AWS Overview

WHAT

is AWS?

WHEN

did AWS start?

WHY

is AWS so popular?

WHAT
is AWS?

WHEN
did AWS start?

WHY
is AWS so popular?

Amazon

Web Services

WHAT
is AWS?

WHEN
did AWS start?

WHY
is AWS so popular?

Amazon

Web Services

WHAT
is AWS?

WHEN
did AWS start?

WHY
is AWS so popular?

Amazon Web Services = Cloud Service Provider

- provides a cloud-based platform or cloud services
- Allows you to **rent out** virtual servers that you access remotely

WHAT
is AWS?

WHEN
did AWS start?

WHY
is AWS so popular?

Cloud Service Provider *is like a* Car Rental

WHAT
is AWS?

WHEN
did AWS start?

WHY
is AWS so popular?

Cloud Service Provider

With different types of **CPU**, **Storage**, **Network** and other components that you can choose from!

Virtual Machines

Physical Servers

Storage Appliances

Network Devices

Available for RENT and accessible online via *Web Service* interfaces (REST, SOAP etc...)

Car Rental

WHAT
is AWS?

WHEN
did AWS start?

WHY
is AWS so popular?

2004

- AWS started out as a department **within** Amazon Inc.
- Used only by early Amazon customers
- Web services are not available publicly

WHAT
is AWS?

WHEN
did AWS start?

WHY
is AWS so popular?

2004

2006

- AWS officially started its operation as a **public cloud** service provider
- Released Amazon S3 (Simple Storage Service)
- Released Amazon SQS (Simple Queue Service)

WHAT
is AWS?

WHEN
did AWS start?

WHY
is AWS so popular?

2004

2006

Today

- Offers hundreds of fully-featured services that are available globally
- Provides a highly reliable, scalable, and **low-cost** infrastructure platform in the cloud
- Boasts a broad set of cloud-based products

WHAT
is AWS?

WHEN
did AWS start?

WHY
is AWS so popular?

Today

aws is the **world's leading cloud platform**.

- Used by **millions** of customers
- Supports various workloads
- Significantly lower your operating costs
- Enables companies to scale globally in minutes!

AWS Global Infrastructure

Has thousands of servers!

These *physical* servers generate *virtual* machines or store your data!

Availability Zone

Region

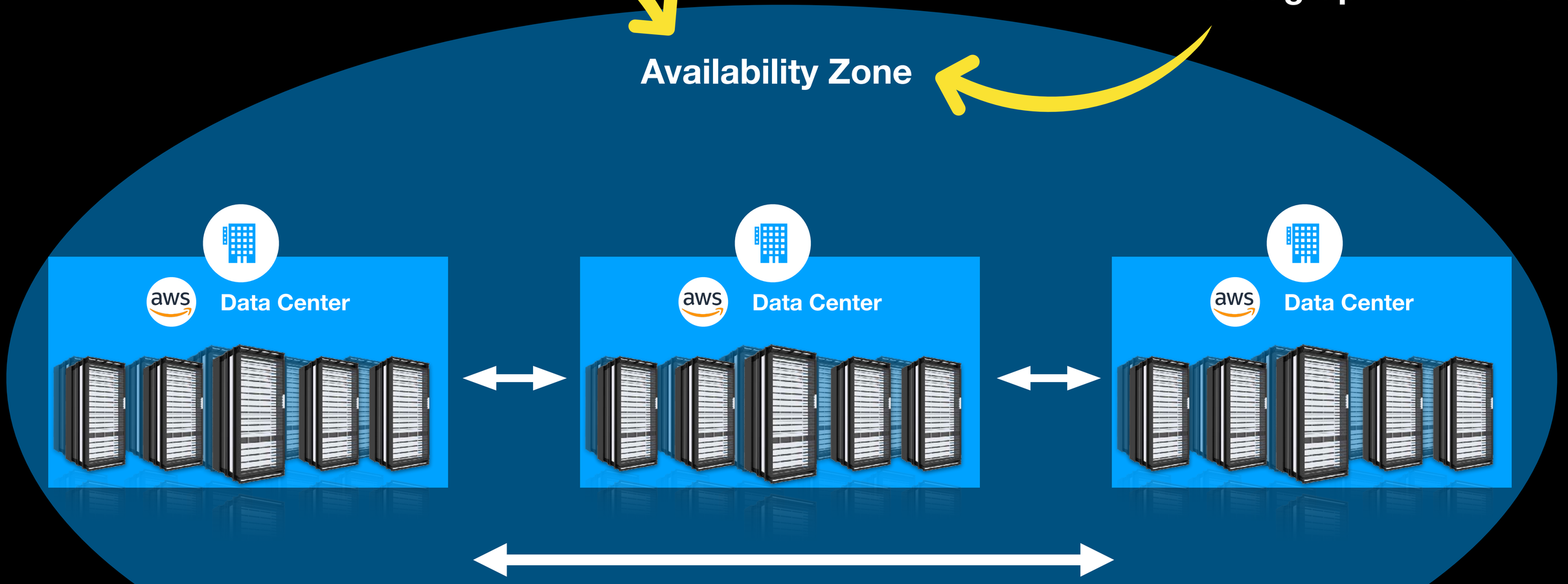
Edge Networks

Edge Networks

Region

Improves the “Availability” of your systems

Literally a Geographic “Zone”

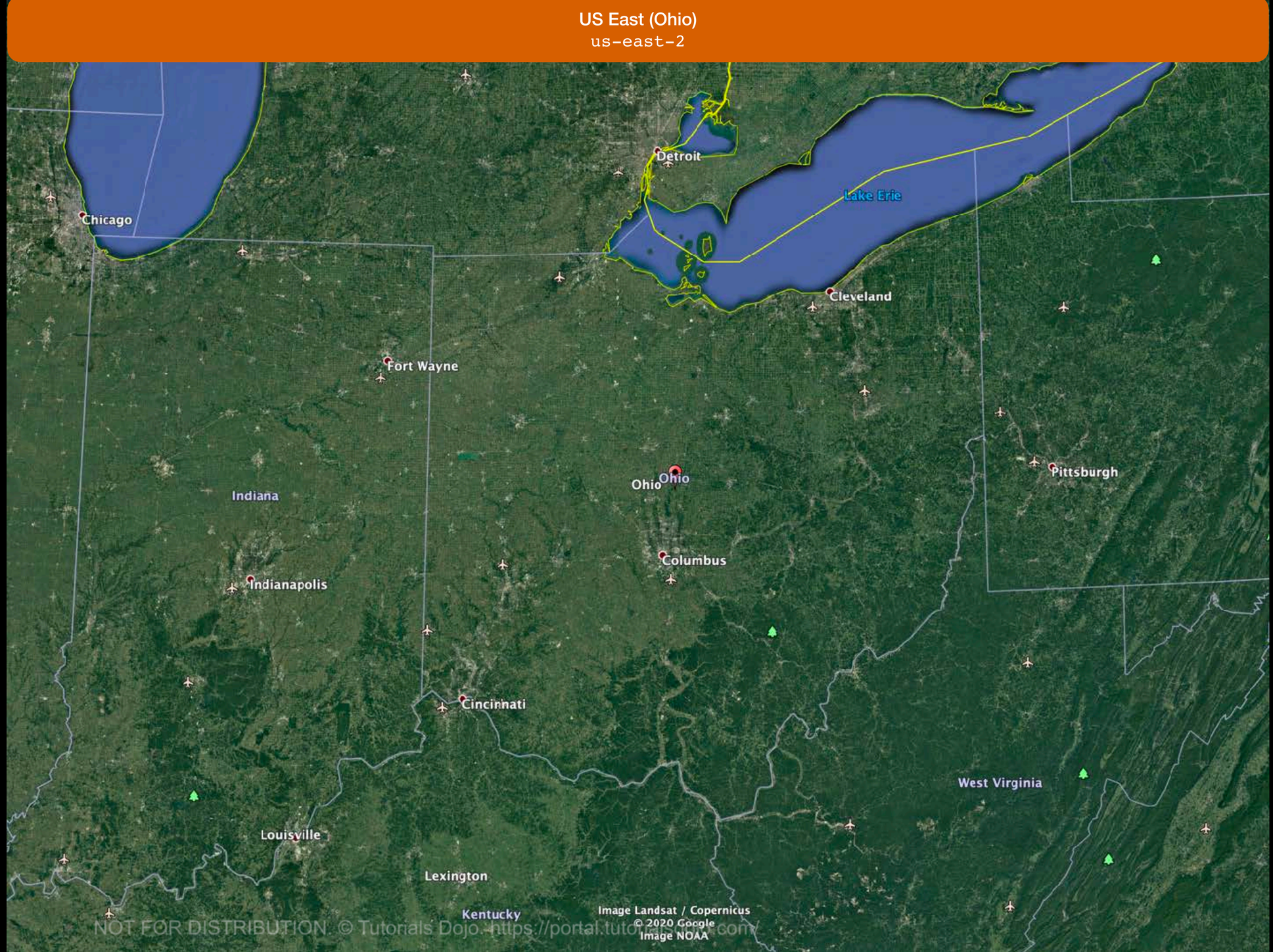


Edge Networks

Region

US East (Ohio)
us-east-2

Availability Zone



Edge Networks

Region

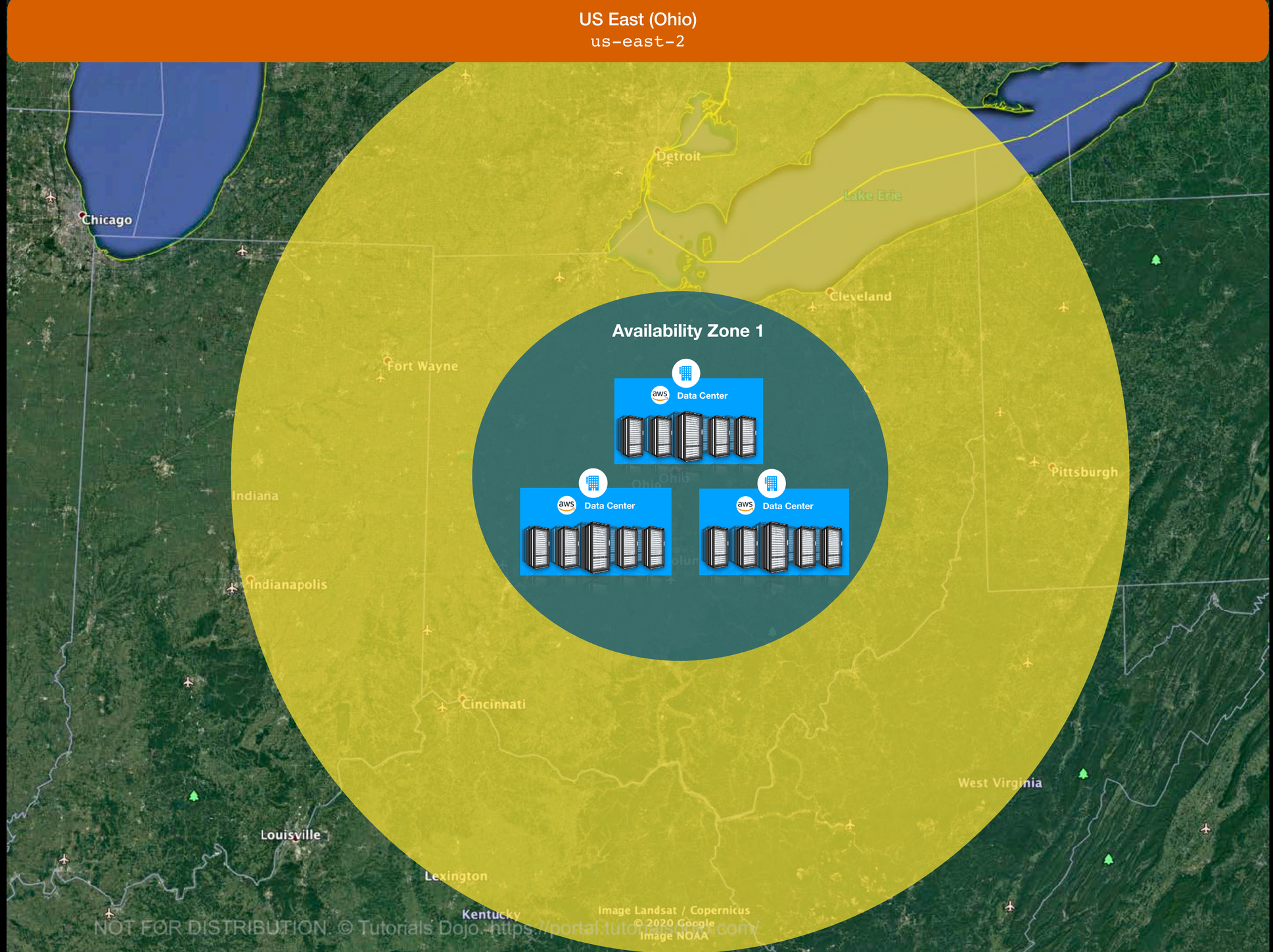
US East (Ohio)
us-east-2

Availability Zone

Edge Networks

Region

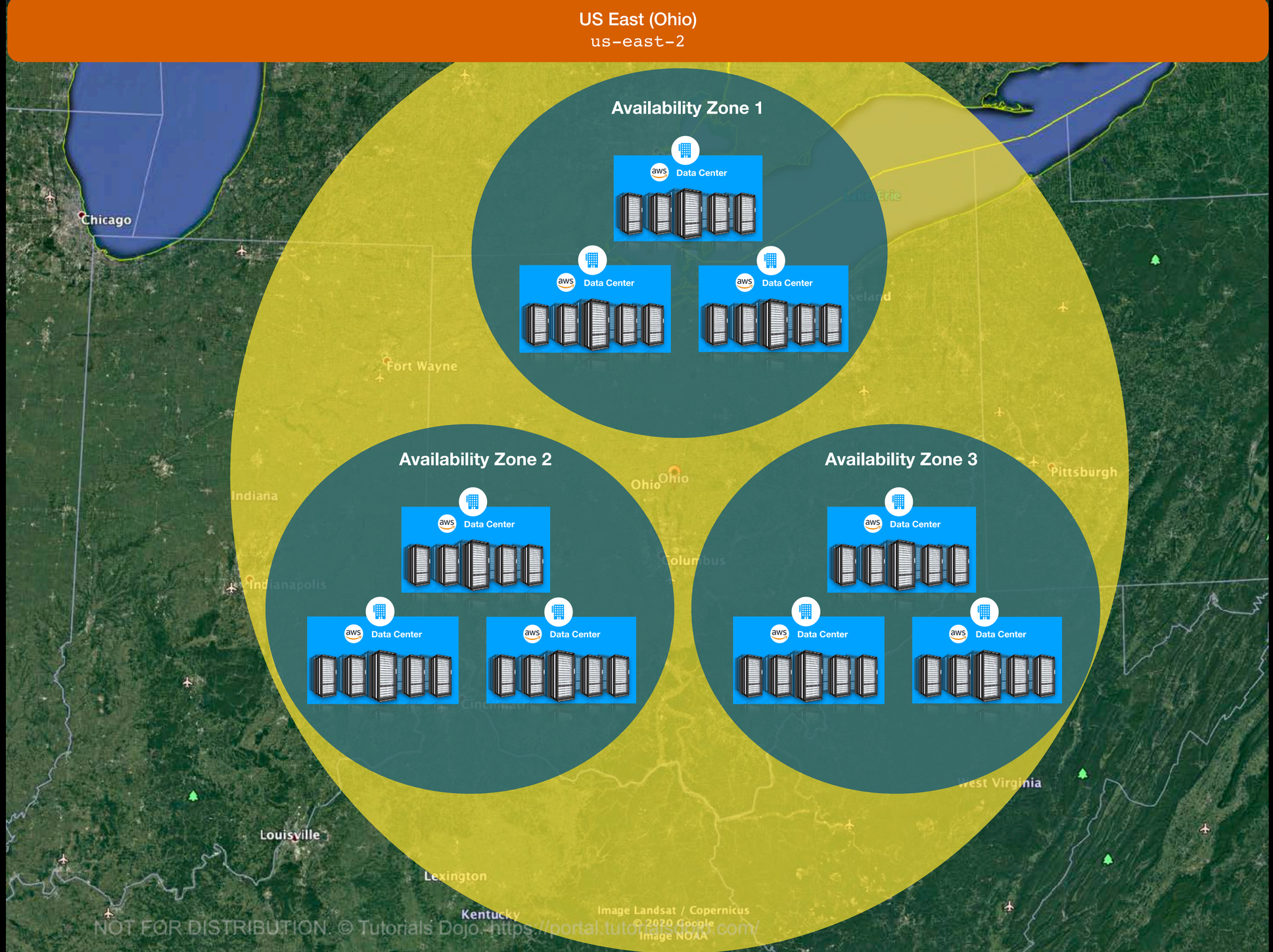
Availability Zone



Edge Networks

Region

Availability Zone

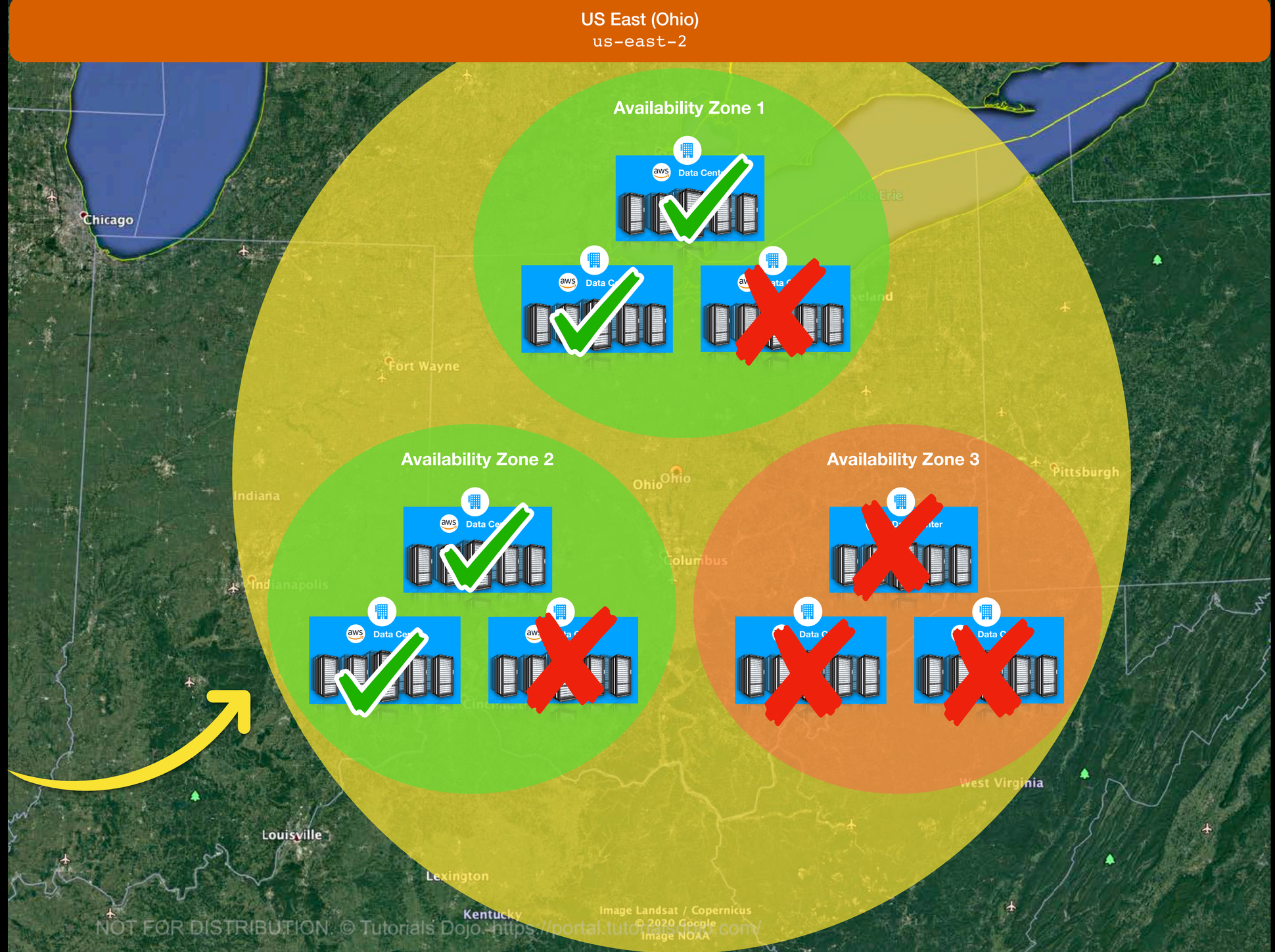


Edge Networks

Region

Availability Zone

Your system will still run even if one or more data centers encountered an outage

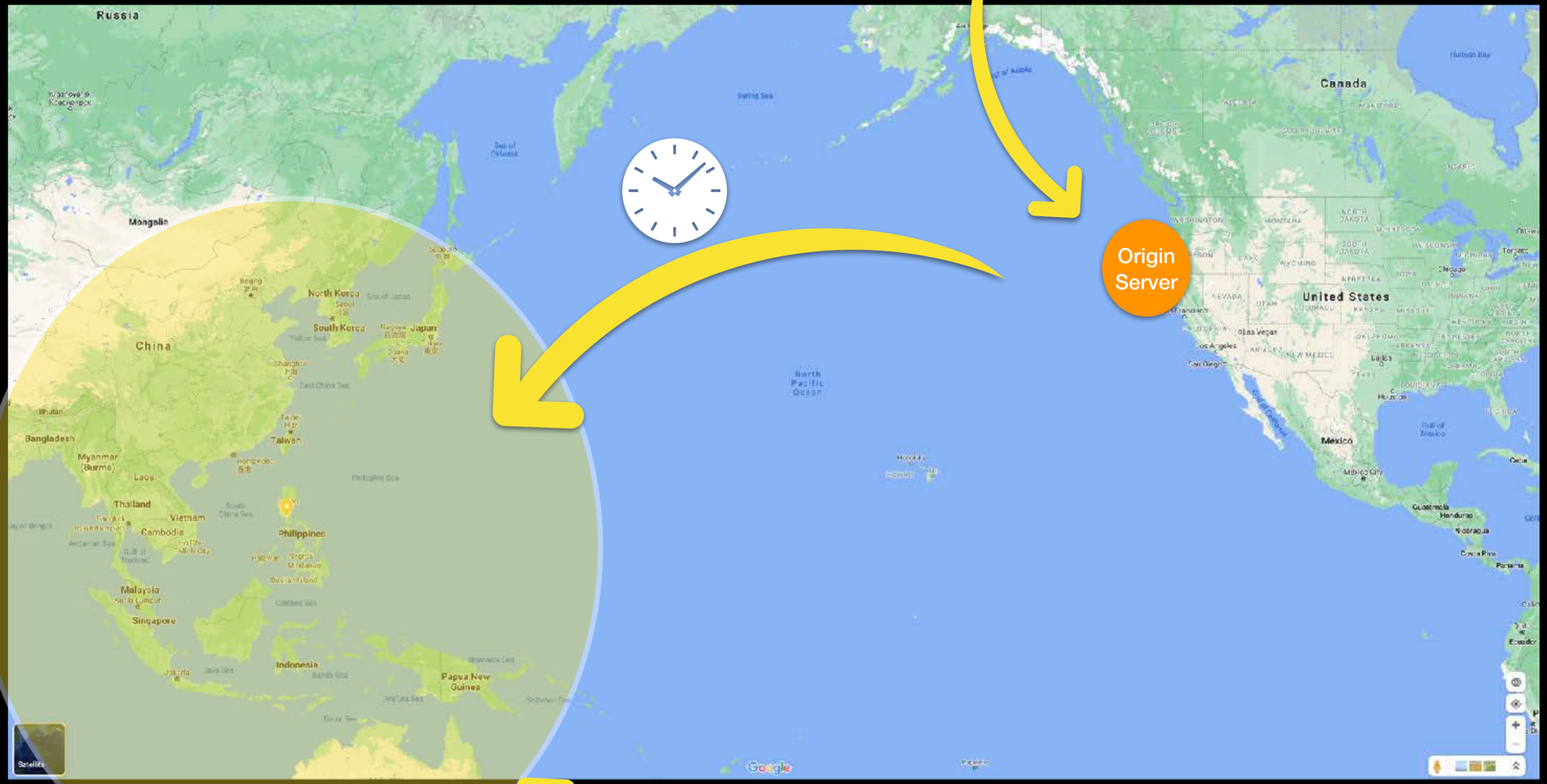


Edge Networks

PoP Point of Presence / Edge Location

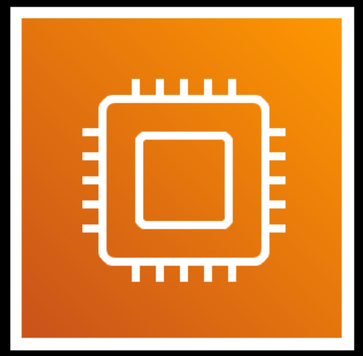
Region

Availability Zone



AWS Services Overview

…and many more!



Host
Web Apps

Run Real-Time
Data Analytics

Develop
Mobile Apps

Store Data
for Backup

Analytics

Application Integration

AWS Cost Management

Blockchain

Business Applications

Compute

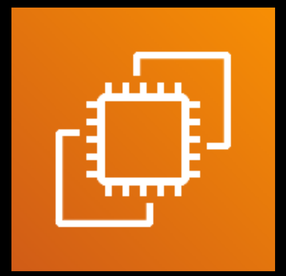
Containers

Customer Engagement

PER CATEGORY

- Analytics
- Application Integration
- AWS Cost Management
- Blockchain
- Business Applications
- Compute
- Containers
- Customer Engagement
- Database
- Developer Tools
- End User Computing
- Front-End Web & Mobile
- Game Tech
- Internet of Things
- Machine Learning
- Management & Governance
- Media Services
- Migration & Transfer
- Networking & Content Delivery
- Quantum Technologies
- Robotics
- Satellite
- Security, Identity & Compliance
- Serverless

COMPUTE SERVICES

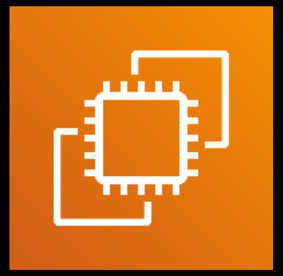


Amazon EC2

AWS Lambda

AWS Outposts

Amazon Lightsail



Amazon EC2

Amazon Elastic **Compute** Cloud

Amazon S3

Amazon Simple **Storage** Service

Amazon RDS

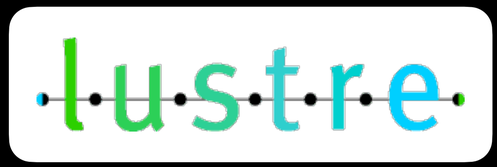
Amazon **Relational Database** Service

Fully Managed *By:* aws

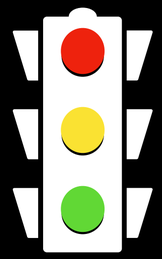
Open Source Technology

Amazon Elastic **Kubernetes** Service (EKS)

Amazon FSx for **Lustre** (FSx)



Amazon **Elasticsearch** Service



Routes Traffic

Amazon Route 53

What's the meaning of this number?

PORT

The number **53** is the TCP and UDP **Port Number** used for the Domain Name System (**DNS**) protocol transport


Featured Services


Analytics


Application Integration


AWS Cost Management

Blockchain


Business Applications


Compute


Containers


Customer Engagement


Database


Developer Tools


End User Computing


Front-End Web & Mobile


Game Tech


Internet of Things


Machine Learning


Management & Governance


Media Services


Migration & Transfer


Networking & Content Delivery


Quantum Technologies


Robotics

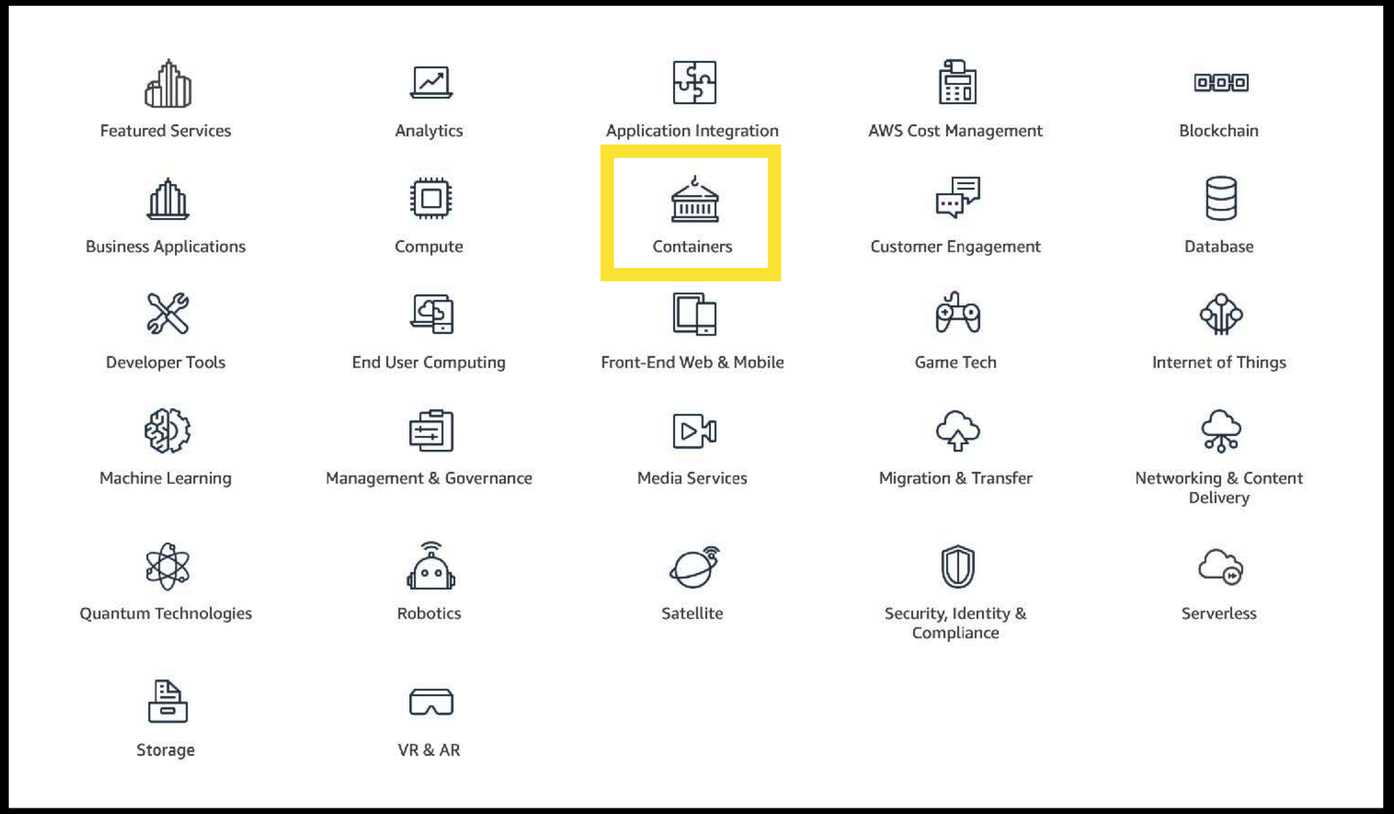

Satellite


Security, Identity & Compliance

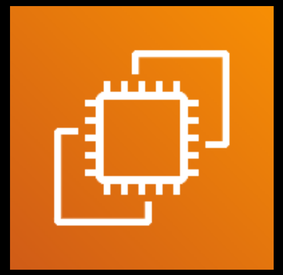

Serverless


Storage

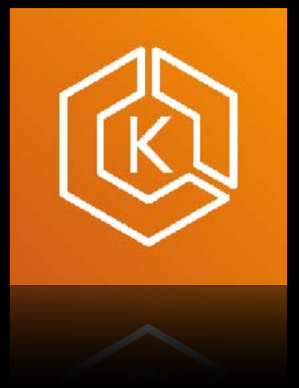

VR & AR



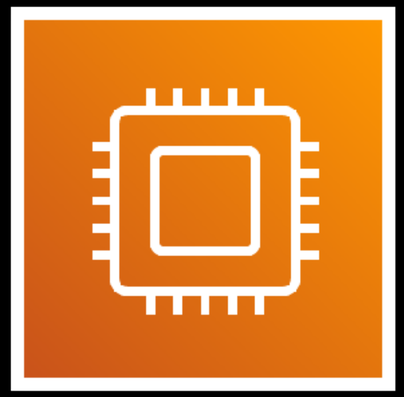
Amazon Elastic Container Service



Amazon EC2



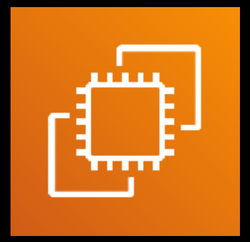
Amazon Elastic Kubernetes Service



AWS Compute Services Overview

AWS Compute Services

Virtual Machines



Amazon EC2

Amazon LightSail

AWS Outposts

Serverless

AWS Lambda

AWS Fargate

Orchestration

AWS Elastic Beanstalk

AWS Batch

Container

Amazon EKS

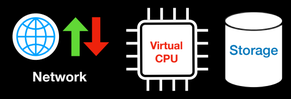
Amazon ECS

Virtual Machines

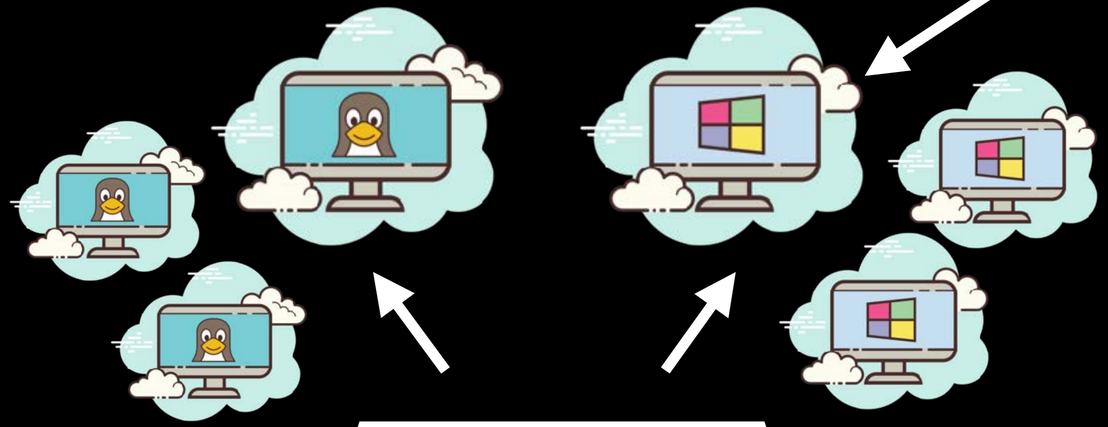
Windows

Used by MULTIPLE Tenants / Customers

Used by a SINGLE Customer



Instance



DEFAULT VIRTUALIZATION

CUSTOM VIRTUALIZATION

VIRTUALIZATION

Also called a
Virtual Machine Monitor
or a
Hypervisor

SHARED

DEDICATED

Serverless

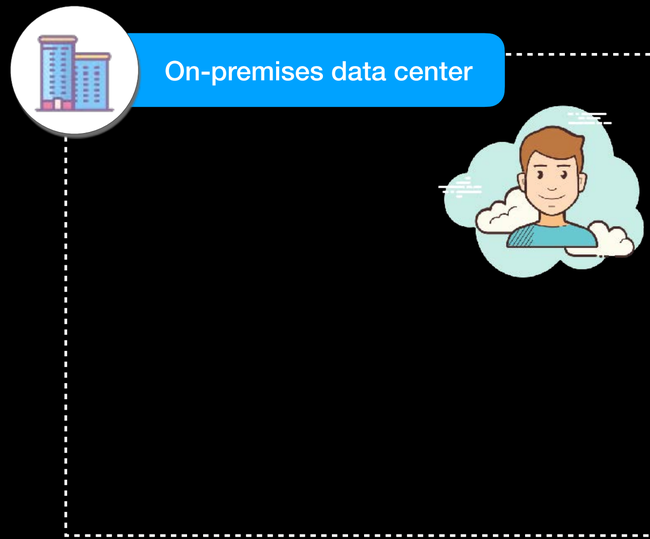
Fully Managed *By:* aws

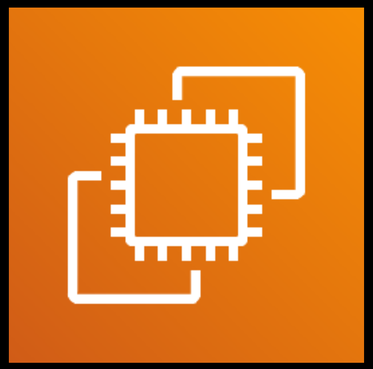
NO DIRECT Server access via:

