**
  - Continuously monitor your AWS environment for suspicious activity (Intelligent Threat Detection).
  - Analyze AWS CloudTrail events, VPC Flow Logs etc.
- **AWS Certificate Manager**
  - Provision, manage, deploy, and renew SSL/TLS certificates on the AWS platform
- **AWS Artifact**
  - Self-service portal for on-demand access to AWS compliance reports, certifications, accreditations, and other third-party attestations.
  - Review, accept, and manage your agreements with AWS.

Other Important Security Services

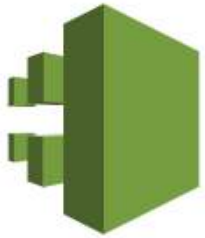
In28 Minutes

- **AWS Security Hub**
  - Consolidated view of your security status in AWS.
  - Automate security checks, manage security findings, and identify the highest priority security issues across your AWS environment.
- **Amazon Detective**
  - Investigate and quickly identify the root cause of potential security issues.
  - Automatically collect log data from your AWS resources and uses machine learning to help you visualize and conduct security investigations.
- **Penetration Testing**
  - Testing application security by simulating an attack.
  - You do not need permission from AWS to do penetration testing on a limited set of services (EC2 instances, ELB, RDS, CloudFront, API Gateway, Lambda, Elastic BeanStalk)

CloudTrail, Config & CloudWatch

AWS CloudTrail

- Track events, API calls, changes made to your AWS resources:
  - Who made the request?
  - What action was performed?
  - What are the parameters used?
  - What was the end result?
- (USE CASE) Compliance with regulatory standards
- (USE CASE) Troubleshooting. Locate a missing resource
- Delivers log files to S3 and/or Amazon cloud watch logs log group (S3 is default)
- You can setup SNS notifications for log file delivery



AWS CloudTrail

AWS Cloud Trail Types

- Multi Region Trail
  - One trail of all AWS regions
  - Events from all regions can be sent to one CloudWatch logs log group
- Single Region Trail
  - Only events from one region
  - Destination S3 bucket can be in any region

AWS CloudTrail

AWS Cloud Trail - Good to know

In 28 Minutes

- Log files are automatically encrypted with Amazon S3 SSE
- You can configure S3 Lifecycle rules to archive or delete log files
- Supports log file integrity
  - You can prove that a log file has not been altered

AWS CloudTrail

AWS Config

- **Auditing**
  - Create a complete inventory of your AWS resources
- **Resource history and change tracking**
  - Find how a resource was configured at any point in time
  - Configuration of deleted resources would be maintained
  - Delivers history file to S3 bucket every 6 hours
  - Take configuration snapshots when needed
- **Governance**
  - Customize Config Rules for specific resources or for entire AWS account
  - Continuously evaluate compliance against desired configuration
  - Get a SNS notification for every configuration change
- **Consistent rules and compliance** across AWS accounts:
  - Group Config Rules and Remediation Actions into Conformance Packs

AWS Config

Predefined Config Rule Examples (80+)

In 28 Minutes

- **alb-http-to-https-redirection-check** - Checks whether HTTP to HTTPS redirection is configured on all HTTP listeners of Application Load Balancers
- **ebs-optimized-instance** - Checks whether EBS optimization is enabled for your EC2 instances that can be EBS-optimized
- **ec2-instance-no-public-ip** - Do EC2 instances have public IPs?
- **encrypted-volumes** - Are all EC2 instance attached EBS volumes encrypted?
- **eip-attached** - Are all Elastic IP addresses used?
- **restricted-ssh** - Checks whether security groups that are in use disallow unrestricted incoming SSH traffic

AWS Config

AWS Config Rules

- (Feature) Create Lambda functions with your custom rules
- (Feature) You can setup auto remediation for each rule
  - Take immediate action on a non compliant resource
  - (Example) Stop EC2 instances without a specific tag!
- Enable AWS Config to use the rules
  - No Free Tier
  - More rules to check => More $$$$

AWS Config

AWS Config + AWS CloudTrail

- AWS Config
  - What did my AWS resource look like?
- AWS CloudTrail
  - Who made an API call to modify this resource?

Monitoring AWS with Amazon CloudWatch

In 28 Minutes

- Monitoring and observability service
- Collects monitoring and operational data in the form of logs, metrics, and events
- Set alarms, visualize logs, take automated actions and troubleshoot issues
- Integrates with more than 70 AWS services:
  - Amazon EC2
  - Amazon DynamoDB
  - Amazon S3
  - Amazon ECS
  - AWS Lambda
  - and ....

Cloudwatch

Amazon CloudWatch Logs

In 28 Minutes

- Monitor and troubleshoot using system, application and custom log files
- Real time application and system monitoring
  - Monitor for patterns in your logs and trigger alerts based on them
  - Example : Errors in a specific interval exceed a certain threshold
- Long term log retention
  - Store logs in CloudWatch Logs for as long as you want (configurable - default:forever)
  - Or archive logs to S3 bucket (Typically involves a delay of 12 hours)
  - Or stream real time to Amazon Elasticsearch Service (Amazon ES) cluster using CloudWatch Logs subscription

Cloudwatch

Amazon CloudWatch Logs

In28 Minutes

- **CloudWatch Logs Agent**
  - Installed on ec2 instances to move logs from servers to CloudWatch logs
- **CloudWatch Logs Insights**
  - Write queries and get actionable insights from your logs
- **CloudWatch Container Insights**
  - Monitor, troubleshoot, and set alarms for your containerized applications running in EKS, ECS and Fargate

Amazon CloudWatch Alarms

- Create alarms based on:
  - Amazon EC2 instance CPU utilization
  - Amazon SQS queue length
  - Amazon DynamoDB table throughput or
  - Your own custom metrics

Amazon CloudWatch Alarms

- Take immediate action:
  - Send a SNS event notification
    - Send an email using SNS
  - Execute an Auto Scaling policy

Amazon CloudWatch Alarm - Example

In28 Minutes

- You set a CPU Utilization alarm on EC2 instance with a threshold of 80% over 3 periods of 10 minutes. If CPU utilization is 90% for 20 minutes, does the alarm get triggered?
  - No

Amazon CloudWatch Dashboards

- Create auto refreshed graphs around all CloudWatch metrics
- Automatic Dashboards are available for most AWS services and resources
- Each Dashboard can have graphs from multiple regions

Amazon CloudWatch Events

In 28 Minutes

- Enable you to take immediate action based on events on AWS resources
  - Call a AWS Lambda function when an EC2 instance starts
  - Send event to an Amazon Kinesis stream when an Amazon EBS volume is created
  - Notify an Amazon SNS topic when an Auto Scaling event happens
- Schedule events - Use Unix cron syntax
  - Schedule a call to Lambda function every hour
  - Send a notification to Amazon SNS topic every 3 hours

Cloudwatch

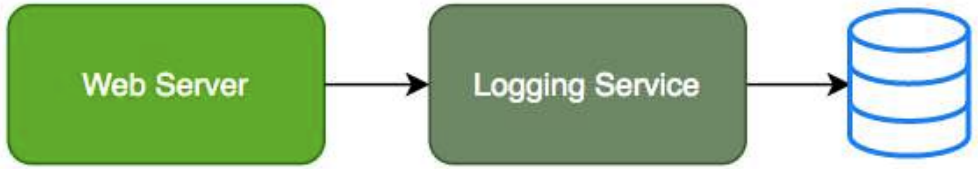
Decoupling Applications with SQS, SNS and MQ

Need for Asynchronous Communication

In28 Minutes

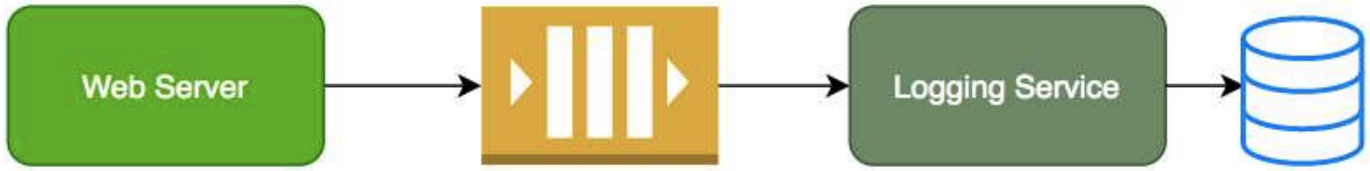
- Why do we need asynchronous communication?

Synchronous Communication



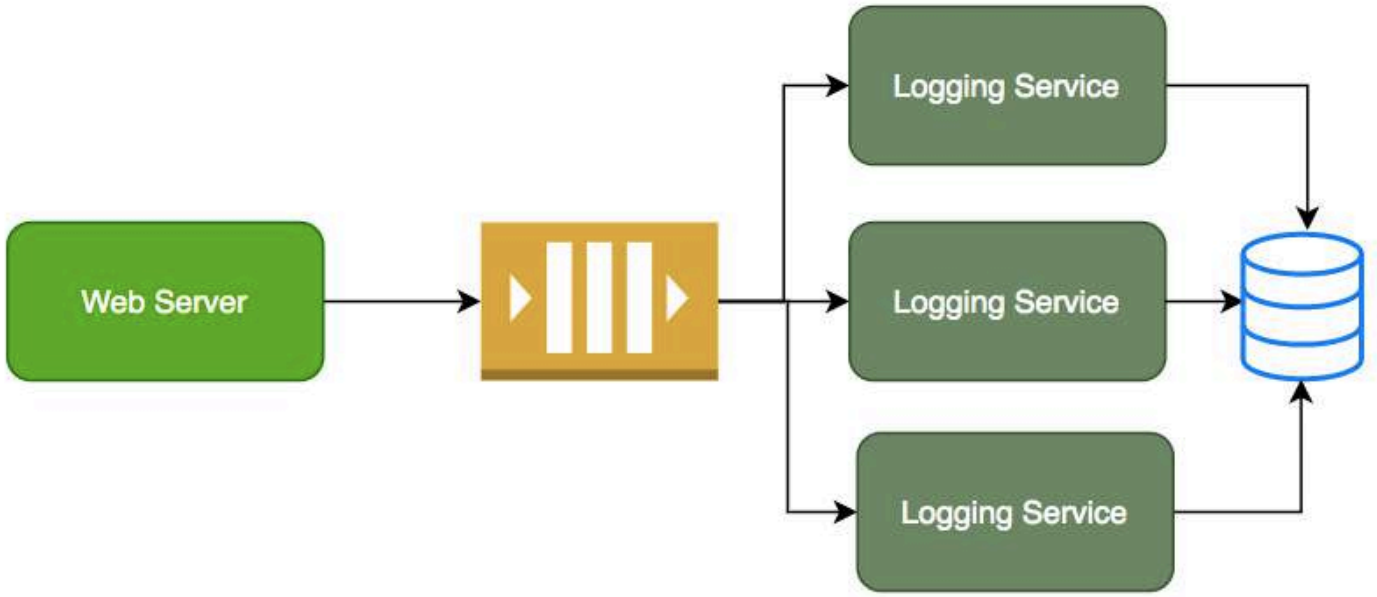
- Applications on your web server make synchronous calls to the logging service
- What if your logging service goes down?
  - Will you applications go down too?
- What if all of sudden, there is high load and there are lot of logs coming in?
  - Log Service is not able to handle the load and goes down very often

Asynchronous Communication - Decoupled



- Create a queue or a topic
- Your applications put the logs on the queue
- They would be picked up when the logging service is ready
- Good example of decoupling!

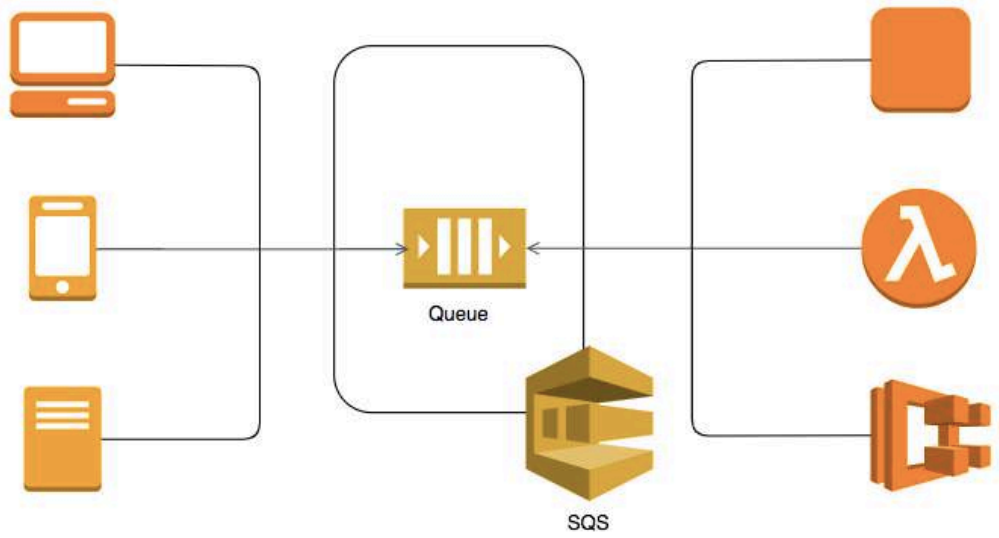
Asynchronous Communication - Scale up



- You can have multiple logging service instances reading from the queue!

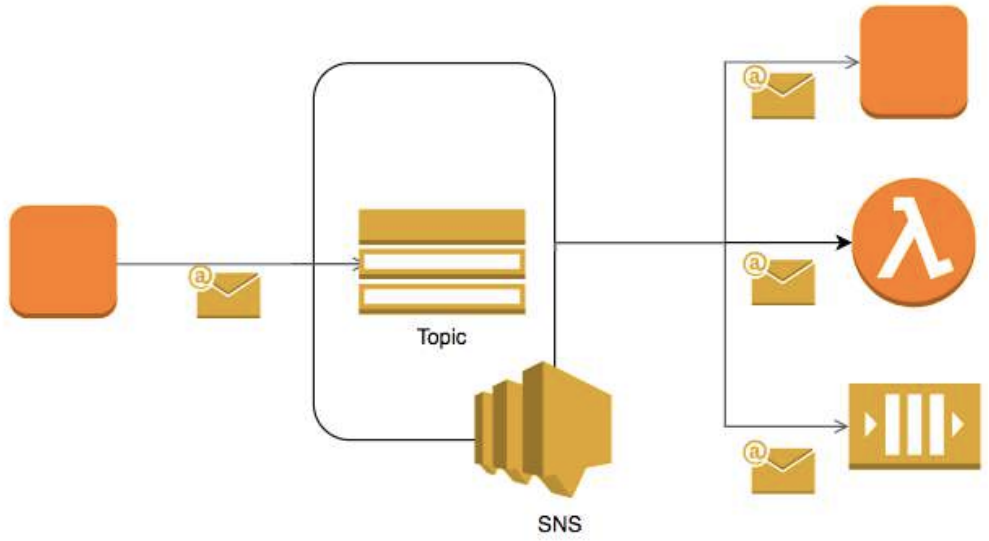
Asynchronous Communication - Pull Model - SQS

- Producers put messages on the queue
- Consumers poll on the queue
  - Only one of the consumers will successfully process a given message
- Scalability
  - Scale consumer instances under high load
- Availability
  - Producer up even if a consumer is down
- Reliability
  - Work is not lost due to insufficient resources
- Decoupling
  - Make changes to consumers without effect on producers worrying about them



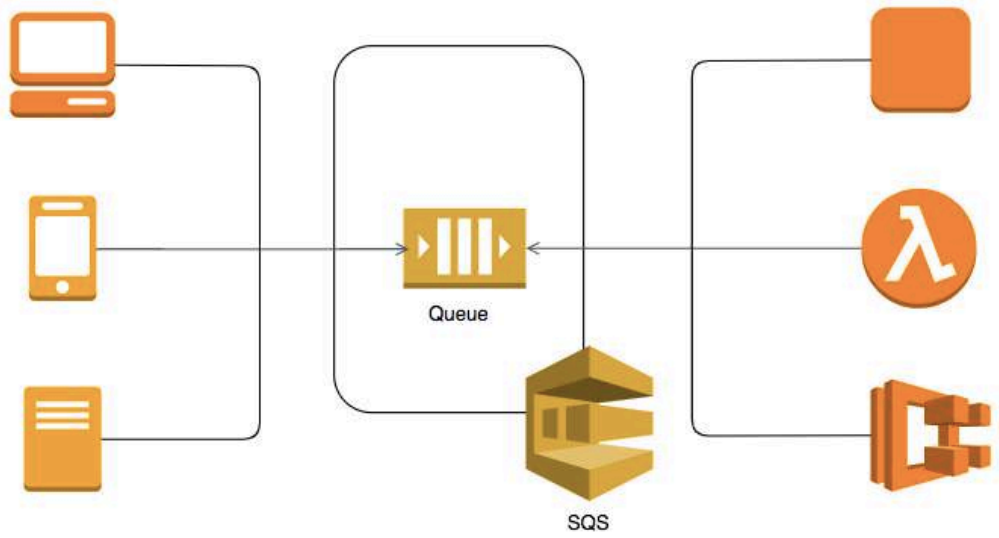
Asynchronous Communication - Push Model - SNS

- Subscribers subscribe to a topic
- Producers send notifications to a topic
  - Notification sent out to all subscribers
- Decoupling
  - Producers don't care about who is listening
- Availability
  - Producer up even if a subscriber is down



Simple Queuing Service

- Reliable, scalable, fully-managed message queuing service
- High availability
- Unlimited scaling
  - Auto scale to process billions of messages per day
- Low cost (Pay for use)



