# 2. Cloud Fundamentals

## 2.1. Cloud Computing

### What is Cloud Computing

Cloud computing involves the delivery of IT resources such as:

- databases,
- compute power,
- application,
- security,
- analytics,
- artificial intelligence
- augmented reality
  etc, over the Internet.

There are three characteristics of Cloud computing.

1. Pay-as-you-go,
2. autoscaling, and
3. serverless.

Pay-as-you-go: there are no long-term contracts, and you pay only for what you use. Sometimes you pay by the minute, hour, or even second.
Autoscaling: resources that grow or shrink or automatically scale based on demand. Servers can be provisioned almost instantly so that you don't have to guess about capacity or make huge capital investments in servers upfront.
Serverless: a Cloud provider manages the servers for you. You, as the developer, simply write code. The Cloud provider executes that code on some server somewhere and you're not concerned with that aspect.

Resources:
Cloud Computing

---

### Types of Cloud Computing

There are three types of Cloud Computing:

- Infrastructure as a service,
- platform as a service,
- software as a service.

Infrastructure as a service: the provider supplies a Virtual Server instance, storage and mechanisms for you to manage the servers.

Examples include Amazon Web Services or AWS, Rackspace, Digital Ocean and Iron Mountain etc.

Platform as a service: the development tools are provided that are hosted on a provider's infrastructure. So they manage the hardware and operating system. You focus on managing and deploying applications.
Examples of this are GoDaddy, salesforcesforce.com etc.

Software as a service: this deliver software applications over the Internet that are run and managed by the service provider.
Examples of this include Google's Gmail or Microsoft's Office 365.

Resources:
Types of Cloud Computing

---

## Cloud Computing Deployment Model

There are three Cloud computing deployment models:

- public,
- private, and
- hybrid.

`Public Cloud`: makes resources available over the internet. Resources could include servers, databases, application development services, etc. Amazon Web Services or AWS is currently the largest Public Cloud provider.

`Private Cloud`: called On-premises, is a proprietary network or an internal data center that supplies services to a limited number of people, and internal to a specific company.

`Hybrid model`: contains a combination of both a public and private Cloud. For example, PII or Personally Identifiable Information about customers may be stored in an On-premise database for security reasons, while a web application to manage that data may be served publicly with orchestration and communication between the two.

The hybrid model is a growing trend in the industry for those organizations that have been slow to adopt the cloud due to being in a heavily regulated industry. The hybrid model gives organizations the flexibility to slowly migrate to the cloud.

On-premises is BEST in describing 'Private Cloud'.

---

## Common Benefits

As a software engineer, the biggest benefit I've seen is the ability to innovate quickly.

- Innovation: I can develop an application and put it in the hands of millions on a global scale with great performance in a relatively short period of time.

- Ability to scale quickly: I also have the ability to fail fast. That is, try an idea and put it out there quickly to see if it's going to work, which is great when you have a lot of innovative ideas to try out.

There are several additional benefits to Cloud computing.

- don't have to guess about capacity and make huge capital investment upfront.
- pay for only what you need and use,
- can quickly go global with good performance with lower latency since content can be delivered from locations close to your user base.
- can also deliver more quickly since you don't have to stand up and manage servers.

---

## Options

### List of AWS Products

Currently, Amazon Web Services or AWS, a subsidiary of Amazon, is the most popular Cloud platform in the enterprise adoption space.
Another popular option is Google Cloud Platform or GCP, and is offered by Google.
There's also Microsoft Azure created by of course, Microsoft.

All of these Cloud providers are in fierce competition with one another.

---

## Services

Now, we will not cover all of these services,
just the more foundational services:

- popular storage and content delivery services,
- networking,
- security, and
- messaging services.

All of the basic building blocks that you'll need for Cloud-based Applications.

### Analytics for Big Data

- `Quick Sight`: analytics for big data and visualization tools
- `Athena`: querying tools for objects or files
- `Redshift`: data warehousing

### Application integration

- `Simple Queue Service` (`SQS`): application integration
- `Simple Notification Service` (`SNS`): Notification and alerting services

### Cost management

- `AWS Budgets`: cost management tools

### Compute services

- `Elastic Cloud Compute` (`EC2`): virtual servers
- `AWS Lambda`: running code in a serverless fashion and in response to events.
- `Elastic Beanstalk`: for running web applications.