SSH or RDP

Unlike Amazon EC2

Hybrid

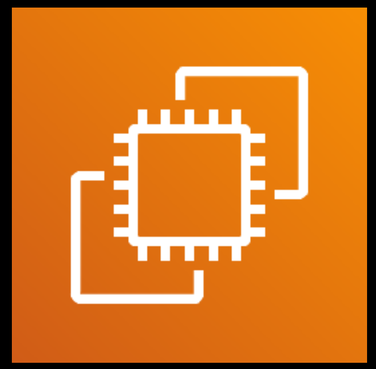




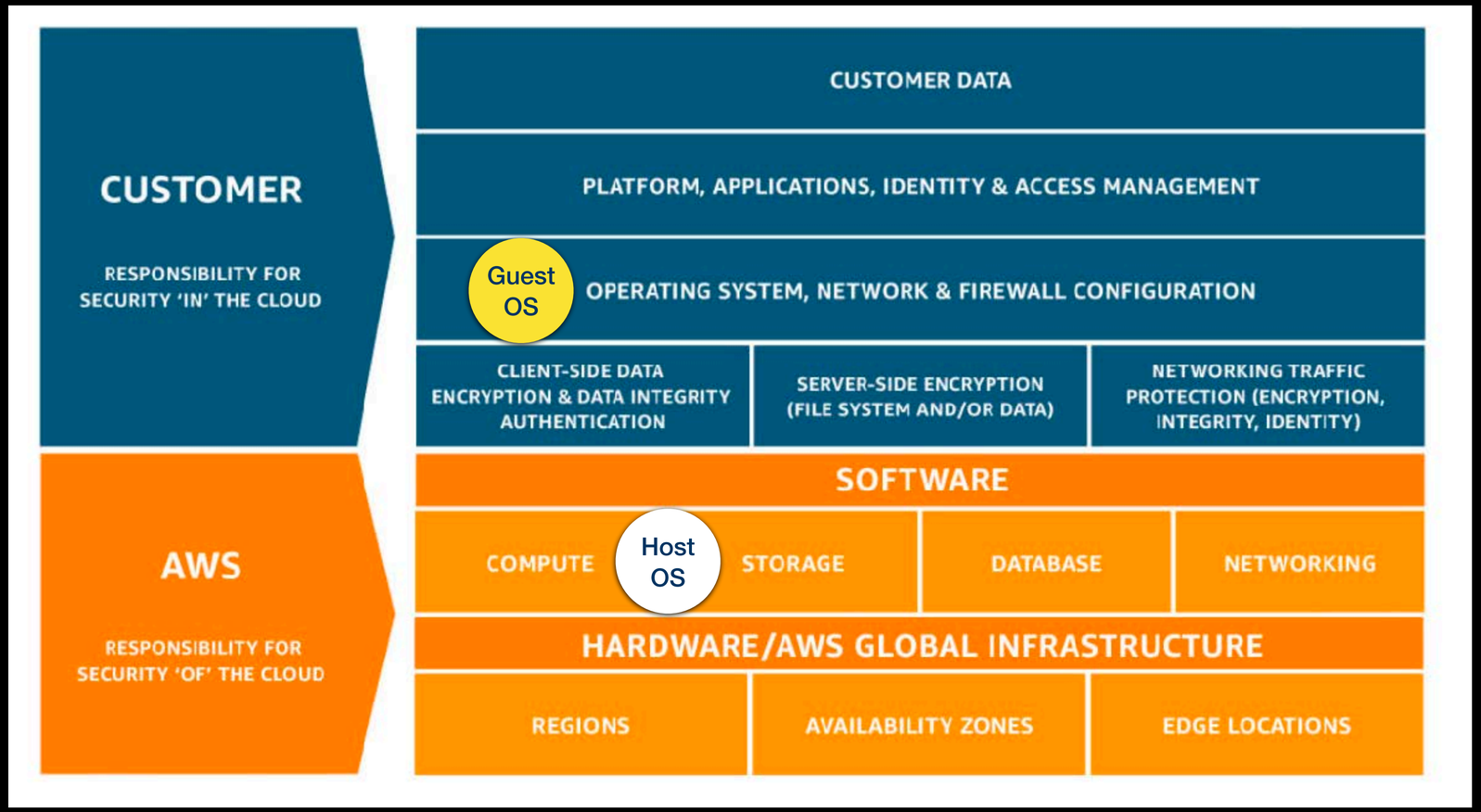
Amazon EC2

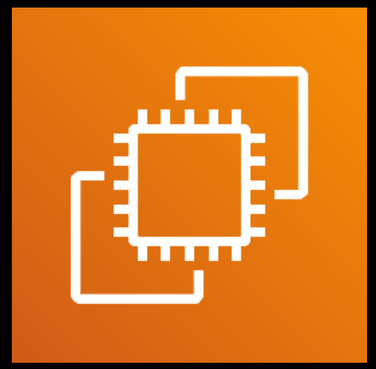
- A **computing service** that runs virtual servers in AWS
- Allows you to launch Windows, Linux or even MacOS virtual machines
- A type of an **Infrastructure as a Service (IaaS)**
- A basic building block for your cloud architecture
- Used by other AWS services as an underlying compute service

Shared Responsibility Model



Amazon EC2

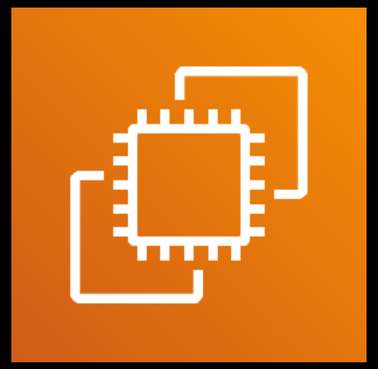




Amazon EC2

Elastic Compute Cloud

- **Flexible**
- **Customizable**
- **Scalable**



Amazon EC2

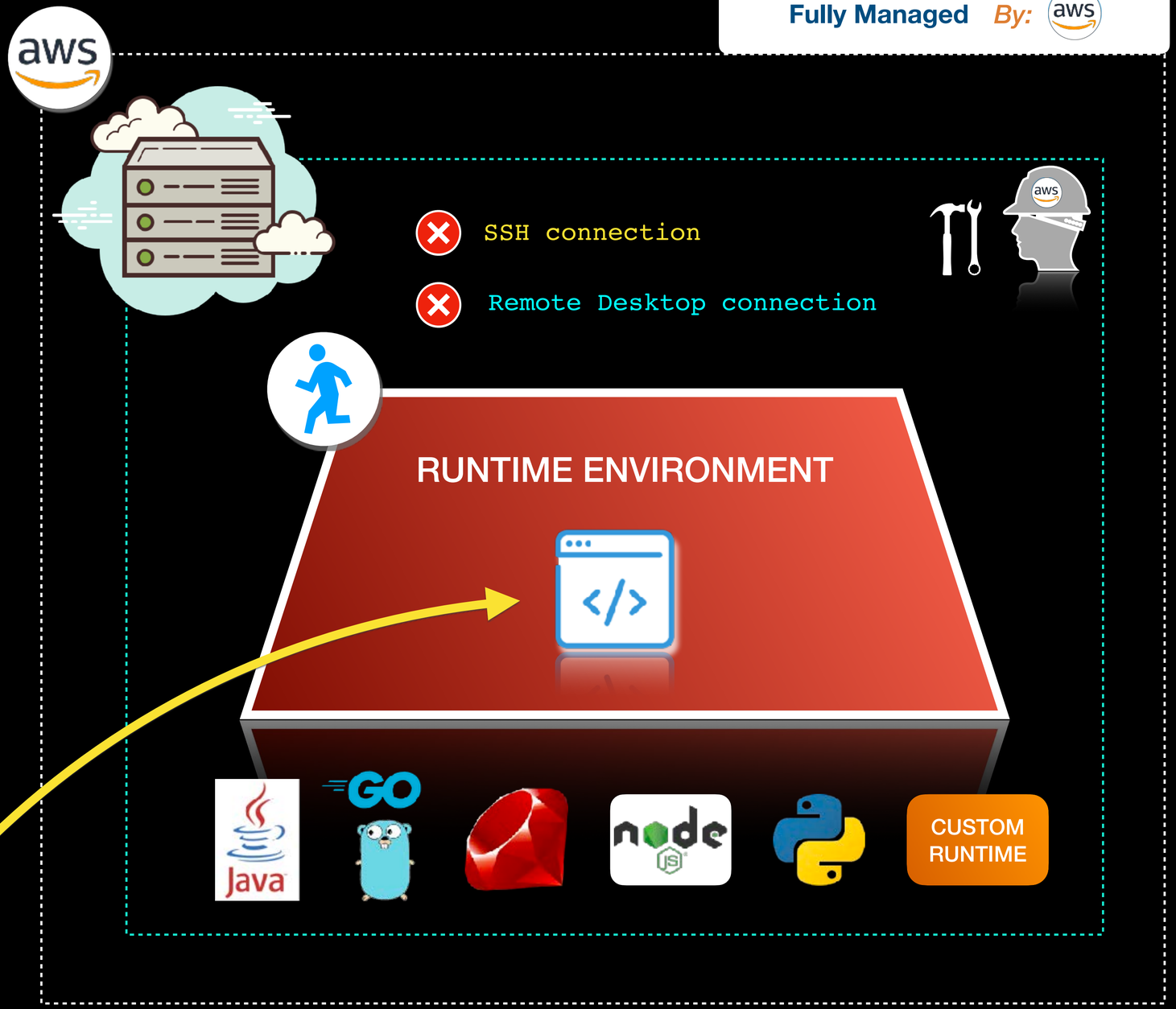
Elastic Compute Cloud

EC2

Serverless

AWS Lambda

Lambda function



Orchestration

AWS Batch

AWS Elastic Beanstalk

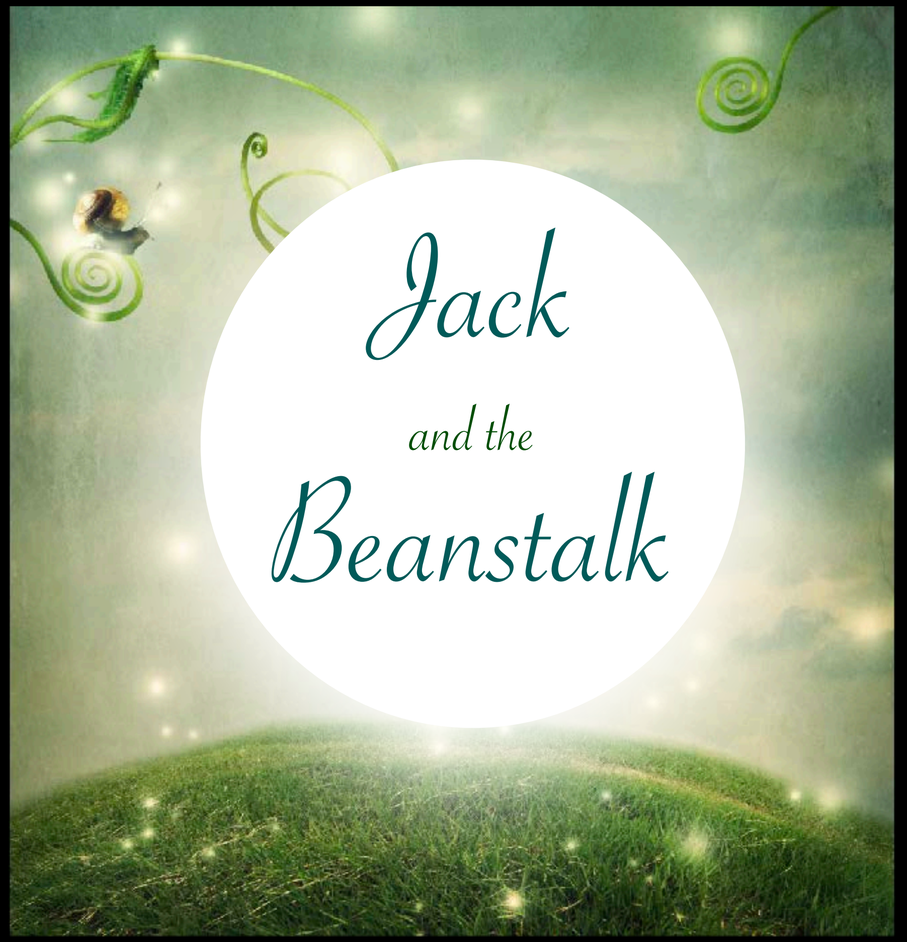
AWS Batch

- Enables you to run **batch** computing workloads
- Dynamically provisions the optimal quantity and type of compute resources, based on the volume and specific resource requirements.
- Does the planning, scheduling, and execution of your batch computing workloads **using Amazon EC2 instances.**

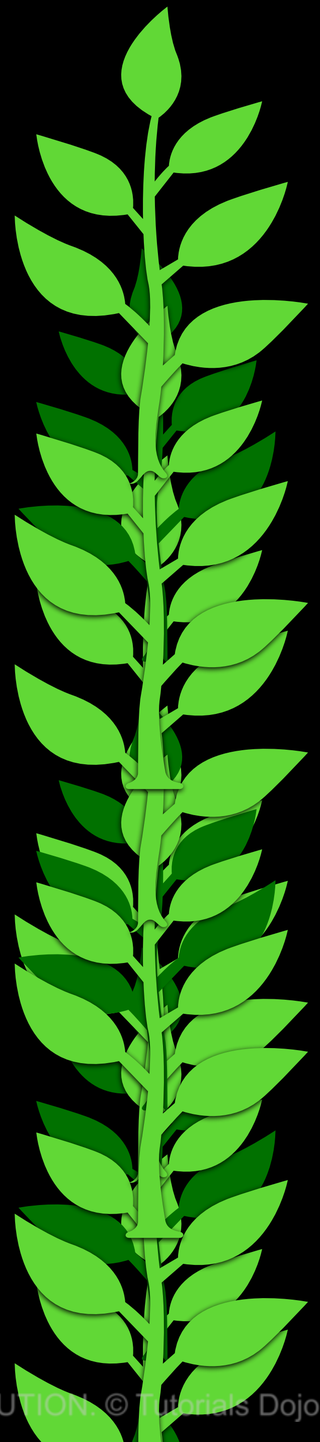
AWS Elastic Beanstalk

- Automates the deployment, management, scaling, and monitoring of your custom applications in AWS
- Just upload your application and it will automatically handle the common tasks to run your application.
- Handles capacity provisioning, load balancing, database management, auto-scaling, and health monitoring

AWS Elastic Beanstalk



AWS Elastic Beanstalk



Your Applications

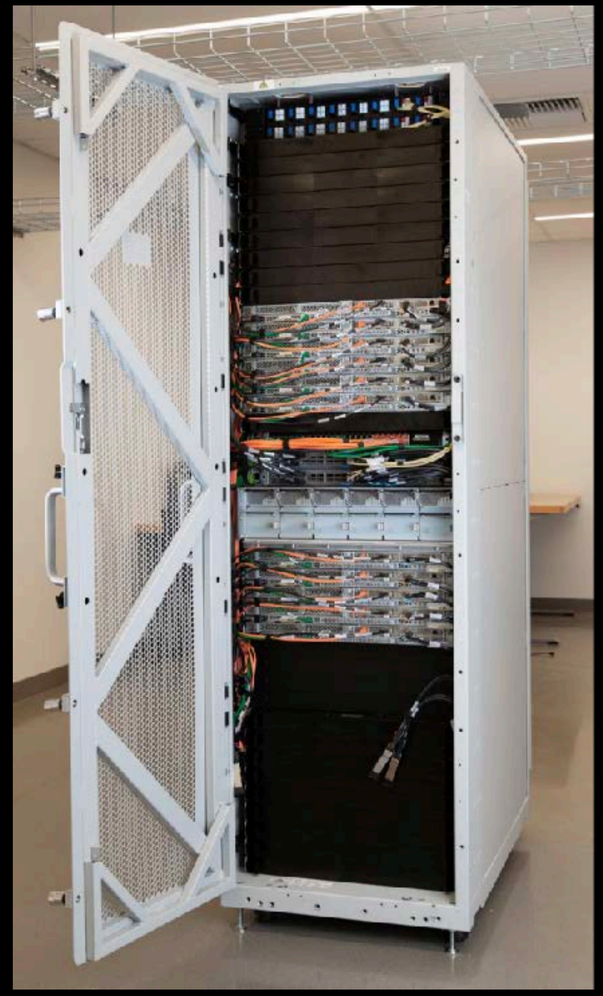
Amazon LightSail

- An easy-to-use **Virtual Private Server** (VPS)
- Has its **own web management console**
- Also provides other services like databases, load balancers, DNS records and many more.

AWS Outposts

- A hybrid service that allows you to run AWS services, like Amazon EC2, in your on-premises data center

AWS Outposts



AWS Container Services Overview

AWS Container Services

Amazon ECS

Amazon EKS

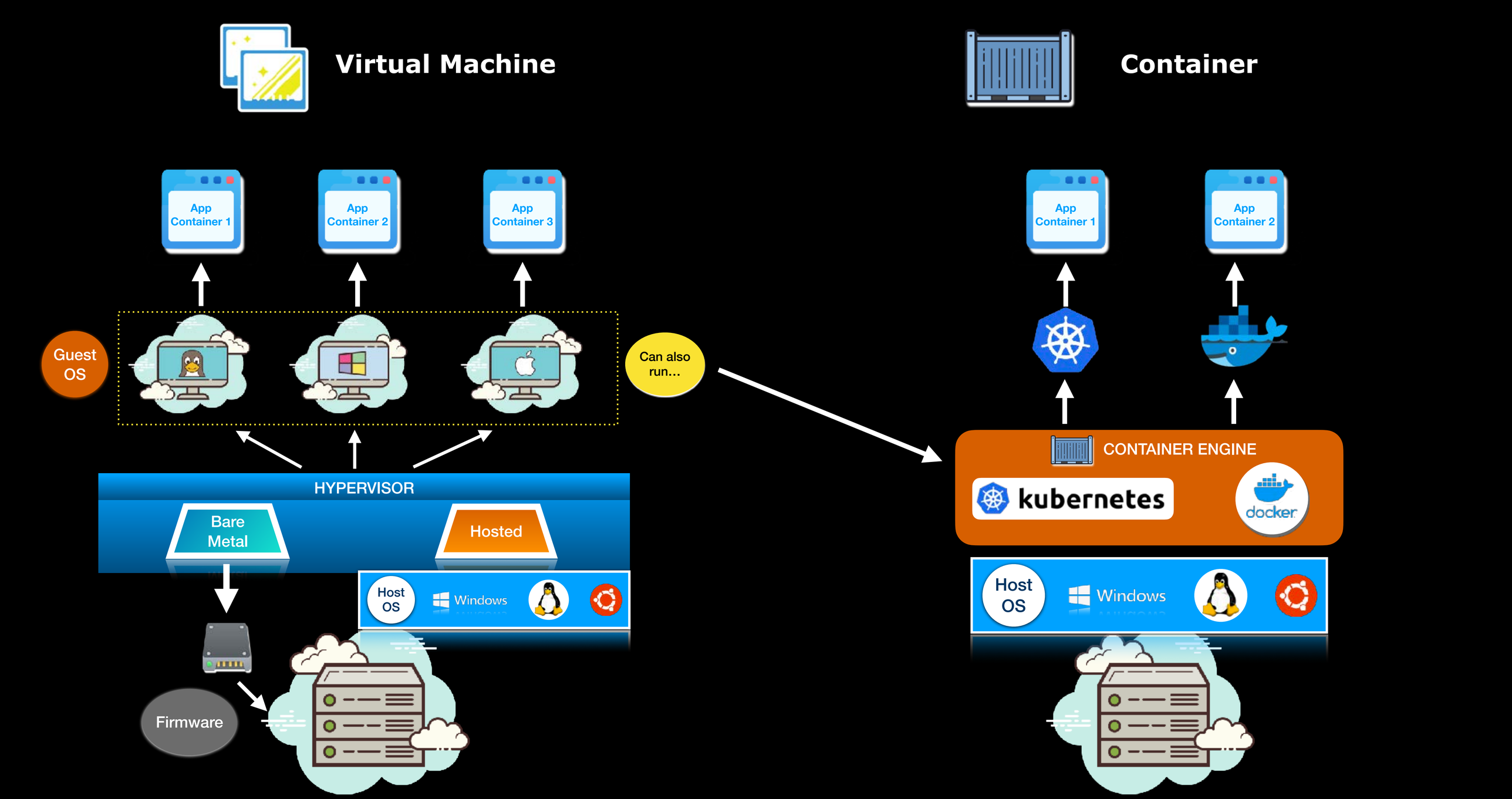
AWS Fargate

Amazon ECR

CLI Tools

AWS App2Container (A2C)

AWS Copilot



Amazon ECS

- Amazon **E**lastic **C**ontainer **S**ervice (Amazon ECS)
- A **container orchestration service** that supports Docker containers.
- Allows you to easily install, operate, and scale your cluster management infrastructure in AWS
- Containers are defined in a **task definition** which you use to run an **ECS task** or are **grouped** together as an **ECS service**

- Runs your ECS tasks using:

Amazon EC2

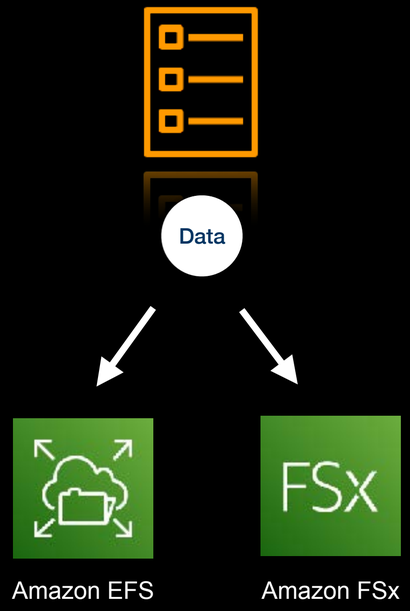
AWS Fargate

- An **IAM Role** can be attached to your ECS task in the `TaskRoleArn` property of your task definition for security control
- Store your Docker Images to:

Amazon ECR

Amazon ECS

Storage

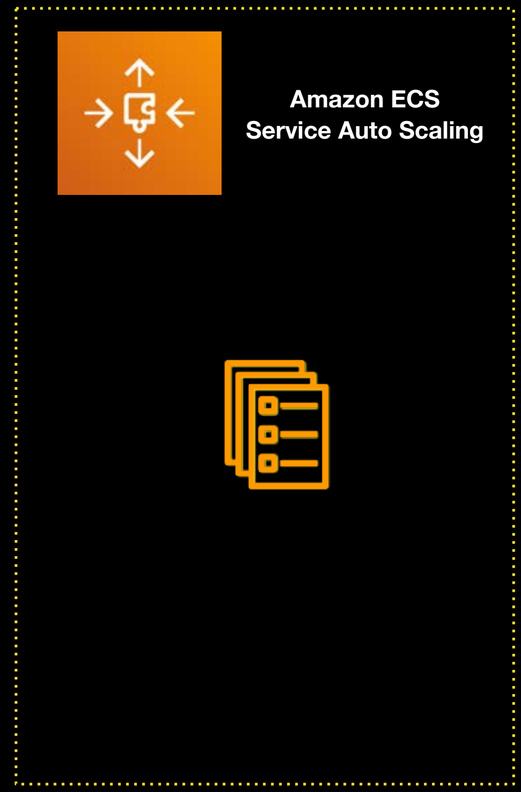


Amazon EFS

Amazon FSx

Integration

Scaling



Amazon EKS

- Amazon **E**lastic **K**ubernetes **S**ervice (Amazon EKS)
- A fully-managed **Kubernetes** service
- Portable, extensible, and open-source platform for managing containerized workloads and services
- Containers are grouped into **Pods** — the basic operational unit for Kubernetes.
- Launches and orchestrates a cluster of compute resources using:

Amazon EC2

AWS Fargate

- Considered as **Cloud-agnostic** as it allows you to easily move your workloads to your on-premises network or to other cloud service providers like Microsoft Azure, Google Cloud Platform (GCP) et cetera.

AWS Fargate

- A **serverless** compute engine
- Works on:

Amazon ECS

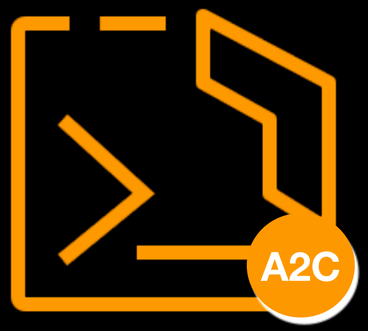
Amazon EKS

- Allows you to focus on building your applications without worrying about server provisioning, scaling, and management
- Provides a more **cost-effective** solution than a container running on **Amazon EC2 launch type**
- Runs each ECS task or Kubernetes pod in its own kernel.
- Provides the tasks and pods in their own isolated compute environment.

Amazon ECR

- Amazon **E**lastic **C**ontainer **R**egistry (Amazon ECR)
- A fully-managed **Docker container registry**
- Allows you to store, manage, and deploy Docker container images.
- Integrated with **Amazon ECS**

- **Stores your docker images** in a highly available and scalable architecture
- You can use IAM to provide resource-level control of each repository.



AWS App2Container
(A2C)

- A **command-line tool**
- Transforms .NET & Java applications to **containerized applications**
- Packages the application artifact and dependencies into container images.
- Configures the network ports and generates the ECS task and Kubernetes pod definitions.

AWS Copilot

- Also a **command-line tool**, just like AWS App2Container (A2C)
- Transforms .NET & Java applications to **containerized applications**
- Enables you to quickly launch and easily manage containerized applications on AWS
- Automates the deployment lifecycle of your containers

AWS Database Services Overview

Relational

A C I D

- Atomicity
- Consistency
- Isolation
- Durability

Amazon RDS

Amazon Aurora

Data warehouse

Amazon Redshift

NoSQL

Amazon DynamoDB

Amazon DocumentDB

In-Memory

Amazon ElastiCache

redis

emcached

Other Databases

Amazon Keyspaces

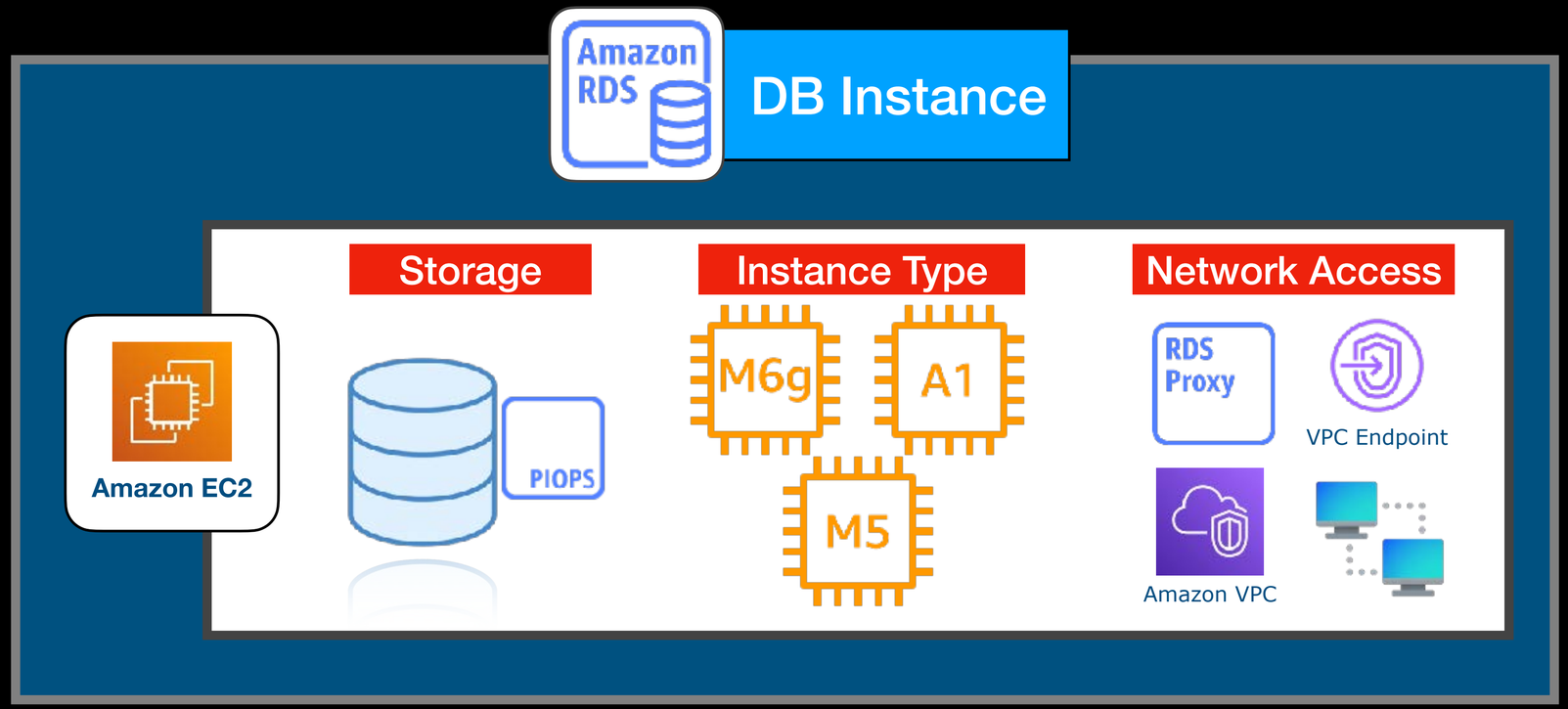
Amazon Neptune

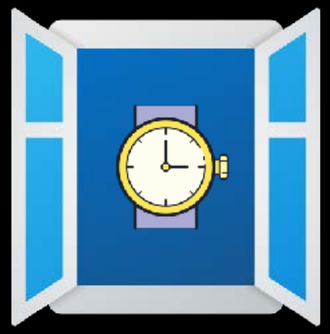
Amazon Timestream

Amazon Quantum Ledger

Amazon Relational Database Service
(Amazon RDS)

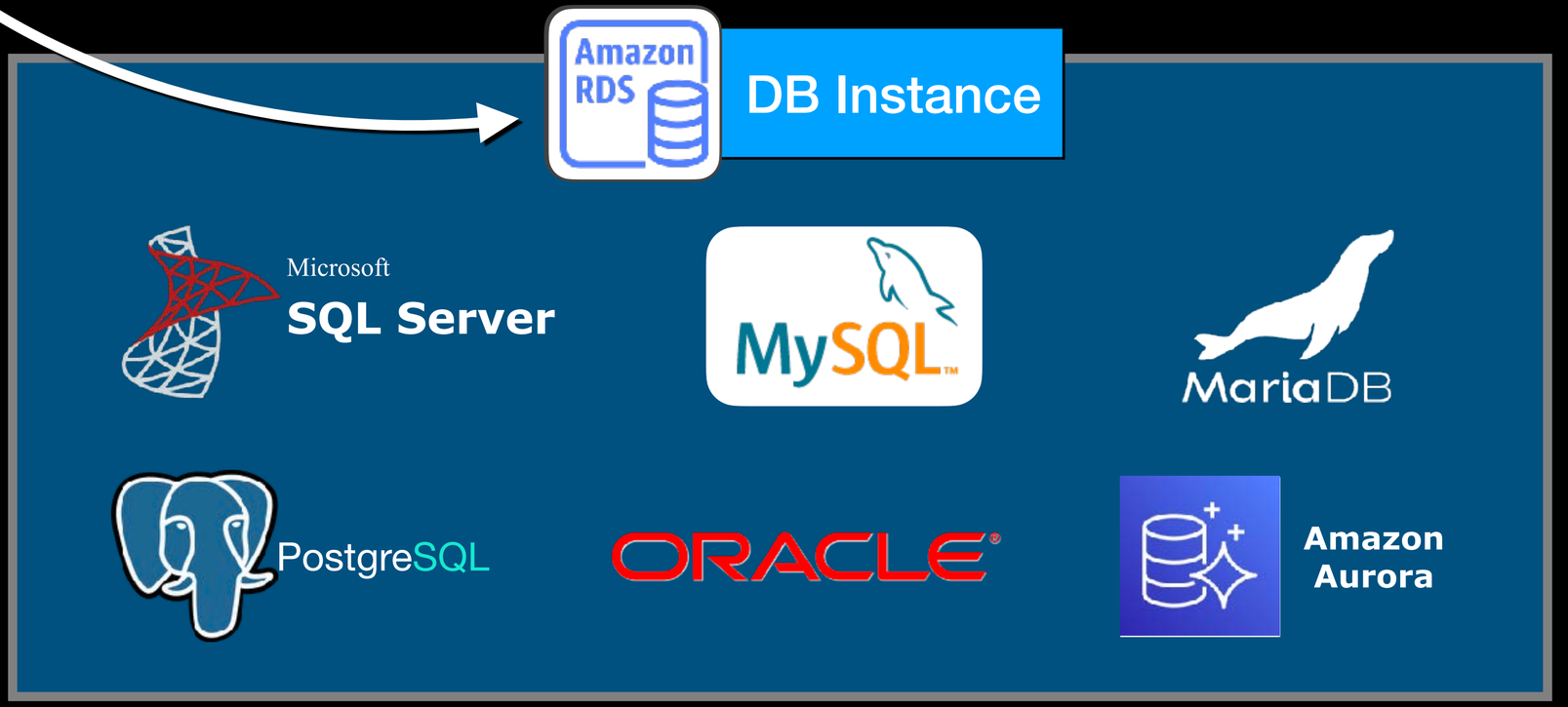
- A **relational** database that is managed by both you (limited access) and AWS.
- **The time-consuming tasks are handled by AWS** — such as hardware provisioning, patching, backups, and maintenance.
- You can **configure the underlying EC2 instance** used by Amazon RDS