Standard and FIFO Queues

In28 Minutes

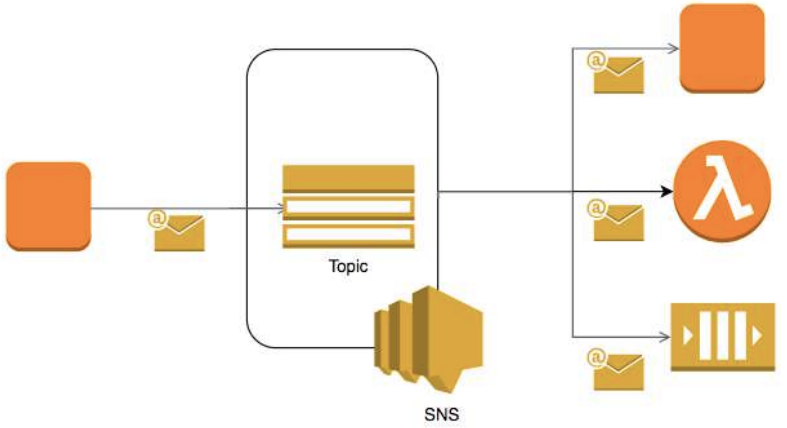
- Standard Queue
  - Unlimited throughput
  - BUT NO guarantee of ordering (Best-Effort Ordering)
  - and NO guarantee of exactly-once processing
    - Guarantees at-least-once delivery (some messages can be processed twice)
- FIFO (first-in-first-out) Queue
  - First-In-First-out Delivery
  - Exactly-Once Processing
  - BUT throughput is lower
    - Upto 300 messages per second (300 send, receive, or delete operations per second)
    - If you batch 10 messages per operation (maximum), up to 3,000 messages per second
- Choose
  - Standard SQS queue if throughput is important
  - FIFO Queue if order of events is important

Amazon SQS

Amazon Simple Notification Service(SNS)

In 28 Minutes

- Publish-Subscribe (pub-sub) paradigm
- Broadcast asynchronous event notifications
- Simple process
  - Create an SNS Topic
  - Subscribers can register for a Topic
  - When an SNS Topic receives an event notification (from publisher), it is broadcast to all Subscribers
- Use Cases : Monitoring Apps, workflow systems, mobile apps



Amazon Simple Notification Service(SNS)

In 28 Minutes

- Provides mobile and enterprise messaging web services
  - Push notifications to Apple, Android, FireOS, Windows devices
  - Send SMS to mobile users
  - Send Emails
- REMEMBER : SNS does not need SQS or a Queue
- You can allow access to other AWS accounts using AWS SNS generated policy

Amazon SNS

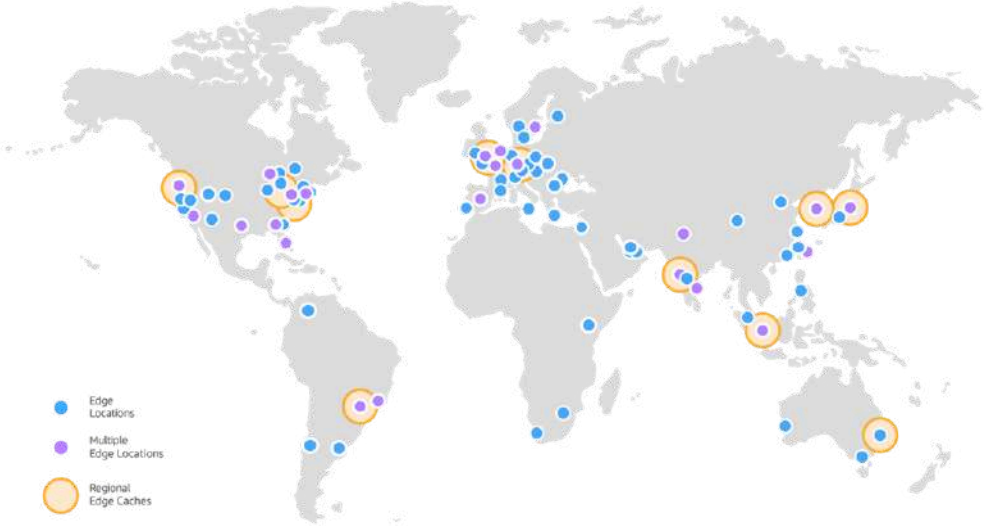
Amazon MQ

- Managed message broker service for Apache ActiveMQ
- (Functionally) Amazon MQ = Amazon SQS (Queues) + Amazon SNS (Topics)
  - BUT with restricted scalability
- Supports traditional APIs (JMS) and protocols (AMQP, MQTT, OpenWire, and STOMP)
  - Easy to migrate on-premise applications using traditional message brokers
  - Start with Amazon MQ as first step and slowly re-design apps to use Amazon SQS and/or SNS
- Scenario: An enterprise uses AMQP (standard message broker protocol). They want to migrate to AWS without making code changes
  - Recommend Amazon MQ

Routing and Content Delivery

Content Delivery Network

- You want to deliver content to your global audience
- Content Delivery Networks distribute content to multiple edge locations around the world
- AWS provides 200+ edge locations around the world
- Provides high availability and performance



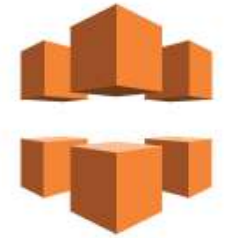
Amazon CloudFront

- How do you enable serving content directly from AWS edge locations?
  - Amazon CloudFront (one of the options)
- Serve users from nearest edge location (based on user location)
- Source content can be from S3, EC2, ELB and External Websites
- If content is not available at the edge location, it is retrieved from the origin server and cached
- No minimum usage commitment
- Provides features to protect your private content

Amazon CloudFront

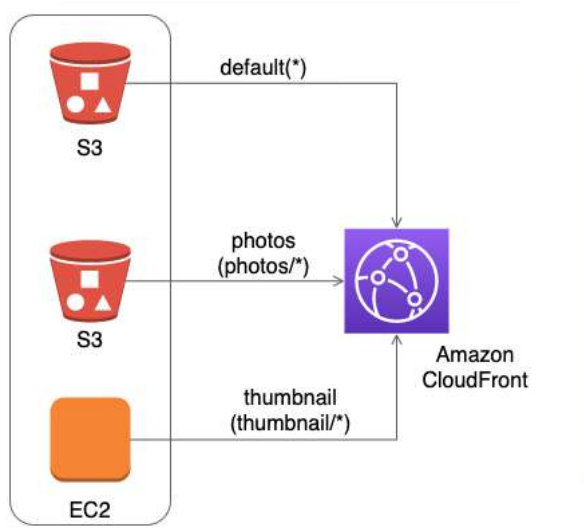
In 28 Minutes

- Use Cases
  - Static web apps. Audio, video and software downloads. Dynamic web apps
  - Support media streaming with HTTP and RTMP
- Integrates with
  - AWS Shield to protect from DDoS attacks
  - AWS Web Application Firewall (WAF) to protect from SQL injection, cross-site scripting, etc
- Cost Benefits
  - Zero cost for data transfer between S3 and CloudFront
  - Reduce compute workload for your EC2 instances



Amazon CloudFront Distribution

- Create a CloudFront Distribution to distribute your content to edge locations
  - DNS domain name - example abc.cloudfront.com
  - Origins - Where do you get content from? S3, EC2, ELB, External Website
  - Cache-Control
    - By default objects expire after 24 hours
    - Customize min, max, default TTL in CloudFront distribution
    - (For file level customization) Use Cache-Control max-age and Expires headers in origin server
- You can configure CloudFront to only use HTTPS (or) use HTTPS for certain objects
  - Default is to support both HTTP and HTTPS
  - You can configure CloudFront to redirect HTTP to HTTPS



AWS Edge Locations: Content Delivery Hubs

In 28 Minutes

- **AWS Edge Locations**: Delivery Hubs for Your Apps
- **Used by Amazon CloudFront**: CloudFront distributes your static content (images, videos, etc.) across a global network of edge locations, minimizing latency and improving delivery speeds for users worldwide
- **Used by Global Accelerator**: Global Accelerator intelligently routes user traffic to the closest AWS edge location, minimizing latency and improving loading times.
- **Used by Amazon S3 Transfer Acceleration**: Accelerates long-distance transfers to and from your Amazon S3 buckets

Route 53

- What would be the steps in setting up a website with a domain name (for example, in28minutes.com)?
  - Step I : Buy the domain name in28minutes.com (Domain Registrar)
  - Step II : Setup your website content (Website Hosting)
  - Step III : Route requests to in28minutes.com to the my website host server (DNS)
- Route 53 = Domain Registrar + DNS
  - Buy your domain name
  - Setup your DNS routing for in28minutes.com

Route 53 - DNS (Domain Name Server)

How should traffic be routed for in28minutes.com?

- Configure Records:
  - Route api.in28minutes.com to the IP address of api server
  - Route static.in28minutes.com to the IP address of http server
  - Route email (ranga@in28minutes.com) to the mail server(mail.in28minutes.com)
  - Each record is associated with a TTL (Time To Live) - How long is your mapping cached at the routers and the client?

Route 53 Hosted Zone

- Container for records containing DNS records routing traffic for a specific domain
- I want to use Route 53 to manage the records (Name Server) for in28minutes.com
  - Create a hosted zone for in28minutes.com in Route 53
- Hosted zones can be
  - private - routing within VPCs
  - public - routing on internet
- Manage the DNS records in a Hosted Zone

Standard DNS Records

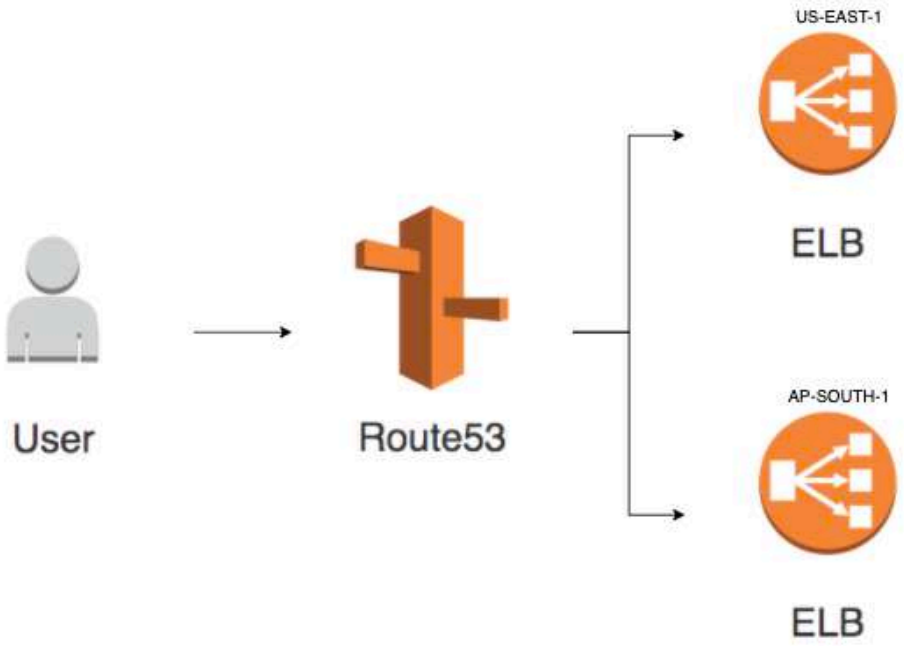
- A - Name to IPV4 address(es)
- AAAA - Name to IPV6 address(es)
- NS - Name Server containing DNS records
  - I bought in28minutes.com from GoDaddy (Domain Registrar)
  - BUT I can use Route 53 as DNS
    - Create NS records on GoDaddy
    - Redirect to Route 53 Name Servers
- MX - Mail Exchange
- CNAME - Name1 to Name2

Name	Type	Value
api.in28minutes.com.	A	192.0.2.235
static.in28minutes.com.	AAAA	2001:0db8:85a3:0:0:8a2e:0370:7334
dummy.in28minutes.com.	CNAME	www.example.com
in28minutes.com.	MX	10 mailserver.in28minutes.com
in28minutes.com.	NS	ns-1423.awsdns-49.org. ns-146.awsdns-18.com. ns-981.awsdns-58.net. ns-1997.awsdns-57.co.uk.
in28minutes.com.	SOA	ns-1423.awsdns-49.org. awsdns-hostmaster.amazo

Route 53 Specific Extension - Alias records

- Route traffic to selected AWS resources
  - Elastic Beanstalk environment
  - ELB load balancer
  - Amazon S3 bucket
  - CloudFront distribution
- Alias records can be created for
  - root(in28minutes.com) and
  - non root domains(api.in28minutes.com)
- COMPARED to CNAME records which can only be created for
  - non root domains (api.in28minutes.com)

Route 53 - Routing



- Route 53 can route across Regions
  - Create ALBs in multiple regions and route to them!
  - Offers multiple routing policies

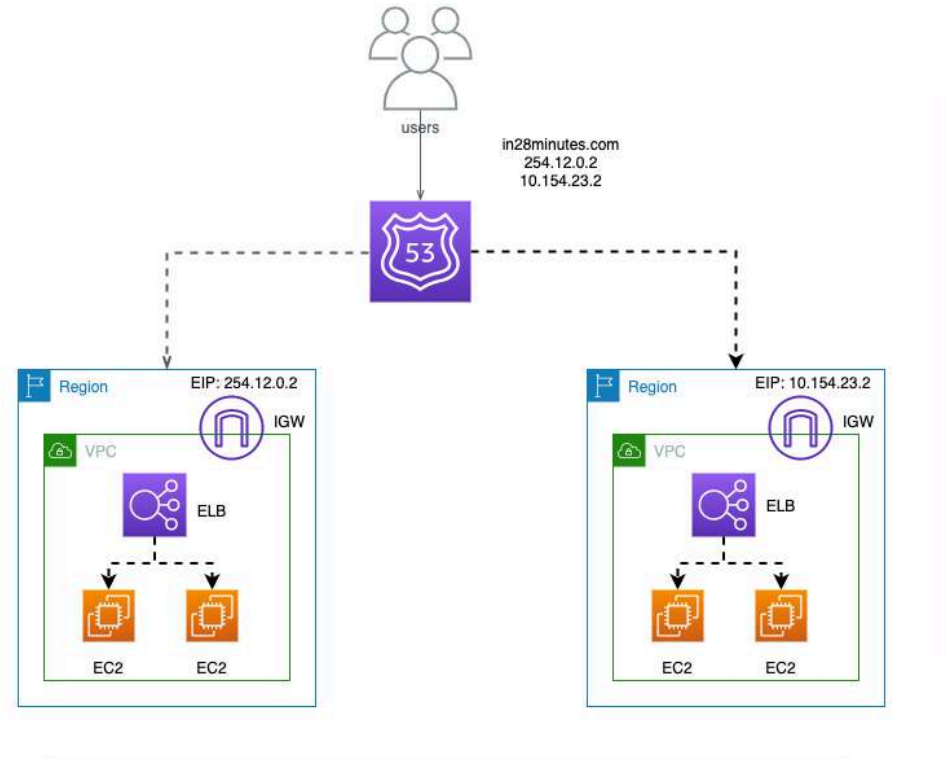
Route 53 Routing Policies

In 28 Minutes

Policy	Description
Simple	Maps a domain name to (one or more) IP Addresses
Weighted	Maps a single DNS name to multiple weighted resources 10% to A, 30% to B, 60% to C (useful for canary deployments)
Latency	Choose the option with minimum latency Latency between hosts on the internet can change over time
Failover	Active-passive failover. Primary Health check fails (optional cloud Watch alarm) => DR site is used
Geoproximity	Choose the nearest resource (geographic distance) to your user. Configure a bias.
Multivalue answer	Return multiple healthy records (upto 8) at random You can configure an (optional) health check against every record
Geolocation	Choose based on the location of the user

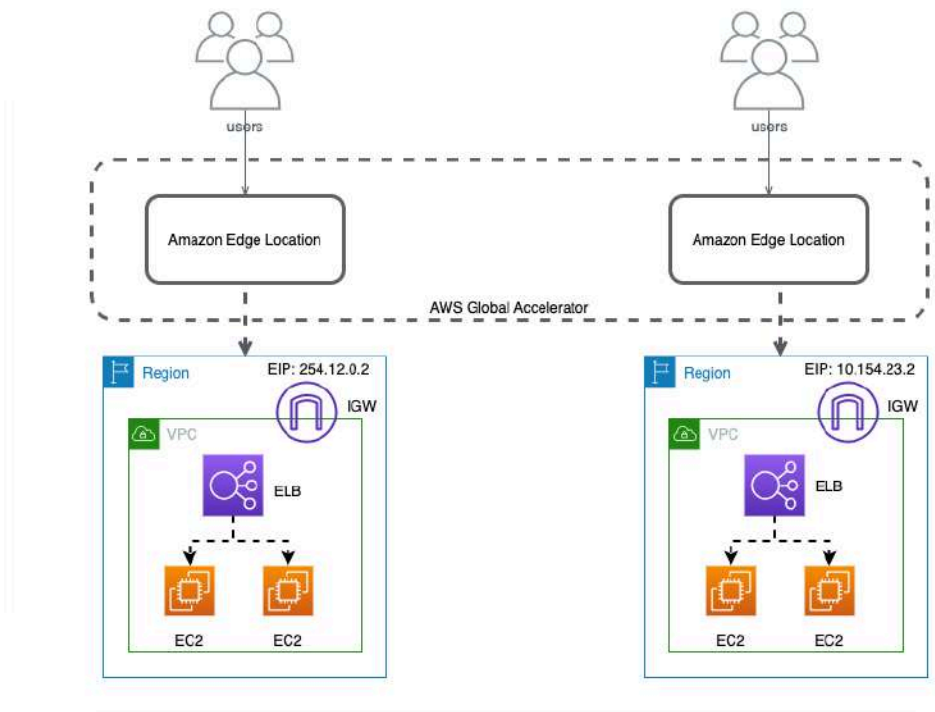
Need for AWS Global Accelerator

- Cached DNS answers
  - clients might cache DNS answers causing a delay in propagation of configuration updates
- High latency
  - users connect to the region over the internet