### Database management services

- `MySQL`
- `Oracle`
- `SQLServer`
- `DynamoDB` (No SQL)
- `MongoDB` (Document based)

### Developer tools

- `Cloud 9`: developer tools / Cloud IDE
- `Code Pipeline`: continuous integration

### Security services

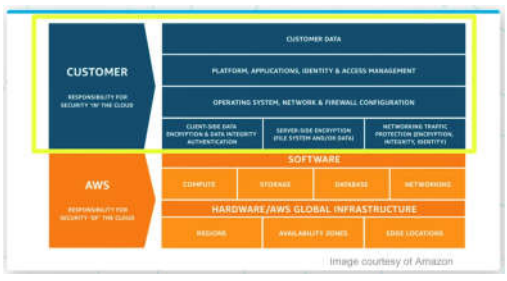
- `Key Management Service` (`KMS`): Key Management Service for data encryption
- `AWS Shield`: DDoS protection
- `Identity and Access Management` (`IAM`): fine grain control over permissions

### Additional Services

- `Blockchain`
- `Machine Learning`
- `Computer Vision`
- `Internet of Things` (`IoT`)
- `AR/VR`

## Shared Responsibility Model

AWS is responsible for security `OF` the cloud, we are responsible for security `IN` the cloud.




AWS is responsible for:

- Securing edge locations

- Monitoring physical device security
- Providing physical access control to hardware/software
- Database patching
- Discarding physical storage devices

You are responsible for:

- Managing AWS Identity and Access Management (IAM)
- Encrypting data
- Preventing or detecting when an AWS account has been compromised
- Restricting access to AWS services to only those users who need it

Shared Responsibility Model - Amazon Web Services (AWS)

# 2.2 Foundational & Compute Service

## Servers in the Cloud

Servers in the cloud have revolutionized the IT industry.

- Scale capacity up and down based on demands.
- Storage, more memory, and computing power can be added as needed.
- Obtain servers in minutes.
- No need for onsite hardware or capital expenses.

## EC2 - Elastic Cloud Compute

EC2 is a foundational piece of AWS cloud computing platform and is a service that provides *servers for rent* in the cloud.

### Pricing Options

There are several pricing options for EC2.

- `On Demand` - Pay as you go, no contract.
- `Dedicated Hosts` - You have your own dedicated hardware and don't share it with others.
- `Spot` - You place a bid on an instance price. If there is extra capacity that falls below your bid, an EC2 instance is provisioned. If the price goes above your bid while the instance is running, the instance is terminated.
- `Reserved Instances` - You earn huge discounts if you pay up front and sign a 1-year or 3-year contract.

### Tips

- EC2 is found under the Compute section of the AWS Management Console.
- Spot instances can save you up to 90% off the on-demand pricing.
- There are several instance types that provide varying combinations of CPU, memory, storage, and networking capacity.

Resources:

- Amazon EC2

## EBS - Elastic Block Store

EBS is a storage solution for EC2 instances and is a physical hard drive that is attached to the EC2 instance to increase storage.
It has to be attached to the server and mounted before you start storing data on it.

Some instance types have EBS volume already attached.

There are 2 types of memory for an EC2 instance:

- In-memory or instance store
- On EBS

The benefit of EBS over instance store is that you're able to persist data after the EC2 instance is terminated / shut-down. Any data stored on the volume are still accessible.

AWS automatically replicate each Amazon EBS volume within its AZ (Availability Zone). This protects against component failure, offering high availability and durability.

**Tips**

- `EBS` is found on the `EC2` Dashboard.
- There are several `EBS` volume types that fall under the categories of Solid State Drives (`SSD`) and Hard Disk Drives (`HDD`).

Resources:

- Amazon Elastic Block Store - EBS

## Security

Security in the cloud allows you to have complete control over your virtual networking environment.

- Configure your virtual network with public or private facing subnets
- Launch your servers in the selected network to secure access

AWS Lambda – Serverless Compute - Amazon Web Services

## Elastic Beanstalk

Elastic Beanstalks is an orchestration service that allows you to deploy a web application at the touch of a button by spinning up (or provisioning) all of the services that you need to run your application.

- Orchestration service

- Deploy an application
- Provisioning services
- Automates the process

The process:

- Instantiate EC2
- Setup Auto-Scaling
- Setup ELB (Elastic Load Balancer)

You still retain control of the services automatically spun up and you can administer them separately.