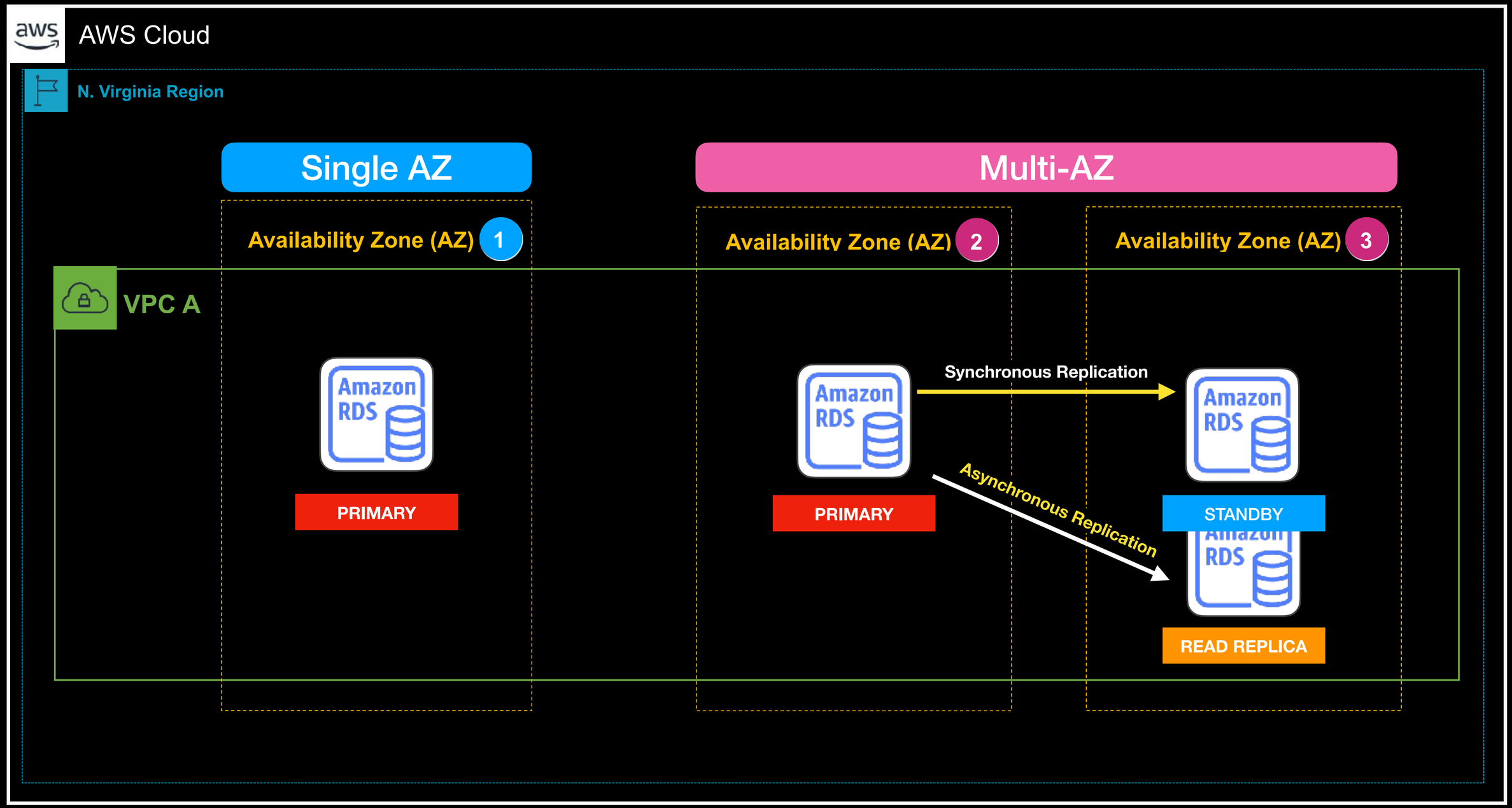


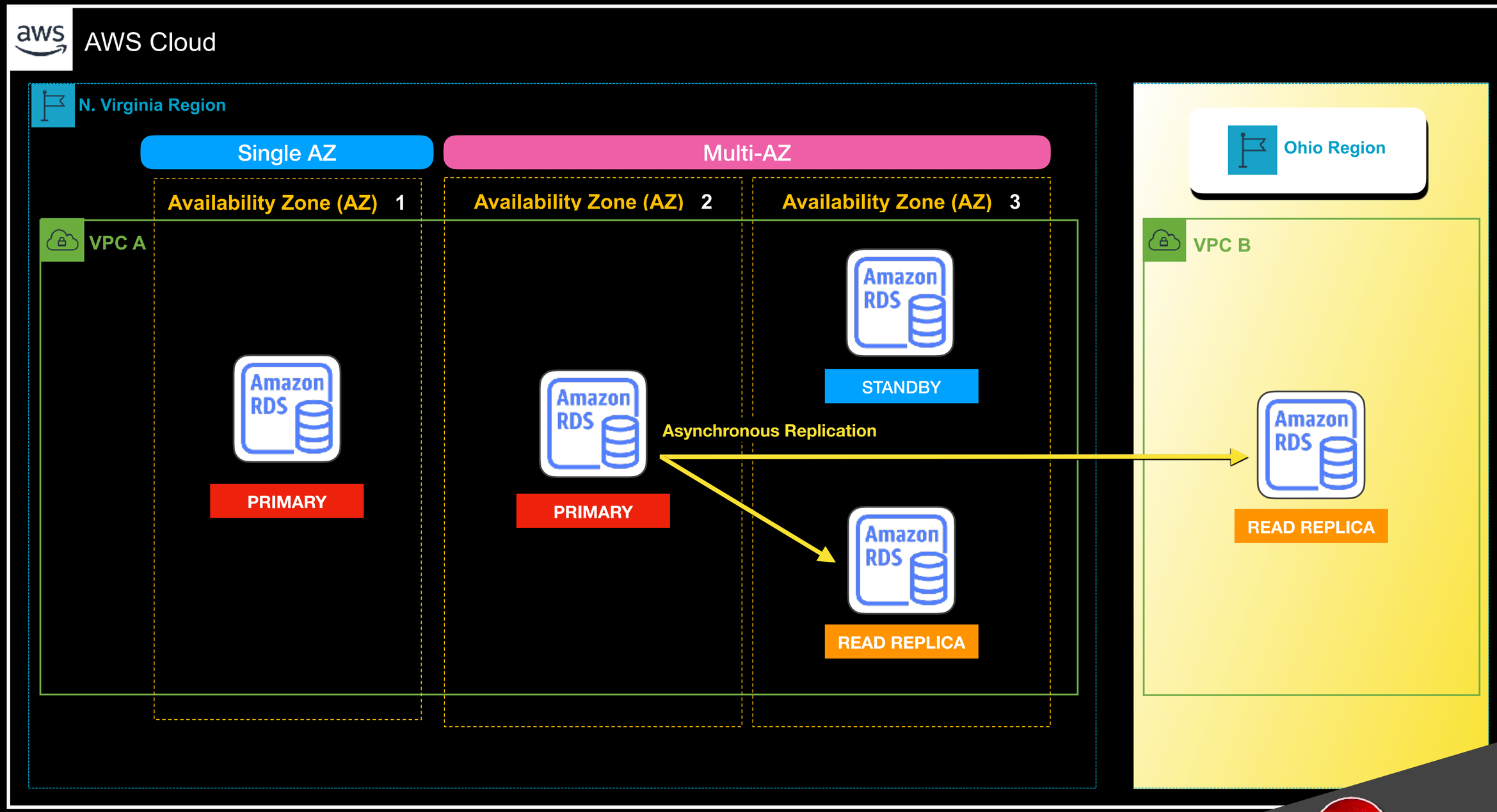


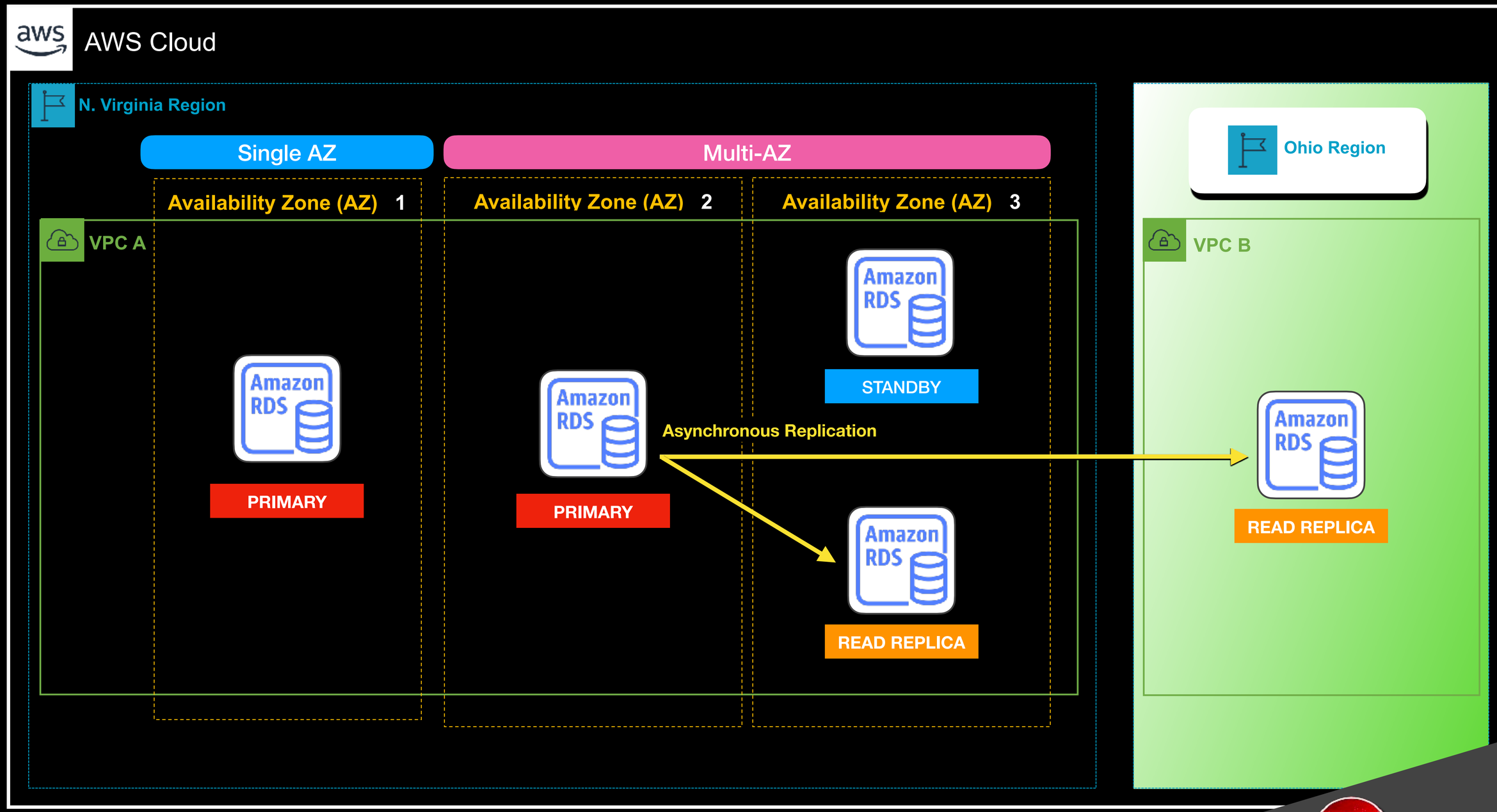
- You decide the **actual time** for the patches to be applied on its **maintenance window**
- Can run various types of **database engines:**

Amazon Relational Database Service
(Amazon RDS)









Amazon Aurora

- A type of a **database engine** (that you can run on Amazon RDS) and a **fully managed database** service.
- Compatible with: MySQL PostgreSQL
- Scales automatically, performs faster, and costs lower than other databases
- Can **automatically grow its data storage**
- Deployed as a **database cluster** that consists of:

- **Similar to Multi-AZ Deployments** in Amazon RDS
- A cluster has a **single-master configuration** where applications can only write data to a single, master DB instance.
- In a **multi-master cluster**, all DB instances have read/write capability.

Amazon Aurora

Amazon Relational Database Service
(Amazon RDS)

- Suitable for applications that read or write **constantly changing data**, such as Online Transaction Processing applications or OLTP.

Data warehouse

Amazon Redshift

- A fully managed **data warehouse**
- Allows you to **analyze all your data using standard SQL** or through your existing Business Intelligence tools
- Optimized to **analyze relational data** coming from transactional systems, business applications, and other sources for fast SQL queries.
- Offers a **concurrency scaling** feature that supports virtually unlimited concurrent users and concurrent queries
- Has a feature called **Redshift Spectrum** that allows you to query and retrieve structured and semistructured data from files stored in:

Amazon S3

Amazon Redshift

- Primarily used for **Online Analytical Processing or OLAP** applications like data reporting and analytics.

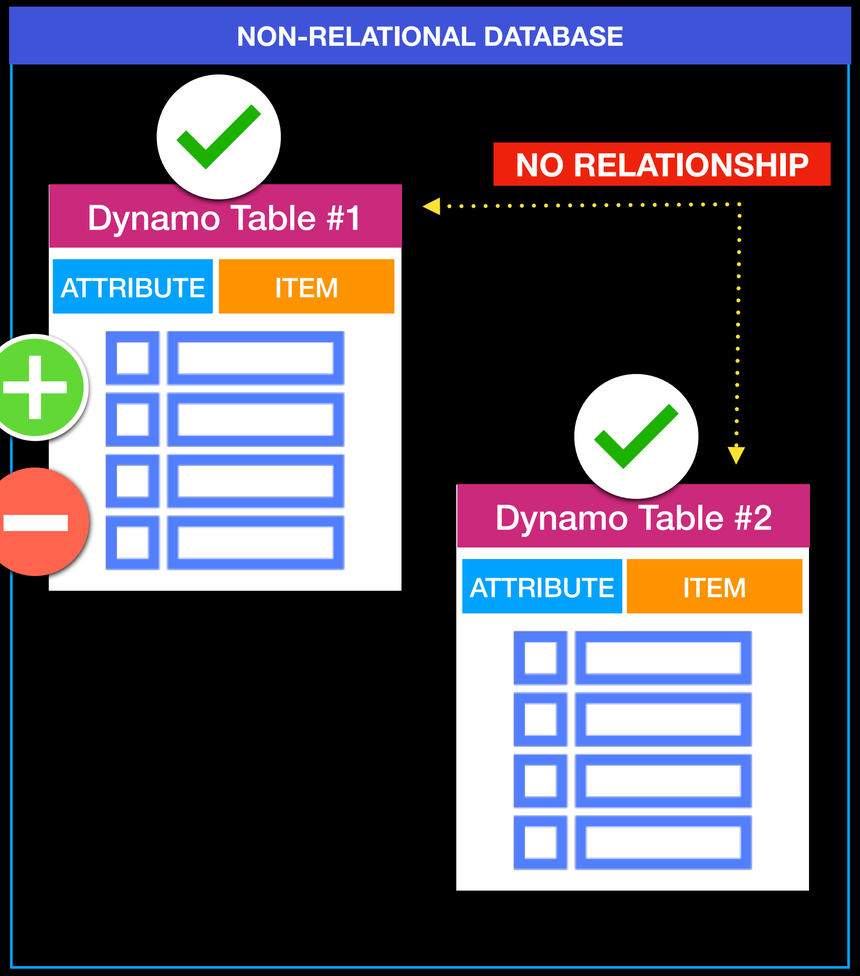
NoSQL Databases

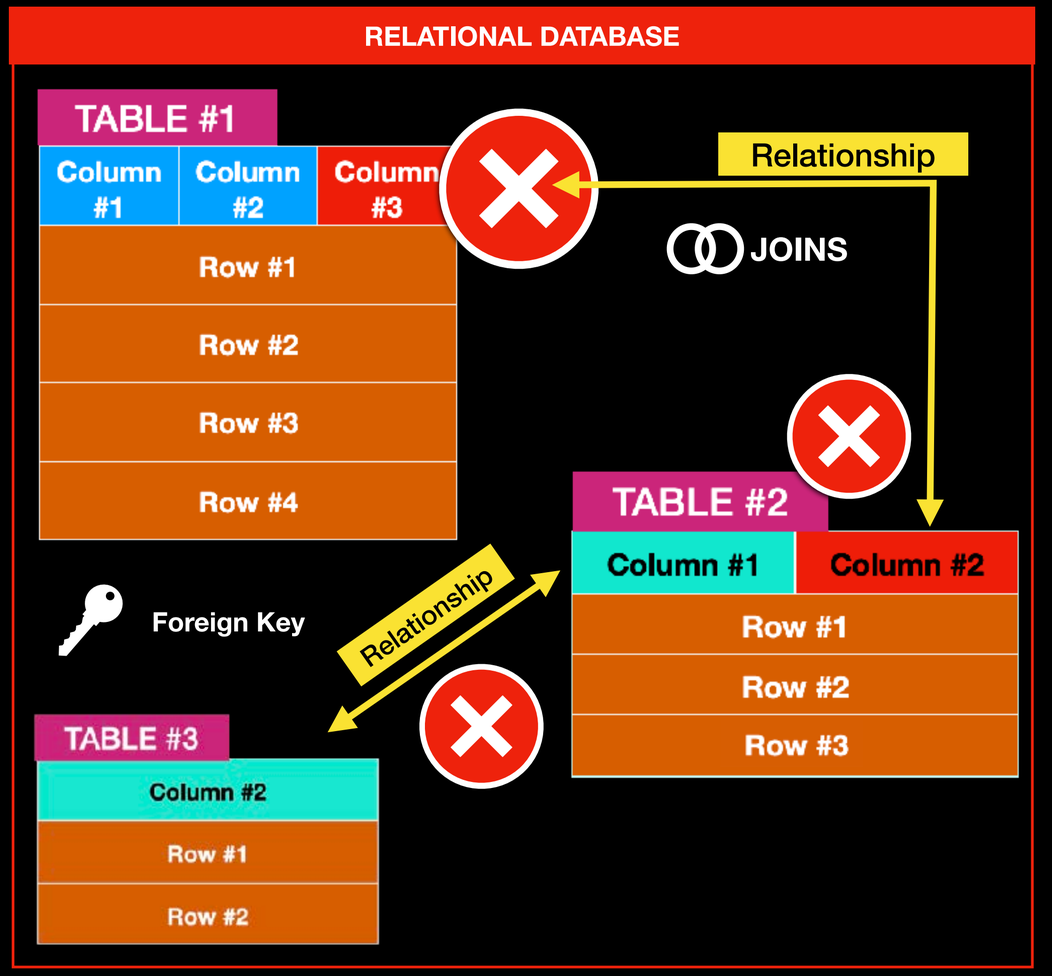
Amazon DynamoDB

Amazon DocumentDB

- A fully managed **NoSQL database** service
- A non-relational database that **does not have a rigid schema** or extensive table relationships.

Amazon DynamoDB





DOCUMENT

```
{
  id: 1898,
  gid: “tutorialsdojo1898”,
  firstName: "Jose",
  lastName: "Rizal",
  profile: {
    nationality: “Filipino,
    country: “Philippines,
    birthPlace: “Laguna"
    }
}
```

- A fast, scalable, highly available **MongoDB-compatible database** service.

- A **document**-oriented database program
- Cross-platform, NoSQL database

- Each **document** contains fields and values in **JSON format** with no rigid schema enforced

Amazon DocumentDB

DOCUMENT DATABASE

COLLECTION

Document #1	Field #1	Value #1
	Field #2	Value #2
	Field #3	Value #3
Document #2	Field #1	Value #1
	Field #2	Value #2
	Field #3	Value #3
Document #3	Field #1	Value #1
	Field #2	Value #2
	Field #3	Value #3

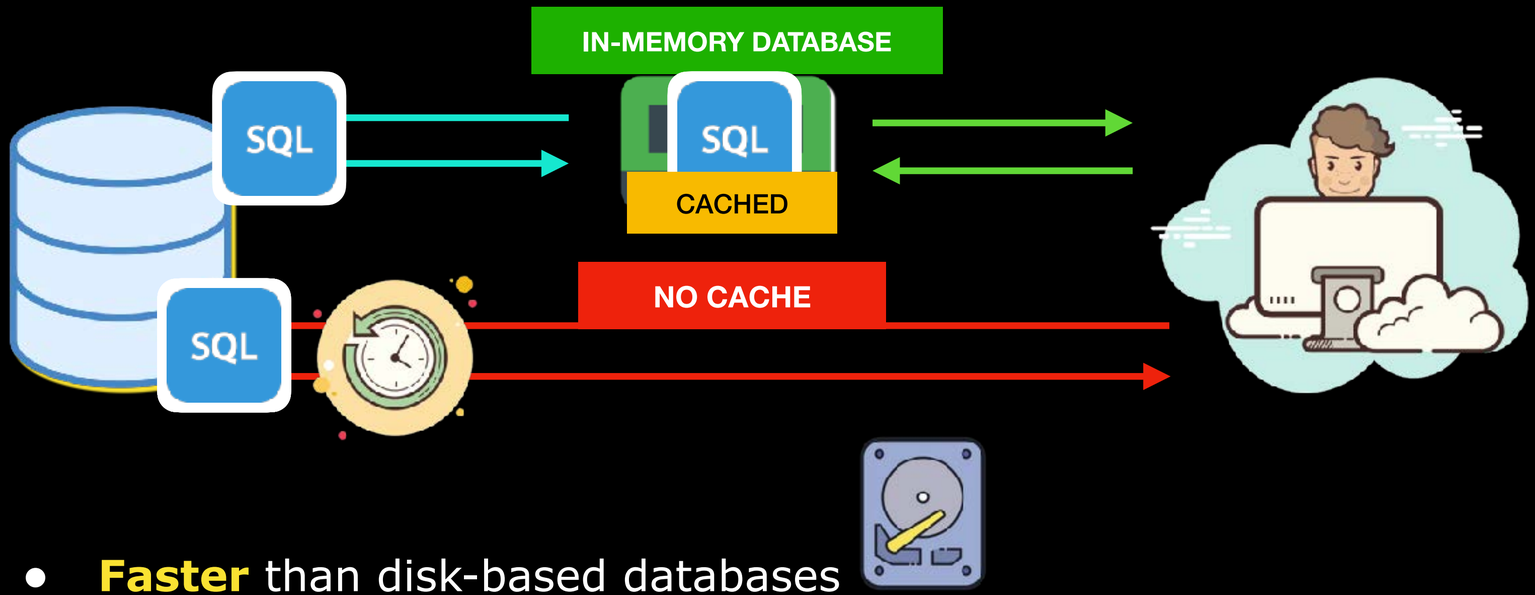
RELATIONAL DATABASE

TABLE #1

Column #1	Column #2	Column #3
Row #1		
Row #2		
Row #3		
Row #4		

Amazon ElastiCache

- A **caching service**
- Allows you to set up, run, and scale **open-source in-memory databases** like: emcached redis



- **Faster** than disk-based databases
- Useful for **database caching** that eliminates unnecessary frequent calls to the database just to **return identical datasets**
- Useful for real-time analytics, distributed session management, geospatial services, and many more

Amazon ElastiCache

emcached

Sub-millisecond latency

Data Partitioning

Can be integrated to your apps with minimal code change

emcached

Amazon ElastiCache for Memcached

- Based on the open-source **Memcached** in-memory data store.
- Suitable for building a simple, scalable caching layer for your data-intensive apps.
- **Multithreaded** — it can utilize multiple processing cores.
- **Lacks data replication capability**
- **Does not:**
  - Support Advanced Data Structures
  - Provide Highly Available Caching Layer

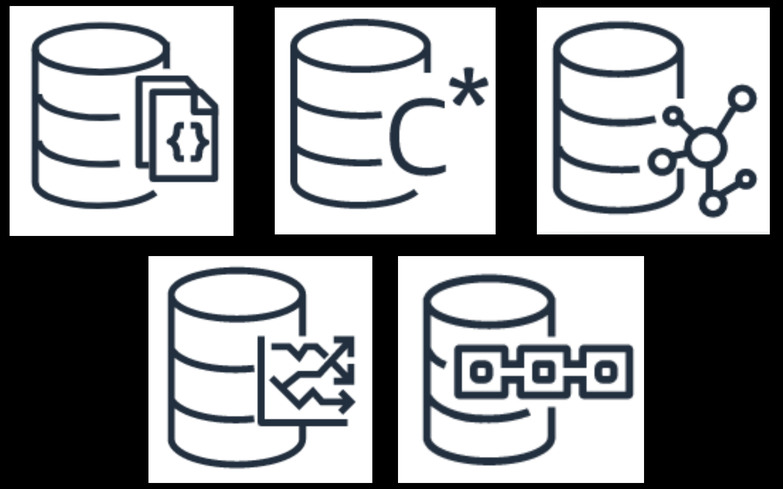
redis

stands for

REmote **DI**ctionary **S**erver

Amazon ElastiCache for Redis

- Based on the open-source **Redis** in-memory data store.
- Provides:
  - Advanced Data Structures
  - Pub/Sub messaging
  - Geospatial support
  - Point-in-Time Snapshot support
- Has a **replication feature** that provides high availability via data replication.
- You can enable the **Cluster Mode** in Redis to have multiple primary nodes and replicas across two or more Availability Zones.



Other Databases

Amazon KeySpaces

- A scalable, highly available, and managed **Apache Cassandra**–compatible database service

- An open-source, **wide column data store** that is designed to handle large amounts of data.

- **Run your Cassandra workloads** on AWS without having to provision, patch, or manage servers.

Amazon Neptune

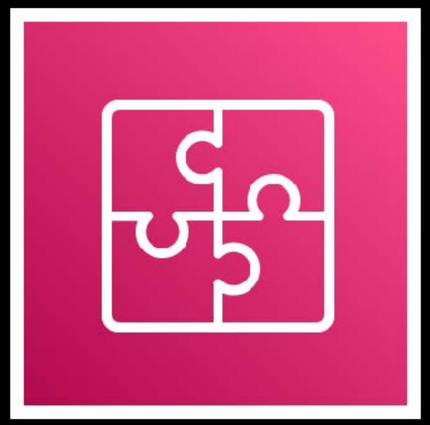
- A fast, reliable, fully-managed **graph database** service
- Makes it easy for you to build and run applications that work with **highly connected datasets**
- Allows you to **store billions of relationships** and query your data graphs with **milliseconds latency.**
- **Uses nodes to store data entities and edges** to store relationships between entities.

Amazon Timestream

- A fast, scalable, and **serverless time series database** service
- Primarily used **for Internet-of-Things and operational applications.**
- **Track the changes of your data**
- Can be used to track stock prices, temperature measurements, and the CPU utilization of an EC2 instance over a specific amount of time.

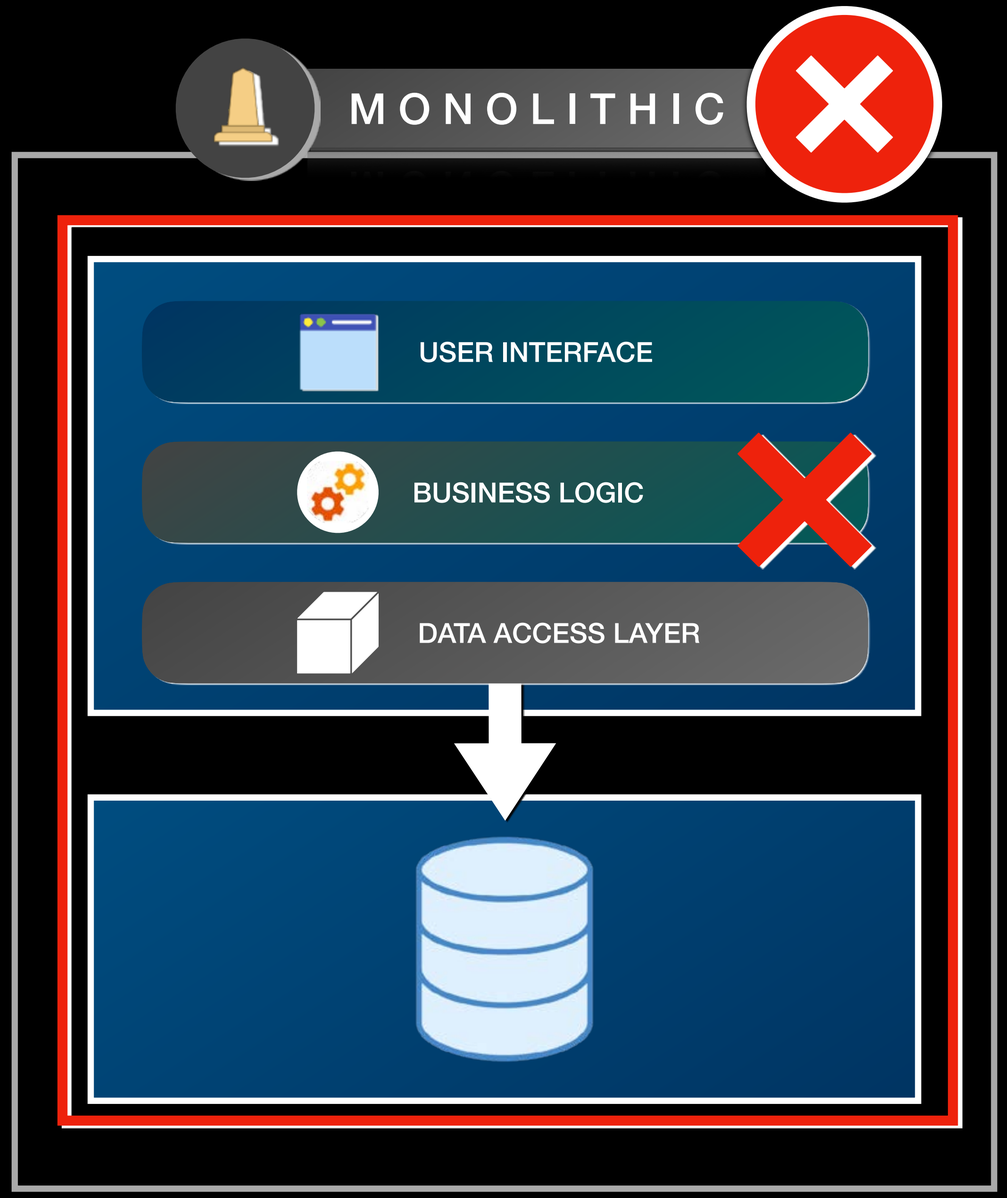
Amazon Quantum Ledger (Amazon QLDB)

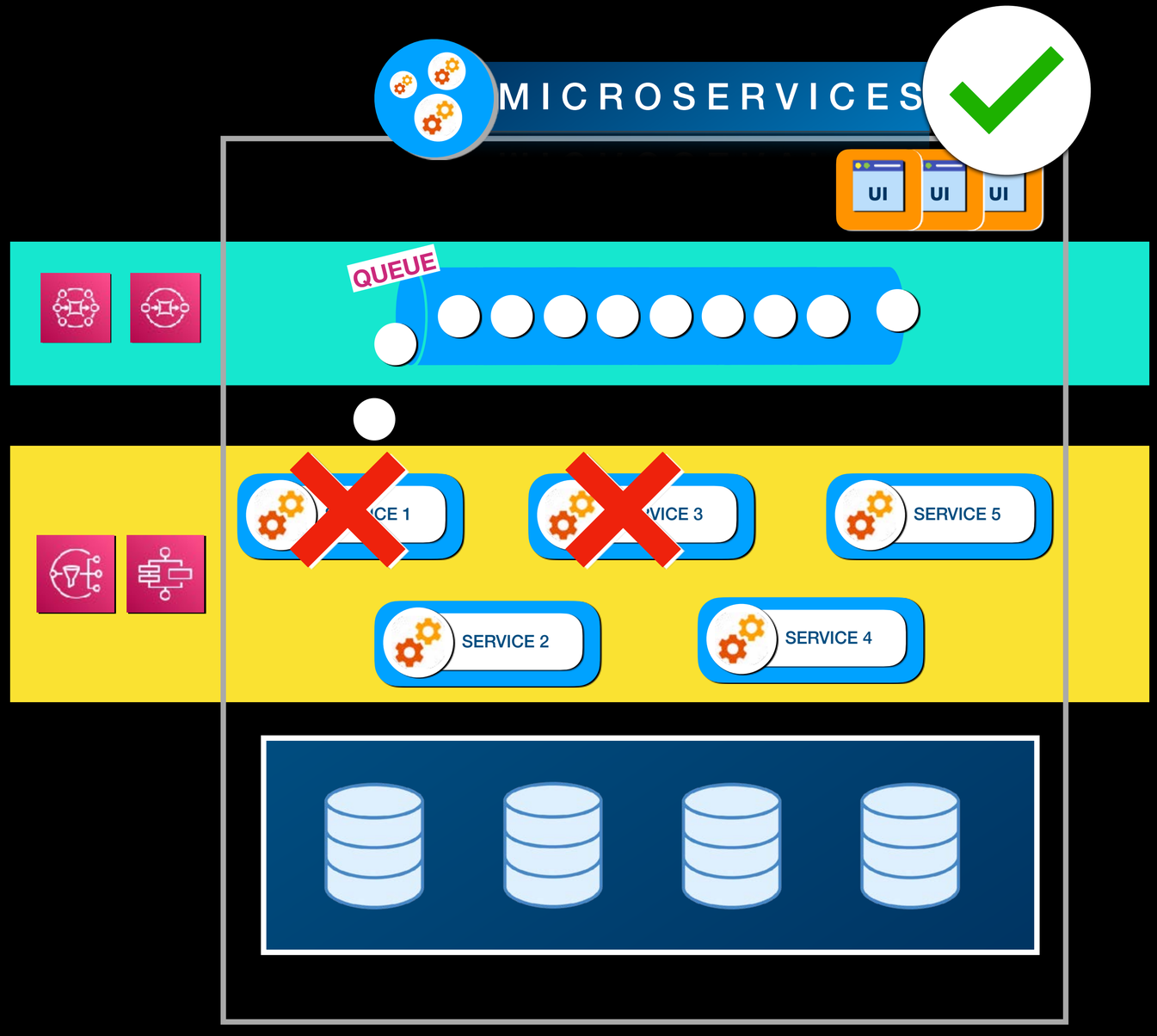
- A fully managed **ledger database** service.
- Provides a transparent and **immutable transaction log** that is owned by a central trusted authority.
- Creates logs that are **cryptographically verifiable**
- Provide an **auditable history of all changes** made to your application data.
- Can be used to track each and every application data change.



Application Integration Services Overview

Application Integration Services

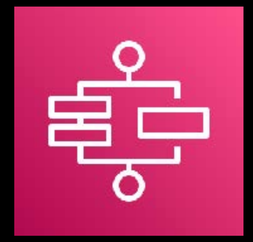




Application Integration Services

Amazon Simple Queue Service (Amazon SQS)

Amazon Simple Notification Service (Amazon SNS)



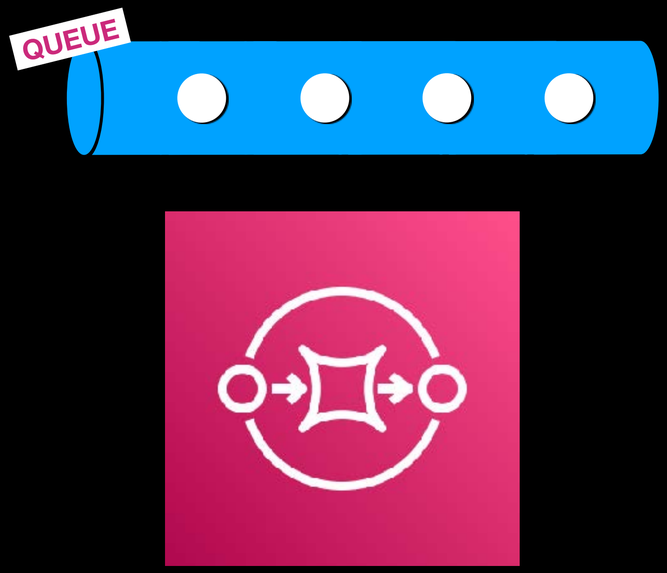
AWS Step Functions

Amazon MQ

Amazon EventBridge

AWS AppSync

Amazon AppFlow



Amazon Simple Queue Service (Amazon SQS)

- A fully managed message queueing service

- The messages can be consumed or processed by:

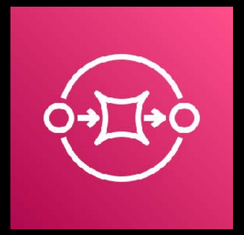

Amazon EC2


AWS Lambda


Amazon ECS

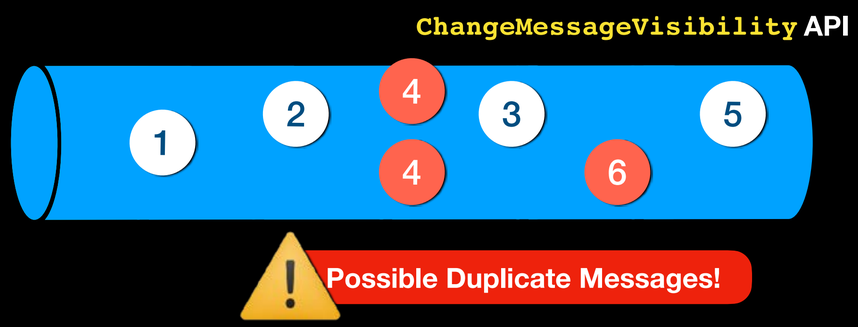

Other Consumers

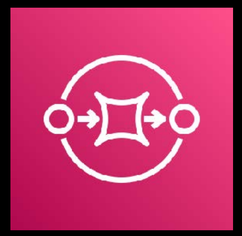
- Can replace your traditional message-oriented middleware without having to manage any servers or resources



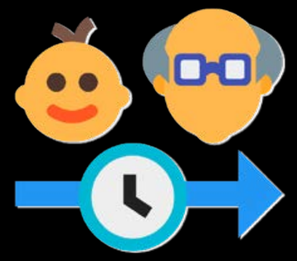
Amazon SQS

TYPES

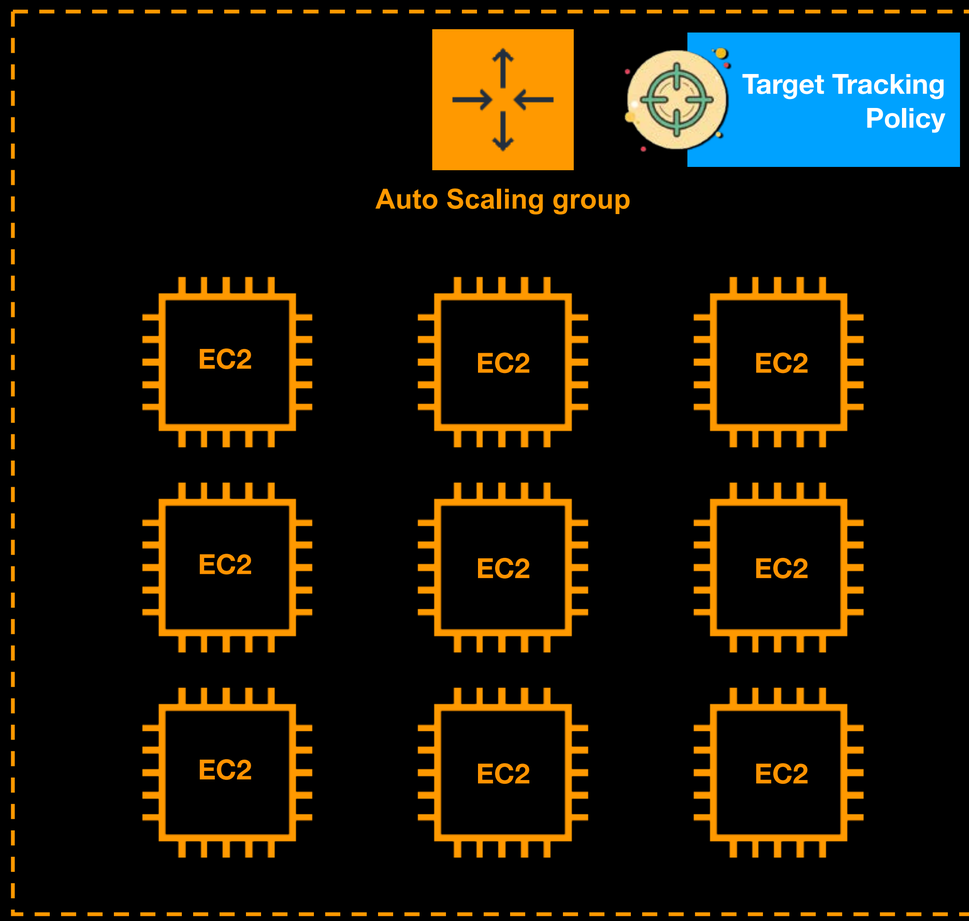
	STANDARD	FIFO
		First In, First Out
DELIVERY		
	At Least Once	Exactly Once
ORDERING	**Best Effort** Messages might be delivered in a **different order**	**Preserves the exact order** **in which the messages are received**
THROUGHPUT	HIGH 	LIMITED 

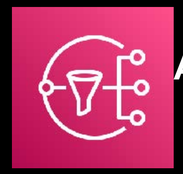


Amazon SQS



- **Age** of the Oldest Message
- **Queue Depth**
- **Number** of Messages

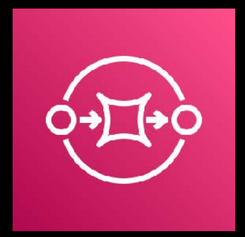




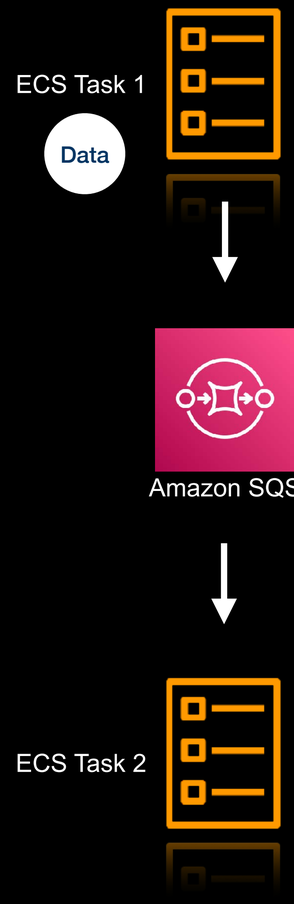
Amazon Simple Notification Service (Amazon SNS)