AWS Global Accelerator

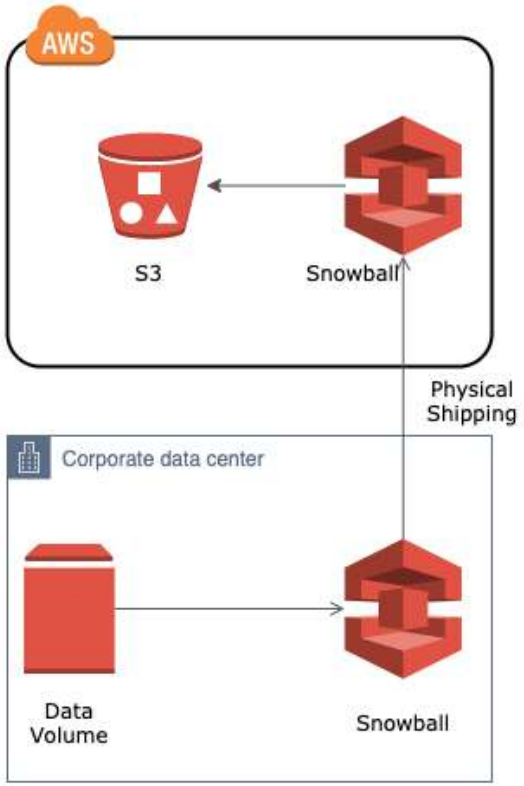
- Directs traffic to optimal endpoints over the AWS global network
- Global Accelerator provides you with two static IP addresses
- Static IP addresses are anycast from the AWS edge network
  - Distribute traffic across multiple endpoint resources in multiple AWS Regions
- Works with Network Load Balancers, Application Load Balancers, EC2 Instances, and Elastic IP addresses



Moving Data between AWS and On-premises

AWS Snowball

- Transfer dozens of terabytes to petabytes of data from on-premises to AWS
- 100TB (80 TB usable) per appliance
  - If needed, request multiple appliances
- Involves physical shipping
- Simple Process
  - Request for Snowball
  - Copy data
  - Ship it back
- Manage jobs with AWS Snowball console
- Data is automatically encrypted with KMS (AES-256)



AWS Snowball

- Current versions of AWS Snowball use Snowball Edge devices
  - Provide both compute and storage
  - Pre-process data (using Lambda functions)
- Choose between
  - Storage optimized (24 vCPUs, 32 GiB RAM)
  - Compute optimized(52 vCPUs, 208 GiB RAM)
  - Compute optimized with GPU
- Choose Snowball if direct transfer takes over a week
  - 5TB can be transferred on 100Mbps line in a week at 80% utilization

AWS Snowball

AWS Snowmobile

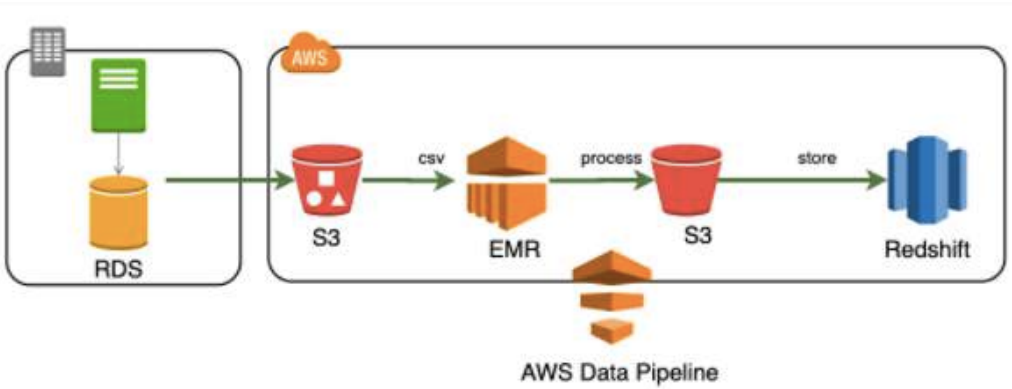
- How do I transfer dozens of petabytes to exabytes of data from on-premises to AWS for cloud migration?
- 100PB storage per truck
- If needed, use multiple trucks in parallel
- Data is automatically encrypted with KMS (AES-256)

AWS DataSync - Transfer File Storage to Cloud

In 28 Minutes

- Secure and 10x faster (100s of TB) data transfers from/to AWS over internet or AWS Direct Connect
- Transfer from onpremise file storage (NFS, SMB) to S3, EFS or FSx for Windows
- Monitor progress using Amazon CloudWatch
- (Use cases) Data Migration, Data replication and Cold data archival
- (Alternative) Use AWS Snowball if you are bandwidth constrained or transferring data from remote, or disconnected
- (Alternative) Use S3 Transfer Acceleration when your applications are integrated with S3 API. If not, prefer AWS DataSync(Supports multiple destinations, built-in retry)
- (Integration) Migrate data using DataSync and use AWS Storage Gateway for ongoing updates from on-premises applications

AWS Data Pipeline



- Process and move data (ETL) between S3, RDS, DynamoDB, EMR, On-premise data sources
- Create complex data processing workloads that are fault tolerant, repeatable, and highly available
- Launches required resources and tear them down after execution
- REMEMBER : NOT for streaming data!

AWS Database Migration Service

- Migrate databases to AWS while keeping source databases operational
  - Homogeneous Migrations (ex: Oracle to Oracle)
  - Heterogeneous Migrations (ex: Oracle to Amazon Aurora, MySQL to Amazon Aurora)
- Free for first 6 months when migrating to Aurora, Redshift or DynamoDB
- (AFTER MIGRATION) Keep databases in sync and pick right moment to switch
- (Use case) Consolidate multiple databases into a single target database
- (Use case) Continuous Data Replication can be used for Disaster Recovery

AWS Schema Conversion Tool

- Migrate data from commercial databases and data warehouses to open source or AWS services
  - Preferred option for migrating data warehouse data to Amazon Redshift
- Migrate database schema (views, stored procedures, and functions) to compatible targets
- Features:
  - SCT assessment report
    - Analyze a database to determine the conversion complexity
  - Update source code (update embedded SQL in code)
  - Fan-in (multiple sources - single target)
  - Fan-out (single source - multiple targets)

Database

Database Migration Service VS Schema Conversion Tool

- (Remember) SCT is part of DMS service
- DMS is preferred for homogeneous migrations
- SCT is preferred when schema conversion are involved
- DMS is for smaller workloads (less than 10 TB)
- SCT preferred for large data warehouse workloads
  - Prefer SCT for migrations to Amazon Redshift
- Only DMS provides continuous data replication after migration

Cloud Migration Approaches

In 28 Minutes

- You have a **combination of following options**:
  - **Rehosting** ("lift and shift")
  - **Replatforming**
    - Few adjustments to suit the cloud
    - Example: Containerizing
  - **Repurchasing**
    - Move to a new, cloud-native product
      - Move to a new database
  - **Refactoring**
    - Example: Serverless Computing
    - Most expensive
  - **Retiring**
    - End of service
  - **Retaining**
    - Do NOT move to Cloud
    - Stay on-premises

Cloud Migration - AWS Managed Services

Service	Description
AWS Migration Hub	Single place to track application migrations across multiple AWS and partner solutions
AWS Application Discovery Service	Discover on-premises application inventory and dependencies
AWS Control Tower	Easiest way to set up and govern a secure multi-account AWS environment (creates your landing zone using AWS Organizations)
AWS Application Migration Service (MGN)	Lift and shift applications to AWS. Earlier version: AWS Server Migration Service (AWS SMS)
AWS Mainframe Modernization	Migrate Mainframe Applications to AWS

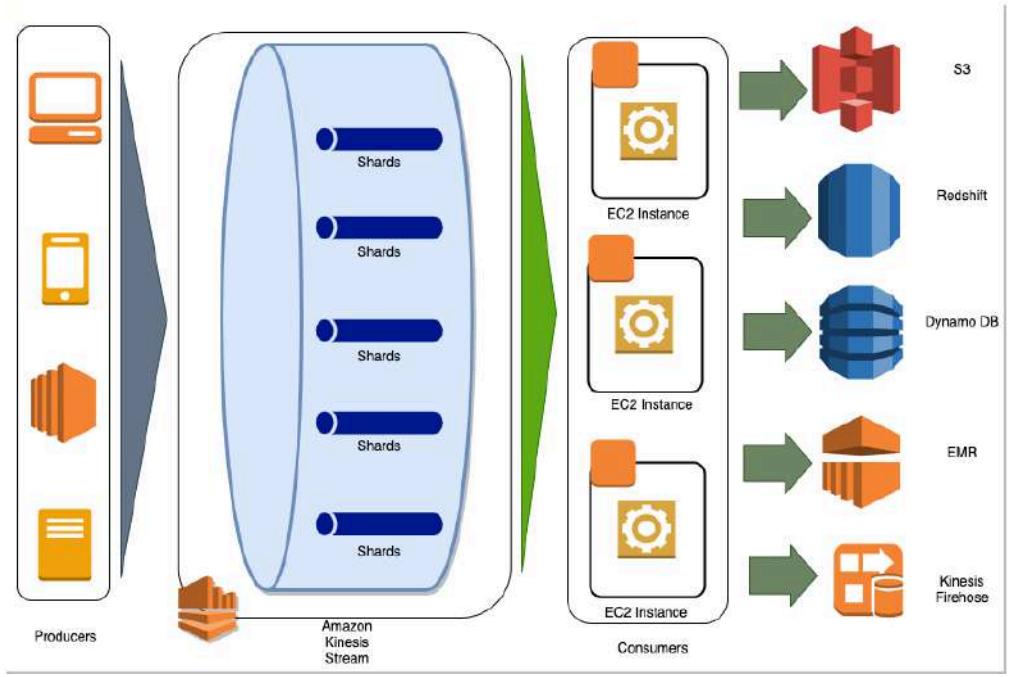
Amazon Kinesis

Amazon Kinesis

- Handle streaming data
  - NOT recommended for ETL Batch Jobs
- Amazon Kinesis Data Streams
  - Process Data Streams
- Amazon Kinesis Firehose
  - Data ingestion for streaming data : S3, Elasticsearch etc
- Amazon Kinesis Analytics
  - Run queries against streaming data
- Amazon Kinesis Video Streams
  - Monitor video streams

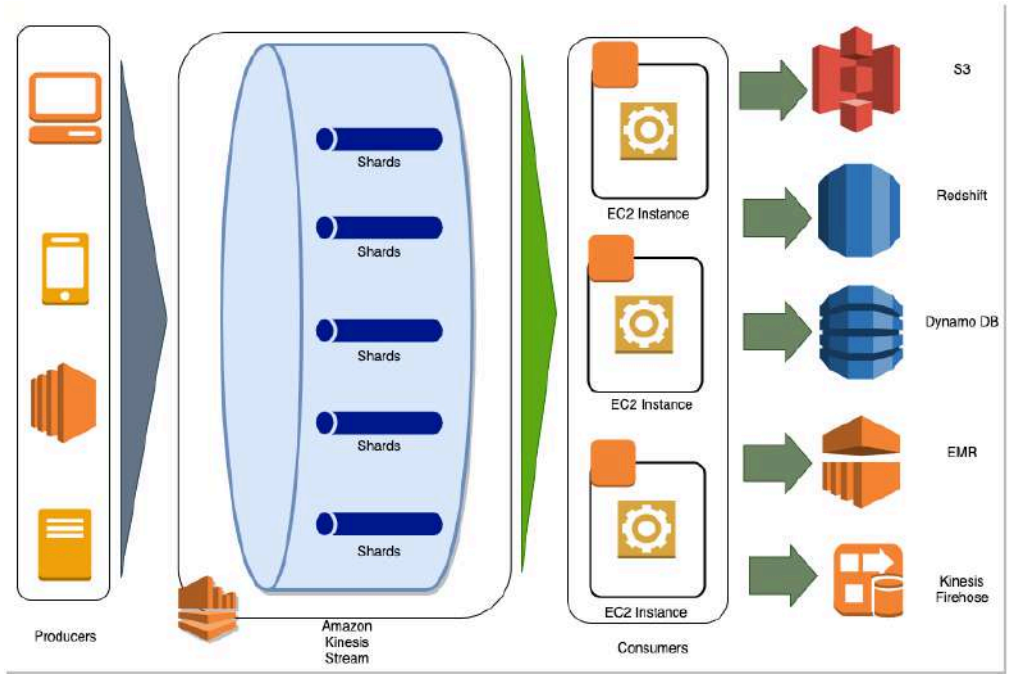
Amazon Kinesis Data Streams

- Limitless Real time stream processing
  - Sub second processing latency
- Alternative for Kafka
- Supports multiple clients
  - Each client can track their stream position
- Retain and replay data (max 7 days & default 1 day)



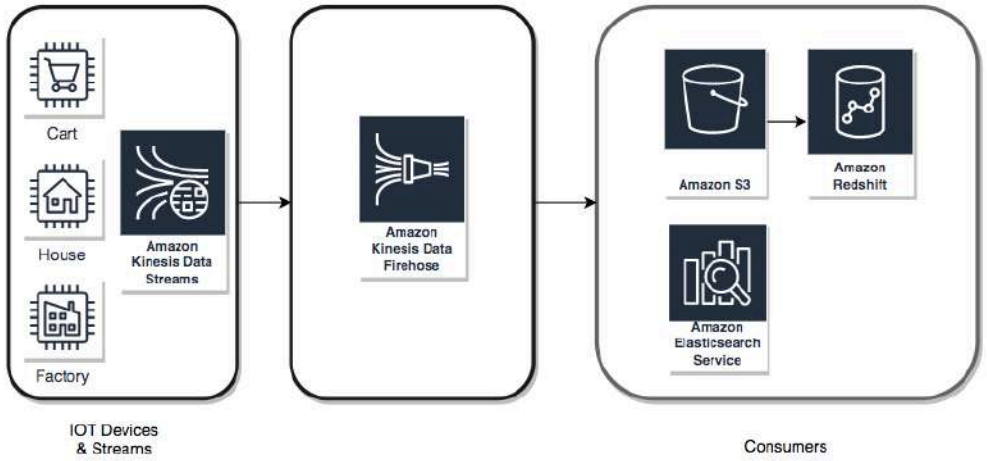
Amazon Kinesis Data Streams - Integrations

- Use application integrations to generate streams
  - Toolkits : AWS SDK, AWS Mobile SDK, Kinesis Agent
  - Service Integrations : AWS IOT, CloudWatch Events and Logs
- Process streams using Kinesis Stream Applications
  - Run on EC2 instances
  - Written using Kinesis Data Streams APIs

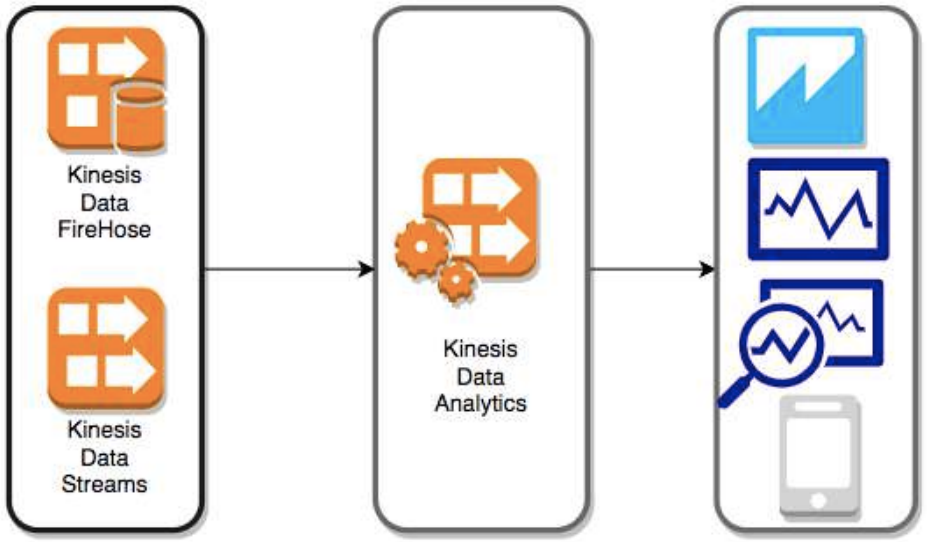


Amazon Kinesis Data Firehose

- Data ingestion for streaming data
  - Receive
  - Process (transform - Lambda, compress, encrypt)
  - Store stream data to S3, Elasticsearch, Redshift and Splunk
- Use existing analytics tools based on S3, Redshift and Elasticsearch
- Pay for volume of data ingested (Serverless)