Elastic Beanstalk supports: Java, PHP, Python, .NET, Node.js, Ruby and Docker.
Also some common servers: Apache, HTTP, Tomcat, Nginx, IIS, etc.

Elastic Beanstalk can spin up databases instances as well, VPC, security groups, all while deploying your code.

Resources:

- AWS Elastic Beanstalk – Deploy Web Applications
- What is AWS Elastic Beanstalk

# 2.3 Storage and Content Delivery

## Introduction

Storage services provide companies facilities to store data, with benefits like `durability`, `availability` and `scalability`.

Durability: guarantees that you will not lose the data you upload to the cloud
Availability: addresses how quickly you can access your data. High availability provides fast and reliable access to the data you stored in the cloud.
Scalability: allows applications running in the cloud to always meet demand seamlessly by adding or removing resources necessary to maintain steady-state and fast response times.

Vertical scaling: or scaling up is modifying your server to meet demand (adding more memory or capacity to a server)
Horizontal scaling: adding or removing servers to meet demand.
Diagonal scaling: is a combination of both vertical and horizontal scaling and offers maximum flexibility.

Storage & Database Services

- Amazon Simple Storage Service (Amazon S3)
- Amazon Simple Storage Service (Amazon S3) Glacier
- DynamoDB
- Relational Database Service (RDS)
- Redshift

- ElastiCache
- Neptune
- Amazon DocumentDB

## S3 & Glacier

- S3 is an object storage system for text documents, image files, HTML files, etc.
- Unique URL is given to each object to enable users to access it.
- All objects/files are stored in a what's called a bucket. One bucket can hold millions of objects.
- S3 buckets live in a region but a bucket name must be globally unique.
- S3 is designed for durability of 99.999999999% (9 digits behind) of objects across multiple AZs
- Availability of 99.99% over a given year.
- Use cases:
  - Hosting static websites
  - Content delivery
  - Backup and recovery
  - Archiving
  - Big data
  - Application data
  - Hybrid cloud storage
- S3 storage classes (i.e. different data access levels for your data at certain price points)
- S3 Glacier is more suited to data-archiving, a place for data that you don't intend to access frequently
- Glacier is cheaper but retrival can be minutes up to hours.
- Glacier's use case:
  - Monthly log files
  - Need to keep them for audit purposes

## DynamoDB

DynamoDB is a NoSQL document database service that is fully managed. Unlike traditional databases, NoSQL databases, are schema-less.
Schema-less simply means that the database doesn't contain a fixed (or rigid) data structure and can easily change on the fly based on the data being passed in. This offers great flexibility.

NoSQL fits well with modern-day server-less web applications, micro-services, gamin, IoT data and mobile back-ends that need to quickly scale and handle large amount of data or millions request per second.

Data stored in JSON or JSON like text.
Each row or record in DynamoDB is called a document.

Tips

- DynamoDB is found under the Database section on the AWS Management Console.
- DynamoDB can handle more than 10 trillion requests per day.
- DynamoDB is serverless as there are no servers to provision, patch, or manage.
- DynamoDB supports key-value and document data models.
- DynamoDB synchronously replicates data across three AZs in an AWS Region.
- DynamoDB supports GET/PUT operations using a primary key.

Resources:

- Amazon DynamoDB - Overview
- What is Amazon DynamoDB

## Relational Database Service (RDS)

RDS (or Relational Database Service) is a service that aids in the administration and management of databases. RDS assists with database administrative tasks that include:

- upgrades,
- patching,
- installs,
- backups,
- monitoring,
- performance checks,
- security, etc.

Database Engine Support:

- Oracle
- PostgreSQL
- MySQL
- MariaDB
- SQL Server
- Aurora

Features:

- failover
- backups
- restore
- encryption
- security
- monitoring
- data replication
- scalability

Resources:

- What is a Relational Database? – Amazon Web Services (AWS)
- Amazon Relational Database Service (RDS) – AWS
- Databases on AWS

## RedShift

Redshift is a cloud data warehousing service to help companies manage big data. Redshift allows you to run fast queries against your data using SQL, ETL, and BI tools. Redshift stores data in a column format to aid in fast querying.