Amazon ECS

Amazon S3 Bucket

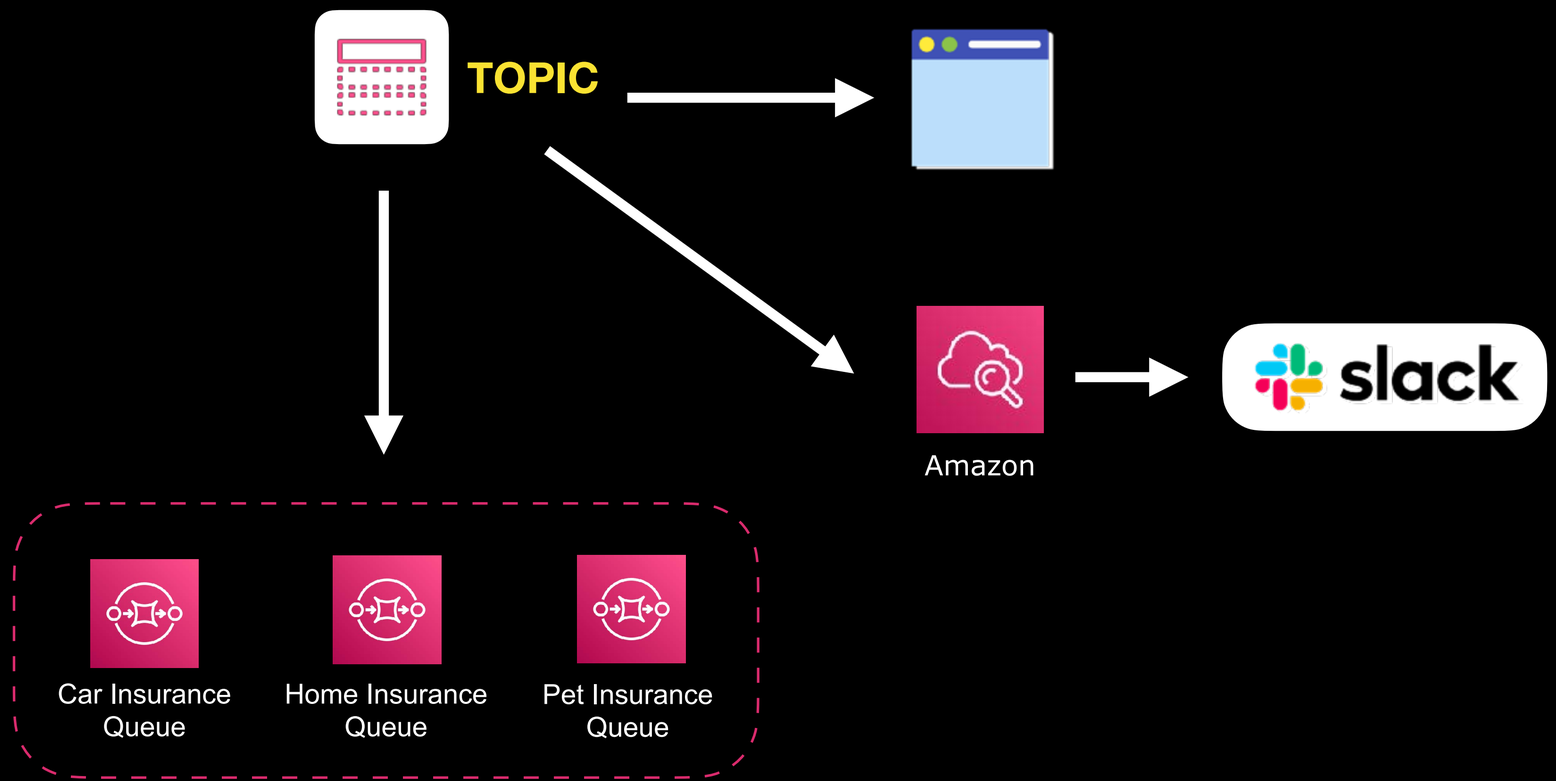


Amazon SQS



Amazon Simple Notification Service (Amazon SNS)

- A fully managed **messaging and notification service**
- **Enables you to communicate between systems** through publish/subscribe patterns or pub/sub messaging
- Messaging via mobile push, email, or SMS



FANOUT EVENT NOTIFICATIONS

Amazon RDS Events

TOPIC

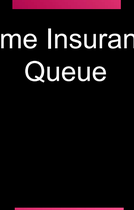
Message Filtering

Filter by

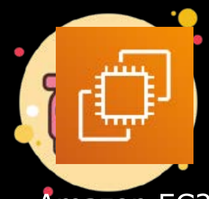
QUOTE Type

Car Insurance Queue

Home Insurance Queue



Pet Insurance Queue



Amazon EC2

Amazon ECS

AWS Lambda

FANOUT EVENT NOTIFICATIONS

SQS QUEUES

CONSUMERS

Amazon SNS with Message Filtering

Custom Events

SNS TOPIC

Message Filter

Filter by

QUOTE Type

Car Insurance Queue

Amazon EC2

Home Insurance Queue

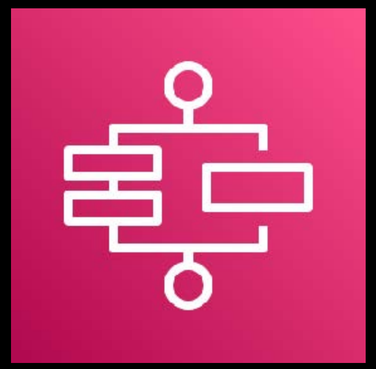
Amazon ECS

Pet Insurance Queue

AWS Lambda

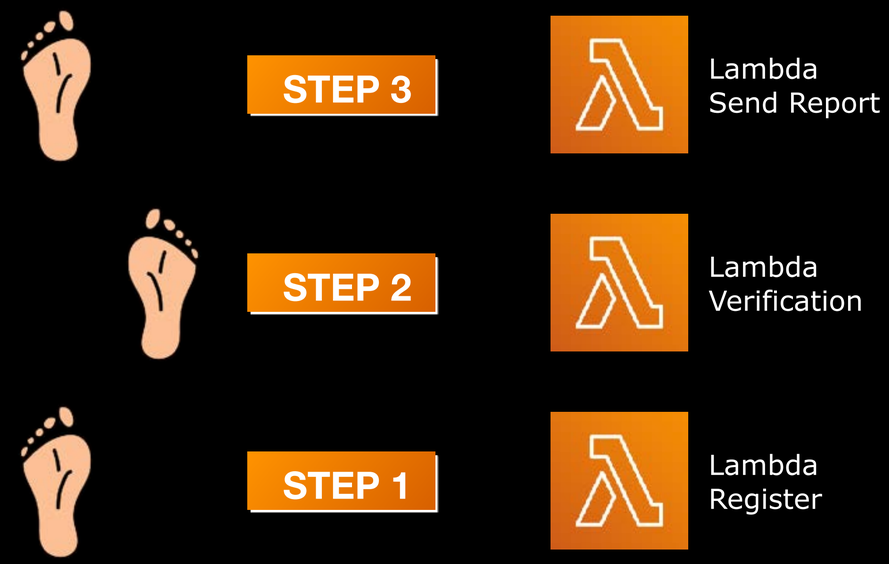
Amazon CloudWatch

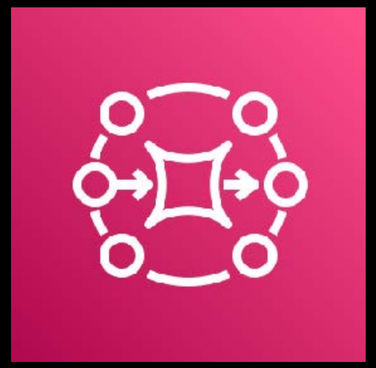
slack



AWS Step Functions

- A **serverless function orchestrator** for AWS Lambda
- Allows you to **orchestrate multiple AWS Lambda functions**, in order to achieve a specific workflow
- Enables you to create a **state machine** containing a combination of steps, activities and service tasks





Amazon MQ

- A managed **message broker** service
- Uses the open-source Apache ActiveMQ message broker
- The "MQ" in Amazon MQ stands for **Message Queue**, which is a form of asynchronous communication
- Works like Amazon SQS but **supports more messaging protocol types**
- Supports Java Message Service (JMS), .NET Message Service (NMS), AMQP, MQTT, WebSocket and many others.

Amazon EventBridge

- A **serverless event bus** service
- Enables you to connect applications together using data from your own applications, Software-as-a-Service (SaaS) applications, and other AWS services.
- **Uses the same service API, endpoint,** and the underlying service infrastructure of:

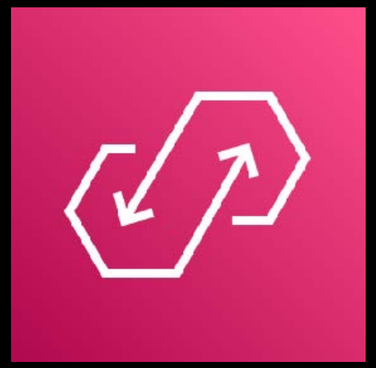
- Recommended to be used for your own applications, 3rd party Software-as-a-Service apps, and other external sources
- Suitable for building **event-driven applications**

AWS AppSync

- A managed service that uses GraphQL
- GraphQL is a **data query language** that basically allows you to query your REST APIs
- Has different types of **schema**

QUERY	Read Data
MUTATION	Write Data
SUBSCRIPTION	Download/Upload Data

- Only fetches the data that you want and not the entire data set
- Unlike REST API, you can query different APIs or resources easily using a single API call
- Uses a **Resolver** which populates the data in your schema
- Simplifies application development by easily integrating GraphQL with your applications



Amazon AppFlow

- A fully managed **integration service**
- Enables you to **securely transfer data between various systems** such as your Software-as-a-Service (SaaS) applications and different AWS Services
- Supports different SaaS apps such as Salesforce, Marketo, Slack, ServiceNow and many more
- Can be integrated with other AWS services
- **Allows you to run your data flows** on-demand, by schedule or as a response to a business event
- Provides you with powerful data transformation capabilities like filtering and validation

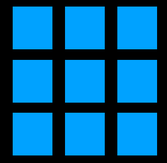
AWS Analytics Services Overview

Data Warehouse

STRUCTURED DATA

Data Lake

STRUCTURED DATA



UNSTRUCTURED DATA

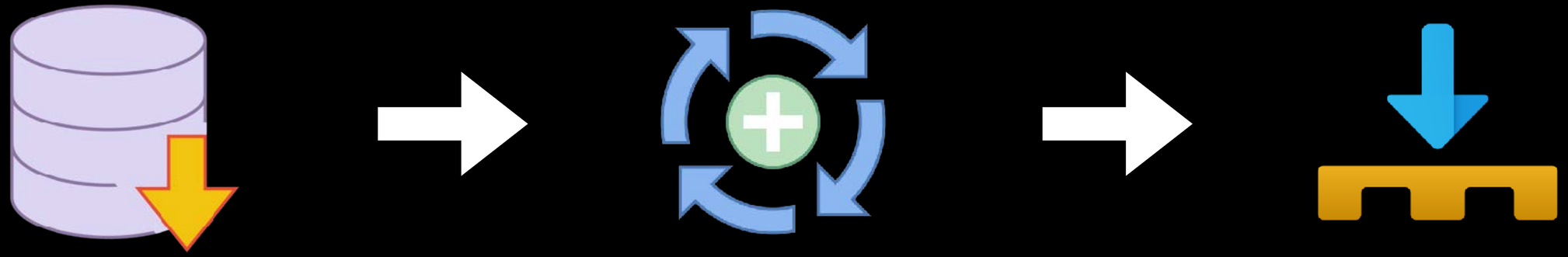
Open Source Technologies used by AWS Analytics Services

...and many other open-source projects!

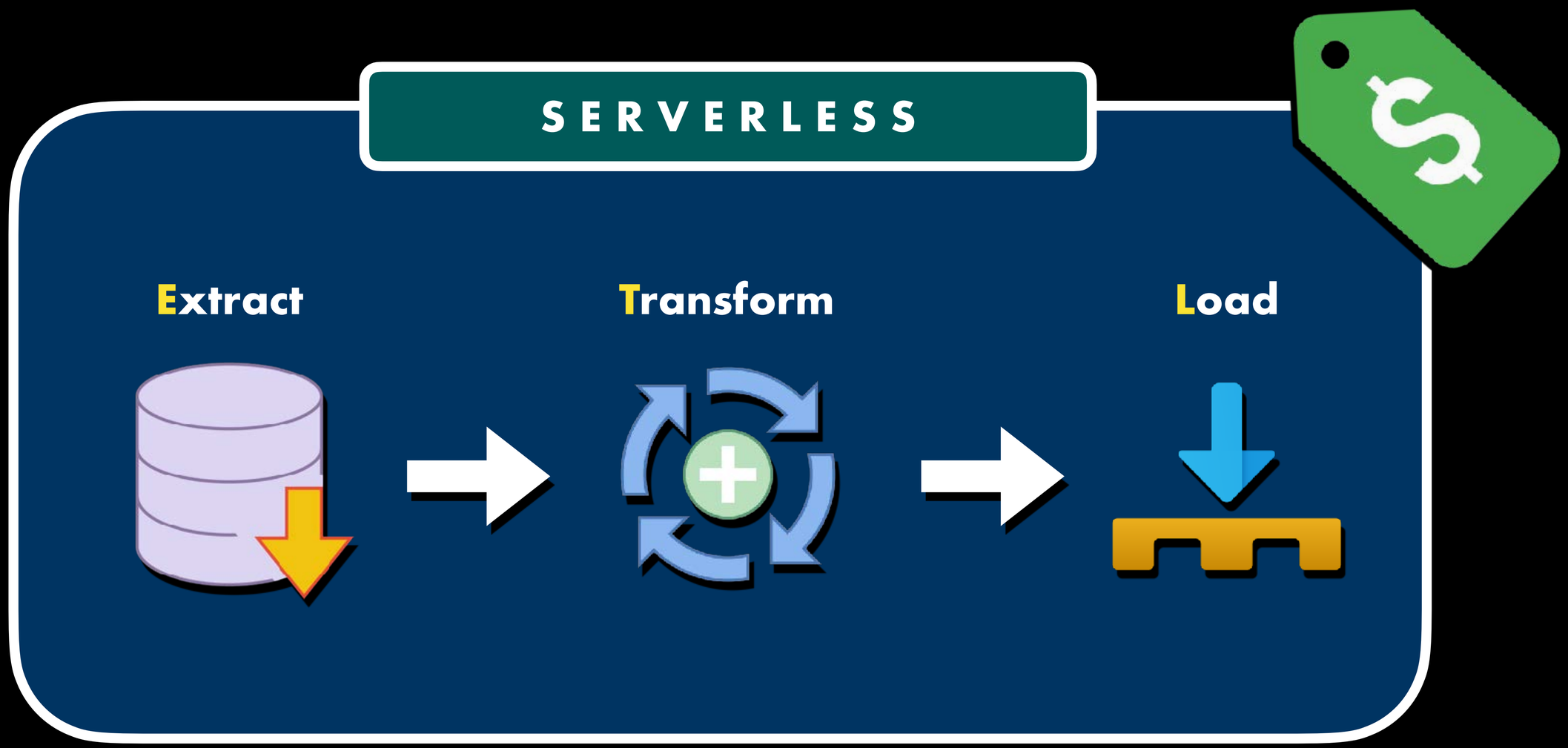
3rd Party Technologies used by AWS Analytics Services

...and many more!

Extract **Transform** **Load**



SERVERLESS



AWS Analytics Services

Amazon Kinesis

Amazon Athena

Amazon Elasticsearch
(Amazon ES)

Amazon Elastic MapReduce
(Amazon EMR)

Amazon QuickSight

Amazon CloudSearch

Amazon Redshift

AWS Data Pipeline

AWS Glue

Amazon Managed Streaming for Apache Kafka

AWS Lake Formation

Amazon Kinesis

- **A suite of services** for processing your data streams
- Analyzes your **data streams** in **real-time**
- Allows you to **collect, transform, process, load, and analyze the streaming data** in real-time to help you acquire the data insights and respond to data changes

CLICK STREAM

Amazon Kinesis

Amazon Kinesis

Amazon Kinesis
Data Streams

Amazon Kinesis
Data Firehose

Amazon Kinesis
Data Analytics

Amazon Kinesis
Video Streams

Amazon Kinesis
Data Streams

- A massively scalable, durable, secure and low-cost **real-time data streaming service**
- Can **continuously capture gigabytes of data** per second from thousands of different sources
- Collects and sends data to your data analytics applications and consumers in real-time

Amazon Kinesis
Data Streams

- Can be used in:
  - Real-time Applications
  - Website Clickstreams
  - Database Event Streams
  - IoT Telemetry
  - Location-tracking Events
  - Predictive Maintenance
  - Mobile Game Data Streams
  - Online Marketplaces
  - Real-time Recommendations Systems
  - …and many more!
- Provides **ordering of records**
- Can **read & replay records** in the same order
- Suitable if you have a requirement where:
  - The data events must be received in an **ordered manner**
  - There's a need to process the data stream of your web applications, or mobile game updates, in **order of receipt**

Amazon Kinesis Data Streams

- Can be used to decouple your cloud architecture like Amazon SQS by accepting data from your data sources and forward it to different compute resources

- Similar to **Amazon SQS** with notable differences:
  - SQS **can't process data in real-time**
  - SQS Standard queue **doesn't maintain the order of data** records by default
  - SQS FIFO queue maintains the order of data records but is **significantly slower** than SQS Standard and doesn't perform in real-time

Amazon SQS

USE CASES

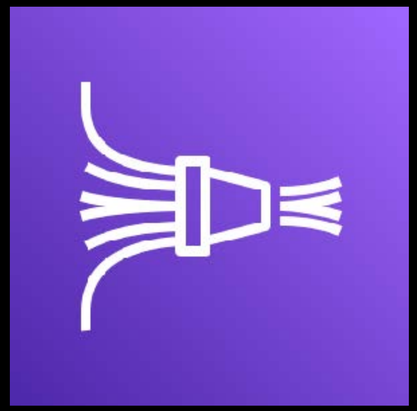
Amazon Kinesis Data Streams

- If you need a solution that **captures the clickstream data from multiple websites in real-time** and analyzes it using batch processing
- For setting up and building a scalable, **near-real-time recommendations** for your users
- For **mobile games that stream score updates** to a backend system and post the results on a leaderboard
- For collecting the mobile game scores in **order of receipt** which can then be processed by an AWS Lambda function and stored in DynamoDB

USE CASES

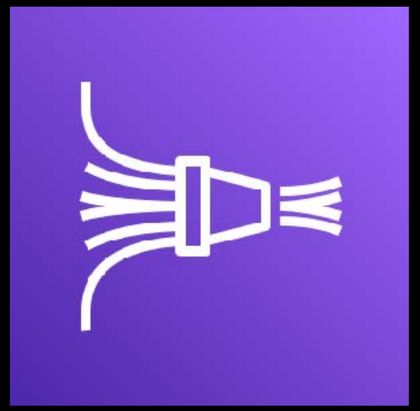
Amazon Kinesis Data Streams

- For implementing **predictive maintenance** on different types of machinery equipment using IoT sensors
- For sending data to AWS in real-time wherein the data stream will **receive events in an ordered manner** for each connected device, data producer or machinery asset
- For implementing a scalable, near-real-time solution in processing millions of financial transactions
- For launching a data stream that can be consumed by Amazon Kinesis Data Analytics which can be queried using SQL queries



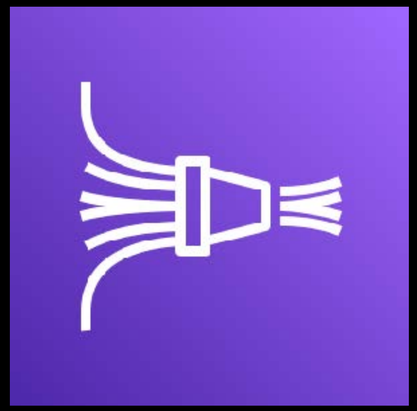
Amazon Kinesis Firehose

- A fully managed service that reliably **transforms and loads your streaming data** into data stores and analytics tools
- Directly delivers data to Amazon S3, Amazon Redshift, Amazon Elasticsearch Service, and any HTTP endpoint
- Can be integrated with your third-party service providers
- Enables your data producers to directly send data to a specific destination or data store that **without any custom applications or consumers**
- Can **transform your data before sending it** to a specified destination to remove sensitive data or for data pre-processing procedures



Amazon Kinesis Firehose

- Similar to Amazon Kinesis Data Stream but with certain differences:
  - Both service can accept streaming data in real-time
  - However, Kinesis Data Stream requires an external consumer to store the records while Kinesis Data Firehose does not
- Acts like a ”firehose” to **immediately send the streams of data to your data store**
- Delivers your data stream directly to your Amazon S3 buckets, Redshift databases, Amazon ES clusters, and others without the need for a consumer



Amazon Kinesis Firehose

- Can transform the data before it is sent to its destination
- Internally **invokes an AWS Lambda function** to transform the incoming source data and deliver the processed data to its destination
- Recommended if you need to parse the data stream to **remove any sensitive data such as personal data or protected health information (PHI)**

Amazon Kinesis Video Streams

- A service that **securely streams video** from connected devices or sources to AWS
- Commonly used for data analytics, machine learning, video playback, and other types of media processing
- Automatically provisions and scales all the required infrastructure to ingest streaming video data from millions of devices
- **Stores, encrypts, and indexes video data** in your streams to improve performance
- Provides access to your video data through a collection of easy-to-use APIs

Amazon Kinesis
Data Analytics

- A serverless service that enables you to **analyze your streaming data**, acquire actionable insights, and respond to events in real-time
- Reduces the complexity of building, managing, and integrating streaming applications with your custom applications and other AWS services
- **Serverless**
- **Uses Apache Flink** to process and analyze streaming data
- Eliminates the manual tasks of setting up and maintaining Apache Flink

Amazon Kinesis
Data Analytics

- Enables you to **author and run code** against streaming sources
- The data **can be analyzed using SQL queries** and the results can be delivered to Amazon S3, Amazon Redshift, and other data stores using Kinesis Data Firehose
- Java or Scala can be used to process and analyze your streaming data

USE CASES

Amazon Kinesis
Data Analytics

- In **near-real-time data processing** and data querying for acquiring timely insights of your application
- For processing your streaming data **with minimal effort and operational overhead**
- For providing scalable and near-real-time data querying with minimal data loss
- For **analyzing the location data points of your GPS application** that tracks the movement of people, bikes, automobiles, or any other moving object
- You can expose a REST API using API Gateway that can be used as an **Amazon Kinesis proxy**

Amazon Athena

- An **interactive query service** for your data that is stored in Amazon S3
- Simplifies data analysis in Amazon S3 using standard SQL queries
- Unlike S3 Select, you can **query the entire data in your Amazon S3 bucket** with Amazon Athena and not just its subset
- **Serverless**

Amazon Athena

- Sample use case:
  - A global eCommerce website stores 250 gigabytes of transactional data each month in Amazon S3
  - You need to identify the number of items sold in each particular region for the previous month in the most cost-effective way
- Athena **costs less** than Amazon Redshift, Amazon EMR, or Amazon ES since it’s serverless
- **Can use an AWS Glue Data Catalog** to store and retrieve table metadata for your Amazon S3 data and provide data visualization using Amazon QuickSight

Amazon Elasticsearch Service
(Amazon ES)

- A fully managed **Elasticsearch** service
- Elasticsearch is a distributed, multitenant-capable full-text search engine based on the Apache Lucene library
- Provides an HTTP web interface that can store data as a **schemaless JSON document**
- Provisions the necessary infrastructure and automatically manages the resources needed to run the Amazon ES cluster

Amazon Elasticsearch Service

(Amazon ES)

- Also allows you to launch an ELK (Elasticsearch, Logstash, and Kibana) stack in AWS
  - ELK Stack:
    - **Elasticsearch** - full-text search engine
    - **Logstash -** server-side data processing pipeline
    - **Kibana** - user interface to visualize Elasticsearch data
- Provides support for open-source Elasticsearch APIs, managed Kibana, integration with Logstash and other AWS services
- Lets you pay only for what you use *(no upfront costs or usage requirements)*

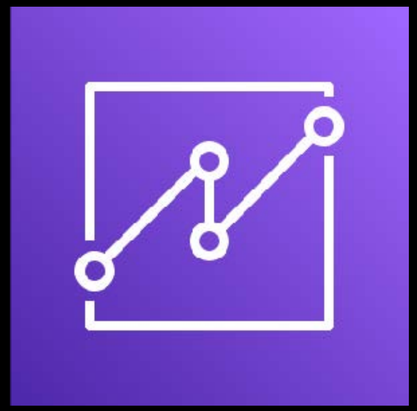
Amazon Elastic MapReduce

(Amazon EMR)

- Allows you to **run different types of big data frameworks in AWS**
- A managed big data platform for processing vast amounts of data using open source tools such as:

Amazon Elastic MapReduce
(Amazon EMR)

- Runs your big data framework on Amazon EC2 instances, Amazon Elastic Kubernetes Service clusters, or in your on-premises EMR cluster via AWS Outposts
- The compute resources launched by Amazon EMR are deployed in your VPC and then grouped as an Amazon EMR cluster
- You can **directly access and control the underlying EC2 instances** of your EMR cluster
- **NOT serverless**
- Automates the server provisioning and management process for you and allows your data to interact with other AWS data stores such as Amazon S3 and Amazon DynamoDB



Amazon QuickSight

- A scalable, serverless, embeddable, machine learning-powered **business intelligence service**
- Allows you to **create and publish interactive dashboards** that can be accessed from different browsers or mobile devices
- Allows you to **embed dashboards** into your applications
- Highly scalable and can easily scale up to thousands of users globally
- **Serverless**

Amazon CloudSearch

- A **managed search service** in AWS
- Can be used to add a search feature in your application or websites
- You can use this to:
  - Retrieve contents of selected fields
  - Provide facet information to categorize results
  - Provide statistics for numeric fields
  - Provide highlights showing search hits in the field data
  - Autocomplete suggestions
  - Geospatial search
  - and many more!

Amazon CloudSearch

- Allows you to **create a search domain**, specify an index and upload your data as documents
- Provisions and manages all the underlying servers and resources needed to build and deploy search indexes
- Simply upload your data to any data store, create a search domain in CloudSearch, and integrate it into your applications

Amazon Redshift

- A fast, scalable **data warehouse**
- Allows you to analyze all your data across your data warehouse and data lake
- Delivers faster performance than other data warehouses through the use of machine learning, massively parallel query execution and **columnar storage** on high-performance disks
- Can **run queries across petabytes of data** in your Redshift data warehouse and analyze exabytes of data in your S3 data lake
- Primarily used **for Online Analytical Processing (OLAP) applications** and reporting tools

Amazon Redshift

- Redshift clusters run in internal Amazon EC2 instances that are configured as nodes
- You can select the particular node type and instance size that you prefer
- **Not a serverless** service
- Has a feature called **Redshift Spectrum** that allows you to query data from Amazon S3 without loading the entire data into Redshift tables
- Redshift Spectrum queries use massive parallelism to **quickly execute large datasets** at a fraction of the cost

Amazon Data Pipeline

- A service that processes and **moves your data between different AWS compute and storage services**
- Enables you to process and move your data in specific intervals that you define to transfer your data to and from your on-premises data center
- Allows you to access, transform and process your data where it's stored at scale
- Empowers you to **transfer and store the results to various AWS services** such as Amazon S3, Amazon RDS, Amazon DynamoDB, and Amazon EMR

AWS Glue

- A fully managed and serverless service that is primarily **used for extract, transform, and load workloads** or ETL
- Simplifies the process of preparing and loading your data before running your data analytics workload
- Creates a **Data Catalog** that allows you to specify and search your data that is stored on Amazon S3 and other AWS services
- Automatically discovers your data and store the associated metadata in the AWS Glue Data Catalog
- The data will be immediately searchable, queryable, and available for ETL once the metadata is stored

Amazon Managed Streaming for Apache Kafka

- A **fully managed Apache Kafka service** in AWS
- Apache Kafka is an open-source platform that allows you to **build real-time streaming data pipelines** and applications
- Allows you to use Apache Kafka APIs to stream changes to and from different databases, populate your Amazon S3 data lakes, and empower machine learning and analytics applications

AWS Lake Formation

- Makes it easy for you to **set up a secure data lake**
- Allows you to **create data catalogs** for your external data just like AWS Glue
- Collects and catalogs your data from different data sources and moves the data into a new Amazon S3 data lake
- **Classifies and processes your data** using machine learning algorithms, and secures access to your sensitive data
- Data can be queried and analyzed using Amazon Athena, Amazon Redshift, Amazon EMR, and other services

AWS Identity Services Overview

AWS Identity Services

AWS Identity Services

AWS Single Sign-On

AWS Identity & Access Management (IAM)

AWS Directory Service

Amazon Cognito

AWS Identity & Access Management (IAM)

- The primary **identity service** in AWS
- Allows you to **manage access to various AWS services** and resources

AWS Identity & Access Management (IAM)

IAM USER

IAM GROUP

PASSWORD

ACCESS KEYS

Permission 1 Permission 2 IAM POLICY Permission 3 Permission 4

Permission 1 Permission 2 IAM ROLE Permission 3 Permission 4

- Let you **add user sign-up, sign-in, and access control** features to your web or mobile apps
- Allows users to log in to your application with their:

and other social media accounts!

Amazon Cognito

Microsoft **Active Directory**

SAML

Security Assertion Markup Language

USER POOL

IDENTITY POOL

Amazon Cognito

For **Authentication**

Users can sign in by authenticating through their **social identity providers**

For **Authorization**

Users can obtain **temporary and limited-privilege AWS credentials** that authorize access to other AWS services

AWS Single Sign-On

- A **single sign-on service** in AWS
- Allows a user to **log in with a single ID and password** to access multiple and independent, software systems
- Provides a **user portal** that allows users to access the roles that they can assume
- Offers pre-configured SAML integrations to many business applications

AWS Directory Service

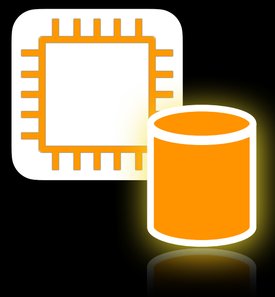
- A managed Microsoft **Active Directory**
- Does not require you to synchronize or replicate data from your existing Active Directory to the cloud
- No need to install and manage an Active Directory domain controller
- Improves security and minimizes administrative overhead
- **Allows you to assign IAM roles** to your Active Directory users and groups
- Allows you to assign IAM roles to your on-premises Microsoft Active Directory using:

AD Connector

AWS Storage Services Overview

AWS Storage Services

Built-in component and NOT a full-fledged AWS Service



Amazon EC2 Instance Store

Amazon Elastic Block Store (Amazon EBS)

Amazon Simple Storage Service (Amazon S3)

Amazon S3 Glacier

Amazon Elastic File System (Amazon EFS)

Amazon FSx for Lustre

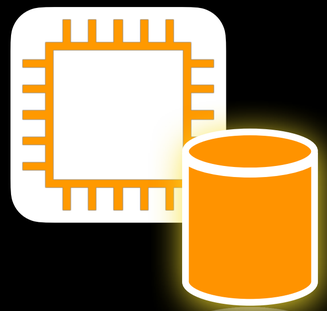
Amazon FSx for Windows File Server

AWS Backup

AWS Storage Gateway

Underlying Host Computer that powers your

Amazon EC2 Instances



Amazon EC2 Instance Store

e·phem·er·al

/əˈfem(ə)rəl/

adjective

lasting for a very short time.
"fashions are ephemeral"

Similar: transitory | transient | fleeting | passing | short-lived | momentary

- A temporary or ephemeral block-level storage
- Uses the local disks or storage volumes that are physically attached to the underlying host computer of the Amazon EC2 instance.
- Provides low-latency access to your data
- Loses its stored data if:
  - The underlying local storage fails
  - The Amazon EC2 Instance:

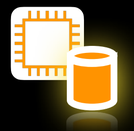
Stops

Hibernates

Terminates

Amazon Elastic Block Store
(Amazon EBS)

- A persistent block-level storage service
- Your data will still be there even if you stop, restart, or terminate your Amazon EC2 instance, unlike:



Amazon EC2
Instance Store

- Also called EBS Volumes
- Mounted or attached to your Amazon EC2 instances
- Zonal in scope — you can only attach a volume to any EC2 instances in the same Availability Zone.
- Can be encrypted at rest using:

AWS Key Management Service
(AWS KMS)

Amazon Elastic Block Store
(Amazon EBS)

Solid State Drive (SSD)

Hard Disk Drive (HDD)

Read & Write Speeds	 Fast !	 Slow…
Use Case	For workloads with frequent read/write operations	For data archiving, backups or throughput-oriented storage
Dominant Performance Attribute	IOPS Input/Out operations Per Second	Throughput Megabit per second (Mbps)
Can be used as **Boot Volume** for  Amazon EC2 ?	Yes	No

Amazon Elastic Block Store
(Amazon EBS)

TYPES

Faster data retrieval than:

Amazon S3

Amazon EFS

Solid State Drive (SSD)

Hard Disk Drive (HDD)

gp — General Purpose SSD

st — Throughput Optimized HDD

io — Provisioned IOPS SSD

sc — Cold HDD

Can only be attached to a single Amazon EC2 at a time

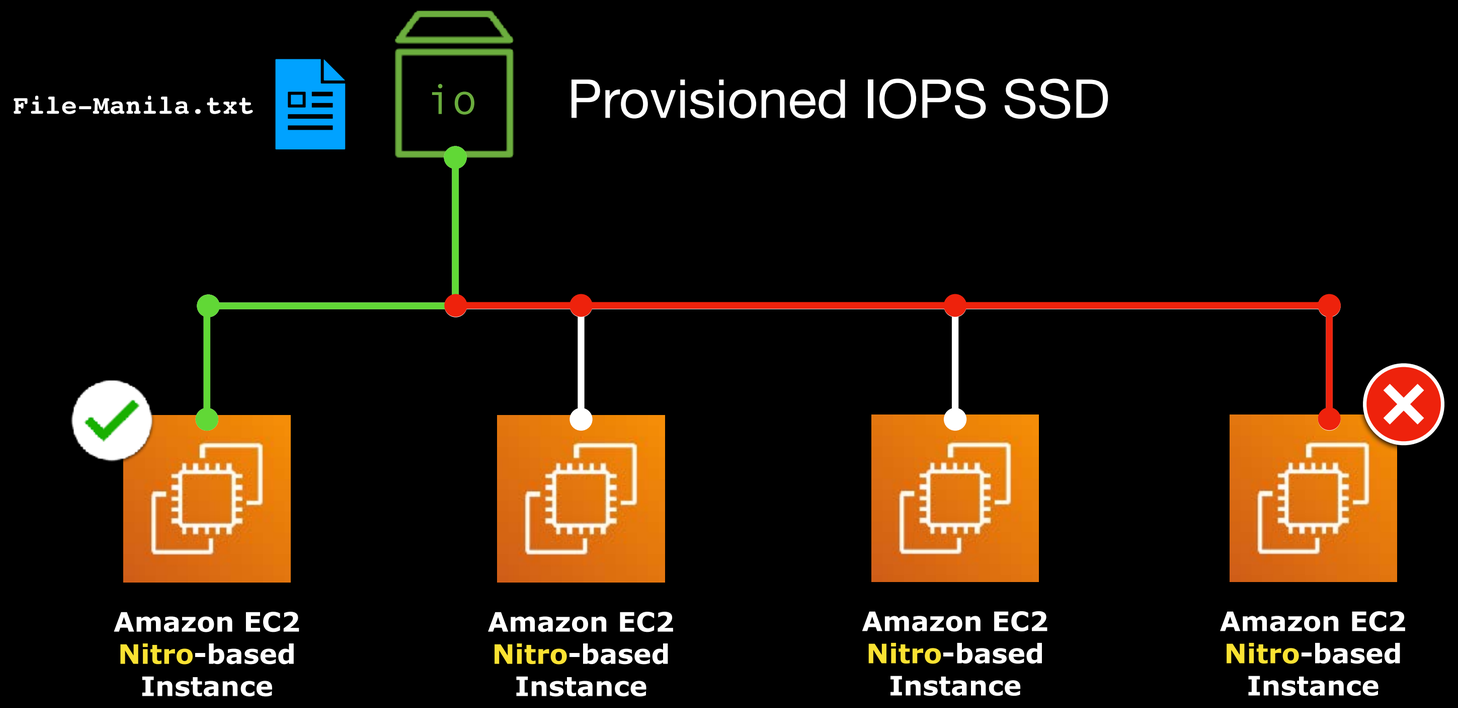
Can be used as **Boot Volume** for Amazon EC2

Cannot be used as a Boot Volume

Amazon EC2

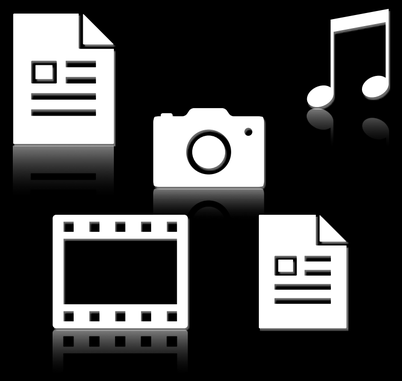
Amazon Elastic Block Store
(Amazon EBS)

EBS Multi-Attach



No concurrent file modification

Amazon EFS



Amazon Simple Storage Service
(Amazon S3)

- An object storage service
- Highly durable and scalable
- Can store virtually unlimited amounts of data
- The files are called “objects” that you upload to an S3 Bucket
- Access files via a REST API call

Amazon S3 Storage Classes

S3 Standard

For **frequently accessed** data

S3 Intelligent-Tiering

For changing or
unknown access patterns

S3 Standard-IA
(Infrequent Access)

S3 **One** Zone-IA
(Infrequent Access)

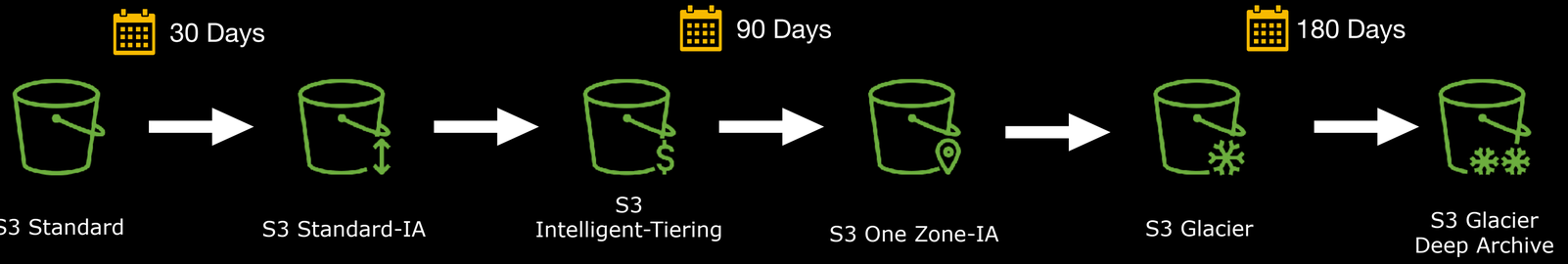
For storing long-lived,
yet **less frequently accessed** data

S3 Glacier

S3 Glacier **Deep** Archive

For **low-cost long-term storage**
and data archiving

Lifecycle Policy



Access Control List
(ACL)

- Secure access to your S3 buckets and objects

Bucket Policy

- Control external access to your Amazon S3 bucket.