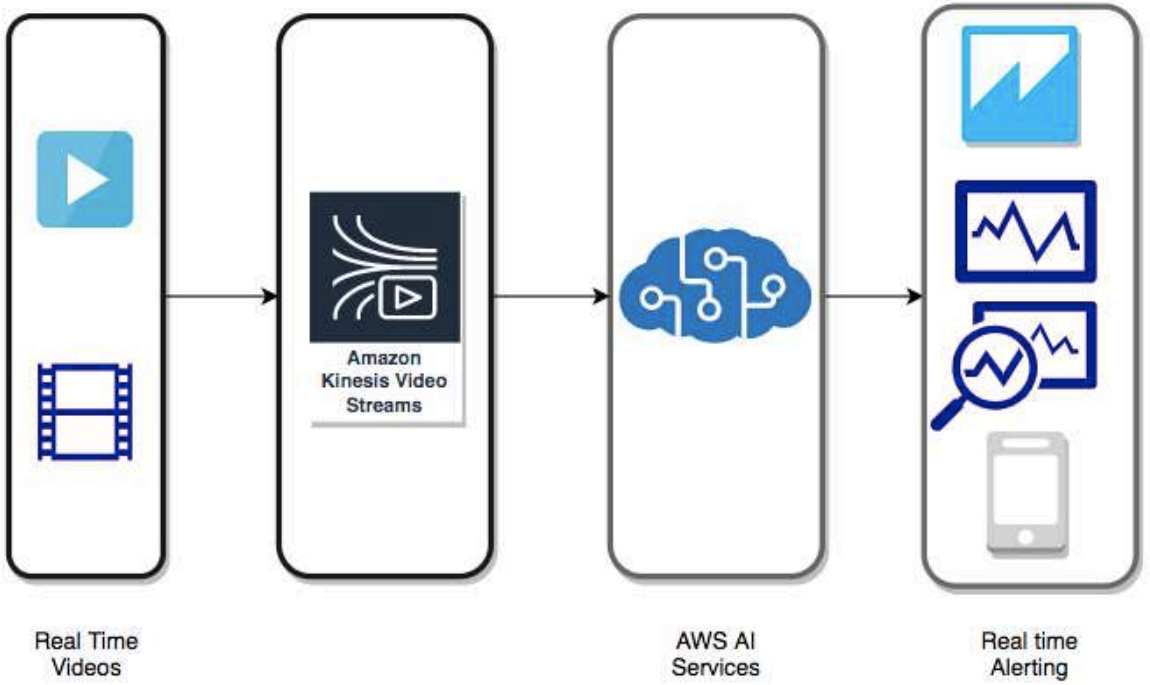


Amazon Kinesis Analytics



- You want to continuously find active number of users on a website in the last 5 minutes based on streaming website data
- With Amazon Kinesis Analytics, you can write SQL queries and build Java applications to continuously analyze your streaming data

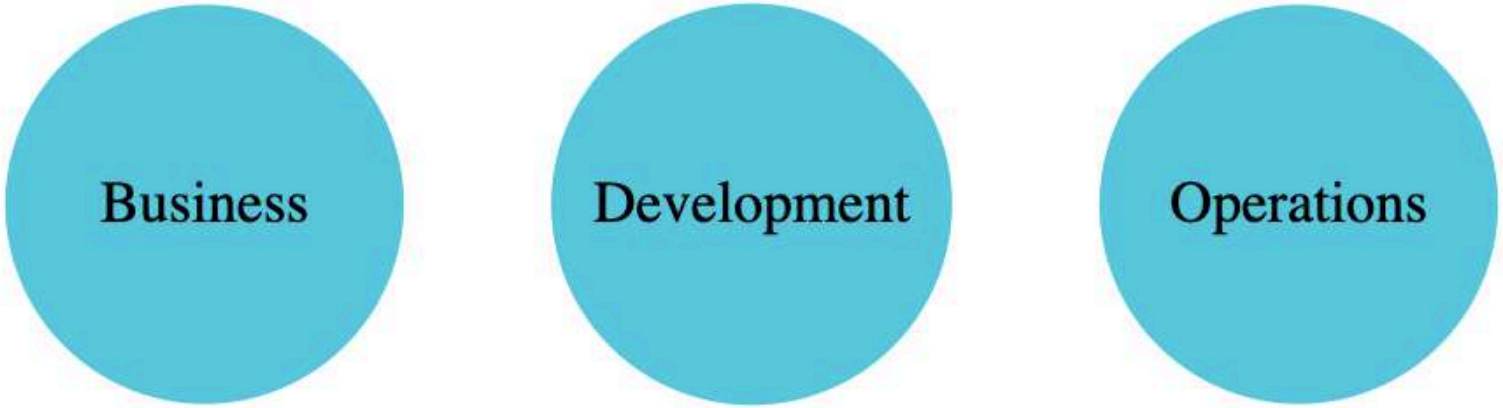
Amazon Kinesis Video Streams



- Monitor video streams from web-cams
- Examples: traffic lights, shopping malls, homes etc
- Integrate with machine learning frameworks to get intelligence

DevOps

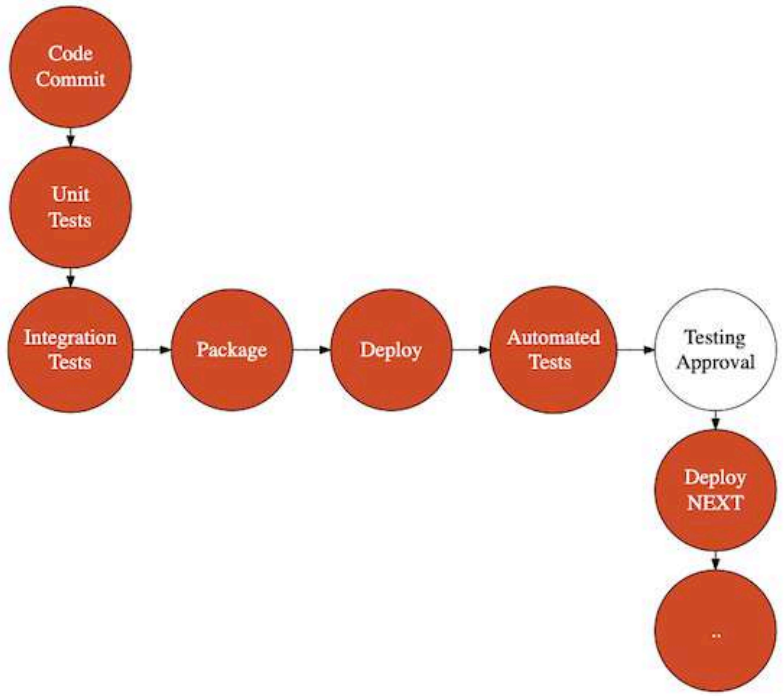
DevOps



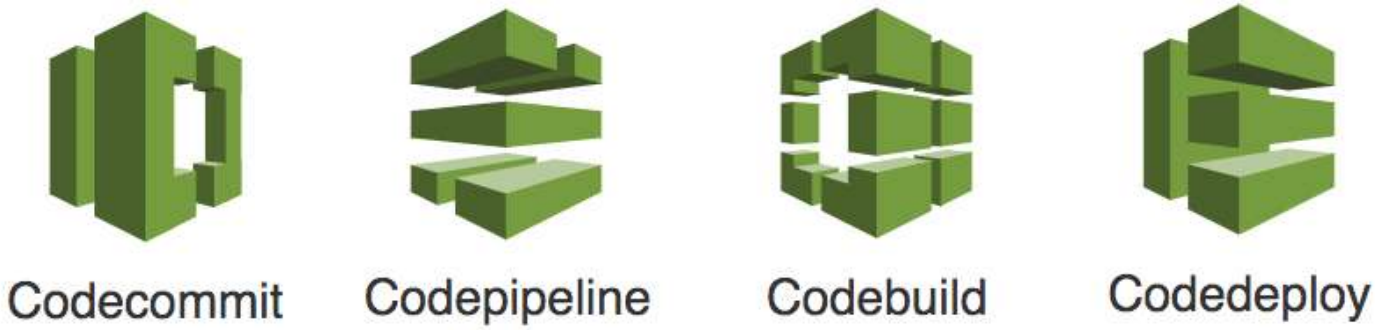
- Getting Better at "**Three Elements of Great Software Teams**"
  - Communication - Get teams together
  - Feedback - Earlier you find a problem, easier it is to fix
  - Automation - Automate testing, infrastructure provisioning, deployment, and monitoring

DevOps - CI, CD

- Continuous Integration
  - Continuously run your tests and packaging
- Continuous Deployment
  - Continuously deploy to test environments
- Continuous Delivery
  - Continuously deploy to production



DevOps - CI, CD Tools



- AWS CodeCommit - Private source control (Git)
- AWS CodePipeline - Orchestrate CI/CD pipelines
- AWS CodeBuild - Build and Test Code (application packages and containers)
- AWS CodeDeploy - Automate Deployment (EC2, ECS, Elastic Beanstalk, EKS, Lambda etc)

DevOps - IAAC

- Treat infrastructure the same way as application code
- Track your infrastructure changes over time (version control)
- Bring repeatability into your infrastructure
- Two Key Parts
  - Infrastructure Provisioning
    - Provisioning compute, database, storage and networking
    - Open source cloud neutral - Terraform
    - AWS Service - CloudFormation
  - Configuration Management
    - Install right software and tools on the provisioned resources
    - Open Source Tools - Chef, Puppet, Ansible

AWS CloudFormation - Introduction

In28 Minutes

- Lets consider an example:
  - I would want to create a new VPC and a subnet
  - I want to provision a ELB, ASG with 5 EC2 instances and an RDS database in the subnet
  - I would want to setup the right security groups
- AND I would want to create 4 environments
  - Dev, QA, Stage and Production!
- CloudFormation can help you do all these with a simple (actually NOT so simple) script!

CloudFormation

AWS CloudFormation - Advantages

In 28 Minutes

- Automate deployment and modification of AWS resources in a controlled, predictable way
- Avoid configuration drift
- Avoid mistakes with manual configuration
- Think of it as version control for your environments

CloudFormation

AWS CloudFormation

CloudFormation

- All configuration is defined in a simple text file - JSON or YAML
  - I want a VPC, a subnet, a database and ...
- CloudFormation understands dependencies
  - Creates VPCs first, then subnets and then the database
- (Default) Automatic rollbacks on errors (Easier to retry)
  - If creation of database fails, it would automatic delete the subnet and VPC
- Version control your configuration file and make changes to it over time
- Free to use - Pay only for the resources provisioned
  - Get an automated estimate for your configuration

AWS CloudFormation - Example 1 - JSON

In 28 Minutes

```
{
    "Resources" : {
        "MyBucket" : {
            "Type" : "AWS::S3::Bucket"
            "Properties" : {
               "AccessControl" : "PublicRead"
            }
        }
    }
}
```

AWS CloudFormation - Example 2 - YAML

In 28 Minutes

```yaml
Resources:
  MyBucket:
    Type: AWS::S3::Bucket
    Properties:
      AccessControl: PublicRead
```

AWS CloudFormation - Example 3

In28 Minutes

```
Resources:
  Ec2Instance:
    Type: 'AWS::EC2::Instance'
    Properties:
      ImageId: "ami-0ff8a91507f77f867"
      InstanceType: t2.micro
      SecurityGroups:
        - !Ref InstanceSecurityGroup
  InstanceSecurityGroup:
    Type: 'AWS::EC2::SecurityGroup'
    Properties:
      GroupDescription: Enable SSH access via port 22
      SecurityGroupIngress:
        - IpProtocol: tcp
          FromPort: '22'
          ToPort: '22'
          CidrIp: 0.0.0.0/0
```

AWS CloudFormation - Terminology

In28 Minutes

- Template
  - A CloudFormation JSON or YAML defining multiple resources
- Stack
  - A group of resources that are created from a CloudFormation template
  - In the earlier example, the stack contains an EC2 instance and a security group
- Change Sets
  - To make changes to stack, update the template
  - Change set shows what would change if you execute
  - Allows you to verify the changes and then execute

AWS CloudFormation - Important template elements

In 28 Minutes

```
{
  "AWSTemplateFormatVersion" : "version date",
  "Description" : "JSON string",
  "Metadata" : {},
  "Parameters" : {},
  "Mappings" : {},
  "Resources" : {},
  "Outputs" : {}
}
```

- Resources - What do you want to create?
  - One and only mandatory element
- Parameters - Values to pass to your template at runtime
  - Which EC2 instance to create? - ("t2.micro", "m1.small", "m1.large")
- Mappings - Key value pairs
  - Example: Configure different values for different regions
- Outputs - Return values from execution
  - See them on console and use in automation

AWS CloudFormation - Mappings Example

In 28 Minutes

```
"Mappings" : {
  "RegionMap" : {
    "us-east-1"      : { "AMI" : "AMI-A"},
    "us-west-1"      : { "AMI" : "ami-B"},
    "eu-west-1"      : { "AMI" : "ami-C"},
    "ap-southeast-1" : { "AMI" : "ami-D"},
    "ap-northeast-1" : { "AMI" : "ami-E"}
  }
}
```

AWS CloudFormation - Remember

- Deleting a stack deletes all the associated resources
  - EXCEPT for resources with DeletionPolicy attribute set to "Retain"
  - You can enable termination protection for the entire stack
- Templates are stored in S3
- Use CloudFormation Designer to visually design templates
- AWS CloudFormation StackSets
  - Create, update, or delete stacks across multiple accounts and regions with a single operation

CloudFormation

CloudFormation vs AWS Elastic Beanstalk

- (Do you know?) You can create an Elastic Beanstalk environment using CloudFormation!
- Think of Elastic Beanstalk as a pre-packaged CloudFormation template with a User Interface
  - You choose what you want
  - (Background) A Cloud Formation template is created and executed
  - The environment is ready!

AWS CDK: Define Cloud Infrastructure Using Code

In 28 Minutes

- **AWS CDK (Cloud Development Kit)**: Provision AWS resources with familiar programming languages
  - **Code as infrastructure**: Use TypeScript, Python, Java, or .NET to define cloud resources
  - **Uses CloudFormation**: Translates your code into AWS CloudFormation templates for reliable and repeatable deployments
  - **Streamline development**: Simplify the creation of complex, multi-component AWS applications with modular, reusable components
  - **Automate deployment**: Integrate with AWS deployment pipelines for CI/CD

Serverless Application Model

- 1000s of Lambda functions to manage, versioning, deployment etc
- Serverless projects can become a maintenance headache
- How to test serverless projects with Lambda, API Gateway and DynamoDB in your local?
- How to ensure that your serverless projects are adhering to best practices?
  - Tracing (X-Ray), CI/CD(CodeBuild, CodeDeploy, CodePipeline) etc
- Welcome SAM - Serverless Application Model
  - Open source framework for building serverless applications
  - Define a YAML with all the serverless resources you want:
    - Functions, APIs, Databases etc
  - BEHIND THE SCENES : Your configuration is used to create a AWS CloudFormation syntax to deploy your application

AWS OpsWorks - Configuration Management

In 28 Minutes

- OpsWorks is used for Configuration Management
  - How do you ensure that 100 servers have the same configuration?
  - How can I make a change across 100 servers?
- Managed service based on Chef & Puppet
- One service for deployment and operations in cloud and on-premise environments
- Configuration - Chef recipes or cookbooks, Puppet manifests
- All metrics are sent to Amazon CloudWatch
- (IMPORTANT) All configuration management tools can also do infrastructure provisioning
  - However, I would recommend NOT doing that as they are not good at infrastructure provisioning

AWS Opsworks

AWS CloudShell: Command Line at Your Fingertips

In 28 Minutes

- **AWS CloudShell**: Instant Command Line I/F
- **Browser-based access**: No setup required, use the AWS CLI directly from your browser
- **Pre-authenticated**: Automatically logs in with your console credentials for immediate access to your resources
- **Built-in tools**: Comes with pre-installed AWS CLI and other useful software to manage your resources
- **No extra cost**: Available at no additional charge, you pay only for the AWS resources you manage with CloudShell

Management Services in AWS

AWS Organizations

Organizations

- Organizations typically have multiple AWS accounts
  - Different business units
  - Different environments
- How do you centralize your management (billing, access control, compliance and security) across multiple AWS accounts?
- Welcome AWS Organizations!
- Organize accounts into Organizational Units (OU)
- Provides API to automate creation of new accounts

AWS Organizations - Features

- One consolidated bill for all AWS accounts
- Centralized compliance management for AWS Config Rules
- Send AWS CloudTrail data to one S3 bucket (across accounts)
- AWS Firewall Manager to manage firewall rules (across accounts)
  - AWS WAF, AWS Shield Advanced protections and Security Groups
- Use Service control policies (SCPs) to define restrictions for actions (across accounts):
  - Prevent users from disabling AWS Config or changing its rules
  - Require Amazon EC2 instances to use a specific type
  - Require MFA to stop an Amazon EC2 instance
  - Require a tag upon resource creation

Organizations

AWS Trusted Advisor

- Recommendations for cost optimization, performance, security and fault tolerance
  - Red - Action recommended Yellow - investigate and Green - Good to go
- All AWS customers get 4 checks for free:
  - Service limits (usage > 80%)
  - Security groups having unrestricted access (0.0.0.0/0)
  - Proper use of IAM
  - MFA on Root Account
- Business or Enterprise AWS support plan provides over 50 checks
  - Disable those you are not interested in
  - How much will you save by using Reserved Instances?
  - How does your resource utilization look like? Are you right sized?