Features:

- Fast query and analysis (not a transaction processing)
- Contains historical data from transactional systems
- Recent transactions placed in relational databases
- Old orders/transactional data archived in a data warehouse

Format:

- Stores data in columnar format
- Not a row store like in a relational database
- This aids in fast query and analysis

Tips

- Redshift can be found under the Database section on the AWS Management Console.
- Redshift delivers great performance by using machine learning.
- Redshift Spectrum is a feature that enables you to run queries against data in Amazon S3.
- Redshift encrypts and keeps your data secure in transit and at rest.
- Redshift clusters can be isolated using Amazon Virtual Private Cloud (VPC).

Resources:

- What is Amazon RedShift
- Amazon Redshift - Amazon Web Services

## Content Delivery Network (CDN)

A Content Delivery Network (or CDN) speeds up delivery of your static and dynamic web content by caching content in an Edge Location close to your user base.

Contents such as:

- web pages
- CSS
- Javascript
- Images

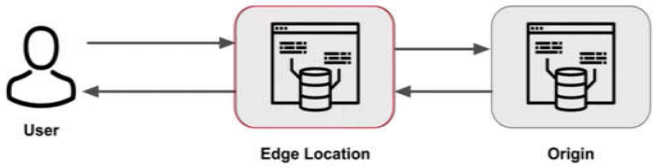
Benefits of a CDN:

- Reduces latency
- decreases the load of server
- better user experience

Typically contents are cached for a certain period of time.

## CloudFront

CloudFront is Amazon service for content delivery. CloudFront speeds up the delivery of contents through Amazon's worldwide network of mini data centres called `edge-locations`.




- You can configure how long an item remain cached before a refresh. Or,
- you can manually expire or remove from the cache should it need to be changed.

CloudFront works with other AWS services, as shown below, as an origin source for your application:

- Amazon S3
- Elastic Load Balancing
- Amazon EC2
- Lambda@Edge
- AWS Shield

Tips

- CloudFront is found under the Networking & Content Delivery section on the AWS Management Console.
- Amazon countinously adds new Edge Locations.
- CloudFront ensures that end-user requests are served from the closest edge location.
- CloudFront works with non-AWS origin sources.
- You can use GeoIP blocking to serve content (or not serve content) to specific countries.
- Cache control headers determine how frequently CloudFront needs to check the origin for an updated version your file.
- The maximum size of a single file that can be delivered through Amazon CloudFront is 20 GB.

Resources:

- Content Delivery Network (CDN) | Low Latency, High Transfer Speeds, Video Streaming | Amazon CloudFront
- What is Amazon CloudFront

# 2.4 Security

It's important for companies to secure data, particularly `PII` (Personally Identifiable Information) data such as:

- Social security numbers
- Bank account information
- Passport numbers, etc.

Along protecting data, it’s also necessary to protect applications that provide access to data.
Cloud infrasturcture:

- protects data
- protects applications that access data
- protects infrastructure

## AWS Shield

AWS Shield is a managed protection service against DDoS (Distributed Denial of Service) attacks.
This attack is an attempt to make a website or an application unavailable by overwhelming it with traffics from multiple sources.
When a server is overwhelmed, it typically crashes or no longer serve requests.

AWS Shield is a service provided out-of-the-box. It always running automatically, and is part of a Free standard tier.

Paid tier is available if customer wanted to utilise more advanced features.

Tips

- AWS Shield can be found under the Security, Identity, & Compliance section on the AWS Management Console.
- AWS Shield Standard is always-on, using techniques to detect malicious traffic.
- AWS Shield Advanced provides enhanced detection.

Resources:

- AWS Shield - Amazon Web Services (AWS)

## AWS WAF (Web Application Firewall)

AWS WAF (or AWS Web Application Firewall) provides a firewall that protects your web applications.

A firewall is a network security mechanism that monitors and controls incoming and outgoing network traffics based on preset security rules.

A firewall stands in front of your applications as a way to guard who can access it.

It protects against common attacks such as:

- SQL injection

- Cross-site scripting
  by reviewing data sent to your applications and stopping well-known attacks.

Tips

- WAF is found under the Security, Identity, & Compliance section on the AWS Management Console.
- WAF can protect web sites not hosted in AWS through Cloud Front.
- You can configure CloudFront to present a custom error page when requests are blocked.

Resources:

- AWS WAF - Web Application Firewall - Amazon Web Services (AWS)

## Identity & Access Management

Along with protecting data and systems, we also need to:

- verify users (they are who they say they are)
- they only have access to specific data they need to, and not to everything. This concept is called `least privileged access`.