S3 Versioning

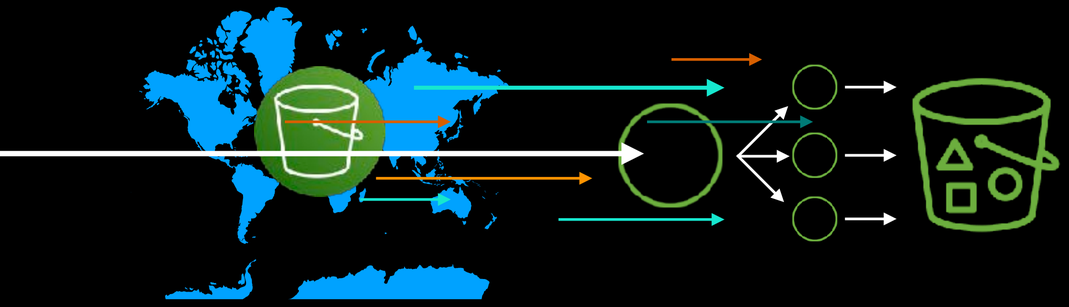
Multi-Factor Authentication
(MFA)

- Prevent accidental data deletion in Amazon S3.

Cross Region Replication (CRR)

- Automatically replicate objects to a different AWS Region for backup purposes

---



Transfer Acceleration

Multipart Upload

- Accelerate or expedite the data transfer (upload/ download) of S3 objects

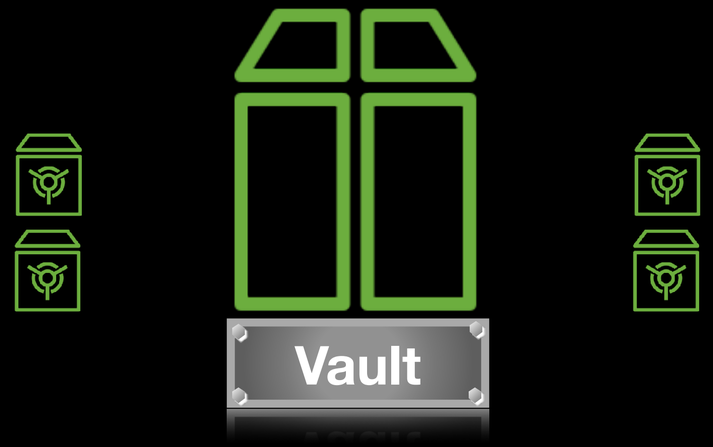
…and many more S3 features!

Amazon S3 Glacier

- One of the storage classes in Amazon S3
- Has its own web management console apart from Amazon S3
- Based on the word — **Glacier**:
  - Rarely Accessed Data (Cold)
  - Frequently Accessed (Hot)

Cold HDD

- Low-cost storage for data archiving and long-term backup.



S3 Glacier		S3 Glacier Deep Archive
LOW $ $	COST	LOWEST $
90 Days	MINIMUM STORAGE DURATION	180 days
You will be billed for the entire 90 Days	DATA DELETED AFTER 1 DAY (24 HOURS)	You will be billed for the entire 180 Days
Normal storage usage charge	DATA DELETED AFTER 90 DAY	You will be billed for the entire 180 Days
Normal storage usage charge	DATA DELETED AFTER 180 DAYS	Normal storage usage charge

S3 Standard

VS

S3 Glacier

HIGHEST $ $ $	COST Timed Storage - Byte Hours	**LOWEST $**
None	MINIMUM STORAGE DURATION	**90 days**
Regular storage usage charge (24 hours)	DATA DELETED AFTER 1 DAY (24 HOURS)	You will be billed for the **entire 90 Days**
Regular storage usage charge (30 days)	DATA DELETED AFTER 30 DAYS	You will be billed for the **entire 90 Days**
Regular storage usage charge (90 days)	DATA DELETED AFTER 90 DAYS	Regular storage usage charge (90 Days)

Archive Retrieval Options

	EXPEDITED	STANDARD	BULK
 **S3 Glacier**	1 - 5 minutes	3 - 5 hours	5 - 12 hours
 **S3 Glacier Deep Archive**	NOT AVAILABLE	Within 12 Hours	Within 48 hours

Amazon Elastic File System
(Amazon EFS)

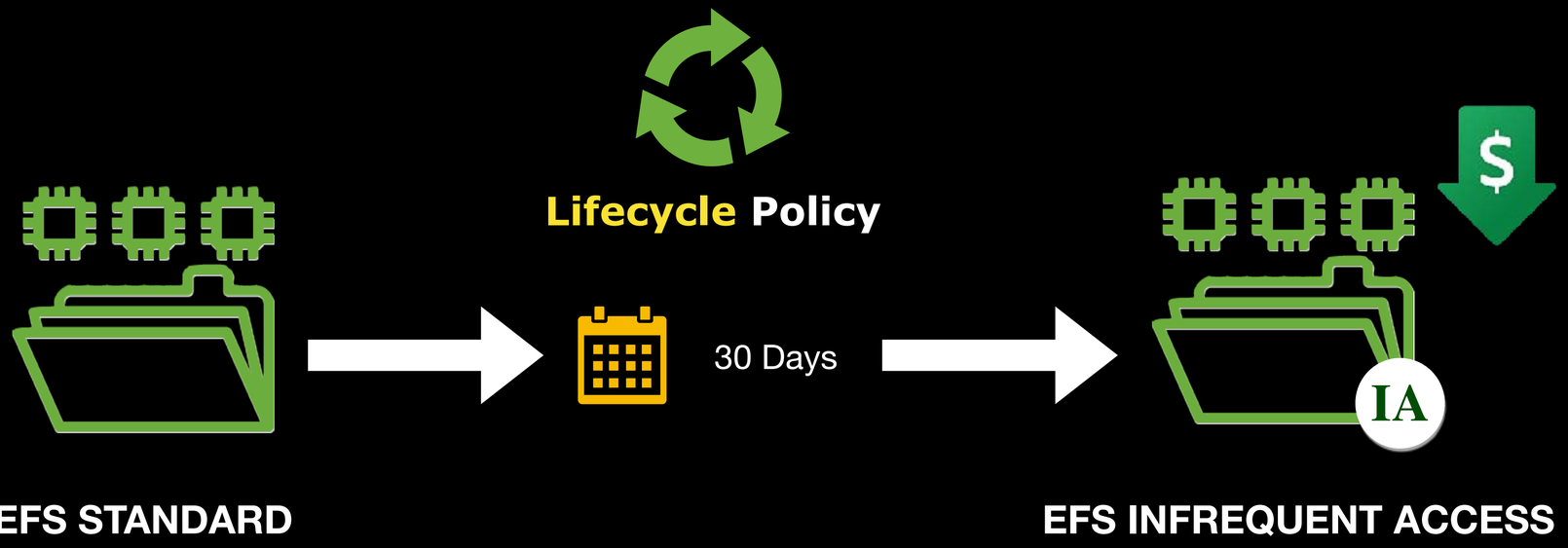
- A scalable shared file storage service
- Provides a POSIX-compliant (Portable Operating System Interface) shared file system
- Can be simultaneously accessed by multiple Amazon Linux EC2 instances in different Availability Zones.
- Uses the Network File System (NFS) protocol. Works as a file share
- Only supports:

Linux Servers

=

Amazon FSx for
Windows File Server

Amazon Elastic File System
(Amazon EFS)



Amazon FSx

Amazon FSx for Lustre

Amazon FSx for
Windows File Server

Amazon Elastic File System
(Amazon EFS)

- A scalable shared file storage service
- Provides a POSIX-compliant (Portable Operating System Interface) shared file system
- Can be simultaneously accessed by multiple Amazon Linux EC2 instances in different Availability Zones.
- Uses the Network File System (NFS) protocol
- Only supports:

Linux Servers

Amazon FSx for Lustre

lustre®
open-source, parallel file system

= Linux Cluster

Amazon FSx for Lustre

- a parallel file system used for large-scale cluster computing.
- Primarily used for High-Performance Computing, Machine Learning, or HPC applications
- For workloads that need high-performance parallel storage for frequently accessed hot 🥵 data.
- Provides a throughput of hundreds of gigabytes per second
- Offers millions of IOPS
- You can mount an Amazon FSX for Lustre file share to:

Amazon EC2 Amazon ECS Amazon EKS

- Use the CSI Container Storage Interface (CSI) to connect to your Amazon EKS cluster.

Amazon FSx for Windows File Server

- A fully managed Microsoft Windows file server service
- Uses the Server Message Block (SMB) protocol
- Can be integrated to your existing:
  - Microsoft **Active Directory**
  - AWS Managed Microsoft AD
- Can be used as shared file storage for your:
  - Microsoft **SharePoint**
  - Microsoft **SQL Server**
  - Microsoft **Containers**

- A fully managed backup service
- Automates your server and database backup processes.

AWS Backup

Service-level backups

- Amazon FSx
- Amazon EFS
- Amazon DynamoDB
- Amazon EC2

Service-level snapshots

- Amazon Aurora
- Amazon RDS
- Amazon EBS
- AWS Storage Gateway

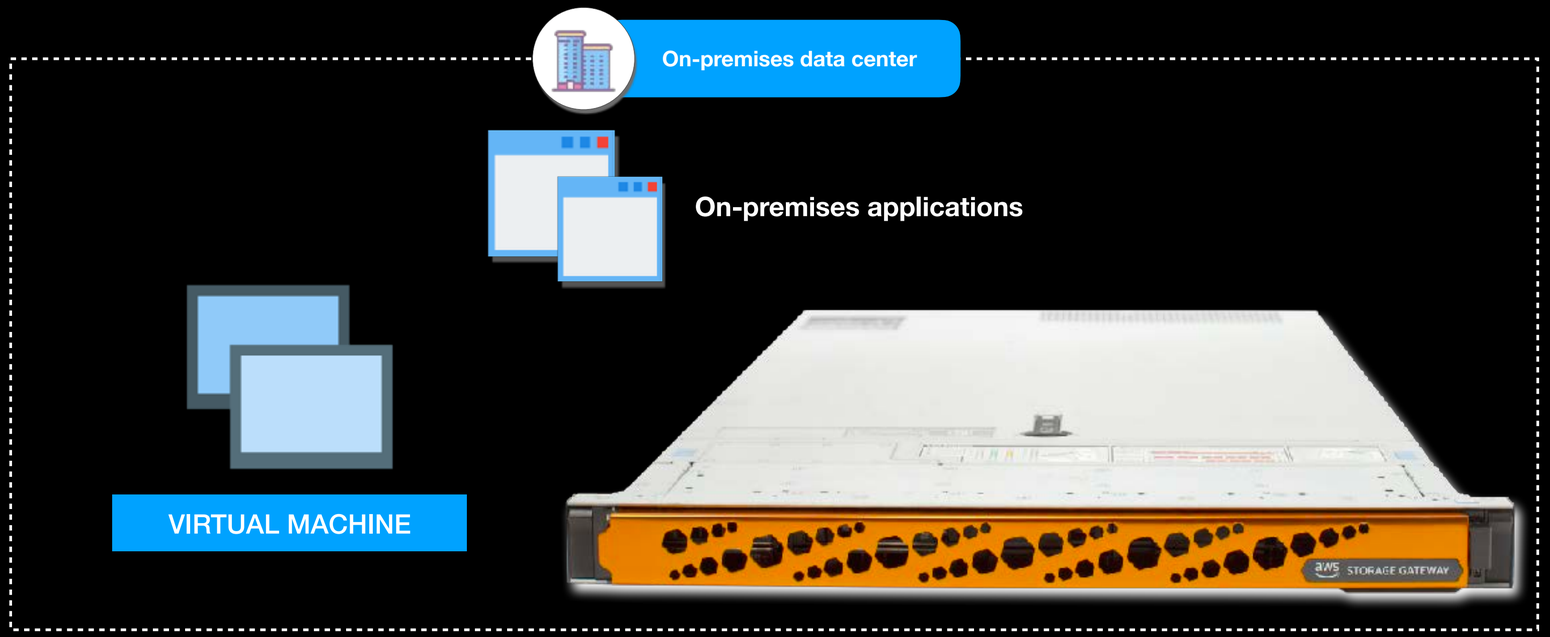
7 Days (Default)

35 Days (Maximum)

90 Days, One Year or even more!

AWS Storage Gateway

- A hybrid cloud storage service
- Connects your on-premises applications and data storage to the AWS Cloud.
- Integrate your local & cloud storage systems by using a gateway.



File Gateway

Store and retrieve objects in Amazon S3

using NFS and SMB protocols

Can be integrated with:

AWS Managed Microsoft AD

Microsoft **Active Directory**

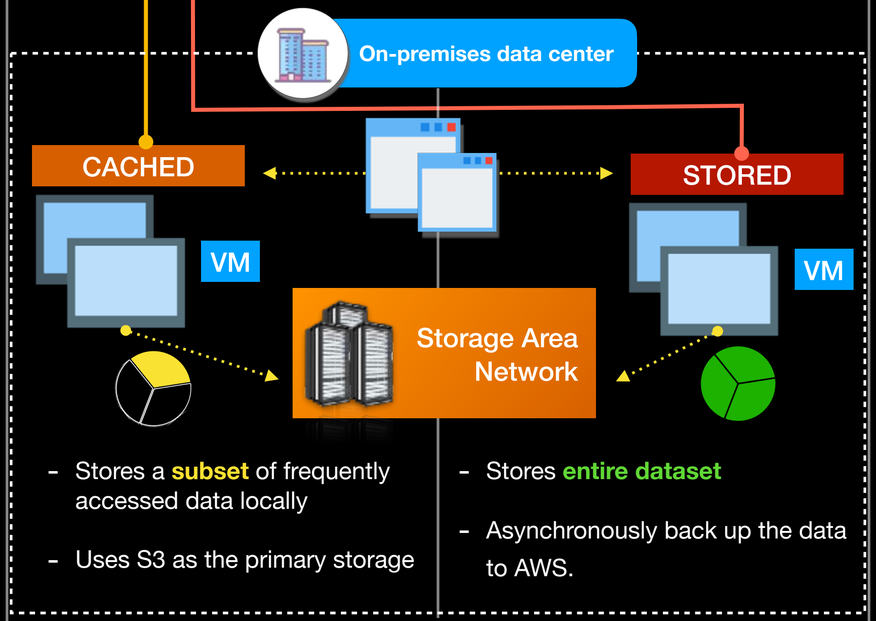
Provides a hardware appliance hosted on-premises

To replicate your local data to Amazon S3

Volume Gateway

Provides block storage to your on-premises apps with low-latency via the Internet Small Computer System Interface (iSCSI)

Uses Amazon S3 for point-in-time snapshots of your EBS Volumes



- Stores a **subset** of frequently accessed data locally
- Uses S3 as the primary storage

- Stores **entire dataset**
- Asynchronously back up the data to AWS.

Tape Gateway

A cloud-based Virtual Tape Library

Uses Amazon S3 to back up the tapes

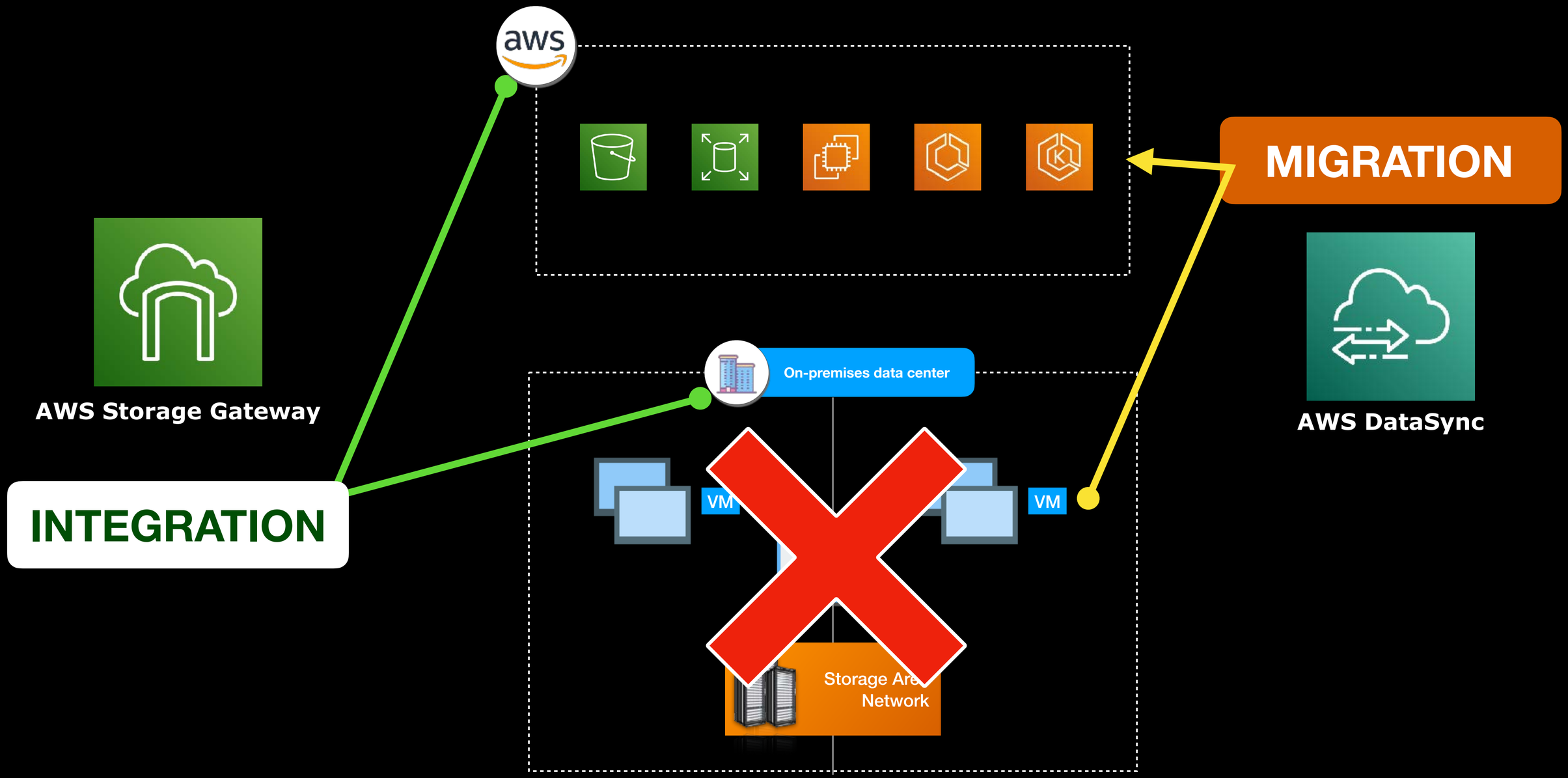
Can store the archived tapes in:

S3 Glacier

S3 Glacier Deep Archive

- On-premises apps can connect to the tape gateway as iSCSI devices

- Reduce costs by eliminating the use of physical backup tapes



REPLICATE DATA

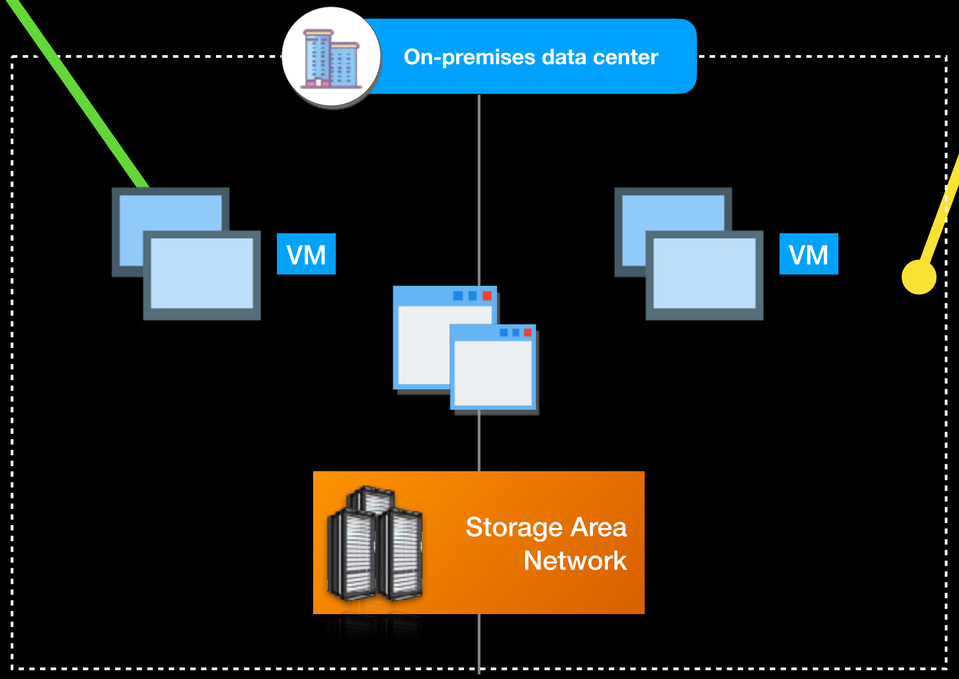
INTEGRATION

MOVE DATA

MIGRATION

AWS Storage Gateway

AWS DataSync



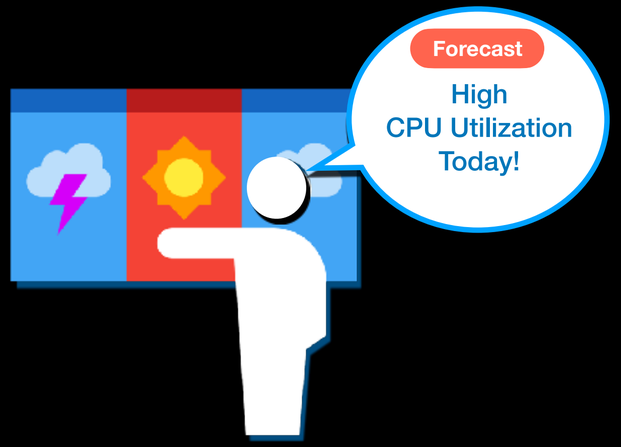
On-premises data will **still be actively used**

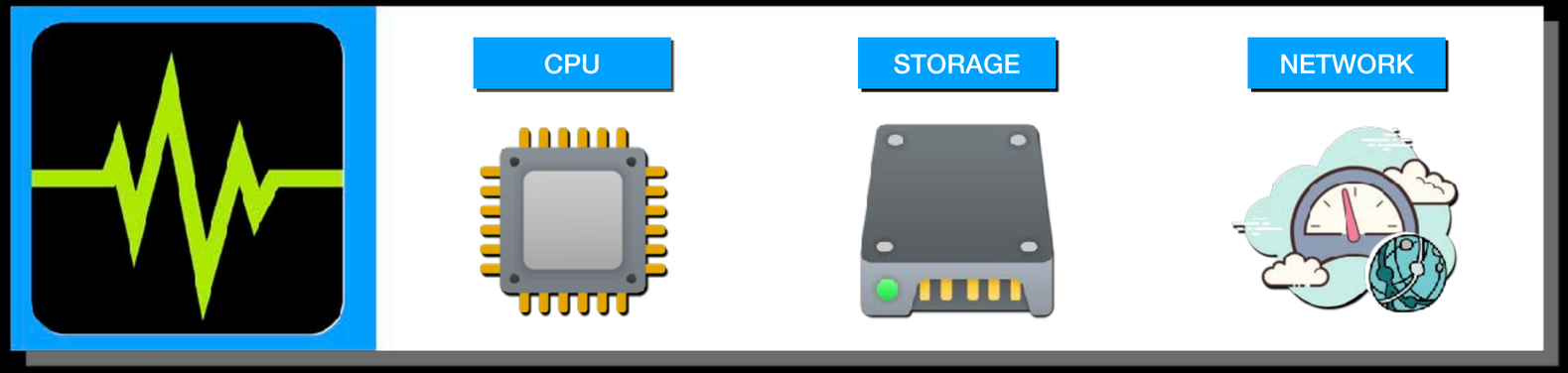
On-premises data would **not be utilized anymore**/ *will be decommissioned*

Tutorials Dojo

AWS Monitoring Services Overview

AWS Monitoring Services





AWS Monitoring Services

Amazon CloudWatch

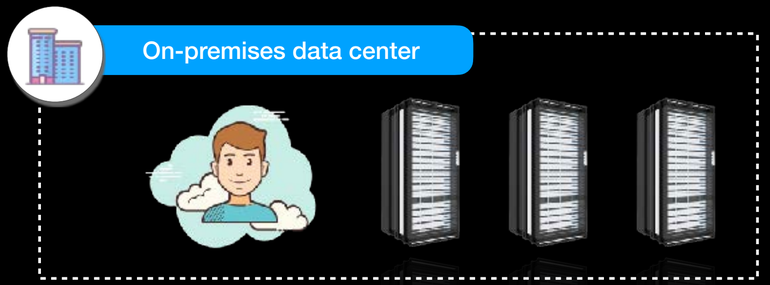
AWS Service Health Dashboard

AWS Personal Health Dashboard

AWS Health API

Amazon CloudWatch

- A **suite of AWS services** used in monitoring your systems on both:



- A **metrics repository** that collects system data from AWS services as well as your custom metrics
- **Monitors** and analyzes system metrics
- **Notifies** you if a certain threshold has been reached
- **Triggers an action** based on a specific threshold or events that you define

Amazon CloudWatch

Metrics

Logs

Alarms

Events

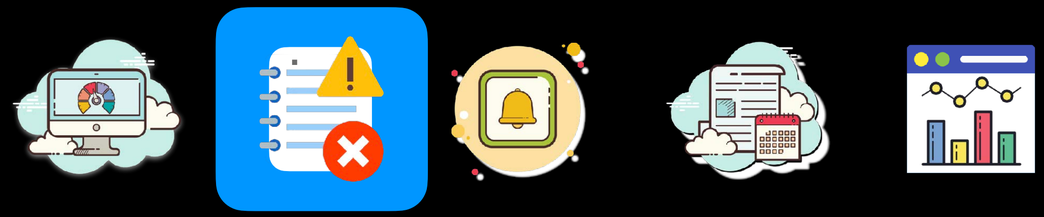
Dashboards

Amazon CloudWatch

METRICS

- **Collect metrics** from various AWS Services and your custom applications
- **Aggregate (combine) metrics** across multiple resources
- Most AWS services send metric data to CloudWatch every 1 minute by default
- For Amazon EC2, the default frequency is every **5 minutes**
- **Detailed Monitoring** sends EC2 metrics data every **1 minute**

- Primarily used for **logs** monitoring
- Allows you to **monitor, store, access, analyze or query the logs** from your AWS resources or from your custom applications
- Install **CloudWatch Logs agent** to your EC2 instances to automatically collect and publish your application logs to CloudWatch



Amazon CloudWatch

ALARMS

- Allows you to create **alarms** for your monitoring
- Performs one or more actions based on a system metric and a **specific threshold**
- Can **notify you or other systems/services** using Amazon SNS
- Can **trigger a custom action**, such as:
  - Auto Scaling your EC2 instances
  - Sending a billing alert
  - Invoking a Lambda function
  - … and many more!

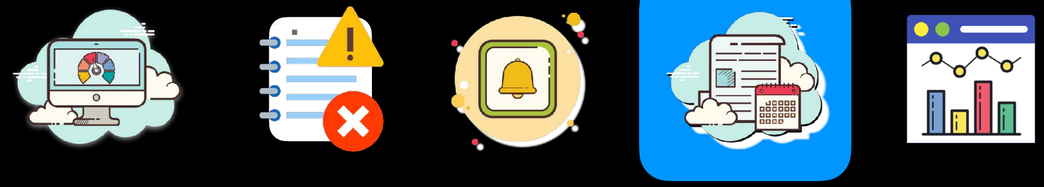
CloudWatch Events and Amazon EventBridge have the **same underlying service and API**, but the latter provides more features.

Amazon EventBridge

Amazon CloudWatch

EVENTS

- **Monitors and responds to the system/service events** of your AWS resource in near real-time
- Allows you to **create a CloudWatch Event rule** to track the changes or the state of your services
- **Invokes a certain action** if a specific event matched your Event rule
- Allows you to create a **scheduled job** that invokes a Lambda function on a regular basis, like every hour, every day, every week, or any schedule that you like.



Amazon CloudWatch

DASHBOARDS

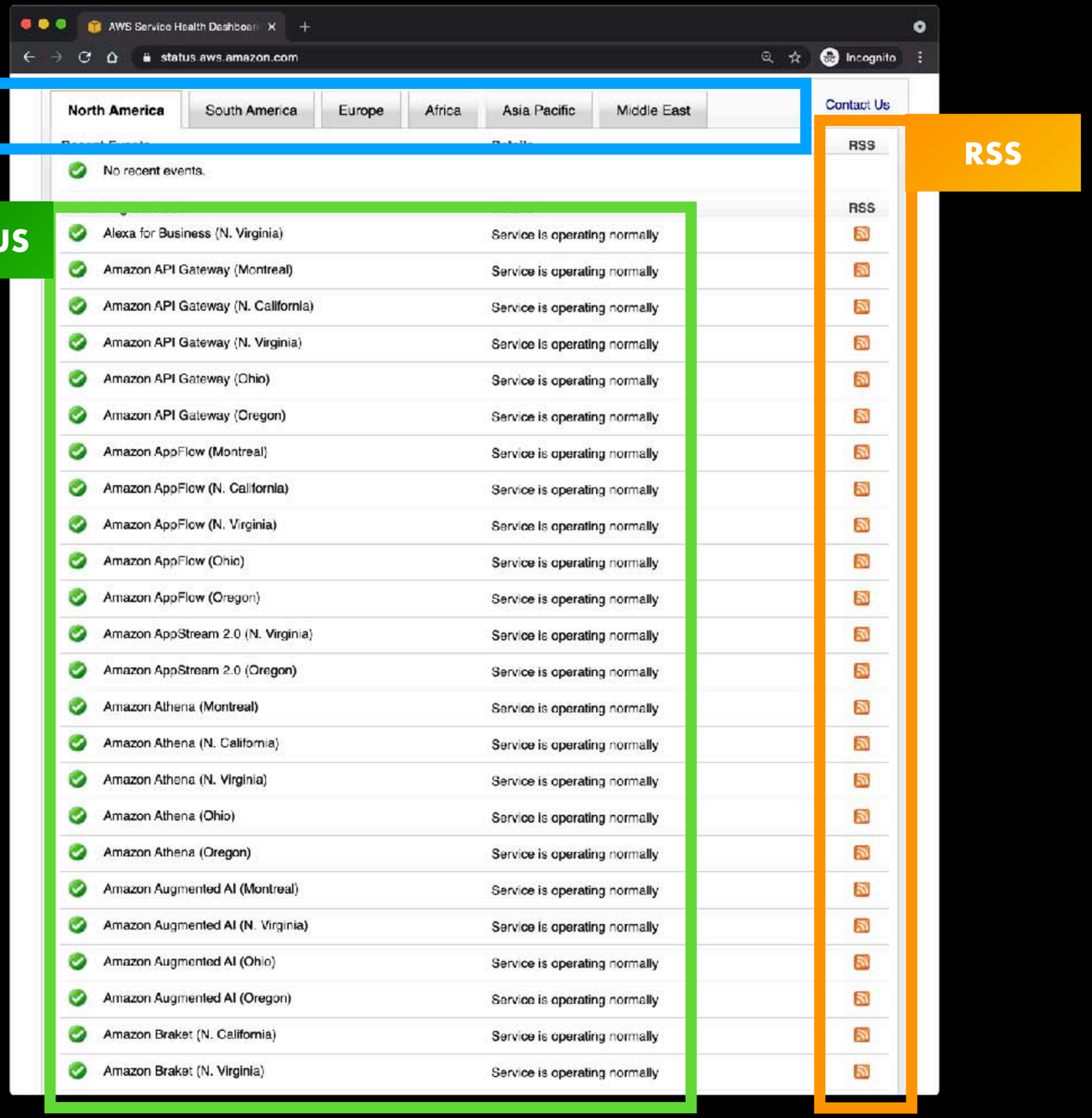
- A customizable **dashboard** containing your AWS system metrics
- **Monitor your resources in a single view**, even if those resources are located across different AWS Regions
- Allows you to publish and view your **custom metrics**

Amazon CloudWatch





AWS Service Health Dashboard

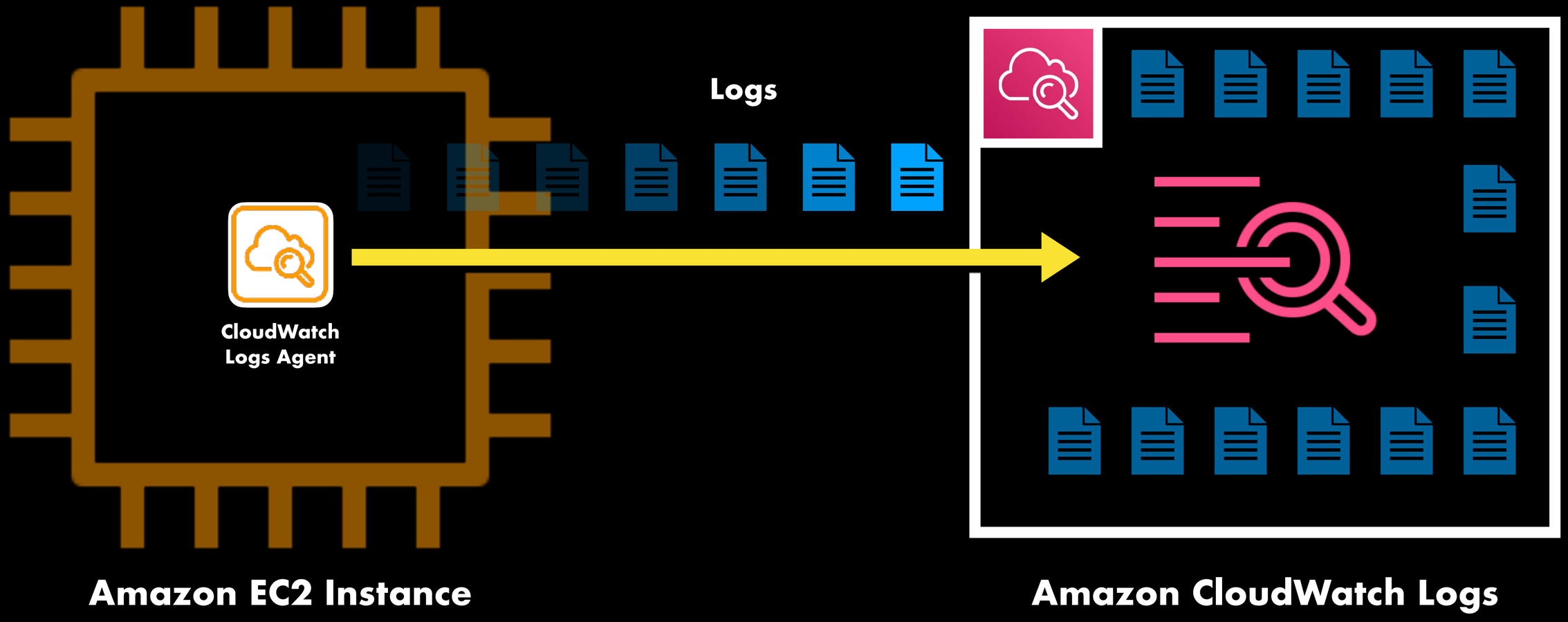


AWS Personal Health Dashboard

- A **personalized** dashboard that shows the **status of the AWS services that you are using**
- Does **NOT** show you the status of all the AWS services globally but only the status of the AWS services that you have in your account.
- **Shows the AWS Health events** that might affect your applications running on AWS such as scheduled maintenance or system outages
- **Allows you to create alerts and notifications** based on the health of your AWS resources

AWS Health API

- Provides **programmatic access to the AWS Health** information that appears in your AWS Personal Health Dashboard
- A **RESTful web service** that you can access via HTTPS
- **NOT** available by default
- Only available in **Business or Enterprise support plans**



AWS Audit & Compliance Services Overview

AWS Audit & Compliance Services

aws API

RESOURCE CHANGES

AWS Audit & Compliance Services

AWS CloudTrail

AWS Artifact

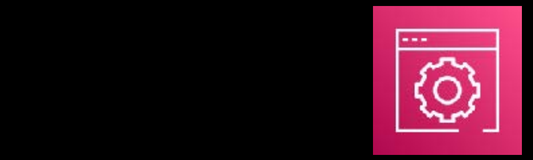
AWS Security Hub

AWS CloudTrail

- **Tracks user activity and API usage** in your AWS account
- **Stores the audit log data** in:

Amazon S3 Bucket

- **Enables risk auditing** by continuously monitoring and logging account activities, such as user actions:



AWS Management Console

AWS SDK

AWS API

AWS Command Line Interface (CLI)

AWS CloudTrail

MANAGEMENT EVENTS

Control Plane

Provide information about the management operations performed on your AWS resources

- **Attaching an IAM Role**
- **Creating a new VPC**
- **Creating a subnet**

DATA EVENTS

Data Plane

Provide information about the resource operations performed ON (e.g. S3 bucket) your resources or performed IN (e.g. S3 objects) your resources

- **Amazon S3 object-level API activities**
- **Invoking an AWS Lambda function**

AWS Artifact

- Provides on-demand **AWS security and compliance reports**
- Acts as a self-service portal to find compliance-related information and reports for:
    - ISO Reports
    - Payment Card Industry (PCI) reports
    - Service Organization Control (SOC) reports
    - . . . and many more!
- Allows you to download AWS security and compliance documents such as **SOC 1 report, ISO certifications, and other reports**

AWS Security Hub

- Provides a **centralized & comprehensive view of the security posture** of your cloud infrastructure across multiple AWS accounts
- Helps you to comply with your company's specific security standards and best practices
- **Collects security alerts and findings** from:

Amazon GuardDuty

Amazon Inspector

Amazon Macie

AWS IAM Access Analyzer

AWS Firewall Manager

AWS Networking & Content Delivery Services Overview

AWS Networking & Content Delivery Services

Amazon VPC

Elastic Load Balancing

Amazon Route 53

AWS Global Accelerator

Amazon CloudFront

AWS PrivateLink

AWS VPN

AWS Direct Connect

AWS Transit Gateway

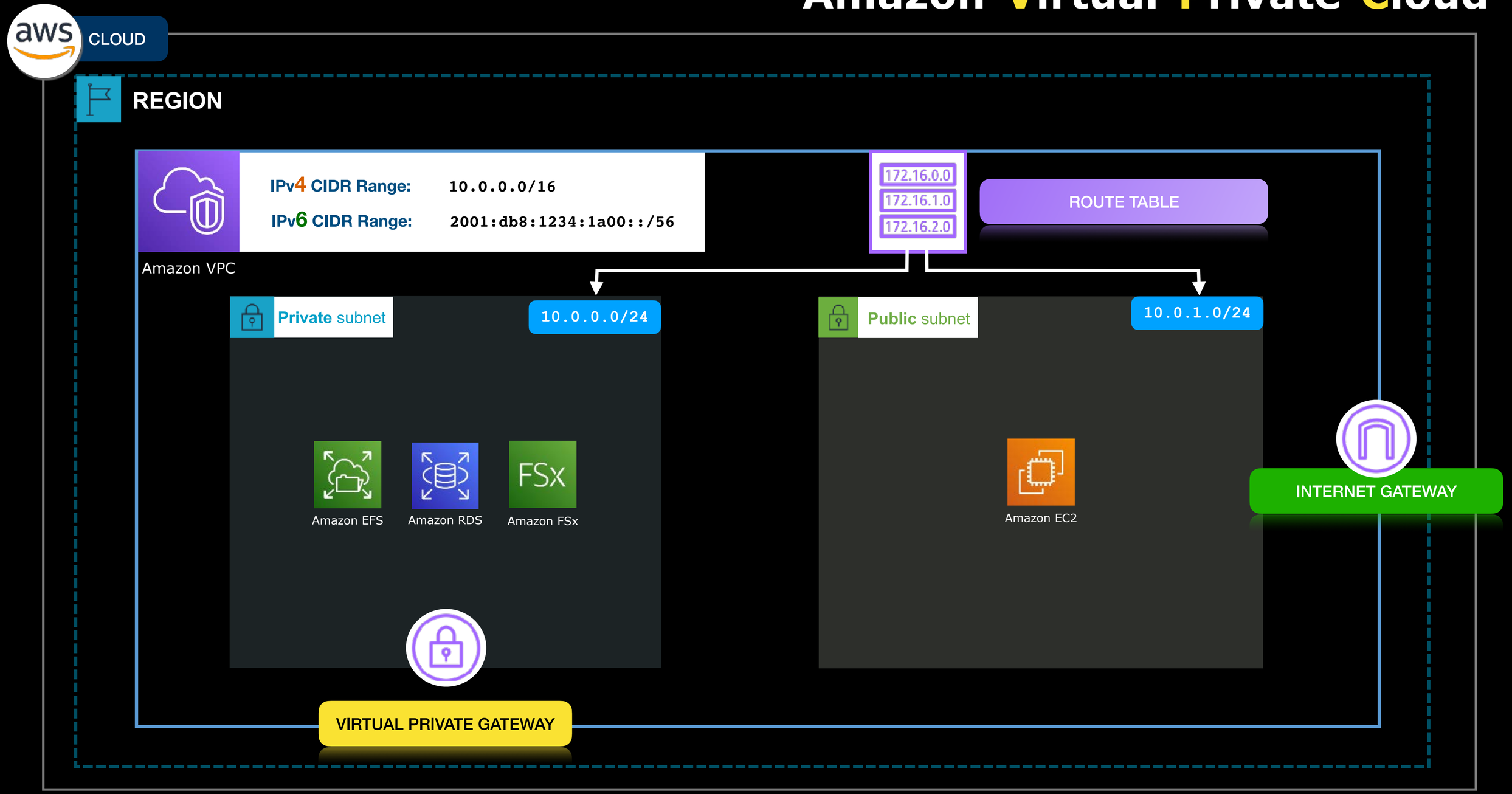
Amazon API Gateway

AWS App Mesh

AWS Cloud Map

Also categorized as an Application Integration Service

Amazon Virtual Private Cloud

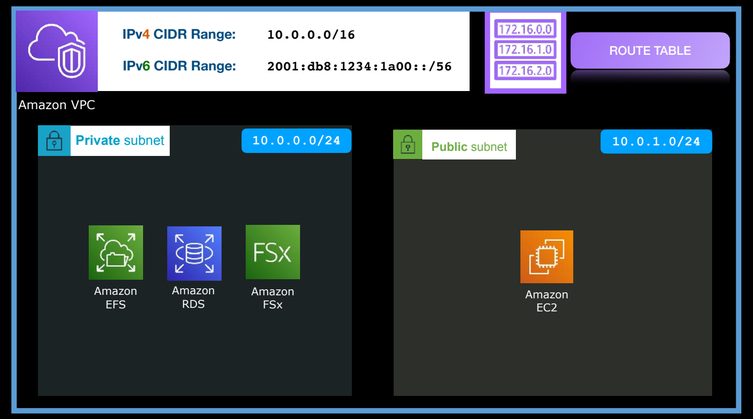


Amazon Virtual Private Cloud

aws CLOUD

ASIA PACIFIC (Singapore)

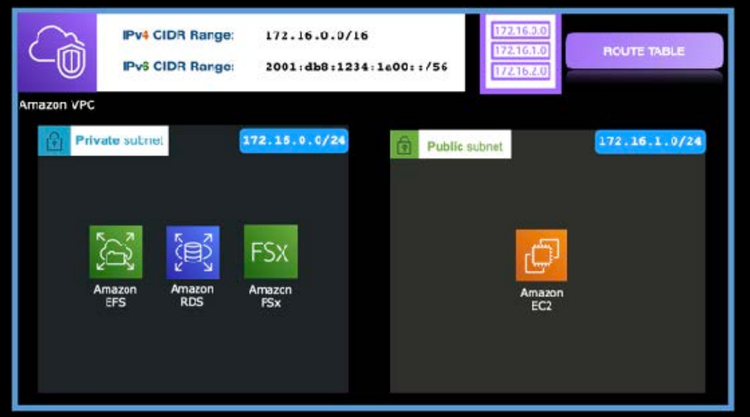
VPC A - Manila Branch



VPC Peering

US EAST (Northern Virginia)

VPC B - New York Branch



Virtual Private Cloud

Amazon VPC

Virtual Devices

VIRTUAL IP address MAC address Firewall

VIRTUAL IP address MAC address Firewall

VIRTUAL IP address MAC address Firewall

NETWORK VIRTUALIZATION

Physical Devices

PCIe Network Interface Card

Nitro Card for VPC

aws Data Center

aws Data Center

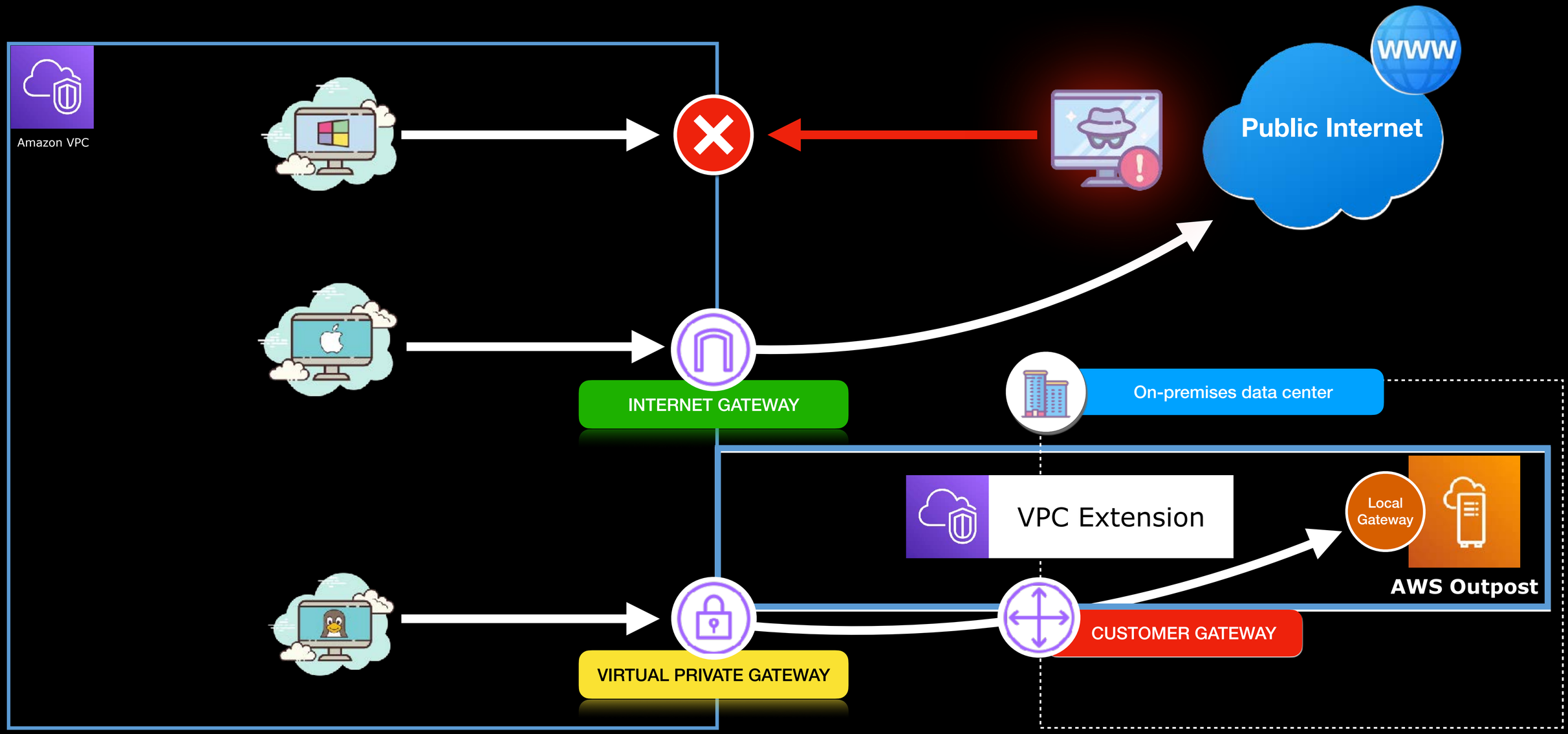
aws Data Center

aws Data Center

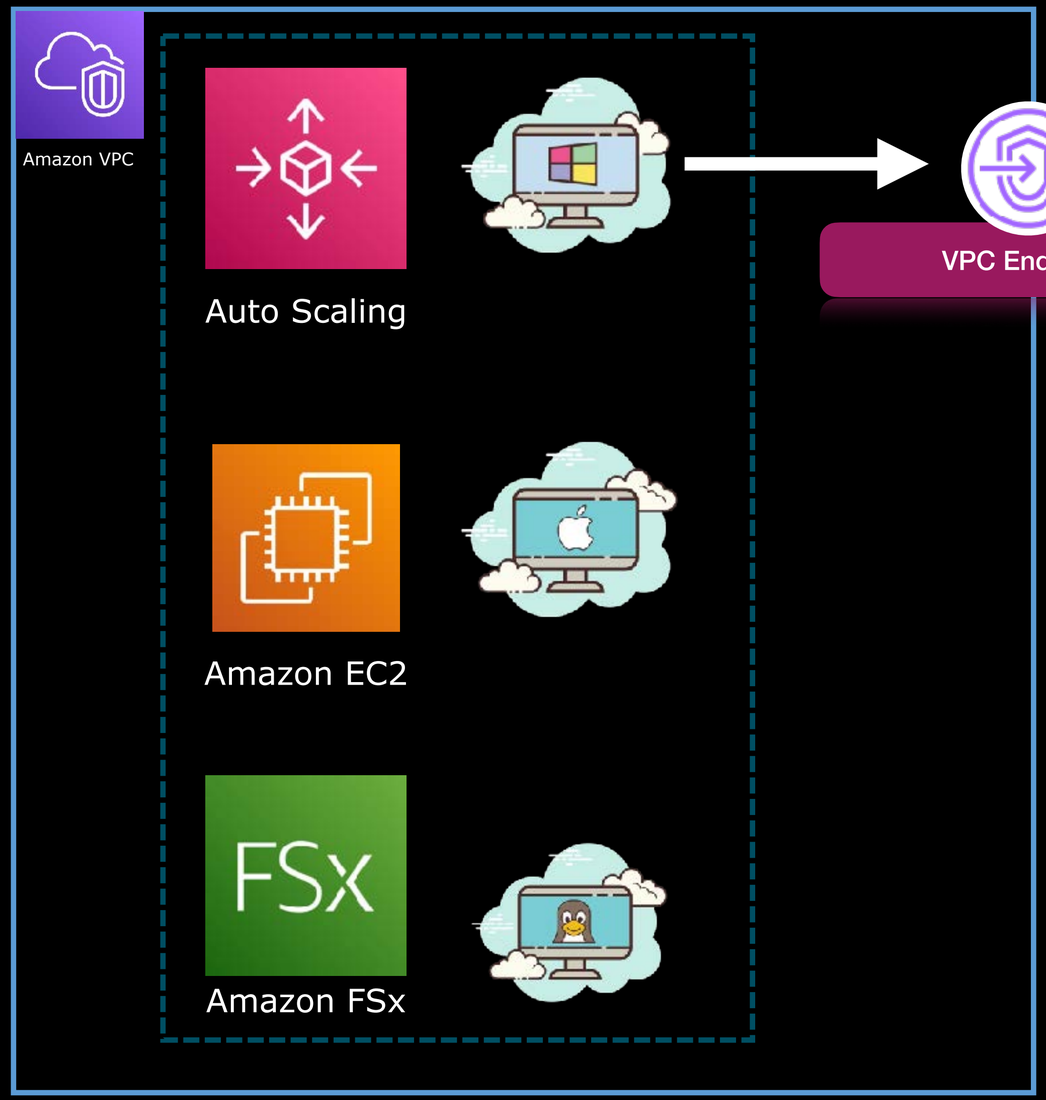
aws Data Center

aws Data Center

Virtual Private Cloud



Virtual Private Cloud



Auto Scaling

Amazon EC2

Amazon FSx

VPC Endpoint

Amazon S3

Amazon DynamoDB

Other aws Services

Elastic Load Balancing

- Automatically distributes incoming traffic across **multiple targets** such as:


Amazon EC2 Instance


Amazon ECS Task


AWS Fargate Task


AWS Lambda Function


IP Address

- It **distributes** (*load balances*) the incoming traffic to your underlying resources
- Provides **high-availability** to your web applications
- if one of your servers or EC2 instances fails (*unhealthy resource*), the request will be routed to another server (*healthy resource*)
- Routes incoming traffic across multiple Availability Zones, within a **single** AWS Region only.

Elastic Load Balancing TYPES

	Application Load Balancer (ALB)	Network Load Balancer (NLB)	Gateway Load Balancer (GWLB)	Classic Load Balancer (CLB)
PROTOCOL LISTENERS	HTTP / HTTPS gRPC	TCP / UDP TLS	IP	HTTP / HTTPS TCP SSL/TLS
USE CASES	For **web apps**, microservices & containers	Handling **millions of requests per second** while maintaining **ultra-low latencies**	Running third-party **virtual appliances** in AWS	For **legacy** applications in AWS For implementing **Custom Security Policies** and **TCP passthrough configuration**

Amazon Route 53

- A **Domain Name System** (DNS) web service
- DNS is a system that routes a **domain name** to a particular **IP address**
- **Map** domain names to:

Elastic IP address

Amazon EC2 Instance

Amazon S3 Static Website

Elastic Load Balancers

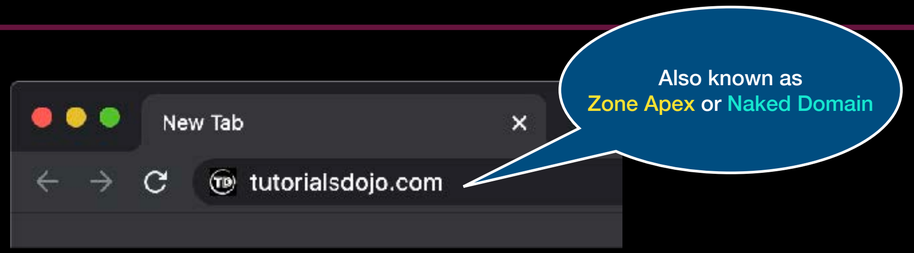
Amazon CloudFront Web Distributions

Amazon Route 53

Buy Domains

Manage Domains

Root Domain



Subdomains

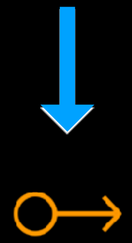
philippines.tutorialsdojo.com

blog.tutorialsdojo.com

portal.tutorialsdojo.com

cdn.tutorialsdojo.com

manila-datacenter.tutorialsdojo.com



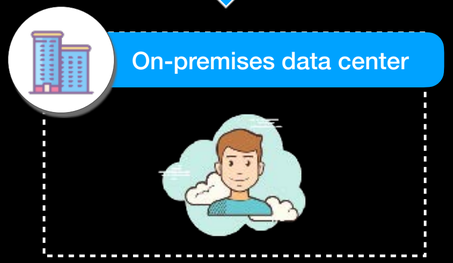
Elastic IP address

Amazon EC2 Instance

Amazon S3 Static Website

Elastic Load Balancers

Amazon CloudFront Web Distributions

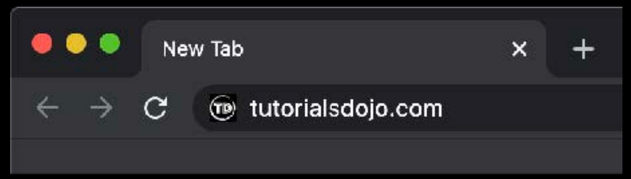


ROUTING POLICIES

Amazon Route 53

Simple | Failover | Geolocation | Geoproximity | Latency-Based | Multivalue Answer | Weighted

Root Domain



Subdomains

philippines.tutorialsdojo.com

blog.tutorialsdojo.com

portal.tutorialsdojo.com

cdn.tutorialsdojo.com

manila-datacenter.tutorialsdojo.com

Elastic IP address

Amazon EC2 Instance

Amazon S3 Static Website

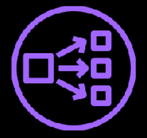
Elastic Load Balancers

Amazon CloudFront Web Distributions

On-premises data center

AWS Global Accelerator

- Provides a set of **static anycast IP** addresses
- The static IP address serves as a **single fixed entry point** to:

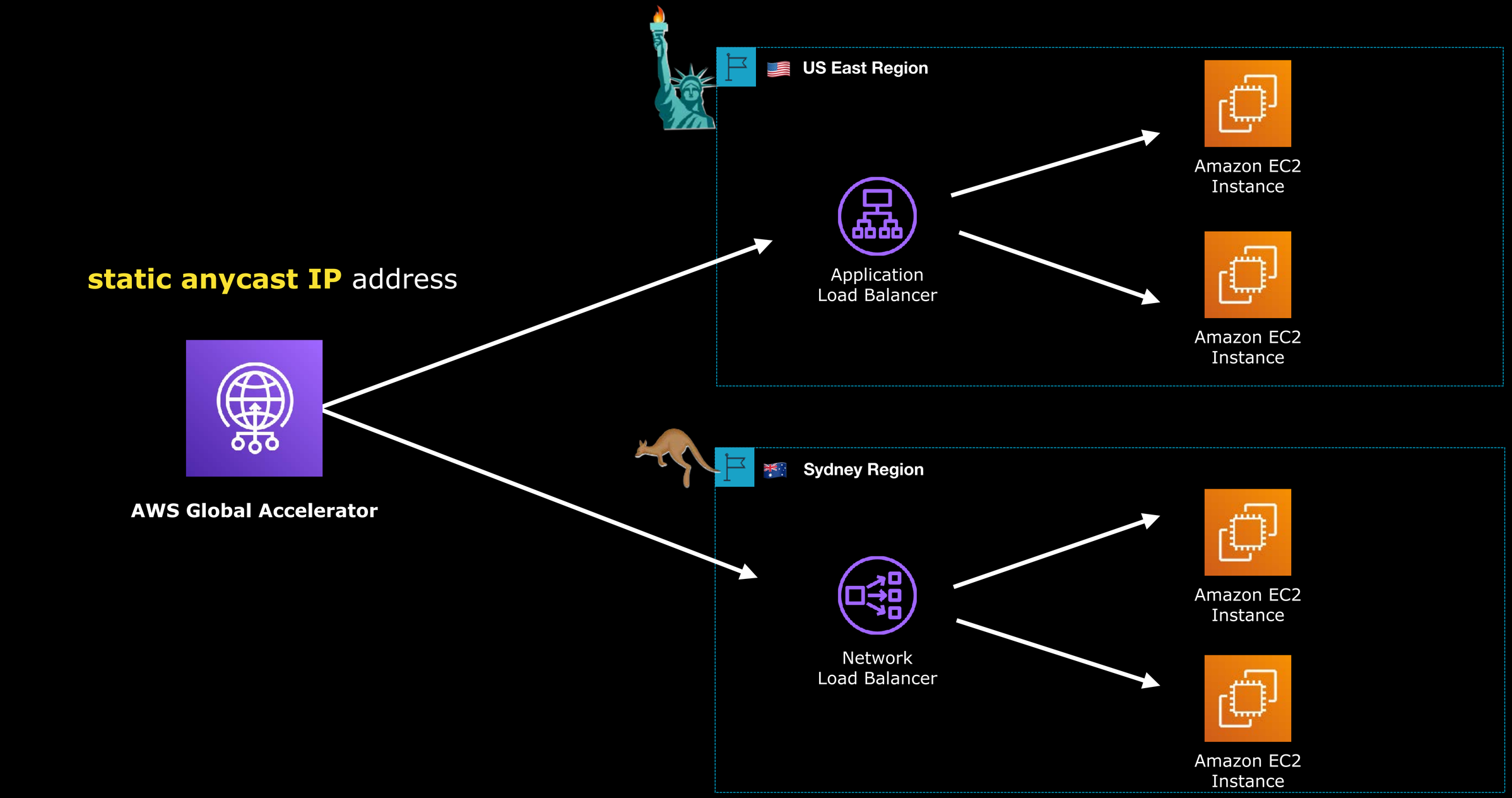


Network
Load Balancer

Application
Load Balancer

Amazon EC2
Instance

Elastic IP
address



Amazon CloudFront

- A **content delivery network** (CDN) service
- Quickly delivers static content and video stream to your clients.
- A CDN is a **globally-distributed network** of services/servers spread around the globe that stores or **caches** your files.
- **Reduces latency** by shortening the time it takes to deliver your data to your users
- **Improves the response time** of your application.
- **Caches** your images, videos, media files, or software packages

AWS PrivateLink

- Allows **private connectivity** to various AWS services
- **Does not pass through the public Internet.**
- Provides a **private endpoint** that you can use for your:

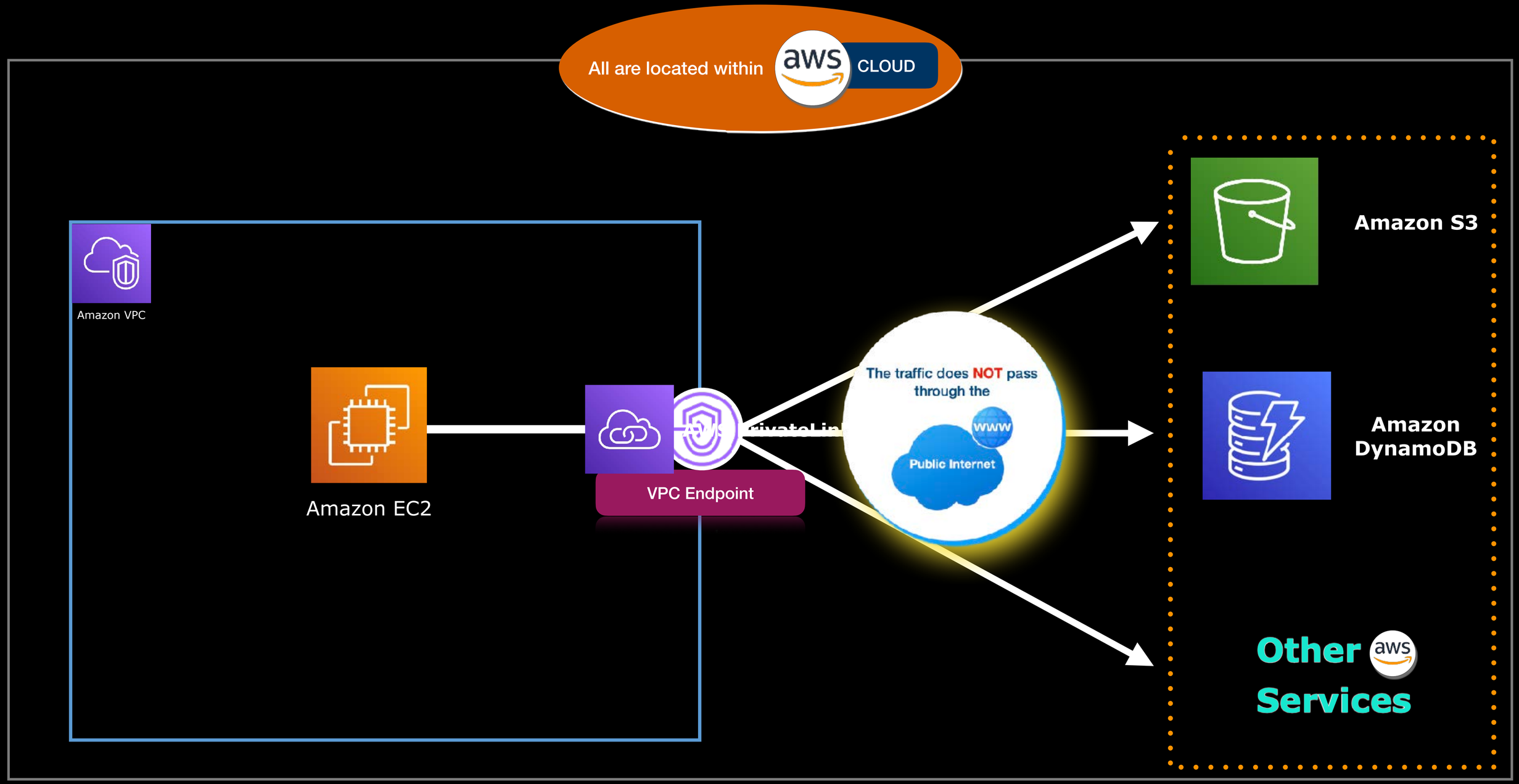
Amazon VPC

Amazon EC2

Amazon S3

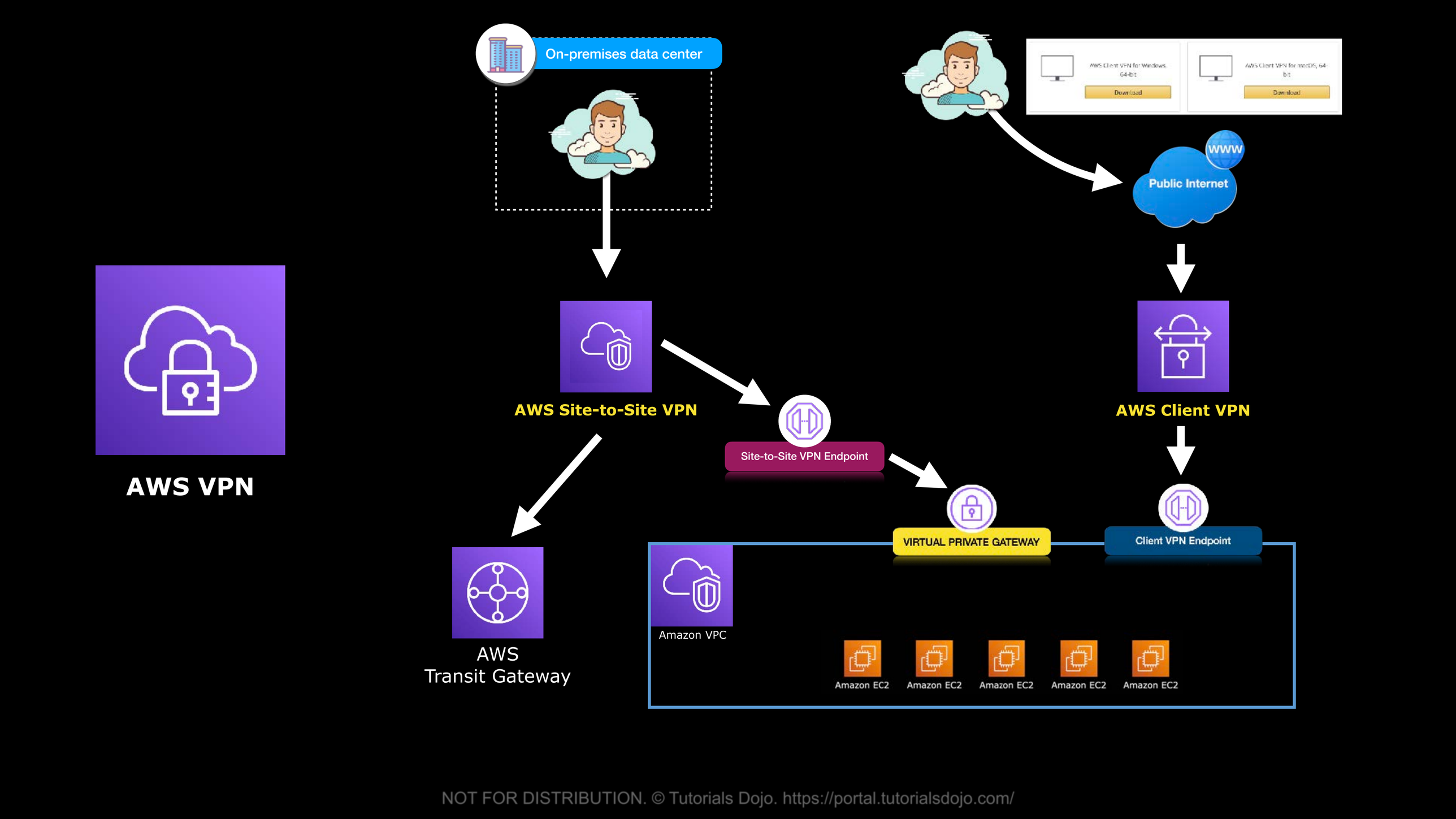
Amazon DynamoDB

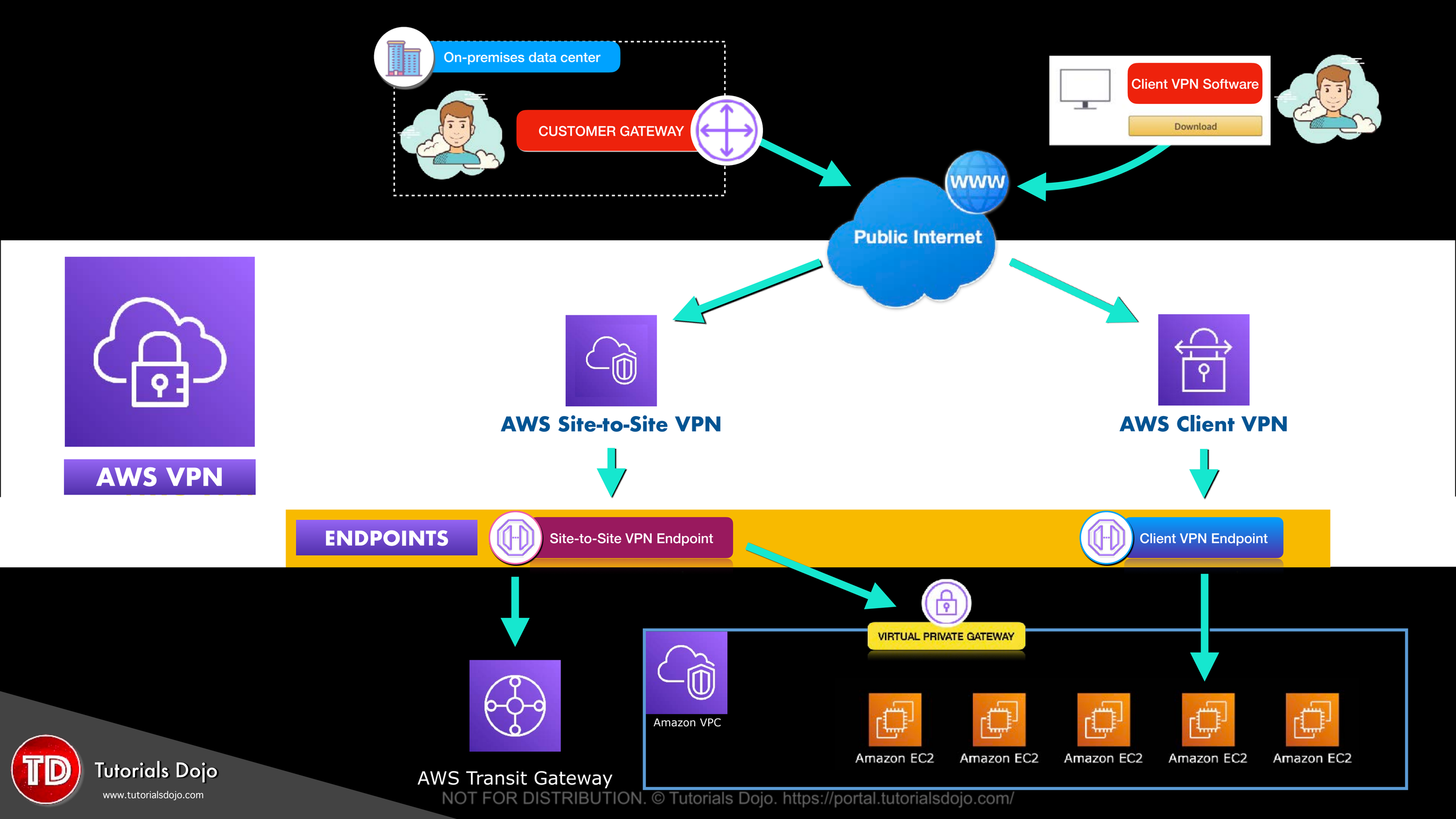
Other Services



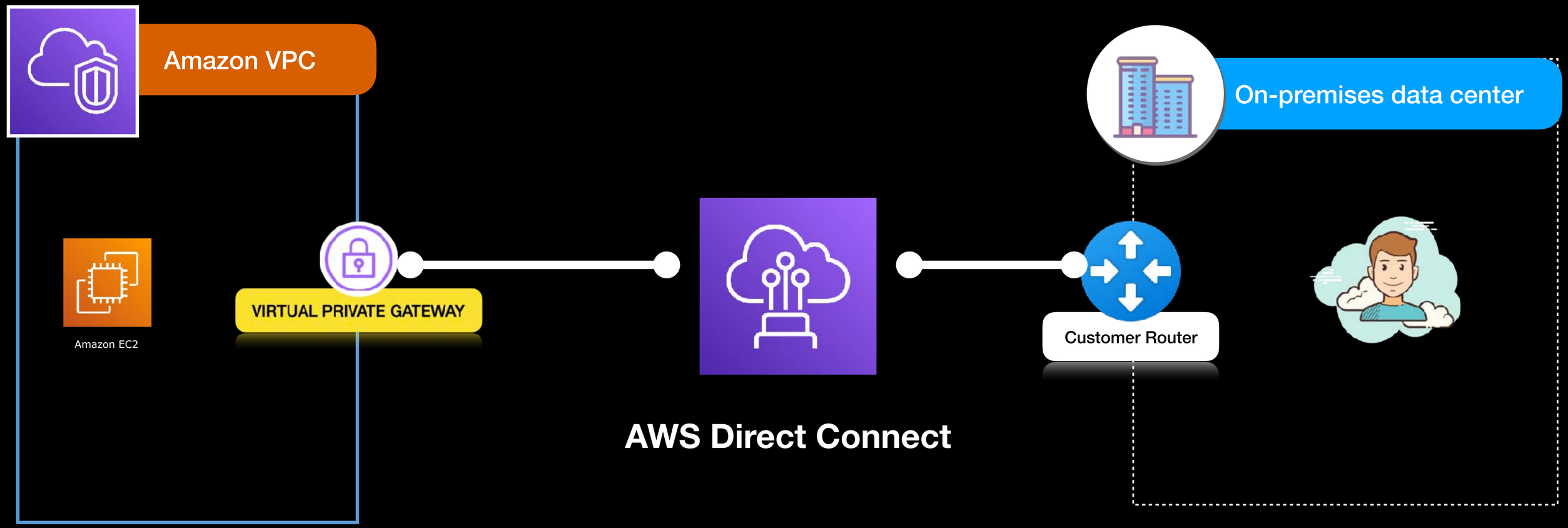
AWS VPN

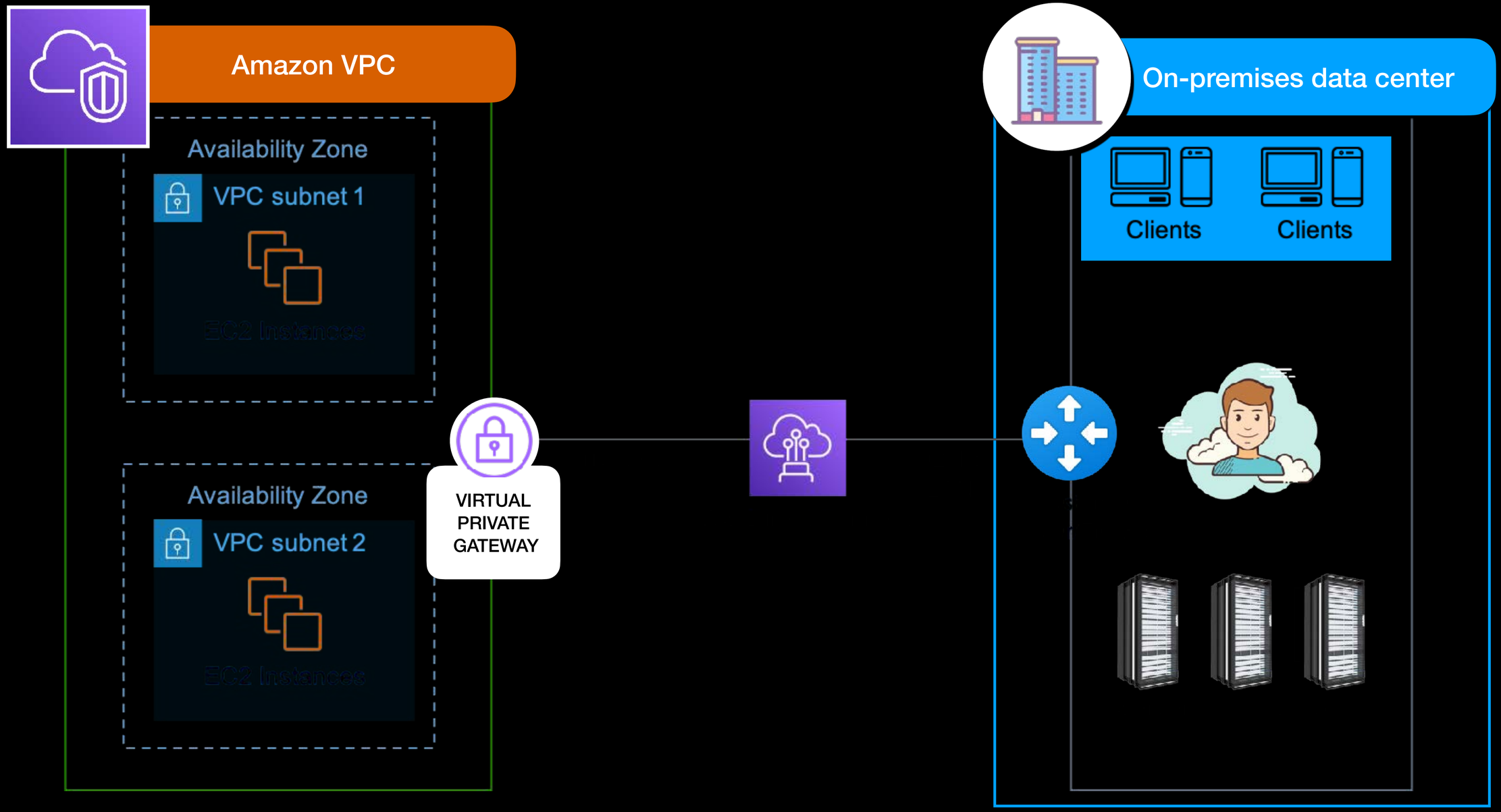
- AWS **V**irtual **P**rivate **N**etwork, or AWS VPN
- Enables you to **connect your on-premises network to AWS.**
- An encrypted connection that **passes through the public Internet.**
- Uses the **IPsec protocol** to authenticate and encrypt your data in transit.