Trusted Advisor

AWS Trusted Advisor Recommendations

In 28 Minutes

- Cost Optimization
  - Highlight unused resources
  - Opportunities to reduce your costs
- Security
  - Settings that can make your AWS solution more secure
- Fault Tolerance
  - Increase resiliency of your AWS solution
  - Redundancy improvements, over-utilized resources
- Performance
  - Improve speed and responsiveness of your AWS solutions
- Service Limits
  - Identify if your service usage is more than 80% of service limits

Trusted Advisor

AWS Service Quotas

In 28 Minutes

- AWS account has Region-specific default quotas or limits for each service
  - You don't need to remember all of them :)
- Service Quotas allows you to manage your quotas for over 100 AWS services, from one location

AWS Directory Service

- Provide AWS access to on-premise users without IAM users
- Managed service deployed across multiple AZs
- Option 1 : AWS Directory Service for Microsoft AD
  - More than 5000 Users
  - Trust relationship needed between AWS and on-premise directory
- Option 2 : Simple AD
  - Less than 5000 users
  - Powered by Samba4 and compatible with Microsoft AD
  - Does not support trust relationships with other AD domains
- Option 3 : AD Connector
  - Use your existing on-premise directory with AWS cloud services
  - Your users use existing credentials to access AWS resources

Directory Service

Billing and Cost Management Services/Tools

In 28 Minutes

- **AWS Billing and Cost Management** - Pay your AWS bill, monitor your usage, and analyze and control your costs
  - Cost Explorer - View your AWS cost data as a graph. Filter by Region, AZ, tags etc. See future cost projection.
  - AWS Budgets - Create a budget. Create Amazon SNS notifications to alert when you go over (or projected to go over) budget.
- **AWS Compute Optimizer** - Recommends optimal AWS Compute resources to reduce costs (Example: Right-sizing - EC2 instance type and EC2 Auto Scaling group configuration)
- **AWS Pricing Calculator** (NEW) - Estimate cost of your architecture solution
- **AWS Simple Monthly Calculator** (OLD) - Estimate charges for AWS services
- **Total Cost of Ownership (TCO) Calculator** (OLD) - Compare Cost of running applications in AWS vs On Premise

Other Management Services

In28 Minutes

- **AWS Marketplace**
  - Digital catalog to find, test, buy, and deploy licensed software solutions using flexible pricing options: Bring Your Own License (BYOL), free trial, pay-as-you-go, hourly, monthly etc.
- **Resource Groups**
  - Group your AWS resources.
  - Automate Tasks using AWS Systems Manager.
  - Get group related insights from AWS Config and CloudTrail.
- **AWS Systems Manager**
  - Run commands(operational tasks) on Amazon EC2 instances.
  - Manage your OS patches.
- **Personal Health Dashboard**
  - Personalized alerts when AWS is experiencing events that may impact you
  - Provides troubleshooting guidance

Getting Help for Your AWS Implementations

In 28 Minutes

Service	Example Use Case	Explanation
AWS Activate for Startups	A newly founded tech startup looking for cloud credits, training, and technical support to kickstart their AWS journey	A program designed to provide startups with the resources they need to get started on AWS. Offers promotional credits, training, technical support, and other benefits.
AWS IQ	A business wanting to find, hire, and work directly with AWS-certified third-party experts for project consultation and implementation	A platform that connects AWS customers with certified service providers and AWS experts. Simplifies the process of finding, hiring, and collaborating with AWS experts.
AWS Managed Services (AMS)	A large enterprise looking to migrate a significant portion of its IT operations to AWS & needs comprehensive operational support	Operates AWS on behalf of customers. Helps in infrastructure operations, such as patch management, and monitoring. Allows businesses to focus on their apps.

Serverless Architecture

REST API Challenges

- Most applications today are built around REST API
- Management of REST API is not easy:
  - You've to take care of authentication and authorization
  - You've to be able to set limits (rate limiting, quotas) for your API consumers
  - You've to take care of implementing multiple versions of your API
  - You would want to monitor your API calls
  - You would want to be able to cache API requests

Amazon API Gateway

- How about a **fully managed service** with auto scaling that can act as a **"front door"** to your APIs?
- Welcome **"Amazon API Gateway"**
  - **"publish, maintain, monitor, and secure APIs at any scale"**
  - Integrates with AWS Lambda, Amazon EC2, Amazon ECS or any web application
  - Supports HTTP(S) and WebSockets (two way communication - chat apps and streaming dashboards)
  - Serverless. **Pay for use** (API calls and connection duration)

Amazon API Gateway - Remember

- Run multiple versions of the same API
- Rate Limits(request quota limits), throttling and fine-grained access permissions using API Keys for Third-Party Developers
- Implement Authorization with:
  - AWS IAM
  - Amazon Cognito
  - Custom Lambda Authorizer

Amazon Cognito

In 28 Minutes

- Want to quickly add a sign-up page and authentication for your mobile and web apps?
- Want to integrate with web identity providers (example: Google, Facebook, Amazon) and provide a social sign-in?
- Do you want security features such as multi-factor authentication (MFA), phone and email verification?
- Want to create your own user database without worrying about scaling or operations?
- Let's go : Amazon Cognito
- Support for SAML

Amazon Cognito

Amazon Cognito - User Pools

In28 Minutes

- Do you want to create your own secure and scalable user directory?
- Do you want to create sign-up pages?
- Do you want a built-in, customizable web UI to sign in users (with option to social sign-in)?
- Create a user pool

Amazon Cognito

Amazon Cognito - Identity pools

- Identity pools provide AWS credentials to grant your users access to other AWS services
- Connect identity pools with authentication (identity) providers
  - Your own user pool OR
  - Amazon, Apple, Facebook, Google+, Twitter OR
  - OpenID Connect provider OR
  - SAML identity providers (SAML 2.0)
- Configure multiple authentication (identity) providers for each identity pool
- Federated Identity
  - An external authentication (identity) provider
  - ex: Amazon, Apple, Facebook, OpenID or SAML identity providers

AWS Step Functions

In 28 Minutes

- Create a serverless workflow in 10 Minutes using a visual approach
- Orchestrate multiple AWS services into serverless workflows:
  - Invoke an AWS Lambda function
  - Run an Amazon Elastic Container Service or AWS Fargate task
  - Get an existing item from an Amazon DynamoDB table or put a new item into a DynamoDB table
  - Publish a message to an Amazon SNS topic
  - Send a message to an Amazon SQS queue
- Build workflows as a series of steps:
  - Output of one step flows as input into next step
  - Retry a step multiple times until it succeeds
  - Maximum duration of 1 year

AWS Step Functions

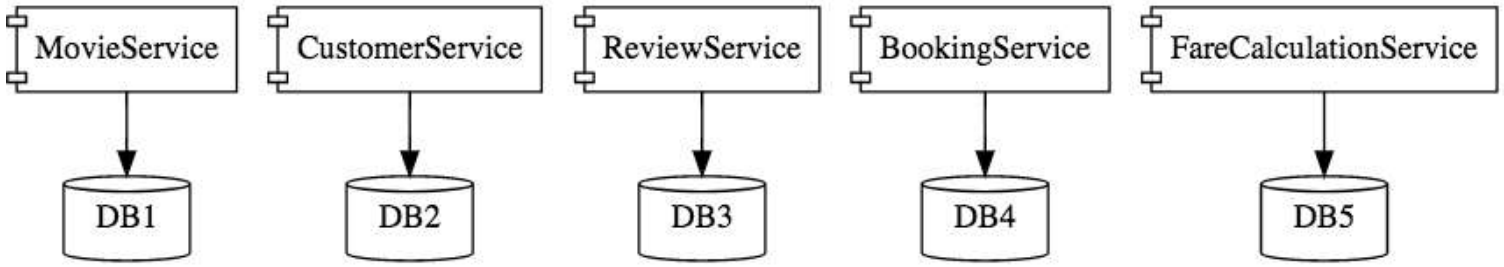
In 28 Minutes

- Integrates with Amazon API Gateway
  - Expose API around Step Functions
  - Include human approvals into workflows
- (Use case) Long-running workflows
  - Machine learning model training, report generation, and IT automation
- (Use case) Short duration workflows
  - IoT data ingestion, and streaming data processing
- (Benefits) Visual workflows with easy updates and less code
- (Alternative) Amazon Simple Workflow Service (SWF)
  - Complex orchestration code (external signals, launch child processes)
- Step Functions is recommended for all new workflows UNLESS you need to write complex code for orchestration

In28 Minutes

Containers and Container Orchestration

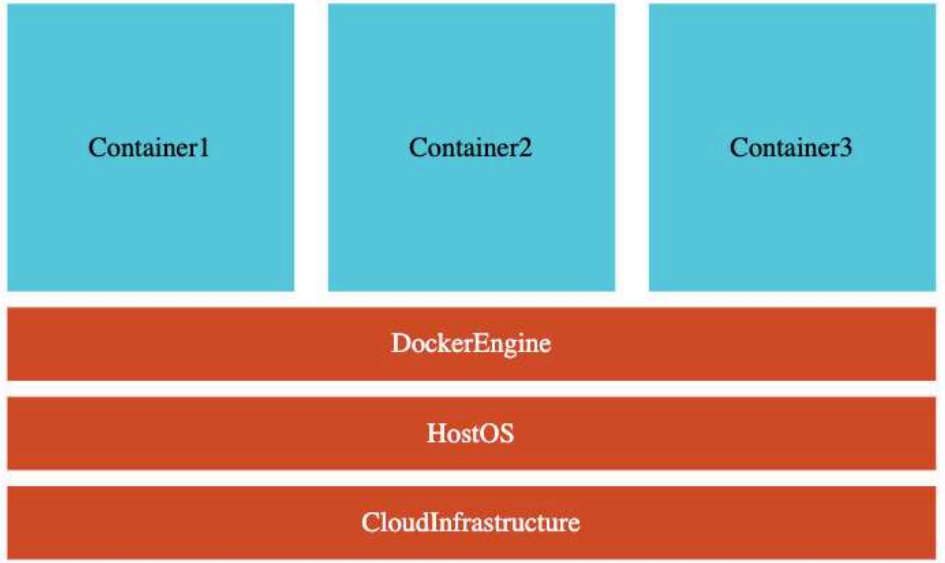
Microservices



- Enterprises are heading towards microservices architectures
- Build small focused microservices
- **Flexibility to innovate** and build applications in different programming languages (Go, Java, Python, JavaScript, etc)
- BUT **deployments become complex**!
- How can we have **one way of deploying** Go, Java, Python or JavaScript .. microservices?
  - Enter **containers**!

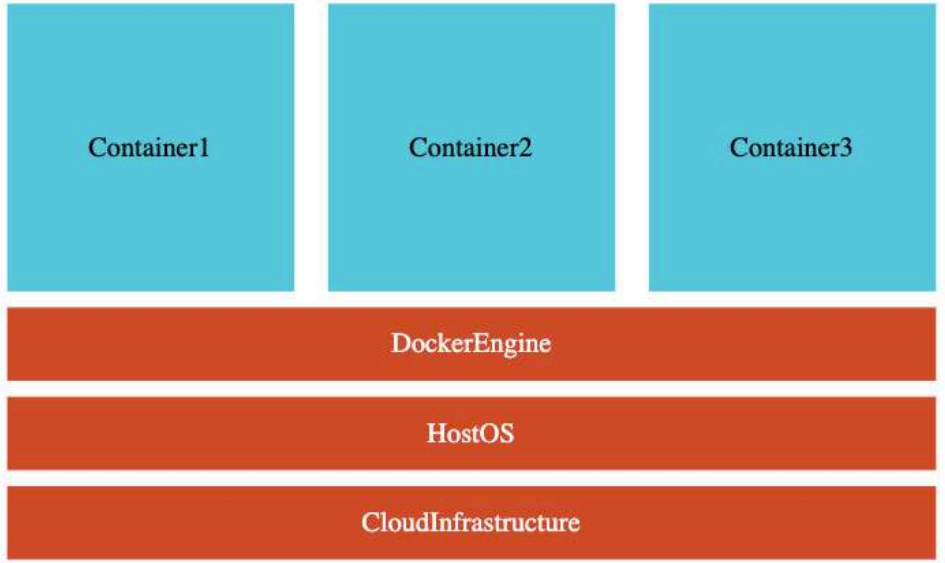
Docker

- Create **Docker images** for each microservice
- Docker image **contains everything a microservice needs** to run:
  - Application Runtime (JDK or Python or NodeJS)
  - Application code
  - Dependencies
- You can run these docker containers **the same way** on any infrastructure
  - Your local machine
  - Corporate data center
  - Cloud



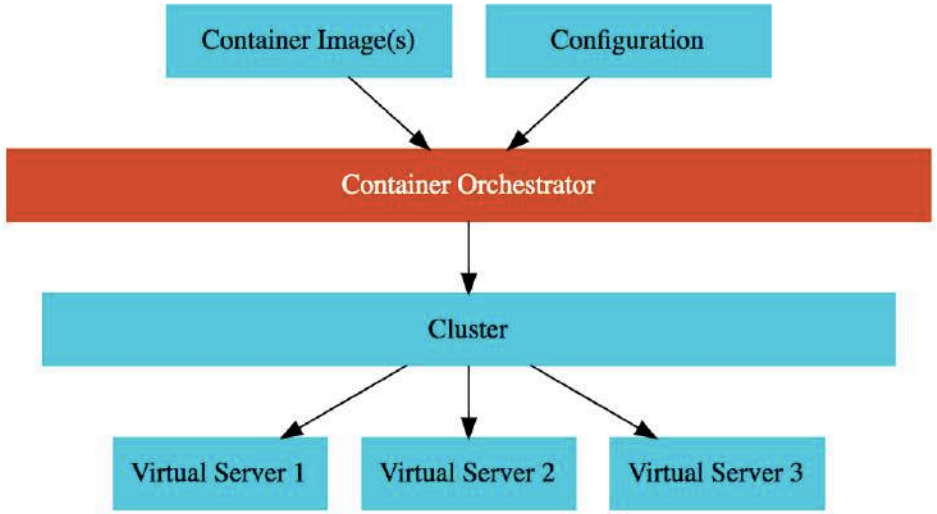
Docker - Advantages

- Docker containers are **light weight** (compared to Virtual Machines)
- Docker provides **isolation** for containers
- Docker is **cloud neutral**
- (NEW CHALLENGE) How do you manage 1000's of containers belonging to multiple microservices?
  - Enter **Container Orchestration**!



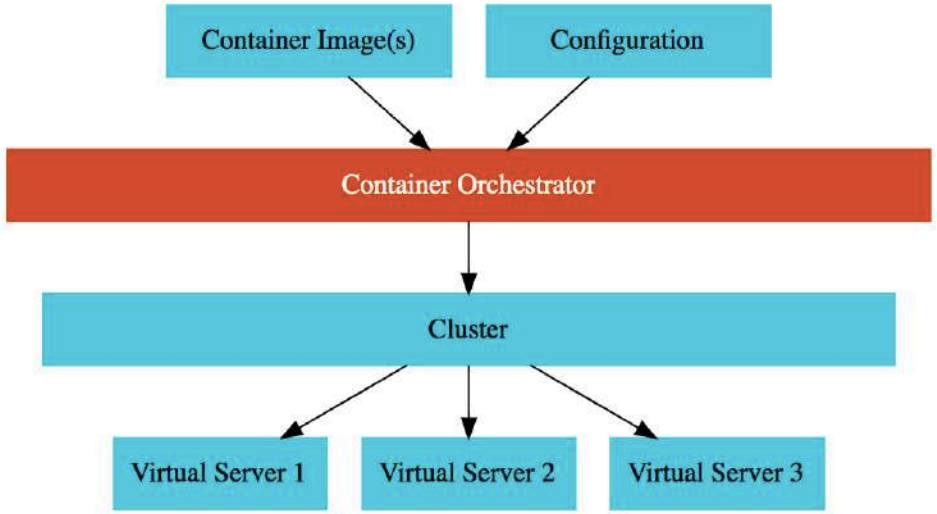
Container Orchestration

- **Requirement** : I want 10 instances of Microservice A container, 15 instances of Microservice B container and ....
- Typical Features:
  - **Auto Scaling** - Scale containers based on demand
  - **Service Discovery** - Help microservices find one another
  - **Load Balancer** - Distribute load among multiple instances of a microservice
  - **Self Healing** - Do health checks and replace failing instances
  - **Zero Downtime Deployments** - Release new versions without downtime



Container Orchestration Options

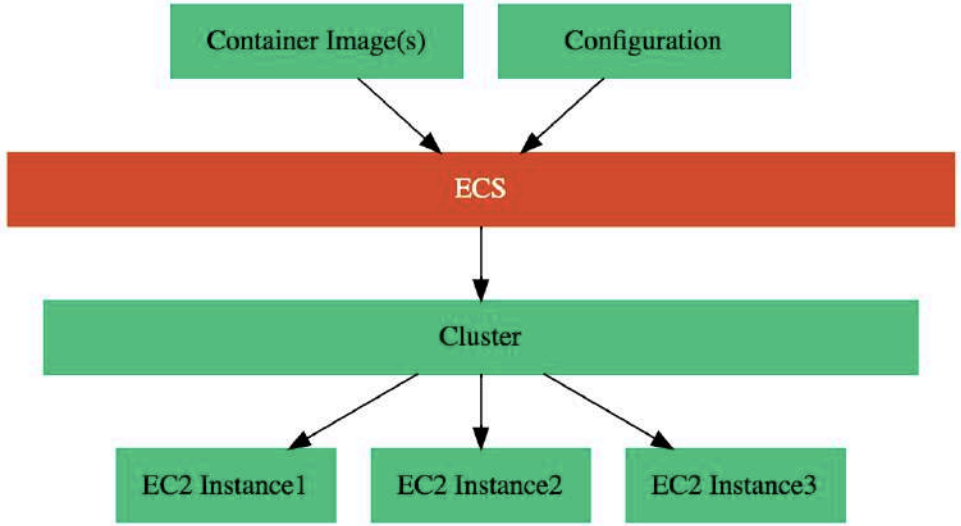
- **Cloud Neutral**
  - Kubernetes
  - AWS service - AWS Elastic Kubernetes Service (EKS)
  - EKS does not have a free tier
- **AWS Specific**
  - AWS Elastic Container Service (ECS)
  - AWS Fargate : Serverless version of AWS ECS
  - AWS Fargate does not have a free tier