Identity & Access Management (IAM) is an AWS service that allows us to configure who can access our AWS `account`, `services`, or even `applications` running in our account.
IAM is a global service and is automatically available across ALL regions.

Email: root account, should not be used for day to day use
To secure root account, use MFA (Multi-Factor Authentication)

IAM User:

- An entity created on AWS that represents a person or a service that interacts with services or applications running in your AWS account.
- A user in AWS consists of `user-name` and `access-credentials` like a console password or an access-key, which include an access-key ID and a secret access-key.

IAM Group:

- A collection of users
- You can specify permissions for a collection, which make permissions easier to manage.

IAM Role:

- An identity with permissions or a set of privileges that are NOT associated with a specific IAM User or IAM Group.
- Roles can be attached to a user, and a user can assume a single role temporarily to perform a specific task.

Policy:

- a way to define granular level permission
- can be attached to Users, Groups, and Roles

AWS provides a predefined list of policies. Additionally, you can create your own custom policies using JSON.

EC2 Security Groups are not a part of IAM Security Group.
EC2 Security Groups are associated with EC2 instance and act as a built-in firewall for your virtual servers to either allow or deny access.

Resources:

- AWS Identity & Access Management - Amazon Web Services
- What is IAM

## 2.4 Networking & Elasticity

The network is the foundation of your infrastructure.

Cloud networking includes:

- Network architecture
- Network connectivity
- Application delivery
- Global performance
- Delivery

Network connectivity includes network services that offer reliable and cost-effective ways to route end-users to Internet applications.

Background on IP Address, Domain Name, DNS (Domain Name Service), Domain Name Authority.

### Route 53

Route-53 is AWS Cloud Domain Name service. It has reliable and scalable DNS servers distributed around the globe.
Route-53:

- scales automatically to manage spikes in DNS queries,
- allows you to register a new domain name
- manage existing ones
- Route internet traffic to the resources for your domain
- Checks the health of your resources
- route users based on the user’s geographic location

Health checks ensure:

- web servers are up and running
- and offer DNS failover to automatically route web visitors to alternate locations to avoid site outages.

Resources:

- Amazon Route 53 - Amazon Web Services

## Elasticity

One of the benefits of the Cloud is it allows you to stop guessing about capacity when running your applications.

With Elasticity, your servers, databases, application resources can automatically scale up or down based on the load or the number of users accessing your applications at a given time.

You can scale up (vertically, by increasing memory, disk, i.o, CPU, etc.) or scale out (horizontally) by increasing resources (servers)

## EC2 AutoScaling

EC2 AutoScaling is a service that monitors EC2 instances and automatically adjust by adding or removing EC2 instances based on conditions you define in order to maintain applications availability and to provide peak performance to your users.

EC2 AutoScaling works with AWS messaging services like the SNS (Simple Notification Services) to alert you when EC2 is launching or terminating your EC2 instances.

Features

- Automatically scale in and out based on needs.
- Included automatically with Amazon EC2.
- Automate how your Amazon EC2 instances are managed.

Tips

- EC2 Auto Scaling is found on the EC2 Dashboard.
- EC2 Auto Scaling adds instances only when needed, optimizing cost savings.
- EC2 predictive scaling removes the need for manual adjustment of auto scaling parameters over time.

Resources:

- Amazon EC2 Auto Scaling
- What is Amazon EC2 Autoscaling

## AWS AutoScaling

There is also AWS AutoScaling service that is different to the EC2 AutoScaling that allows you to setup other services such as DynamoDB to automatically scale.

## Elastic Load Balancing

Elastic Load Balancer is a service that balances the load between 2 or more servers. It stands in front of servers and provide redundancy and good performance.

Redundancy: if you lose a server the Load Balancer will send request to other working server
Good Performance: if a server starts having issues or bottlenectks, the Load Balancer will add more servers to the pool of available servers.

Tips

- Elastic Load Balancing can be found on the EC2 Dashbaoard.
- Elastic Load Balancing works with EC2 Instances, containers, IP addresses, and Lambda functions.
- You can configure Amazon EC2 instances to only accept traffic from a load balancer.