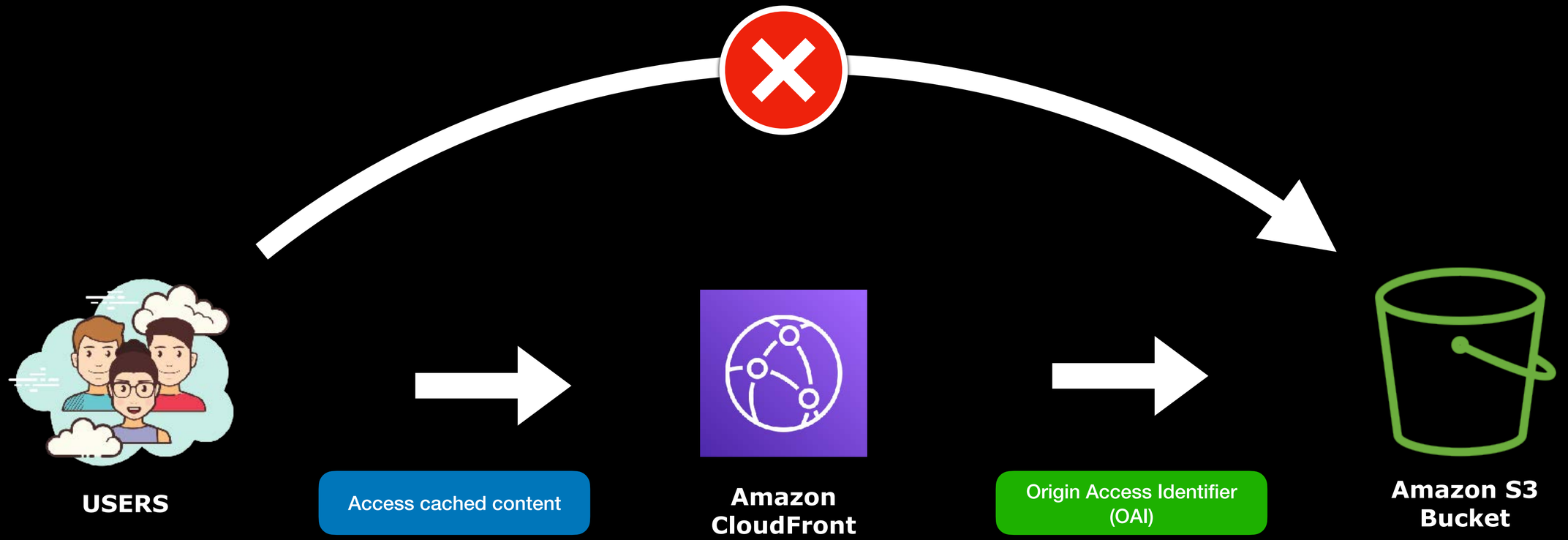


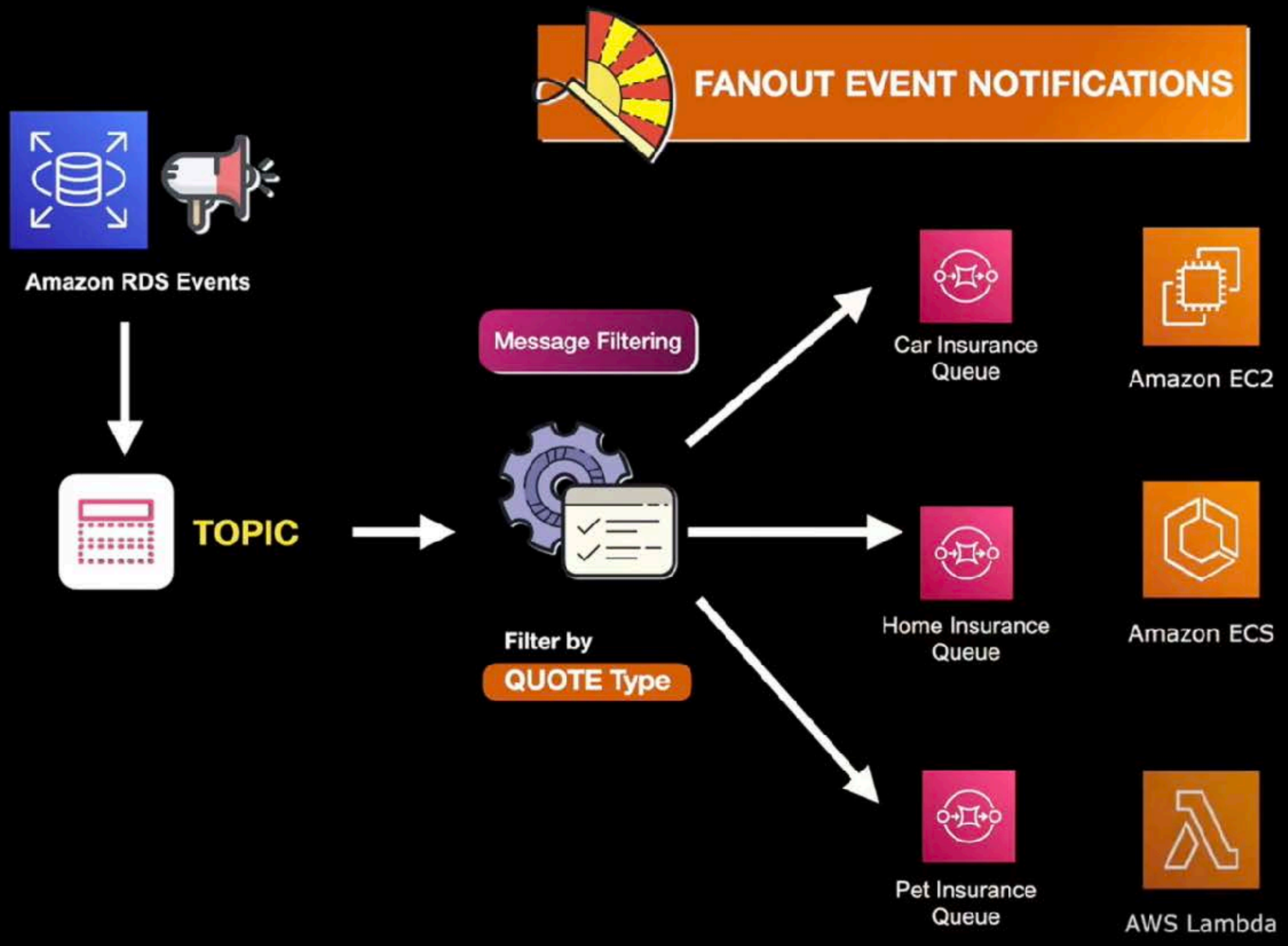


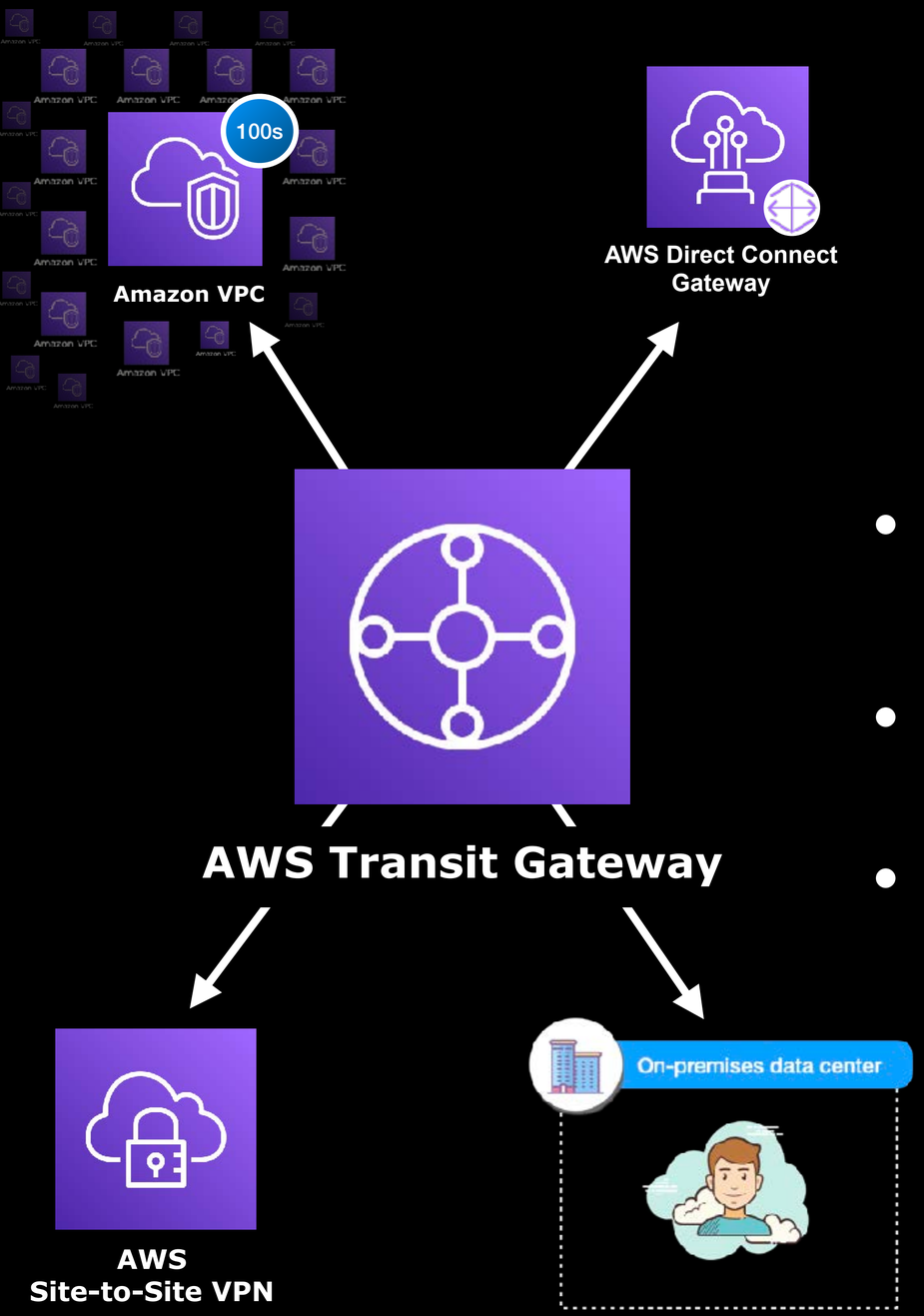
- Allows you to establish a **dedicated network connection** from your on-premises network to AWS
- Provides a **more consistent network experience** over Internet-based connections such as a VPN, and a **higher bandwidth**.
- You can create a **private virtual interface** to enable your on-premises servers to connect to the virtual private gateway of your Amazon VPC.
- You can group your virtual private gateways and private virtual interfaces using a **Direct Connect Gateway**.
- You can also use a **public virtual interface** to connect to your Amazon S3 buckets and other public resources in AWS.
- **The traffic does NOT pass through the public Internet.**



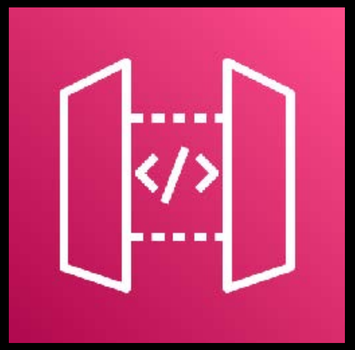








- **Connects your cloud networks** (e.g. Amazon VPCs, VPNs, Direct Connect Gateways, and on-premises networks) **to a single gateway.**
- Recommended for large organizations with 100s **hundreds of Amazon VPCs, site-to-site VPNs, and external networks.**
- Reduces the complexity of your infrastructure and makes scaling easier



Amazon API Gateway

- Allows you to publish, maintain, monitor, and secure your **RESTful APIs.**
- Also supports **WebSockets** for real-time message communication
- **Acts as a front door for your back-end services** that are running on:


Amazon EC2


Amazon ECS


AWS Fargate


AWS Lambda


AWS Elastic Beanstalk

- Works as a **Proxy** — similar to APIGEE, Mulesoft and other proxies/integration platforms

AWS App Mesh

- A **service mesh** *(an infrastructure layer that handles communication between microservices)*
- Provides application-level networking for the different types of **containerized applications** in AWS**.**
- **Allows your services to communicate with each other** across multiple types of computing infrastructure.
- Uses envoy *(an open-source service mesh proxy)*
- Can be used with microservice containers managed by:


Amazon ECS


Amazon EKS


AWS Fargate

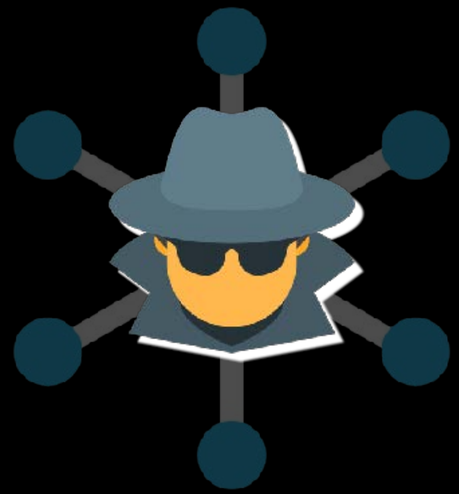

Amazon EC2

AWS Cloud Map

- A **cloud resource discovery** service.
- Commonly used in microservices and containerized applications that have **dynamically changing resources.**
- You can **name your containerized application resources** with **custom names.**
- Improves your containerized applications in AWS by always discovering the **most up-to-date locations of your resources**
- **Improves the availability** of your system.

AWS Security Services Overview

AWS Security Services



7 Open Systems Interconnection (OSI) Model Layers

DDOS

Distributed **D**enial-**O**f-**S**ervice Attack

UDP TCP

IP

SYN SYN SYN SYN SYN SYN SYN SYN SYN

SYN ACK

ACK ACK ACK

AWS Security Services

AWS Web Application Firewall (AWS WAF)

AWS Firewall Manager

AWS Shield

Amazon GuardDuty

AWS CloudHSM

AWS Key Management Service (AWS KMS)

AWS Secrets Manager

AWS Certificate Manager (AWS ACM)

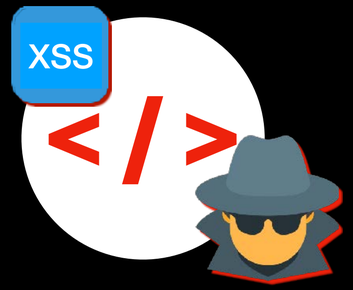
Amazon Macie

Amazon Inspector

Amazon Detective

AWS Web Application Firewall (AWS WAF)

- A **web application firewall** service
- Protects your web applications from **common web exploits**
- Allows you to create **custom rules** that block common attack patterns such as:



- **Can be integrated with**:

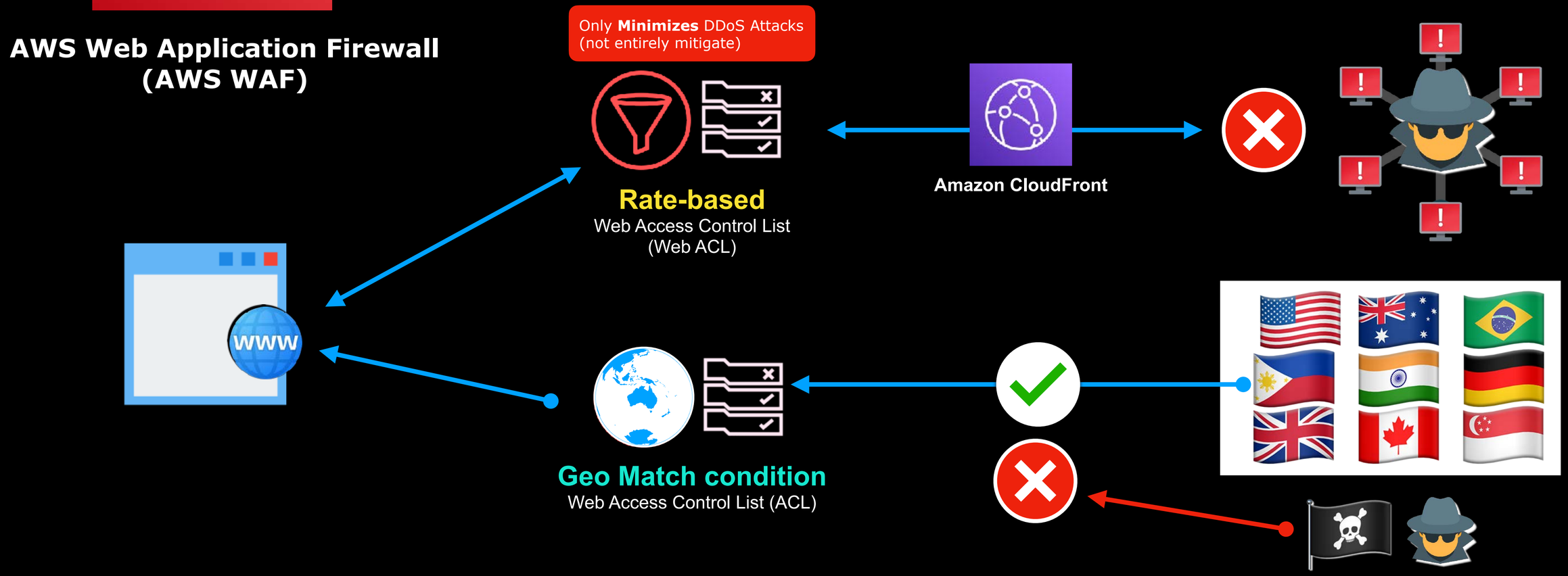
Amazon CloudFront

Application Load Balancer

Amazon API Gateway

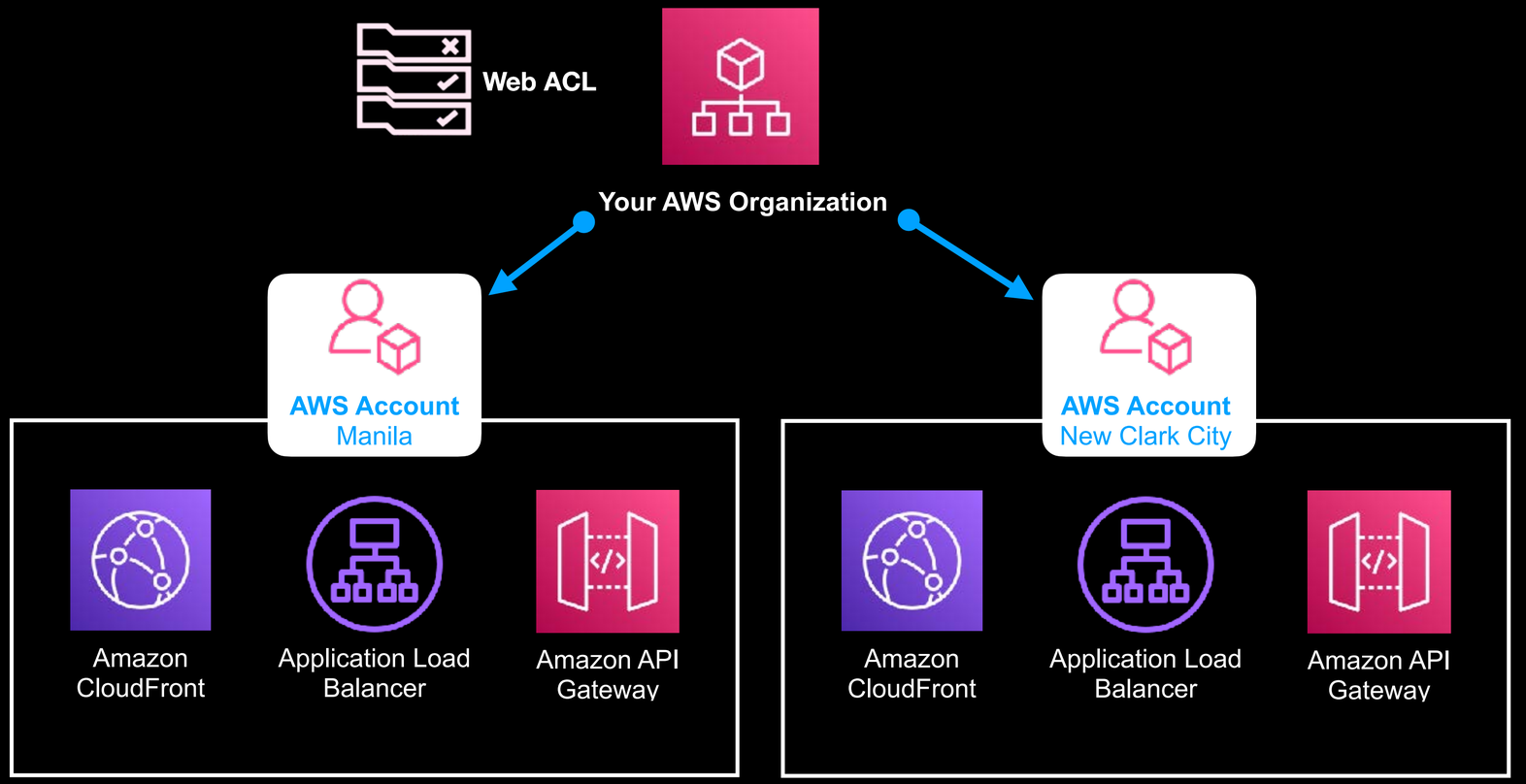
AWS Web Application Firewall (AWS WAF)

- Has an **IP Match condition** feature, you can block malicious requests from a recurring set of IP addresses.
- Can protect your application from **illegitimate requests** sent by illegitimate external systems, through its **rate-limiting rule.**



- A **security management service** designed for: AWS WAF Rules
- Allows you to **centrally configure and manage WAF rules across multiple AWS accounts** and applications.
- Enables you to roll out your custom rules to your **AWS Organization**

AWS Firewall Manager



AWS Shield

- A **managed DDoS protection** service
- Provides **detection and automatic mitigations** that minimize application downtime and latency.
- **Mitigate different types of flood attacks** such as UDP reflection, SYN flood, DNS Query flood, and HTTP flood attacks.
- Protects your applications that use:


Amazon EC2


Elastic Load Balancer


Amazon CloudFront


AWS Global Accelerator

Amazon Route 53

- Two Tiers:
  - **Standard**
    - Built-in by default
    - No extra charge
  - **Advanced**
    - Has an additional charge
    - Provides access to real-time DDoS attack notification
    - DDoS Response Team (DRT) supports you during DDoS Attack

Amazon GuardDuty

- A **managed threat detection** service
- **Identifies malicious or unauthorized activities** in your AWS accounts and workloads.
- **Monitors activities** such as unusual API calls, cryptocurrency mining, or potentially unauthorized deployments that indicate a possible account compromise.
- Also detects potentially compromised:

Amazon EC2 Instances

- Produces security reports called:

Findings

- Able to **send notifications using CloudWatch Events** when a change was detected
- **NOT capable of doing any resource changes** by itself, like rate-limiting protection or DDoS attack mitigation.

AWS CloudHSM

AWS Key Management Service (AWS KMS)

- A fully managed, **cloud-based hardware security module** or HSM.
- The **HSM** in Cloud**HSM** means: **Hardware Security Module**

AWS CloudHSM

- Enables you to easily **generate and use your own encryption keys.**
- Encryption keys can be in 128-bit or 256-bit

HSM

Hardware Security Module

- A **physical hardware device**
- Performs **cryptographic operations**
- Securely stores cryptographic **key material**

AWS CloudHSM

Leading HSM Providers

THALES

utimaco®

IBM

SWIFT

yubico

Hewlett Packard Enterprise

- A random, **Base64 or hexadecimal string**
- **Binary** format (`.bin`)
- Used by your **encryption key.**

AWS CloudHSM

- The **CloudHSM clients** is installed and **hosted in your**:

Amazon EC2 Instances

- The HSM cluster is **deployed in your:**

Amazon VPC

- **Single Tenant** — Only used by one tenant or user (you)

- Can be used to:
  - Offload SSL Processing
  - Enabling Transparent Data Encryption (TDE) for Oracle databases
  - Protecting the private keys for an Issuing Certificate Authority (CA).

- Integrate CloudHSM and  AWS KMS to create a **custom key store.**

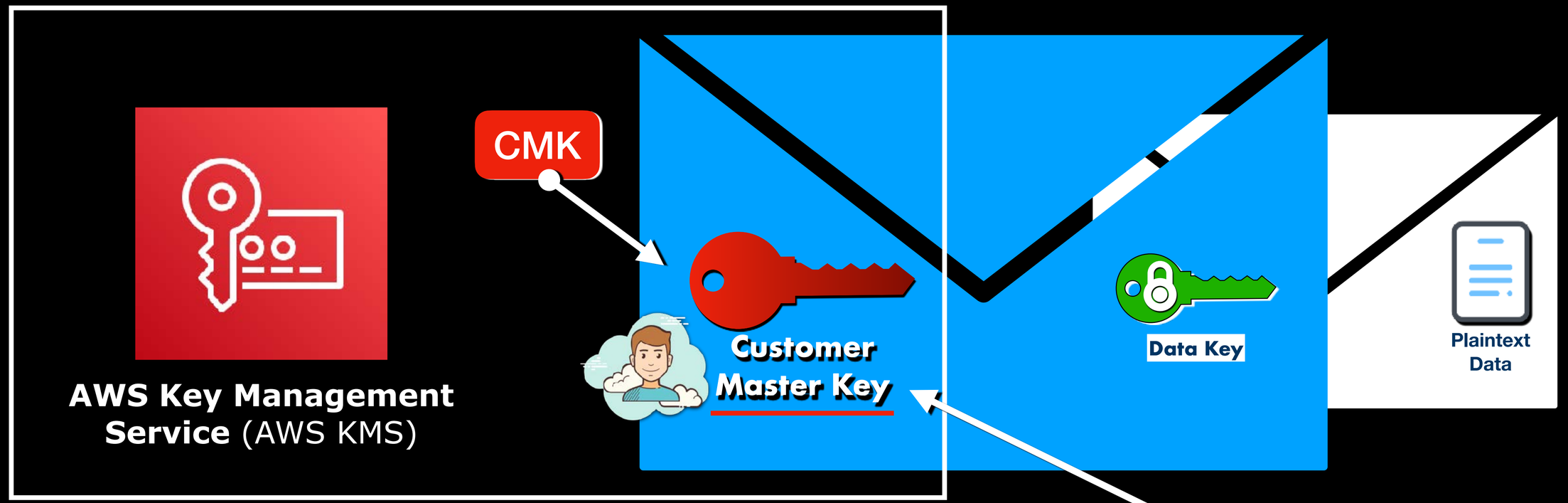
AWS Key Management Service (AWS KMS)

- A managed service that **works like:** **AWS CloudHSM**
- Internally, it **also uses hardware security modules (HSMs)** for creating and controlling your encryption keys.
- Has **multi-tenant access**

You **share** the HSM with other tenants or AWS customers

- Unlike CloudHSM, you **cannot launch the HSM to Amazon VPC or EC2 instances** *(as clients with direct HSM access)* that you own.
- Can be integrated with other AWS services to help you protect the data you store with these services.

ENVELOPE ENCRYPTION



- AWS KMS **automatically rotates** your CMK

AWS Key Management Service (AWS KMS)

- You can also create a **custom key store** in AWS KMS with

AWS CloudHSM

- Provides complete control over your **encryption key lifecycle management**
- Allows you to **remove the key material** of your encryption keys.

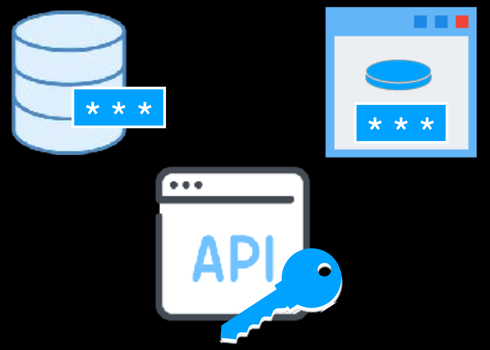
- You can **audit key usage independently** of:

AWS CloudTrail

AWS KMS

AWS Secrets Manager

- Protect the **secrets** of your applications, services, and IT resources.
- Enables you to easily **rotate, manage, and retrieve** your secrets
- A **secret** can be:
  - A database password
  - API key
  - Authentication token
  - Other sensitive data



- Eliminates hardcoded sensitive information in plain text in:

AWS Lambda

- Offers **secret rotation** with built-in integration for:

Amazon RDS

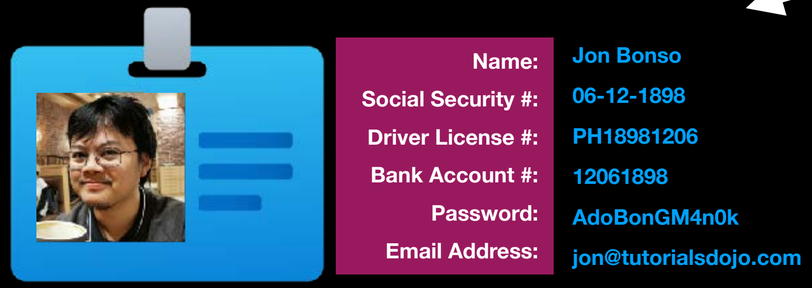
Amazon Redshift

Amazon DocumentDB

Other Services

- **Control access to secrets** using fine-grained permissions and centrally audit your secrets.
- **Not recommended** for storing encryption keys or key materials since it does not use an HSM

Amazon Macie



- A fully managed **data security and data privacy** service
- Automatically **recognizes and classifies sensitive data** or intellectual property
- **Uses machine learning** to automatically discover, classify, and protect sensitive data stored in your:

Amazon S3 bucket

Other Services

- Recognizes sensitive data such as **personally identifiable information** or PII.
- **Provides dashboards and alerts** that give visibility into how sensitive data is being accessed or moved.

AWS Certificate Manager
(AWS ACM)

- Provisions, manages, and deploys public and private Secure Sockets Layer/Transport Layer Security **(SSL/TLS) certificates**
- Enables you to **create private certificates** for your internal resources and **manage the certificate lifecycle centrally**
- SSL Certificates are **free of charge for ACM-integrated services** such as:

Amazon API Gateway

Elastic Load Balancing

Amazon Inspector

- An **automated security assessment** service
- Improves the security and compliance of applications deployed on your AWS cloud infrastructure
- **Automatically assesses applications for vulnerabilities** or deviations from best practices.
- Produces a detailed list of security findings prioritized by level of security risk severity
- Provides an **automated security assessment report** that will identify unintended network access to your:

Amazon EC2 Instances

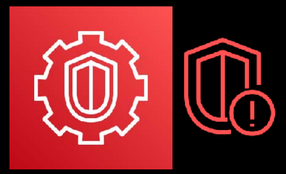
- The detailed assessment reports are available via the Amazon Inspector console or API

Amazon Detective

- Helps you **detect the root cause of your security issues** easier
- It analyzes, investigates, and quickly identifies the potential security issues or suspicious activities in your AWS infrastructure
- **Automatically collects log data from various AWS resources** such as:

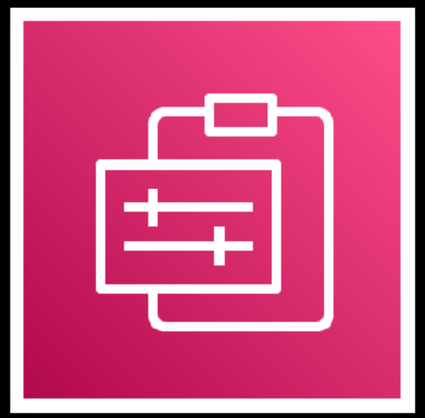
AWS CloudTrail

VPC Flow Logs



GuardDuty Findings

- **Uses machine learning** to analyze and conduct security investigations.



AWS Management & Governance Services Overview

AWS Management & Governance Services

S O P

Standard Operating Procedures

H I P A A

Health Insurance Portability and Accountability Act of 1996

G D P R

General Data Protection Regulation

AWS Management & Governance Services

– control resources

AWS Management Console

AWS Command Line Interface (AWS CLI)

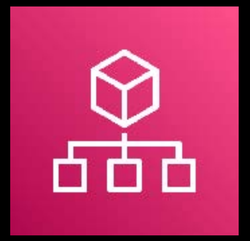
AWS Console Mobile Application

AWS Resource Access Manager

AWS Systems Manager (SSM)

– enforce standards
– ensure compliance

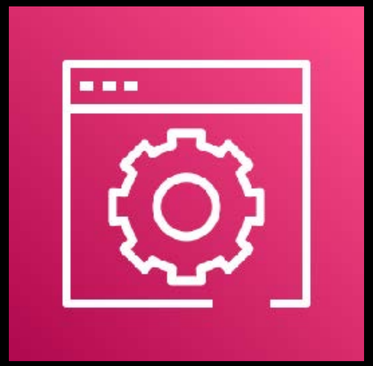
AWS Config



AWS Organizations

AWS Service Catalog

AWS Control Tower



AWS Management Console

- A **web interface** to control your AWS resources
- Accessible through your **web browser**
- Log in using your IAM username and password
- Supports **Multi-Factor Authentication** (MFA)
- Accessible via this URL: `https://console.aws.amazon.com`

AWS Command Line Interface
(AWS CLI)

- A **command-line interface** to control your AWS resources
- Accessible through your terminal, command prompt or Windows PowerShell

- Allows you to **develop custom shell scripts** that invoke different AWS CLI commands

AWS Console Mobile Application

- The **official mobile app** provided by Amazon Web Services
- Allows you to monitor your resources through a dedicated dashboard
- Enables you to view your configuration details, metrics, and alarms of **select AWS services** (not all services) on your mobile device
- Provides an overview of the account status, real-time CloudWatch metrics, Personal Health Dashboard, and AWS Billing
- Has **limited capabilities** compared with:

AWS Management Console

AWS CLI

AWS Systems Manager
(SSM)

- A **suite of services** that allows you to manage your resources
- Allows you to **control both of your AWS Cloud and on-premises** infrastructure
- Composed of:

Session Manager

State Manager

Patch Manager

Automation

Maintenance Windows

Run Command

Parameter Store

Others

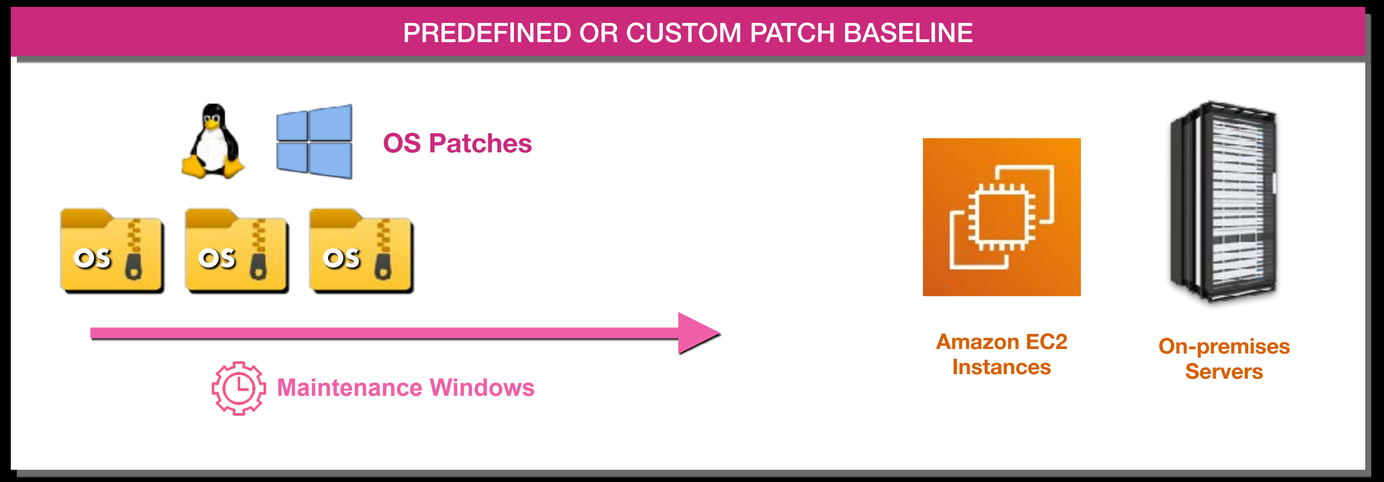
- Also has an **SSM agent** that you can install on your EC2 instances or on-premises servers to centrally manage your resources

Amazon EC2 Instances

On-premises Servers

AWS Systems Manager
(SSM)

Patch Manager



State Manager

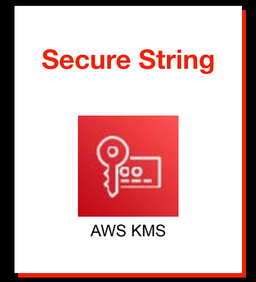
STATE

- Installed softwares (e.g. startup script, antivirus etc)
- Server configurations
- Firewall settings
- Associate Ansible playbooks, Chef recipes, PowerShell modules, and other SSM Documents

Parameter Store

PARAMETER

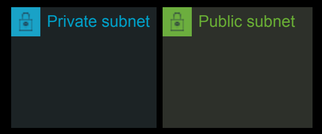
- Passwords
- Database Strings
- Amazon Machine Image (AMI) IDs
- License Codes
- Environment Variables



AWS Resource Access Manager (AWS RAM)

- Enables you to easily and **securely share your AWS resources** with any AWS account or within your AWS Organization
- Allows you to share:

AWS Transit Gateway



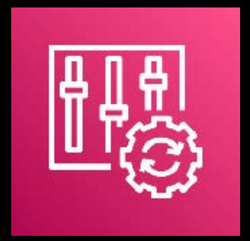
Subnets

AWS License Manager

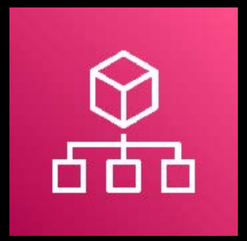
Amazon Route 53 Resolver

Other AWS Resources

- **Eliminates the need to create duplicate resources** in multiple accounts
- **Reduces the operational overhead** of managing multiple resources in each and every single account you own.