Amazon Elastic Container Service (Amazon ECS)

In 28 Minutes

- Fully managed service for container orchestration
- Serverless option - **AWS Fargate**
- Use cases:
  - Microservices Architectures - Create containers for your microservices and orchestrate them using ECS or Fargate
  - Batch Processing. Run batch workloads on ECS using AWS Batch
- DEMO



Architecture and Best Practices

Well Architected Framework

In28 Minutes

- Helps cloud architects build application infrastructure which is:
  - Secure
  - High-performing
  - Resilient and
  - Efficient
- Five Pillars
  - Operational Excellence
  - Security
  - Reliability
  - Performance Efficiency
  - Cost Optimization

Operational Excellence

- Avoid/Minimize effort and problems with
  - Provisioning servers
  - Deployment
  - Monitoring
  - Support

Operational Excellence - Solutions and AWS services

In 28 Minutes

- Use Managed Services
  - You do not need to worry about managing servers, availability, durability etc
- Go serverless
  - Prefer Lambda to EC2!
- Automate with Cloud Formation
  - Use Infrastructure As Code
- Implement CI/CD to find problems early
  - CodePipeline
  - CodeBuild
  - CodeDeploy
- Perform frequent, small reversible changes

AWS Lambda

CloudFormation

Codepipeline

Codebuild

Codedeploy

Operational Excellence - Solutions and AWS services

In 28 Minutes

- Prepare: for failure
  - Game days
  - Disaster recovery exercises
  - Implement standards with AWS Config rules
- Operate: Gather Data and Metrics
  - CloudWatch (Logs agent), Config, Config Rules, CloudTrail, VPC Flow Logs and X-Ray (tracing)
- Evolve: Get intelligence
  - Use Amazon Elasticsearch to analyze your logs


AWS Config


Cloudwatch


AWS CloudTrail


AWS X-Ray


Amazon ES

Security Pillar


AWS IAM


AWS Shield


AWS WAF


AWS KMS


Cloud HSM

- Principle of least privilege for least time
- Security in Depth - Apply security in all layers
- Protect Data in Transit and at rest
- Actively monitor for security issues
- Centralize security policies for multiple AWS accounts

Security Pillar - Principle of least privilege for least time

AWS IAM

- Use temporary credentials when possible (IAM roles, Instance profiles)
- Use IAM Groups to simplify IAM management
- Enforce strong password practices
- Enforce MFA
- Rotate credentials regularly

Security Pillar - Security in Depth

- VPCs and Private Subnets
  - Security Groups
  - Network Access Control List (NACL)
- Use hardened EC2 AMIs (golden image)
  - Automate patches for OS, Software etc
- Use CloudFront with AWS Shield for DDoS mitigation
- Use WAF with CloudFront and ALB
  - Protect web applications from CSS, SQL injection etc
- Use CloudFormation
  - Automate provisioning infrastructure that adheres to security policies


VPC


EC2 AMI


AWS Shield


AWS WAF


CloudFormation

Security Pillar - Protecting Data at Rest

- Enable Versioning (when available)
- Enable encryption - KMS and Cloud HSM
  - Rotate encryption keys
- Amazon S3
  - SSE-C, SSE-S3, SSE-KMS
- Amazon DynamoDB
  - Encryption Client, SSE-KMS
- Amazon Redshift
  - AWS KMS and AWS CloudHSM
- Amazon EBS, Amazon SQS and Amazon SNS
  - AWS KMS
- Amazon RDS
  - AWS KMS, TDE

AWS KMS

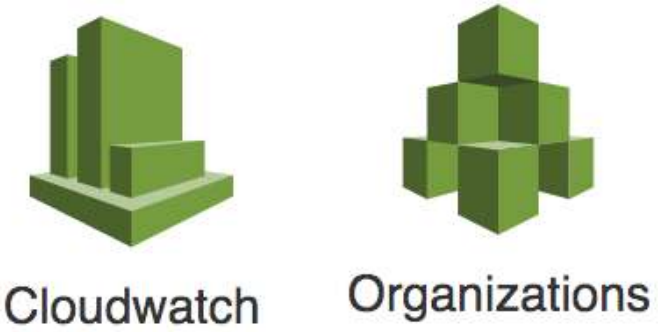
Cloud HSM

Security Pillar - Protecting Data in Transit

Certificate Manager

- Data coming in and going out of AWS
- By default, all AWS API use HTTPS/SSL
- You can also choose to perform client side encryption for additional security
- Ensure that your data goes through AWS network as much as possible
  - VPC Endpoints and AWS PrivateLink

Security Pillar - Detect Threats



- Actively monitor for security issues:
  - Monitor CloudWatch Logs
  - Use Amazon GuardDuty to detect threats and continuously monitor for malicious behavior
- Use AWS Organization to centralize security policies for multiple AWS accounts

Reliability

- Ability to
  - Recover from infrastructure and application issues
  - Adapt to changing demands in load

Reliability - Best Practices

AutoScaling

CW Alarm

Cloudwatch

- Automate recovery from failure
  - Health checks and Auto scaling
  - Managed services like RDS can automatically switch to standby
- Scale horizontally
  - Reduces impact of single failure
- Maintain Redundancy
  - Multiple Direct Connect connections
  - Multiple Regions and Availability Zones

Reliability - Best Practices

- Prefer serverless architectures
- Prefer loosely coupled architectures
  - SQS, SNS
- Distributed System Best Practices
  - Use Amazon API Gateway for throttling requests
  - AWS SDK provides retry with exponential backoff

Loosely coupled architectures

- ELB
  - Works in tandem with AWS auto scaling
- Amazon SQS
  - Polling mechanism
- Amazon SNS
  - Publish subscribe pattern
  - Bulk notifications and Mobile push support
- Amazon Kinesis
  - Handle event streams
  - Multiple clients
  - Each client can track their stream position

Troubleshooting on AWS - Quick Review

In 28 Minutes

Option	Details	When to Use
Amazon S3 Server Access Logs	S3 data request details - request type, the resources requested, and the date and time of request	Troubleshoot bucket access issues and data requests
Amazon ELB Access Logs	Client's IP address, latencies, and server responses	Analyze traffic patterns and troubleshoot network issues
Amazon VPC Flow Logs	Monitor network traffic	Troubleshoot network connectivity and security issues

Troubleshooting on AWS - Quick Review

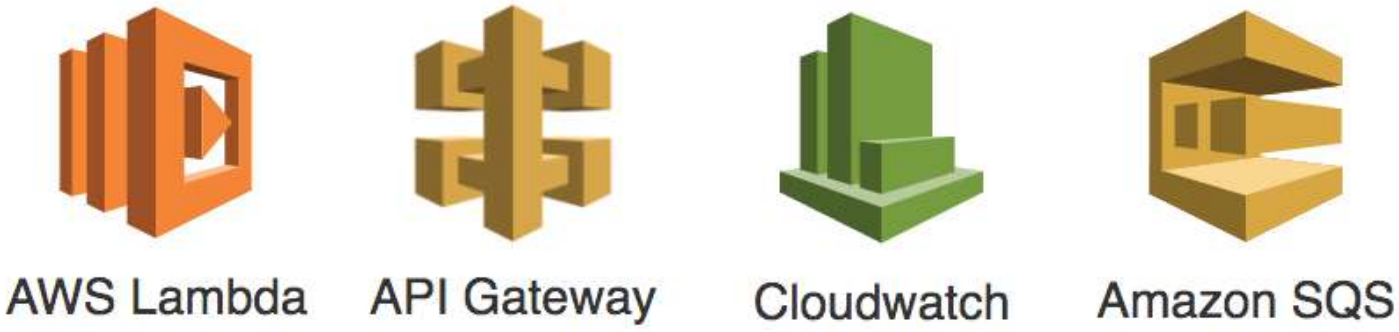
In 28 Minutes

Option	Details	When to Use
Amazon CloudWatch	Monitor metrics from AWS resources	Monitoring
Amazon CloudWatch Logs	Store and Analyze log data from Amazon EC2 instances and on-premises servers	Debugging application issues and Monitoring
AWS Config	AWS resource inventory. History. Rules.	Inventory and History
Amazon CloudTrail	History of AWS API calls made via AWS Management Console, AWS CLI, AWS SDKs etc.	Auditing and troubleshooting. Determine who did what, when, and from where.

Performance Efficiency

- Meet needs with minimum resources (efficiency)
- Continue being efficient as demand and technology evolves

Performance Efficiency - Best Practices



- Use Managed Services
  - Focus on your business instead of focusing on resource provisioning and management
- Go Serverless
  - Lower transactional costs and less operational burden
- Experiment
  - Cloud makes it easy to experiment
- Monitor Performance
  - Trigger CloudWatch alarms and perform actions through Amazon SQS and Lambda

Performance Efficiency - Choose the right solution

In 28 Minutes

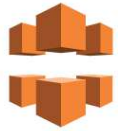
- Compute
  - EC2 instances vs Lambda vs Containers
- Storage
  - Block, File, Object
- Database
  - RDS vs DynamoDB vs RedShift ..
- Caching
  - ElastiCache vs CloudFront vs DAX vs Read Replicas
- Network
  - CloudFront, Global Accelerator, Route 53, Placement Groups, VPC endpoints, Direct Connect
- Use product specific features
  - Enhanced Networking, S3 Transfer Acceleration, EBS optimized instances


AWS Lambda


Amazon S3


DynamoDB


ElastiCache


CloudFront

Cost Optimization

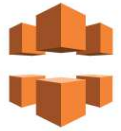
- Run systems at lowest cost

AutoScaling

AWS Lambda

Trusted Advisor

Cloudwatch



CloudFront

Cost Optimization - Best Practices

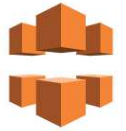
- Match supply and demand
  - Implement Auto Scaling
  - Stop Dev/Test resources when you don't need them
  - Go Serverless
- Track your expenditure
  - Cost Explorer to track and analyze your spend
  - AWS Budgets to trigger alerts
  - Use tags on resources


AutoScaling


AWS Lambda


Trusted Advisor


Cloudwatch


CloudFront

Cost Optimization - Choose Cost-Effective Solutions

In 28 Minutes

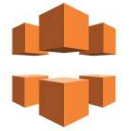
- Right-Sizing : Analyze 5 large servers vs 10 small servers
  - Use CloudWatch (monitoring) and Trusted Advisor (recommendations) to right size your resources
- Email server vs Managed email service (charged per email)
- On-Demand vs Reserved vs Spot instances
- Avoid expensive software : MySQL vs Aurora vs Oracle
- Optimize data transfer costs using AWS Direct Connect and Amazon CloudFront

AutoScaling

AWS Lambda

Trusted Advisor

Cloudwatch



CloudFront

[ADDITION] AWS Well-Architected Pillar - Sustainability

In 28 Minutes

- **Sustainability**: "meeting the needs of the present without compromising the ability of future generations to meet their own needs." (United Nations Definition)
  - **Various Reports**: Moving to cloud helps you reduce your carbon footprint upto 80%
  - **Design principles** for sustainability in the cloud
    - **1**: Understand your impact (AWS Customer Carbon Footprint Tool)
    - **2**: Establish sustainability goals
    - **3**: Maximize utilization
    - **4**: Adopt new, more efficient hardware & software offerings
      - Make your designs flexible
    - **5**: Use managed services
    - **6**: Reduce the downstream impact of your cloud workloads
      - Reduce need for device upgrades

Digital Transformation

What has changed in last decade or so?

In 28 Minutes

- How consumers make purchase decisions? (**Social**)
- How we do things? (**Mobile**)
- How much data we have? (**Big Data**)
  - How much intelligence we can get? (**AI/ML**)
- How much access startups have to technology at scale? (**Cloud**)

Enterprises have to adapt (or get disrupted)

- **Enterprises can ADAPT** by:
  - Providing awesome (omni-channel - social, mobile) customer experiences
  - Getting intelligence from data (Big Data, AI/ML)
    - Example: Personalize consumer offerings
  - Enabling themselves to make changes faster
    - Cultural change from "traditional Datacenter, SDLC, manual IT Ops" to "Cloud, Containers, DevOps/SRE, Automation"
- **Digital Transformation**: Using modern technologies to create (or modify) business processes & customer experiences by innovating with technology and team culture
  - Focus on WHY (NOT HOW)
    - Increase pace of change
    - Revenue Growth
    - Cost Savings
    - Higher customer engagement/retention

Cloud - Enabler for Digital Transformation

In 28 Minutes

- Cloud can **ENABLE** Digital Transformations
  - Lower cost
  - Reduced responsibilities
  - Higher capabilities
  - Increased speed to market
- **BUT needs a change** in skills, mindset and culture
  - Modern Architectures (Microservices, Serverless, Containers, Kubernetes)
  - More Agile Processes (DevOps, SRE)
  - Right Talent
  - Right Culture (of data driven experimentation and innovation)

Cloud Mindset

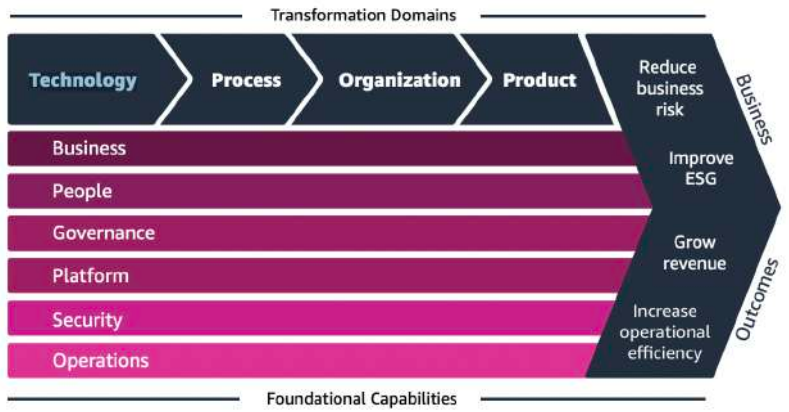
In28 Minutes

Factor	Data Center	Cloud
Infrastructure	Buy	Rent
Planning	Ahead of time	Provision when you need it
Deployment	VMs	PaaS or Containers or Serverless
Team	Specialized skills	T-shaped skills
Releases	Manual	CI/CD with flexible release options (Canary, A/B Testing, ...)
Infrastructure Creation	Manual	Infrastructure as Code
Attitude	Avoid Failures	Move Fast by Reducing Cost of Failure (Automation of testing, releases, infrastructure creation and monitoring)

Exploring AWS Cloud Adoption Framework

In 28 Minutes

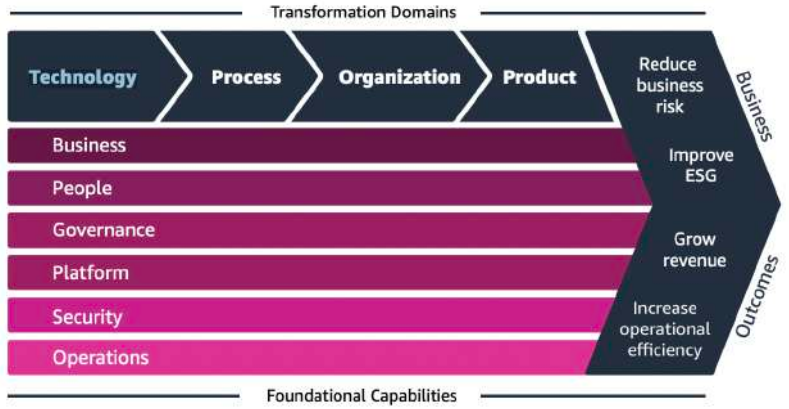
- You have decided to move to the cloud:
  - How do you ensure that **you are following the best practices** while adopting the cloud?
  - **Solution**: The AWS Cloud Adoption Framework (AWS CAF)
- Leverage **AWS experience and best practices**
- **Transformation Domains**:
  - **Technology**: Platform modernization
  - **Process**: Automate, and optimize your business operations (using AI/ML insights,...)
  - **Organization**: Restructure business & tech teams
  - **Product**: Re-imagine your business model