Resources:

- Elastic Load Balancing - Amazon Web Services

## 2.5 Messaging & Containers

Users of your application need to be notified when certain events happen. E.g.:

- if a large withdrawal is made from a bank account
- someone from outside your country login to your netflix account

Notifications such as `text messages` and `emails` can be send through services in the Cloud. The use of the Cloud offers benefits like lower costs, increased storage and flexibility.

Through Cloud services, you can:

- send notifications and even
- track the lifecycle of those notifications.

`Messaging` is a form of notifications. But instead received by human, messaging's notification is received by an application. Messaging typically happens between Internet-based applications and devices. One system can send messages to another system.

### SNS

Amazon SNS is a cloud service that allows you to send notification to the users of your application.
It allows you to:

- decouple notification logic from being embedded in your application, and
- allows your notifications to be published to a large number of subscribers

SNS uses a Publish/Subscribe model. This means, in order for users to receive messages they will have to sign-up or subscribe first.

Subscribers can be:

- a person or
- other AWS services: Amazon SQS queues, AWS Lambda functions, and HTTP/S webhooks.

Notifications can be sent to users using mobile-push, text-messages or email.

Tips

- SNS is found under the Application Integration section on the AWS Management Console.
- SNS Topic names are limited to 256 characters.
- A notification can contain only one message.

Resources:

- Amazon Simple Notification Service (SNS) | AWS
- What is Amazon SNS

## Queues

A queue is a data structure that holds requests, sometimes called messages.
A queue is similar in concept to waiting in line at a department store. People join the line and wait for their turn to check out. Similarly, messages join a queue and wait their turn to be processed.

Messages in a queue are commonly processed in order first-in first out or FIFO.

Example, In a money-transfer app with millions of users sending funds back and forth.

- instead of making users wait for confirmation while money is being transferred through the platform
- you let them know their request was submitted and allow them to go about their day
- Behind the scenes, you please the money transfer request on a queue and process that requests when system resources are available - send the money transfer process along with doing security check could take a while to process.
- The benefit of using a queue is that the user doesn’t have to sit there waiting.
- The use of messaging queue will help improve performance and scalability.

The use of asynchronous processing where a user doesn’t wait for a response improves the overall user experience.

## SQS (Simple Queue Service)

SQS is a fully managed message queue service that allows you to integrate queuing functionality into your application.

With SQS, you can send, store, and receive messages between applications without losing messages.

SQS offers two types of message queues:

- standard queues: offer `best-effort-ordering`
- FIFO queues: designed to guarantee that messages are processed exactly once in the exact order they were added to the queue.

Example:

- A course registration system.
- To improve scalability, you send account creation and course registration messages to a queue.


- In order for a student to register for a course, their account must first be created then they can be registered.
- The ordering of these steps is very important. You do not want students to register for a course before their account is created.
- Since ordering is important a FIFO queue should be used in this case.

Tips

- The Simple Queue Service (SQS) is found under the Application Integration on the AWS Management Console.
- FIFO queues support up to 300 messages per second.
- FIFO queues guarantee the ordering of messages.
- Standard queues offer best-effort ordering but no guarantees.
- Standard queues deliver a message at least once, but occasionally more than one copy of a message is delivered.

Resources:

- Amazon Simple Queue Service (SQS) | Message Queuing for Messaging Applications | AWS
- What is Amazon SQS

## Containers in the Cloud

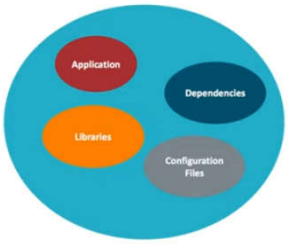
Enterprises are adopting container technology at an explosive rate.
Docker is the leading container technology.
There are several container-orchestration-services that help you manage your docker clusters. `Kubernetes` is one of them. Another is `Docker Swarm`.

Why container is so popular?

A container consists of everything an application needs to run:

- the application and its dependencies like:
  - libraries,
  - utilities,

    - configuration files,
      all bundled into one package.




Benefits:

- Migration - Don’t have to rebuild.
    - Instead of having to rebuild your application and its environment from scratch, as it moves from environment to environment, say from development to production, you can easily *move the entire container from environment to environment* without any issues.
    - *Each container is an independent component that can run on its own*.
    - Say you have a huge monolith application that you’re migrating to a microservices architecture.
        - *The `microservices-architecture` decomposes large, complex, monolith system into discrete, individual, stand-alone component that can communicate among themselves, working together, or with external systems*.
    - *A docker container works well with microservices use case because each microservice is its own independent component with no inter-dependencies*.

Resources:

- Docker overview | Docker Documentation
- What is a Container? | App Containerization | Docker

## Elastic Container Service (ECS)

ECS is an orchestration service used for:

- *automating deployment*,
- *scaling*, and
- *managing of containerised applications*.

ECS works well with Docker container by:

- *launching* and *stopping* Docker containers
- *scaling* applications
- *querying the state* of applications

Tips:

- ECS is under the Compute section
- You can schedule long-running applications, services, and bath processes using ECS
- Docker is the only container platform supported by Amazon ECS.

ECS is userd for:

- automating deployment
- automatic scaling
- managing containerized applications

Resources:

- Amazon ECS - Run containerized applications in production
- What is Amazon ECS

# 2.6 AWS Management

## Logging & Auditing in the Cloud

It's important to have visibility into your cloud resources and applications.

Visibility into:

- How is this server performing?
- What is the current load on the server?
- What is the root cause of an application error that a user is seeing?
- What is the path that leads to this error?

To proactively monitor your resources and applications in the Cloud, you'll need access to `logging` and `auditing` services.

## Cloud Trail

Cloud Trail allows you to *audit (or review)* everything that occurs in your AWS account. For example:

- Who has logged in
- Services accessed
- Actions performed
- Parameters for the actions
- Responses returned

Cloud Trail does this by:

- *recording all the AWS API calls* occurring in your account and
- delivering *a log file* to you.

Cloud trail logs actions performed through the *AWS management console* and the *AWS SDK* (Software Development Kit). This means, any application that uses the software development kit to interact with AWS services, command-line tools, and other AWS services will be logged via Cloud Trail.

You can also setup alerts and alarms to notify you should certain activities occur.

Tips

- CloudTrail shows results for the last 90 days
- You can create up to 5 trails in an AWS region

AWS CloudTrail – Amazon Web Services

## CloudWatch

Cloud watch is a service that monitors resources and applications that run on AWS.

There are several useful features:

- collecting and tracking metrics
- collecting and monitoring log files
- setting alarms and creating triggers to run your AWS resources
- reacting to changes in your AWS resources

The ability to review log files is crucial during development and support of applications.

You can routinely use CloudWatch logs written from my lambda functions, to diagnose issues and monitor application flow.

You can also use CloudWatch as a trigger for your lambda functions causing a lambda to run on a given schedule.

Tips

- Metrics are provided automatically for a number of AWS products and services.

Amazon CloudWatch - Application and Infrastructure Monitoring
What is Amazon CloudWatch

## Infrastructure as Code

Infrastructure as Code allows you to describe and provision all the infrastructure resources in your cloud environment. You can stand up servers, databases, runtime parameters, resources, etc. based on scripts that you write. Infrastructure as Code is a time-saving feature because it allows you to provision (or stand up) resources in a reproducible way.

You can manage a collection of related resources and treat them as one logical unit.

Let’s say, in order to setup a development environment, you’ll have to do the following steps:

- configure a vpc security group
- launch an EC2 instance
- Create load balancers
- Create an RDS instance

- Create auto scaling

You can write a script to create the above resources for you in a repeatable way.

Infrastructure as code - Wikipedia

## CloudFormation

AWS CloudFormation is AWS infrastructure as code service, allowing you to model your entire infrastructure in a text file template which then can be used to provision AWS resources based on the scripts you write.

We are able to provision EC2 instances, VPCs, and sub-nets.

The templates are written using text, either `JSON` or `YAML`. You can write from scratch or re-use a blueprint provided by AWS.

CloudFormation scripts can be created using a visual designer with drag and drop components.

Since your infrastructure is now code, you can check your scripts into version control and even review the files with your team in order to receive feedback.

You can still individually manage AWS resources that are part of a CloudFormation stack.

AWS CloudFormation - Infrastructure as Code & AWS Resource Provisioning
What is AWS CloudFormation

# Project - Hosting Static Website

Steps:

1. Create a S3 bucket and upload the website files to your bucket.
2. Configure the bucket for website hosting and secure it using IAM policies.
3. Speed up content delivery using AWS' content distribution network service, CloudFront.
4. Access your website in a browser using the unique CloudFront endpoint.