AWS Cloud Adoption Framework (AWS CAF) - An Overview

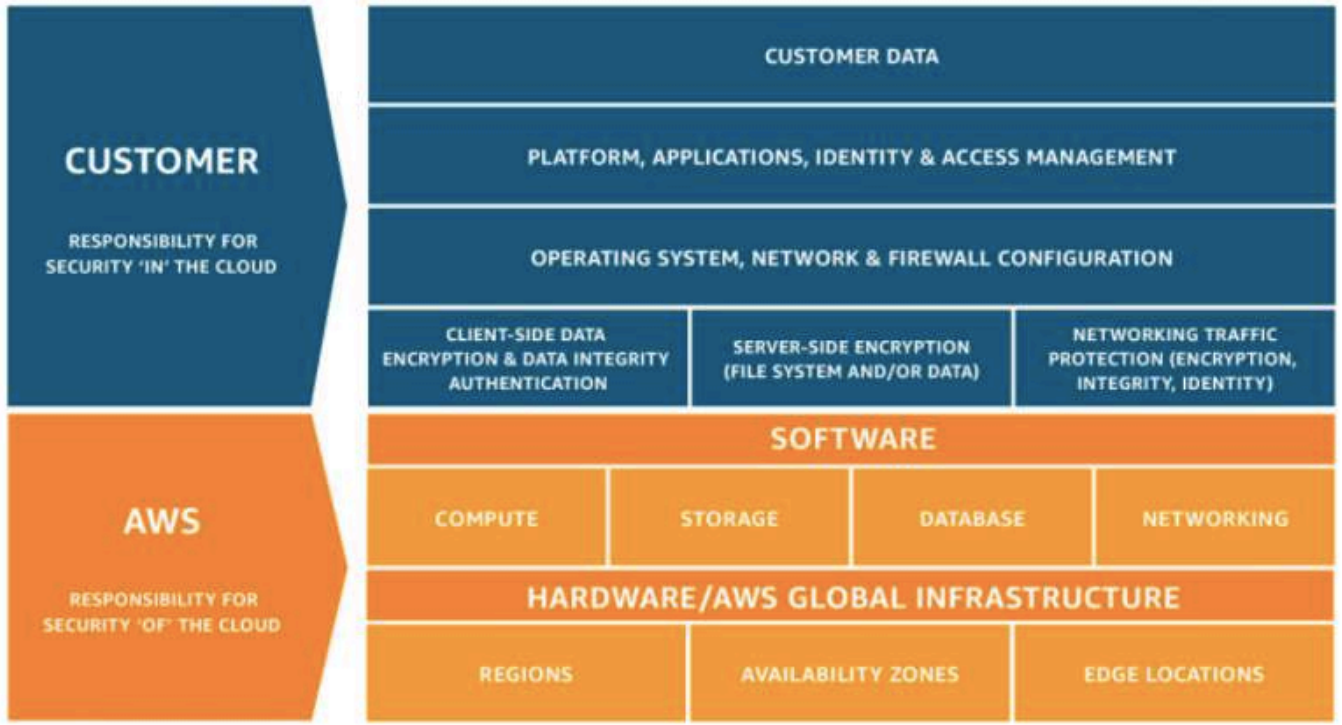
In 28 Minutes

- Capabilities grouped in **SIX perspectives**
  - **1: Business**: Strategy, Product Mgmt,..
  - **2: People**: Cloud Fluency, Culture Evolution, ..
  - **3: Governance**: Program/Project/Risk/Data Mgmt ..
  - **4: Platform**: Modern App. development (Container orchestration, Serverless), CI/CD/IaaC/Observability...
  - **5: Security**: IAM, Threat Detection, Incident Response..
  - **6: Operations**: Observability, Incident/Release Mgmt..
- **4 Key Phases**
  - **Envision phase**: Identify & prioritize opportunities
  - **Align phase**: Identify capability gaps
  - **Launch phase**: Deliver pilot initiatives
  - **Scale phase**: Expand & deliver business value



Shared Responsibility Model

Shared Responsibility Model



https://aws.amazon.com/compliance/shared-responsibility-model/

Security & Compliance is shared responsibility between AWS and customer

Shared Responsibility Model

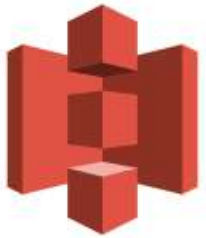
- AWS manages security of the cloud
  - AWS operates, manages and controls components from the Host OS and virtualization layer down to the physical security.
- YOU are responsible for security in the cloud
  - Customer assumes responsibility and management of
    - Guest operating system (including updates and security patches)
    - Application software
    - Configuration of Security Groups
    - Choosing and Integrating AWS Services with their IT environments

Shared Responsibility Model - Amazon EC2

- Amazon EC2 instances is Infrastructure as a Service (IaaS)
- You are responsible for
  - Guest OS (incl. security patches)
  - Application software installed
  - Configuring Security Groups (or firewalls)
- AWS is responsible for infrastructure layer only

Shared Responsibility Model - Managed Services

- Amazon S3 & DynamoDB are managed services
- AWS manages infrastructure layer, OS, and platform
- You are responsible for
  - Managing your data
  - Managing security of data at rest(encryption)
  - Managing security of data in transit
    - Mandating SSL/HTTPS
    - Using the right network - AWS global network or dedicated private network when possible
  - Managing access to the service
    - Configure right permissions (IAM users/roles/user policies/resource policies)
    - (FOR AWS RDS) Managing in database users
    - Configuring the right security groups (control inbound and outbound traffic)
    - Disabling external access (public vs private)



Amazon S3

DynamoDB

Shared Responsibility Model - Compliance

In28 Minutes

- Compliance responsibilities will be shared
- AWS ensures adherence of IT Infrastructure with IT security standards
  - SOC 1/ISAE 3402, SOC 2, SOC 3
  - FISMA, DIACAP, and FedRAMP
  - PCI DSS Level 1
  - ISO 9001, ISO 27001, ISO 27017, ISO 27018
- AWS provides a wide range of information on its IT control environment (white papers, reports, certifications etc)
- Customers can use the AWS control and compliance documentation to perform their control evaluation and verification procedures as required.

Shared Responsibility Model - IT controls

In28 Minutes

- Inherited Controls (Customer fully inherits from AWS)
  - Physical and Environmental controls
- Shared Controls (Controls shared by AWS and Customer)
  - Patch Management
    - AWS - Infrastructure Patches
    - Customer - Guest OS Patches and Application Software Patches
  - Configuration Management
    - AWS - Infrastructure (AWS)
    - Customer - Guest operating systems, databases, and applications
  - Awareness & Training
- Customer Owned Controls
  - Controls based on the applications deployed to AWS
  - Data Security Requirements

More AWS Services

One Enterprise - Many AWS accounts - AWS Control Tower

In 28 Minutes

- Having **multiple AWS accounts** helps businesses meet their business, governance, security, and operational requirements:
  - 1: Group workloads based on their business purpose
  - 2: Create environment specific security controls
  - 3: Make cost management easier
    - Each business unit responsible for their costs
  - **AWS Organizations**: Centralized Governance of multiple AWS accounts
    - Ex: Billing, Config Rules, Service control policies (SCPs), Firewall Manager rules, CloudTrail data
- **AWS Control Tower**: Easiest way to set up and govern a secure, multi-account AWS environment (also called a landing zone)
  - Sets up AWS Organizations
  - Ensures that all new accounts adhere to organization policies
  - Configure guardrails (AWS Config Rules, Service control policies (SCPs) etc) and check compliance

More Security Services in AWS

In28 Minutes

Service	Description
Certificate Manager	Provision, manage, deploy, and renew SSL/TLS certificates on the AWS platform
AWS Firewall Manager	Central management of firewall rules across multiple AWS accounts 1: AWS WAF rules 2: AWS Shield Advanced protections 3: Security groups and NACLs
AWS Artifact	No cost, self-service portal for on-demand access to AWS's compliance reports (Service Organization Control (SOC) reports, Payment Card Industry (PCI) reports...)
AWS Audit Manager	Enterprise infrastructure undergoes regular auditing(SOC, PCI DSS, GDPR...) Reduce Manual effort in generating auditor-friendly reports

More Security Services in AWS - 2

In28 Minutes

Service	Description
Amazon GuardDuty	Continuously monitor AWS environment for suspicious activity (Intelligent Threat Detection) Analyze AWS CloudTrail events, VPC Flow Logs etc
Amazon Inspector	Scan AWS workloads - EC2, Containers - for software vulnerabilities and unintended network exposure with a single click
AWS Security Hub	Single dashboard for comprehensive view of your security state within AWS Consolidated findings from enabled security services - Amazon GuardDuty, Amazon Inspector, Amazon Macie, ..
Amazon Detective	Analyze, and quickly identify the root cause of potential security issues Collects log data from your AWS resources Uses machine learning enabling you to perform more efficient security investigations.

AWS Transit Gateway

- How to connect multiple Amazon VPCs?
  - VPC Peering is NOT transitive => Complex Setup
- How can you connect multiple Amazon VPCs with on-premises networks (Direct Connect and VPN)?
  - Setting up individual connections might be difficult to maintain
  - **AWS Transit Gateway**: Connect multiple VPCS with VPN and DC
    - Supports Global inter-Region peering
    - Traffic between an Amazon VPC and AWS Transit Gateway remains on the AWS global private network
    - Create Route Table and associate Amazon VPCs and VPNs
  - **Remember**: No overlapping CIDRs
  - **Pricing**: No of connections to Transit Gateway (per hour) +
    - Amount of traffic thru Transit Gateway +
    - Inter-Region data transfer charges

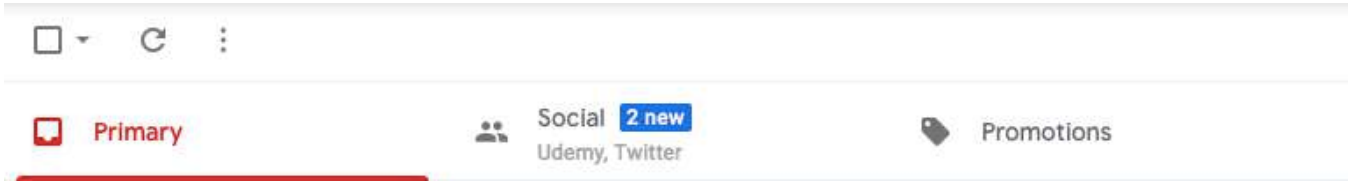
Hybrid Cloud - Compute

- Many enterprises are going **hybrid-cloud** and **multi-cloud**
- How about **running AWS services** on **customer-managed infrastructure**?
  - **Amazon ECS Anywhere**: Run ECS applications in on-premises environments and the cloud with centralized management
  - **Amazon EKS Anywhere**: Create and operate Kubernetes clusters on customer-managed infrastructure
    - Makes use of **Amazon EKS Distro**:
      - Amazon EKS customized Kubernetes distribution
      - Provides extended support even after community support expires
  - **Advantages**:
    - Meet your compliance needs
    - Use familiar ECS and EKS tooling
    - Intermediate step before workloads can be shifted to cloud

Databases - A Quick Review

Service	Description
RDS	Managed Relational Database Service
DynamoDB	Managed NoSQL Database
Amazon DocumentDB	Fully-managed MongoDB-compatible database service
ElastiCache	In-Memory Cache
Amazon Keyspaces	Serverless Cassandra-compatible Column-family database
Neptune	Fast, reliable graph database built for the cloud
Amazon Timestream	Fast, scalable, and serverless time series database for IoT and operational applications
Amazon Quantum Ledger Database (Amazon QLDB)	Fully managed ledger database Immutable, cryptographically verifiable log of data changes Usecases: Financial transactions, System-of-record applications

Artificial Intelligence - All around you



- Self-driving cars
- Spam Filters
- Email Classification
- Fraud Detection

What is AI? (Oxford Dictionary)

In 28 Minutes

The theory and development of computer systems able to perform tasks normally requiring human intelligence, such as visual perception, speech recognition, decision-making, and translation between languages

Understanding Types of AI

- **Strong artificial intelligence (or general AI)**: Intelligence of machine = Intelligence of human
  - A machine that can solve problems, learn, and plan for the future
  - An expert at everything (including learning to play all sports and games!)
  - Learns like a child, building on it's own experiences
  - We are far away from achieving this! (Estimates: few decades to never)
- **Narrow AI (or weak AI)**: Focuses on specific task
  - Examples: Self-driving cars and virtual assistants
  - **Machine learning**: Learn from data (examples)

Tags:

Water 100% confidence Sky 100% confidence
Lake 95% confidence Outdoor 95% confidence
Skyscraper 89% confidence Reflection 61% confidence
Overlooking 33% confidence Day 12% confidence

Description:

a city skyline with water 27% confidence

Racy Content: False 75% confidence

Adult Content: False 78% confidence

Exploring Machine Learning vs Traditional Programming

In 28 Minutes

- **Traditional Programming**: Based on Rules
  - IF this DO that
  - Example: Predict price of a home
    - Design an algorithm taking all factors into consideration:
      - Location, Home size, Age, Condition, Market, Economy etc
- **Machine Learning**: Learning from Examples (NOT Rules)
  - Give millions of examples
  - Create a Model
  - Use the model to make predictions!
- **Challenges**:
  - No of examples needed
  - Availability of skilled personnel
  - Complexity in implementing MLOps

Home size (Square Yds)	Age	Condition (1-10)	Price $$$
300	10	5	XYZ
200	15	9	ABC
250	1	10	DEF
150	2	34	GHI

Machine Learning - 3 Approaches

- **Use Pre-Trained Models**
  - Get intelligence from text, images, audio, video
  - Amazon Comprehend, Amazon Rekognition, ...
- **Build simple models**: Without needing data scientists
  - Limited/no-code experience
  - Example: Amazon SageMaker Auto ML
- **Build complex models**: Using data scientists and team
  - Build Your Own ML Models from ZERO (code-experienced)
  - Example: Amazon SageMaker

Tags:

Water 100% confidence Sky 100% confidence
Lake 95% confidence Outdoor 95% confidence
Skyscraper 89% confidence Reflection 61% confidence
Overlooking 33% confidence Day 12% confidence

Description:

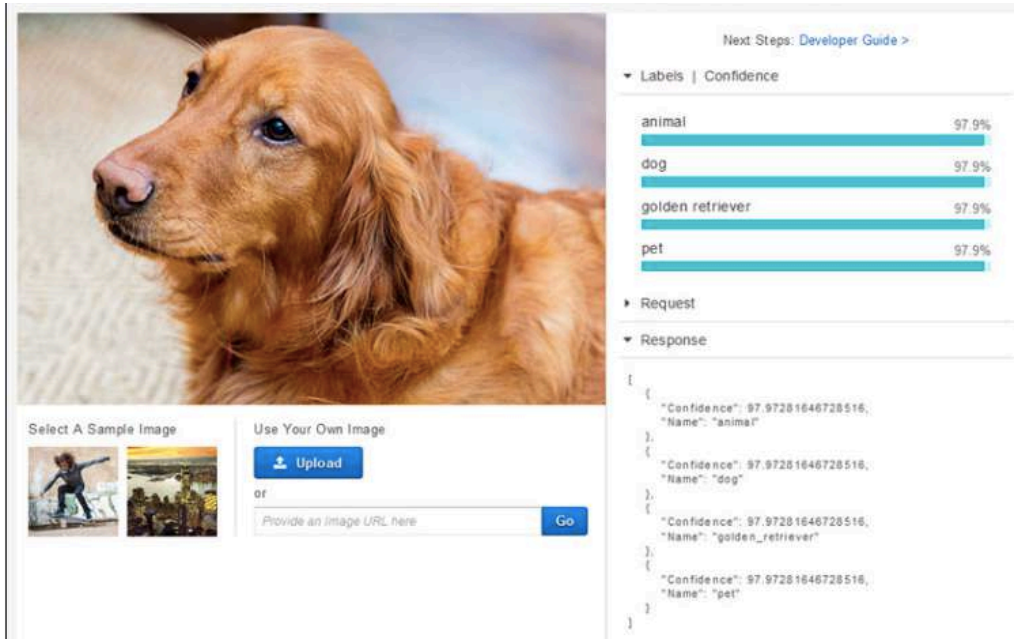
a city skyline with water 27% confidence

Racy Content: False 75% confidence

Adult Content: False 78% confidence

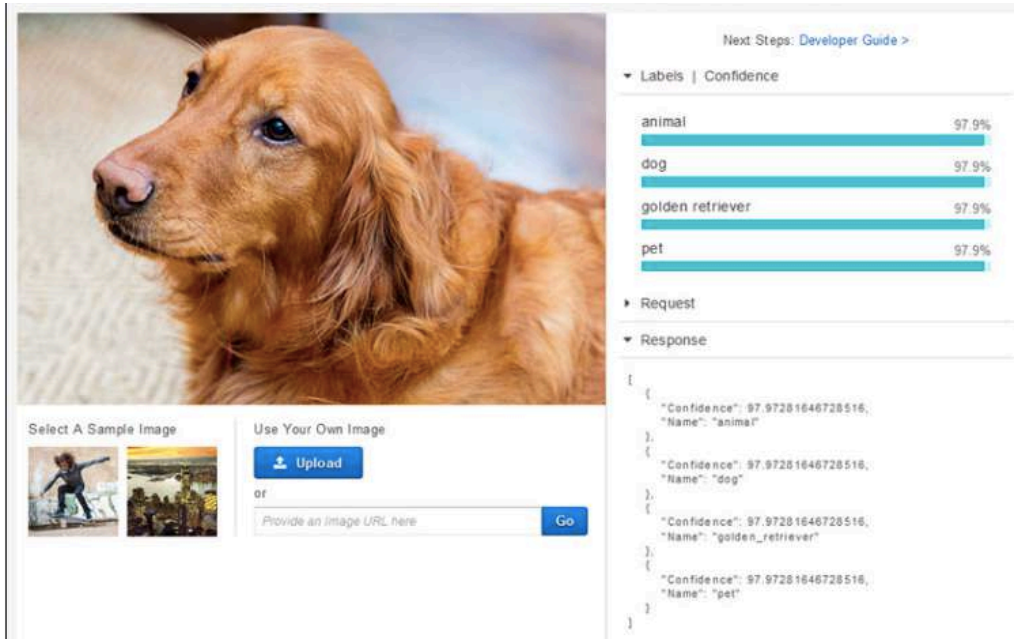
Pre-Trained Models in AWS

- **Amazon Comprehend** - Analyze Unstructured Text
- **Amazon Textract** - Easily extract text and data from virtually any document
- **Amazon Rekognition** - Search and Analyze Images and Videos
- **Amazon Transcribe** - Powerful Speech Recognition
- **Amazon Polly** - Turn Text into Lifelike Speech
- **Amazon Translate** - Powerful Neural Machine Translation



Pre-Trained Models in AWS

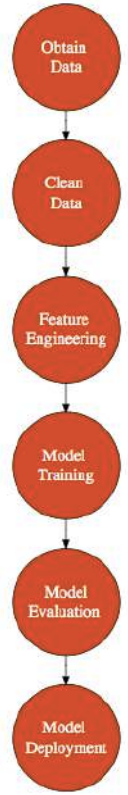
- **Amazon Personalize** - Add real-time recommendations to your apps
- **Amazon Fraud Detector** - Detect online fraud faster
- **Amazon Forecast** - Time-series forecasting service
- **Amazon Kendra** - Intelligent search service (Search from scattered content - multiple locations and content repositories)
- **Amazon Lex** - Build Voice and Text Chatbots



Creating Machine Learning Models - Steps

In 28 Minutes

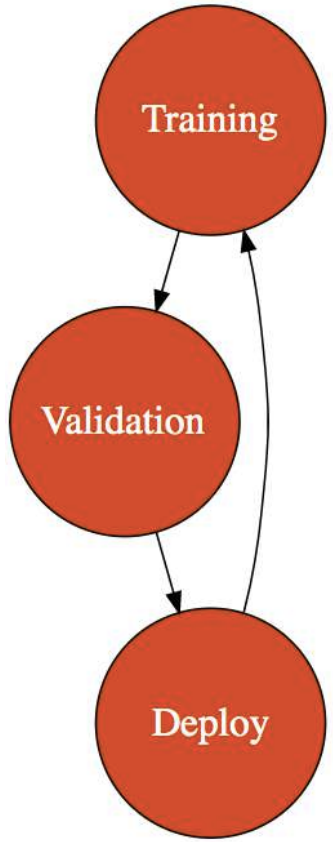
- **1:** Obtain Data
- **2:** Clean Data
- **3:** Feature Engineering: Identify Features and Label
- **4:** Create a Model using the Dataset and the ML algorithm
- **5:** Evaluate the accuracy of the model
- **6:** Deploy the model for use



Amazon SageMaker

In 28 Minutes

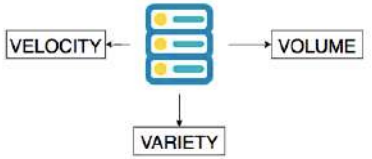
- **Amazon SageMaker**: Simplifies creation of your models
  - Manage data, code, compute, models etc
  - Prepare data
  - Train models
  - Publish models
  - Monitor models
- **Multiple options to create models**
  - **AutoML/Autopilot**: Build custom models with minimum ML expertise
  - **Build Your Own Models**: Data Scientists
    - Support for deep-learning frameworks such as TensorFlow, Apache MXNet, PyTorch, and more (use them within the built-in containers)
    - Data and compute management, pipelines

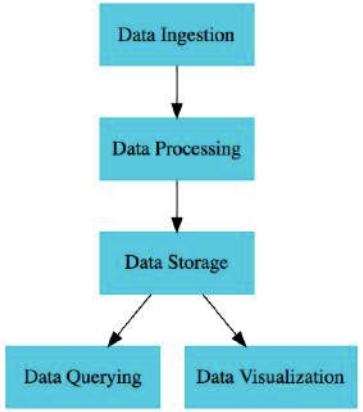


Big Data - Terminology and Evolution

In 28 Minutes

- **3Vs** of Big Data
  - **Volume**: Terabytes to Petabytes to Exabytes
  - **Variety**: Structured, Semi structured, Unstructured
  - **Velocity**: Batch, Streaming ..
- **Terminology**: Data warehouse vs Data lake
  - **Data warehouse**: PBs of Storage + Compute (Typically)
    - Data stored in a format ready for specific analysis! (processed data)
      - Examples: Teradata, BigQuery(GCP), Redshift(AWS), Azure Synapse Analytics
    - Typically uses specialized hardware
  - **Data lake**: Typically retains all raw data (compressed)
    - Typically object storage is used as data lake
      - Amazon S3, Google Cloud Storage, Azure Data Lake Storage Gen2 etc..
    - Flexibility while saving cost
    - Perform ad-hoc analysis on demand
    - Analytics & intelligence services (even data warehouses) can directly read from data lake
      - Azure Synapse Analytics, BigQuery(GCP), Redshift Spectrum(AWS), Amazon Athena etc..





Big Data & Datawarehousing in AWS

In 28 Minutes

Service	Scenario
Amazon Redshift	Run complex queries (SQL) against data warehouse - housing structured and unstructured data pulled in from a variety of sources
Amazon EMR	Managed Hadoop. Large scale data processing with high customization (machine learning, graph analytics) Important tools in Hadoop ecosystem are natively supported (Pig, Hive, Spark or Presto)
Amazon Kinesis	Data Streams + Firehose (ingestion) + Analytics + Video Streams
Amazon Managed Streaming for Apache Kafka (Amazon MSK)	Managed Kafka Service in AWS
Amazon Timestream	Fast, scalable, and serverless time series database for IoT and operational applications

Big Data & Datawarehousing in AWS - 2

In 28 Minutes

Service	Scenario
Amazon S3	Can be used as a Data Lake
AWS Lake Formation	Makes it easy to set up a secure data lake 1: Collect data from databases and object storage 2: Move it to S3 3: Clean data and secure access
Amazon Redshift Spectrum	Run queries directly against S3 without worrying about loading entire data from S3 into a data warehouse. Scale compute and storage independently.
Amazon Athena	Quick ad-hoc queries without provisioning a compute cluster (serverless) Amazon Redshift Spectrum is recommended if you are executing queries frequently against structured data
AWS Glue	Fully managed extract, transform, and load (ETL) service Simplify data preparation (capturing metadata) for analytics Transform data between formats (.csv to .parquet)

Big Data & Datawarehousing in AWS - 3

In 28 Minutes

Service	Scenario
AWS Data Exchange	Securely find and use third-party data in the cloud. Ideal for organizations that need access to a diverse range of datasets. Example Datasets: 20 Years of End-of-Day Stock Data for Top 10 US Companies by Market Cap, ShareThis Consumer Behavior & Interest Data
Amazon QuickSight	A fully managed business intelligence service. Create and publish interactive dashboards. Scales to hundreds of thousands of users. Can be embedded into applications, portals, and websites. Important capabilities: modern interactive dashboards, paginated reports, embedded analytics, and natural language queries

Observability in AWS

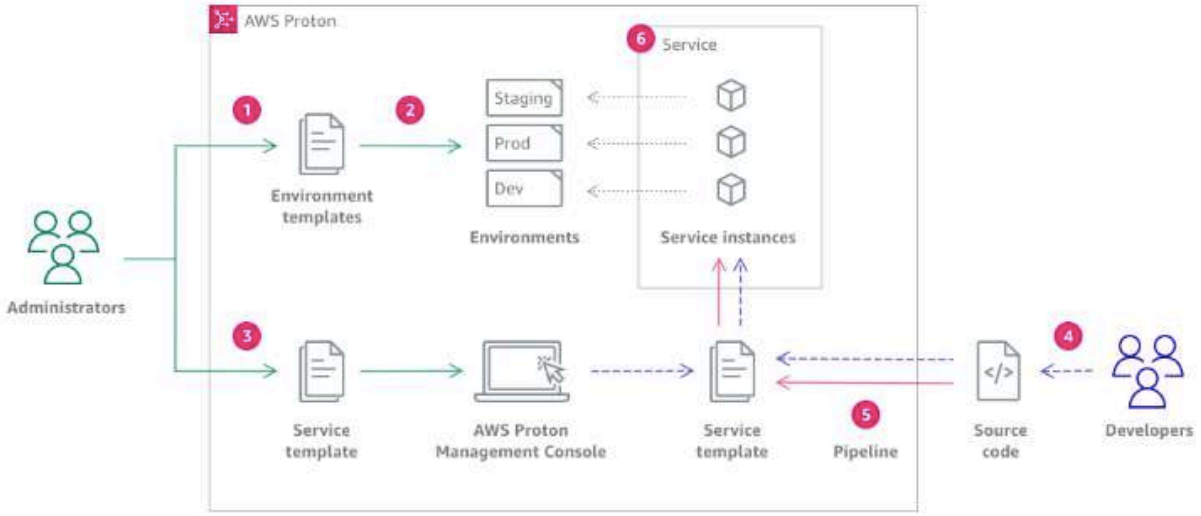
- **Amazon CloudWatch, AWS X-Ray**: Monitoring, Alarms,.. & Tracing
- **AWS Distro for OpenTelemetry**: AWS-supported OpenTelemetry dist.
  - Instrument your apps with it > Automate all telemetry (metrics, logs, and traces)
- **Amazon Managed Service for Prometheus** - Fully managed service for Prometheus.
  - Monitoring & alerting for containerized applications (Amazon EKS, Amazon ECS)
- **Amazon OpenSearch Service** - Perform interactive log analytics, real-time application monitoring, website search (ELK stack)
- **Amazon Managed Grafana** - Fully managed service for Grafana
  - Visualize and analyze your metrics, logs, and traces
  - Integrate with multiple data sources in your observability stack (CloudWatch, Amazon Managed Service for Prometheus, Amazon Elasticsearch/OpenSearch)!

More Management Services

Service	Description
AWS Marketplace	Digital catalog to find, test, buy, and deploy licensed software solutions using flexible pricing options: Bring Your Own License (BYOL), free trial, pay-as-you-go, hourly, monthly etc.
AWS Personal Health Dashboard	Personalized alerts when AWS is experiencing events that may impact you Provides troubleshooting guidance
AWS Service Catalog	Manage catalogs of IT services that are approved for use on AWS (virtual machine images, servers, software, and databases)
AWS License Manager	Manage your software licenses from vendors such as Microsoft, SAP, Oracle, and IBM across AWS and on-premises environments

AWS Proton

- Enhanced IaaC for serverless & containerized applications
  - Infrastructure, CI/CD pipeline, observability tools
- Create Std. Environments & Services across Enterprise:
  - Steps:
    - 1: Create Environment Templates
    - 2: Create Environment
    - 3: Create Service Template and Map a Source Repo
    - 4: Commit code
    - 5: Pipeline triggered
    - 6: Service deployed



https://docs.aws.amazon.com/proton/latest/userguide/Welcome.html

Important Things to Remember

Three ways to use AWS

- AWS Management Console
  - Mobile App
- AWS CLI (Command Line Interface)
  - Execute Commands
  - Create Scripts
- AWS SDKs (Software Development Kits)
  - Write Code (Java, JavaScript, Python, Go etc) using AWS APIs
  - Integrate into Existing Applications

AWS Support Plans - 1

Feature	Basic	Developer	Business	Enterprise
AWS Trusted Advisor	7 core checks	7 core checks	All checks	All checks
Account Assistance				Concierge Support Team
Technical Account Management				Designated Technical Account Manager (TAM) to pro-actively monitor your environment and assist with optimization and coordinate access to programs and AWS experts
Cost		Starts from $29	Starts from $100	Starts from $15,000

AWS Support Plans - 2

Feature	Basic	Developer	Business	Enterprise
Technical Support	24x7 access to customer service and forums	Business hours email access to Cloud Support Associates Unlimited cases / **1 primary contact**	24x7 phone, email, and chat access to Cloud Support Engineers Unlimited cases / unlimited contacts (IAM supported)	24x7 phone, email, and chat access to Cloud Support Engineers Unlimited cases / unlimited contacts (IAM supported)
Response Times	NA	General guidance - 24 hours System impaired: < 12 hours	General guidance: < 24 hours System impaired: < 12 hours Production impaired: < 4 hours Production down: < 1 hour	General guidance: < 24 hours System impaired: < 12 hours Production impaired: < 4 hours Production down: < 1 hour Business-critical system down: < 15 minutes

Amazon Workspaces: Cloud Based Virtual Desktop

In 28 Minutes

- **Amazon Workspaces**: Your Virtual Office in the Cloud
- **Cloud-based desktops**: Creates virtual desktops on powerful computers in the cloud
- **Ready to start**: Your own cloud-based computer, pre-loaded with all your software and settings
- **Very Flexible**: Choose from pre-built workspaces with popular software like Microsoft Office
- **Access from anywhere**: Use your Workspace from any computer, tablet, or even a phone

Amazon AppStream: Your favorite App Anywhere

In 28 Minutes

- **Amazon AppStream**: Use Your Favorite Apps Anywhere, on Any Device
- **Through Browser**: Access with your web browser, even on a simple laptop or tablet
- **Enable Remote workers**: Access work software from home, or even while traveling
- **Enable Students**: Use specialized design or engineering software even without a powerful computer
- **Happy Gamers**: Play demanding games on any device without needing a high-end gaming PC

10 More AWS Services For CLF-C02

In 28 Minutes

Service	Example Use Case	Explanation
Amazon Connect	A call center looking to provide cloud-based customer support with easy scalability	Cloud-based contact center service. Makes it easy for businesses to deliver better customer service at lower cost. Scalable and easy to set up. Integrates with various AWS services.
Amazon Simple Email Service (Amazon SES)	An e-commerce platform wanting to send transactional emails to its user base	A cloud-based email sending service. Reliable. Cost-effective. Highly scalable.
Amazon AppStream	An educational institution wanting to provide students with access to specific desktop applications without installing them on individual devices	Fully managed service that allows you to stream desktop applications to any device through a web browser. Centralizes application management. Provides users with instant access.

10 More AWS Services For CLF-C02 - 2

In 28 Minutes

Service	Example Use Case	Explanation
AWS Device Farm	A mobile app development company wanting to ensure their app works seamlessly across device types and OS's	Test mobile apps on range of devices. Ensure compatibility across devices.
AWS License Manager	A company that uses licensed software and wants to ensure compliance across its cloud and on-premises environments	Manage software licenses from software vendors across AWS and on-premises
AWS Launch Wizard	A business wanting to deploy (Microsoft SQL Server Always On or HANA based SAP systems or ..) on AWS but unsure about the optimal resources and configurations	Guided experience for deploying AWS resources for specific apps.
AWS Resource Access Manager (AWS RAM)	A company with multiple AWS accounts that wants to share specific resources across these accounts	Share AWS resources across accounts or within an organization Simplifies resource sharing while ensuring security and compliance.

10 More AWS Services For CLF-C02 - 3

In 28 Minutes

Service	Example Use Case	Explanation
AWS IAM Identity Center (AWS Single Sign-On)	A large enterprise with multiple AWS accounts and applications wanting to streamline user access with a single set of credentials	Provides centralized access and identity management across AWS accounts and applications. Simplifies user access to AWS accounts and business applications through a single sign-on experience.
AWS Network Firewall	An organization looking to protect its AWS VPCs from malicious traffic & common web threats	Provides network-level protection against malicious & unauthorized traffic. Offers fine-grained traffic filtering and threat detection capabilities.
AWS Elastic Disaster Recovery	An enterprise that wants to ensure business continuity by quickly recovering its critical applications after unforeseen disasters or outages	Enables quick recovery from infrastructure or service disruptions. Minimizes downtime and data loss by automating recovery processes.

10 More AWS Services For CLF-C02 - Scenarios

In 28 Minutes

Scenario	Recommendation
Share AWS resources across accounts or within an organization	AWS Resource Access Manager (AWS RAM)
A business wanting to deploy a HANA based SAP systems on AWS but unsure about the optimal resources and configurations	AWS Launch Wizard
A company that uses licensed software and wants to ensure compliance across its cloud and on-premises environments	AWS License Manager
Test mobile apps on a range of devices	AWS Device Farm
Provide students with access to specific desktop applications without installing them on individual devices	Amazon AppStream
Cloud-based contact center service	Amazon Connect

Get Ready

Certification Resources

Title	Link
Certification - Home Page	*https://aws.amazon.com/certification/certified-cloud-practitioner/*
Overview of Amazon Web Services	*https://d0.awsstatic.com/whitepapers/aws-overview.pdf*
How AWS Pricing Works	*https://d0.awsstatic.com/whitepapers/aws_pricing_overview.pdf*
AWS Well-Architected Framework	*https://d1.awsstatic.com/whitepapers/architecture/AWS_Well-Architected_Framework.pdf*
The Total Cost of (Non) Ownership of Web Applications in the Cloud	*https://media.amazonwebservices.com/AWS_TCO_Web_Applications.pdf*
Compare AWS Support Plans	*https://aws.amazon.com/premiumsupport/plans/*

Certification Exam

In 28 Minutes

- Multiple Choice Questions
  - Type 1 : Single Answer - 4 options and 1 right answer
  - Type 2 : Multiple Answer - 5 (or more) options and 2 (or more) right answers
- No penalty for wrong answers
  - Feel free to guess if you do not know the answer
- 65 questions and 90 minutes
- Result immediately shown after exam completion
- Email with detailed scores (a couple of days later)

Certification Exam - My Recommendations

In28 Minutes

- Read the entire question
  - Identify the key parts of the question
- Read all answers at least once
- If you do NOT know the answer, eliminate wrong answers first
- Mark questions for future consideration and review them before final submission

You are all set!

Let's clap for you!

- You have a lot of patience! Congratulations
- You have put your best foot forward to be an AWS Certified Cloud Practitioner
- Make sure you prepare well and
- Good Luck!

Do Not Forget!

In 28 Minutes

- Recommend the course to your friends!
  - Do not forget to review!
- Your Success = My Success
  - Share your success story with me on LinkedIn (Ranga Karanam)
  - Share your success story and lessons learnt in Q&A with other learners!

What Next?

FASTEST ROADMAPS

in28minutes.com

In28 Minutes

Google Cloud Certifications

Azure Certifications

AWS Certifications

DevOps

Java Full Stack

Java Microservices