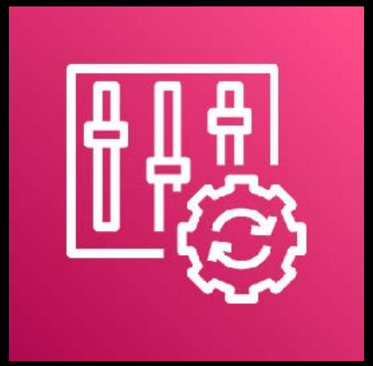
AWS Config



AWS Organizations

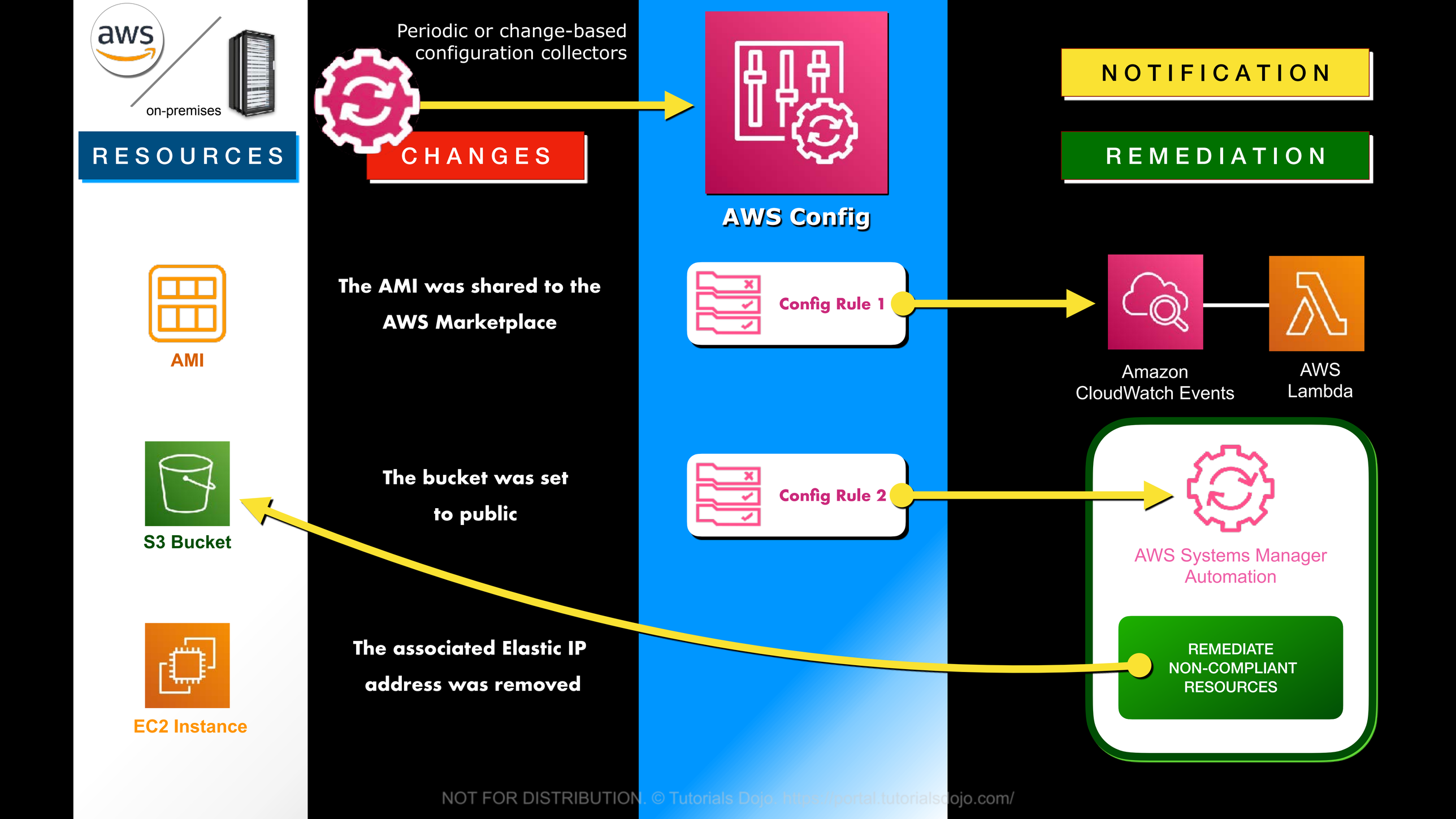
AWS Service Catalog

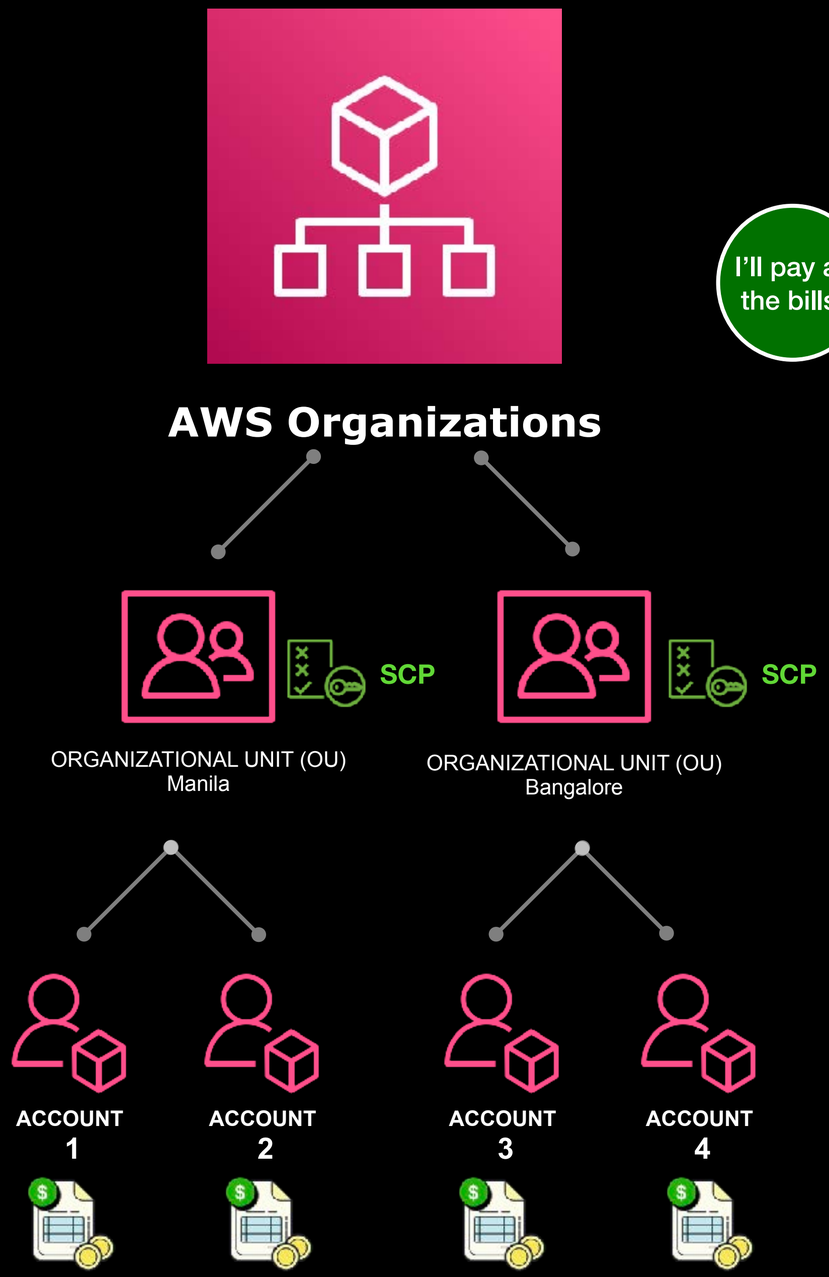
AWS Control Tower

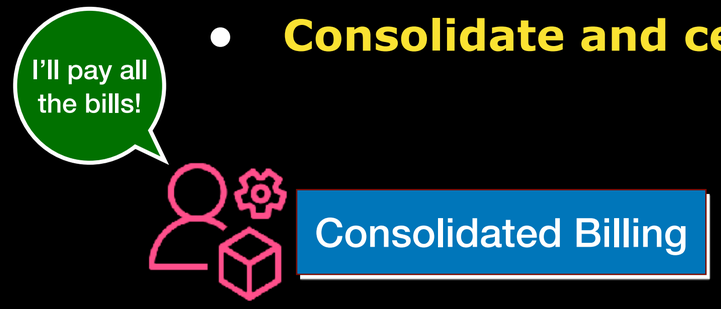


AWS Config

- Enables you to **assess, audit, and evaluate the configurations** of your AWS resources
- Automates your **compliance assessment** process
- **Provides visibility on the existing configurations** of your various AWS services and third-party resources (such as your on-premises servers)
- Enables you to **identify the changes** made to a specific resource over time

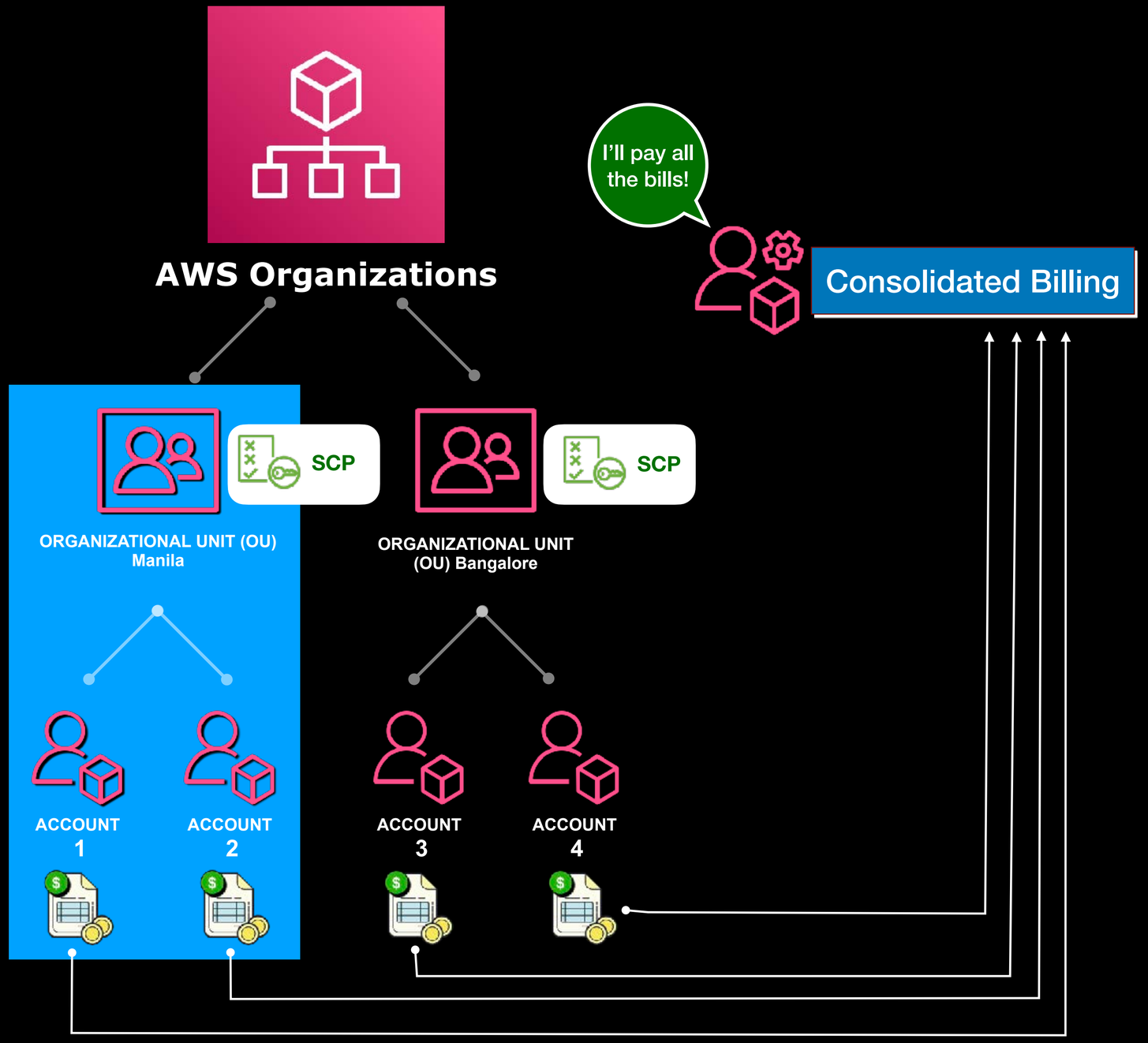






- **Consolidate and centrally manage** multiple AWS accounts
  - Combines the bills of multiple AWS accounts
  - Provides **volume discounts** to further lower down your costs
- Uses **Service Control Policies (SCP)** to control access and ensure organizational compliance across your AWS accounts
- Offers **Central Logging** to monitor all activities performed across your organization using AWS CloudTrail
- **Aggregate data from all your AWS Config rules** to quickly audit your environment for compliance.

A single AWS Organization can have two or more Organizational Unit (OU) and underlying AWS accounts with Service Control Policies (SCPs) attached



AWS Service Catalog

- Empowers you to **set up and centrally manage catalogs** of approved IT services
- Allows you to manage various IT services, referred to as "**products**" in Service Catalog then group them in a portfolio

PRODUCT

- Machine image (AMI)
- Application server
- Program
- Tool
- Database
- Other services

- Assists you in meeting your compliance requirements
- **Enforce granular access control** to your resources

AWS Control Tower

- Helps you **set up and govern a secure multi-account** AWS environment
- Automates the setup of your multi-account AWS environment
- Uses **blueprints** that follow AWS best practices for security and management
- Provides mandatory high-level rules called **guardrails**
- Help enforce your policies using service control policies (SCPs)
- **Detect policy violations** using AWS Config rules

IAM Overview

Identity and Access Management

AUTHENTICATION

AUTHORIZATION

Identity

AUTHENTICATION

IAM ENTITIES

IAM USER

IAM GROUP

IAM ROLE

TYPES:

- Root User
- Regular IAM User

Access Management

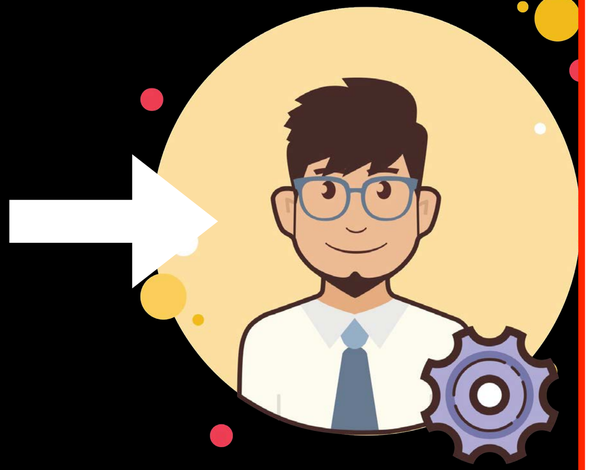
AUTHORIZATION

IAM POLICY

- AWS-managed Policy
- Customer-managed Policy
- Inline Policy

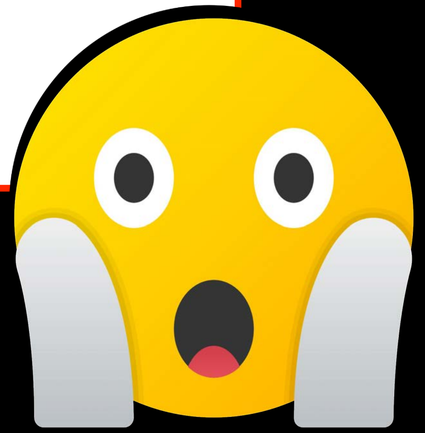
Grant Least Privilege

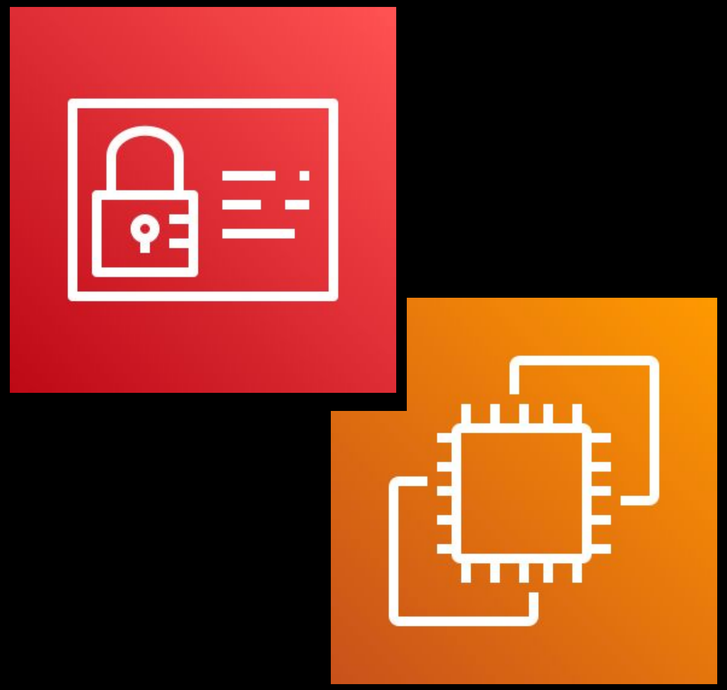
Does not grant the least privilege



IAM ROLE

ROOT USER ACCESS





Amazon EC2 and AWS IAM

- Use the **Instance Profile** to pass a specific IAM role to your Amazon EC2 instance for it to perform certain actions
- IAM roles attached to your instance can also be viewed on your EC2 metadata.

```
curl http://169.254.169.254/latest/meta-data/iam/info
```

Amazon S3 and AWS IAM

- You can set up a **bucket policy** to grant IAM users and other AWS accounts the access permissions for your bucket and its objects.
- In AWS Organization, you can set up an S3 bucket policy that allows cross-account access to other departments of your organization.

AWS Databases and AWS IAM

- For DynamoDB, you can design an IAM policy that allows access to put, update, and delete items in one specific table.
- **IAM DB Authentication** is a feature available for Amazon RDS and Aurora. This allows you to use IAM to centrally manage access to your database resources

Amazon SQS and AWS IAM

- An **Access Policy** can be provisioned to control external access to your SQS queue.
- Helps you grant permissions to an external company to access your queue.
- An SQS access policy can allow external companies to poll the queue without giving up the permissions of your own account.

IDENTITY-BASED POLICY

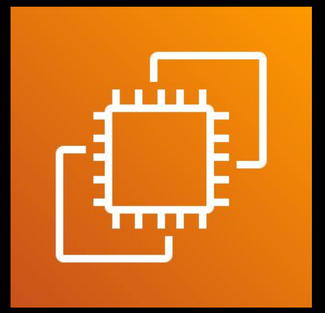
IAM ENTITIES

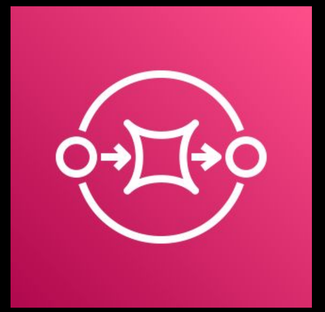
IAM USER

IAM GROUP

IAM ROLE

RESOURCE-BASED POLICY

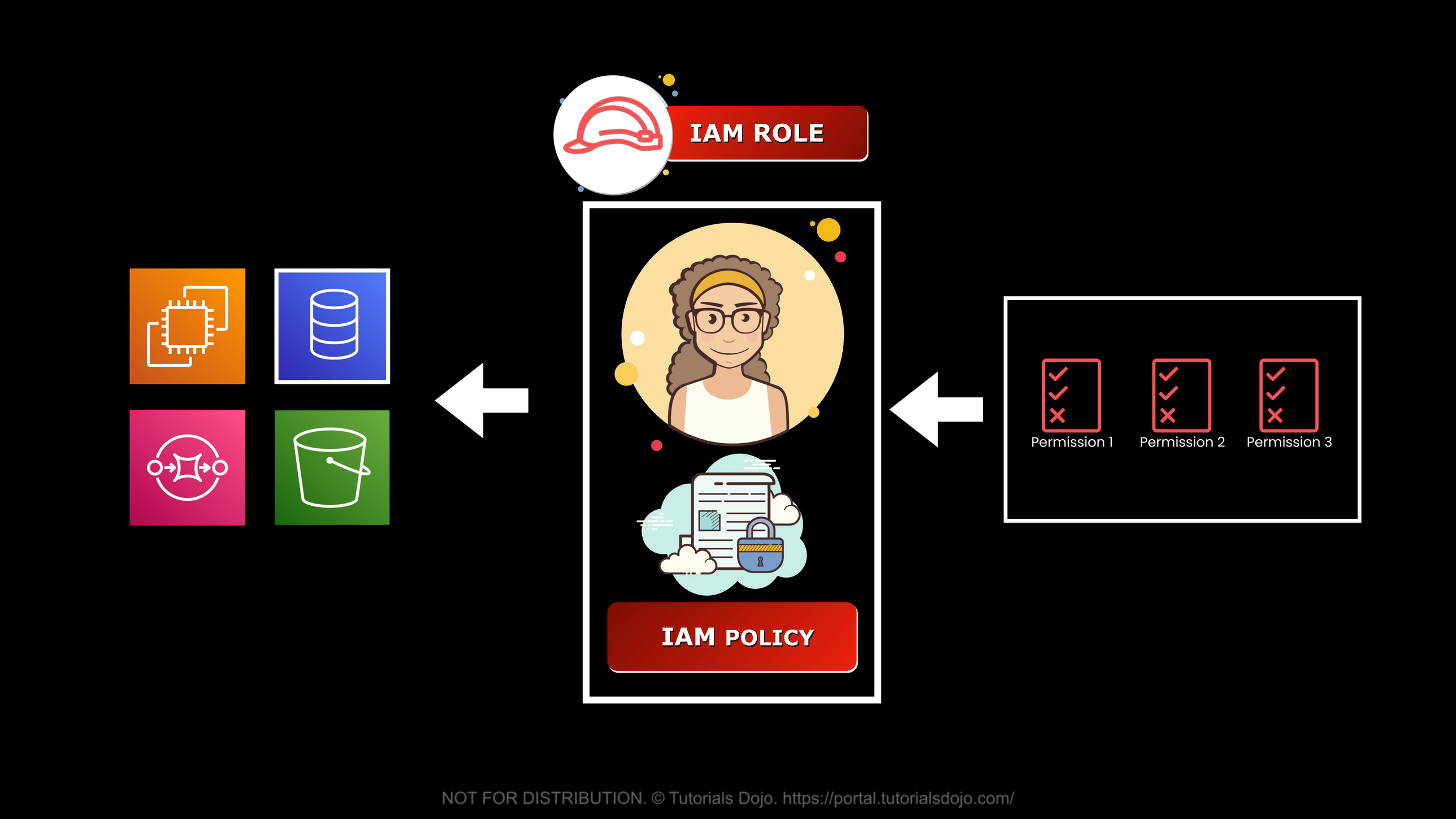




PERMISSIONS BOUNDARY

- Allows you to set the **maximum permissions** that an identity-based policy can grant to an IAM entity.
- Ensure that the entity can only perform the actions that are allowed by both its identity-based policies and its permissions boundaries.

IAM Identities



IAM IDENTITIES

IAM USER

IAM GROUP

IAM ROLE

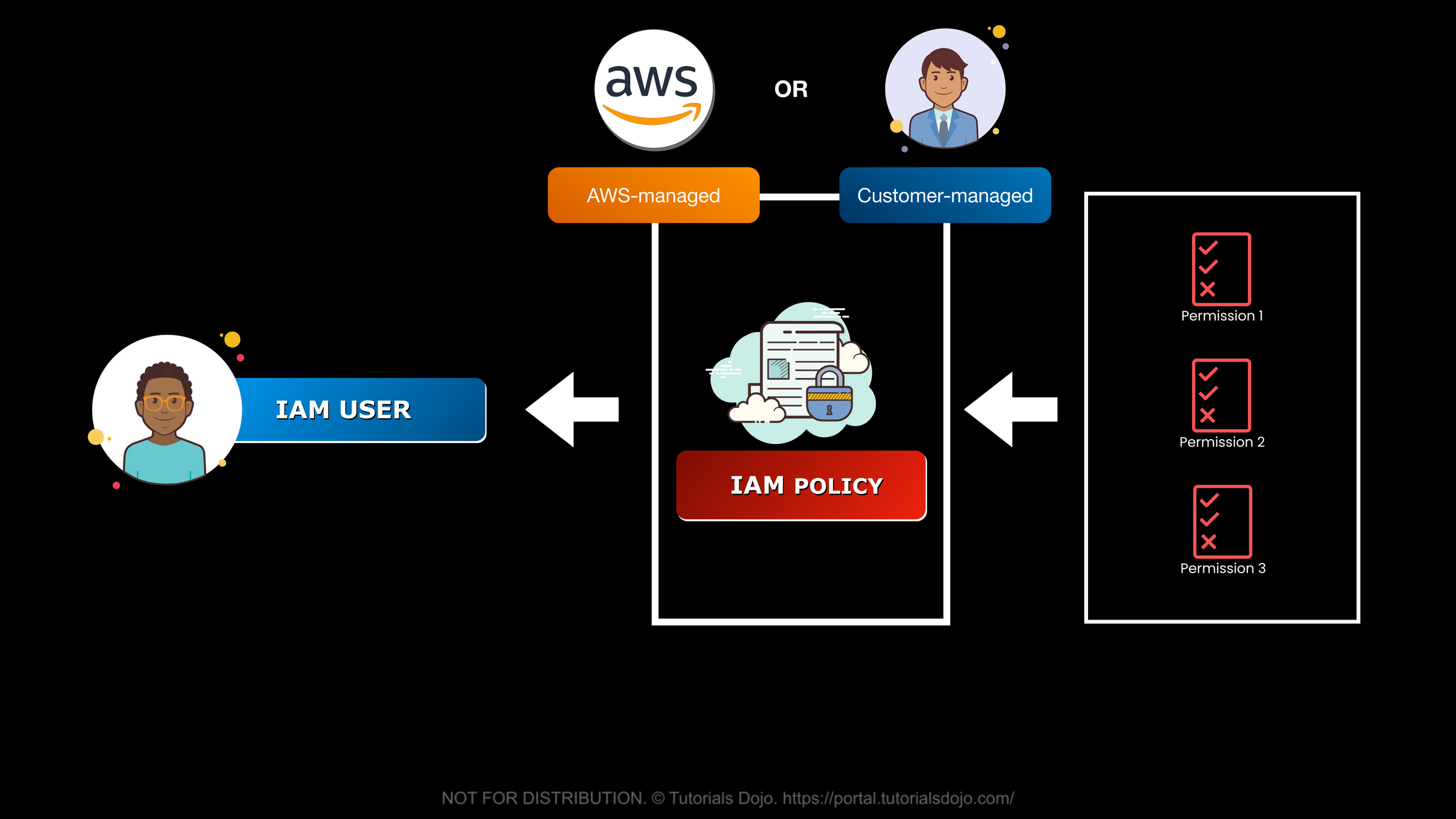
- An entity that **represents an actual person or a service**
- **Can interact with your AWS resources** using the AWS command-line interface, AWS API, or through the AWS management web console
- Provides someone the ability to sign in to the AWS Management Console and programmatic access to AWS APIs

Consists of:

- **NAME**
- **PASSWORD**

- **ACCESS KEY PAIR**
  - **Access Key ID**
  - **Secret Access Key**

- AWS CLI
- AWS APIs
- AWS SDKs
- AWS CDKs



IAM POLICY TYPES

AWS-managed

- Managed by **AWS**
- **Cannot** be fully customized
- Has AWS Managed-Policies for Job Functions that you can readily use:
  - Administrator
  - Support User
  - Security Auditor
  - Network Administrator
  - Developer Power User
  - Billing
  - …and others

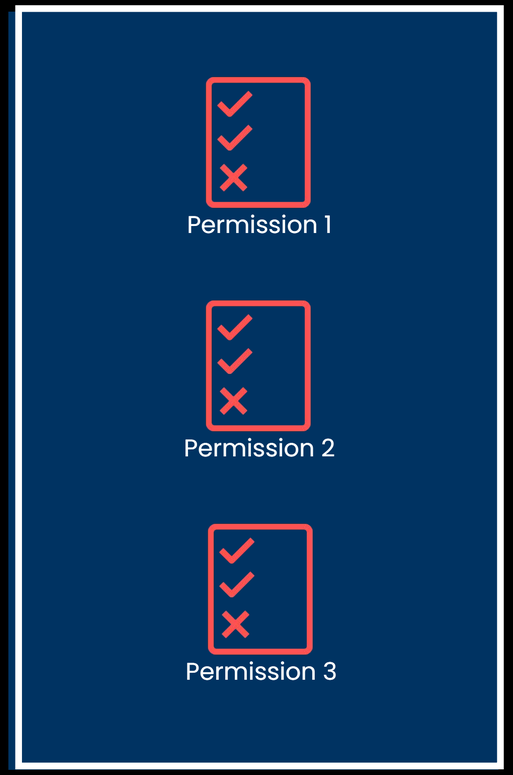
Customer-managed

- Managed by **you** (the customer)
- **Can** be fully customized
- You have to manually create a policy for a particular job function

IAM USER

IAM GROUP

- Can **contain multiple IAM Users**
- A single IAM User can belong to multiple IAM Groups
- **Cannot be nested**
- It can only contain IAM users and not other IAM Groups
- There is no default user group that automatically includes all of the IAM Users in your AWS account



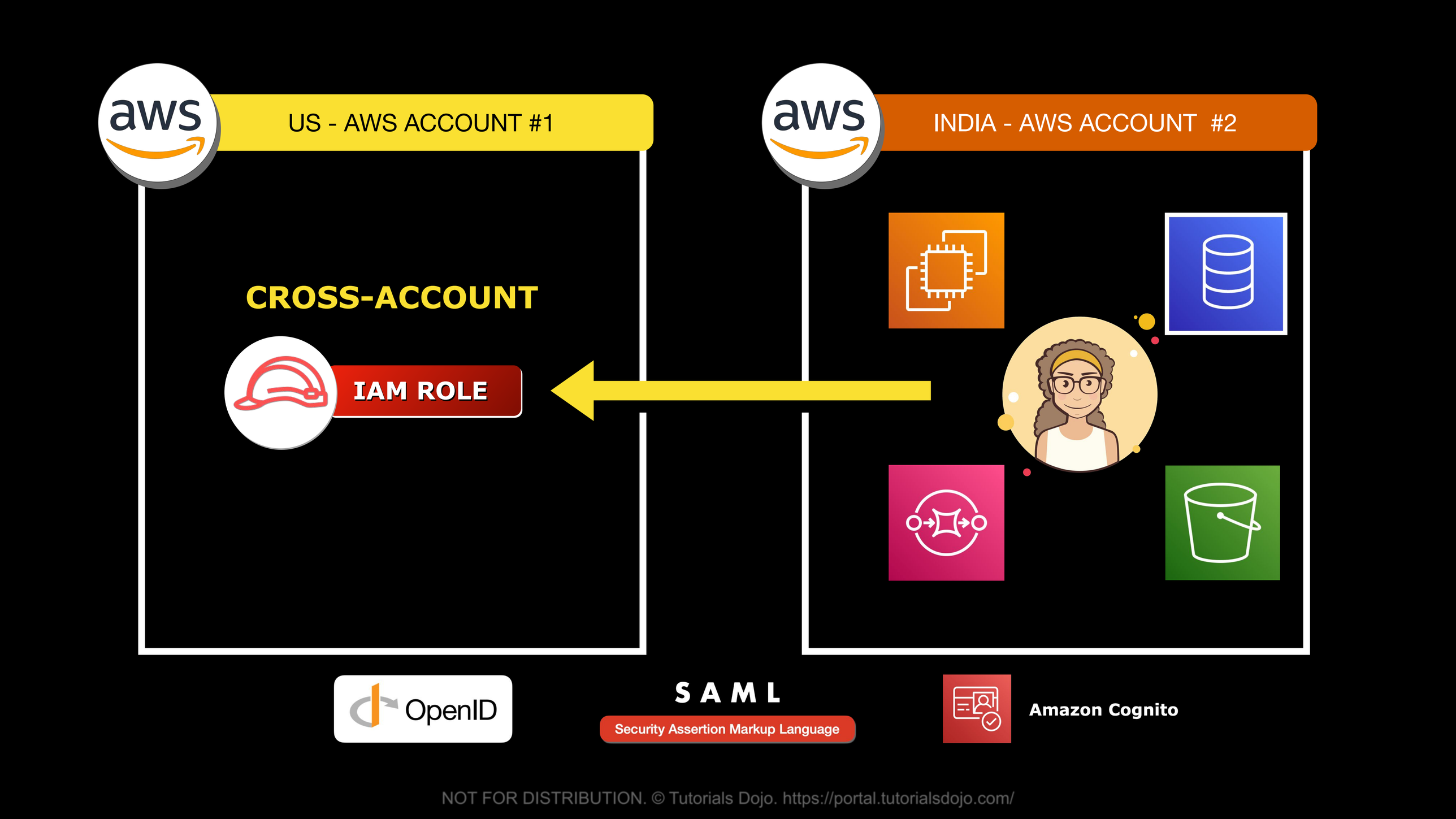
assumed by

IAM ROLE

- Intended to be **assumed** by one or more AWS resources
- No long-term credentials

IAM USER

- Uniquely **associated** with one single person only
- Has long-term credentials:
  - AWS Management Console password
  - Access Keys



CROSS-ACCOUNT

- Grants access to your resources in one account to a trusted principal in a **different AWS account**

AWS SERVICE ROLE

- **Assumed by an AWS service** or applications running in your EC2 instance
- Limited within your AWS account only
- The **custom applications hosted in Amazon EC2 can assume** an AWS service role to perform certain actions

AWS SERVICE-LINKED ROLE

- A predefined role that is **directly linked to an AWS service**

IAM Policy Types

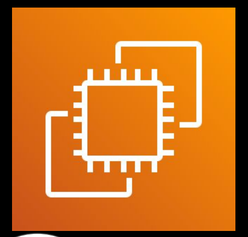
IAM USER

IAM GROUP

IAM ROLE

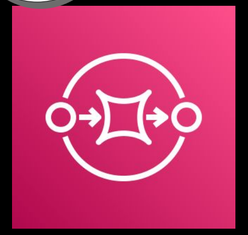
IAM IDENTITIES

IAM POLICY



aws

RESOURCES



IAM POLICY

- Contains permissions that **explicitly ALLOW or DENY access** to certain AWS services
- It provides fine-grained **access control to specific API actions** as well as the AWS resources that the policy should be applied to

IAM POLICY

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "TutorialsDojo",
            "Effect": "Allow",
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::tutorialsdojo-manila/*"
        }
    ]
}
```

ALLOWS THE API ACTIONS YOU SPECIFY

API action

IAM POLICY

```
{
    "Version": "2012-10-17",
    "Statement": {
        "Sid": "tutorialsdojo",
        "Effect": "Deny",
        "Action": [
            "lambda:CreateFunction",
            "lambda:DeleteFunction"
            ],
        "Resource": "*",
        "Condition": {
            "IpAddress": {
                "aws:SourceIp": "220.110.16.0/20"
            }
        }
    }
}
```

DENIES THE API ACTIONS

API actions

IP Condition

IAM POLICY

Multi-Factor Authentication
(MFA)

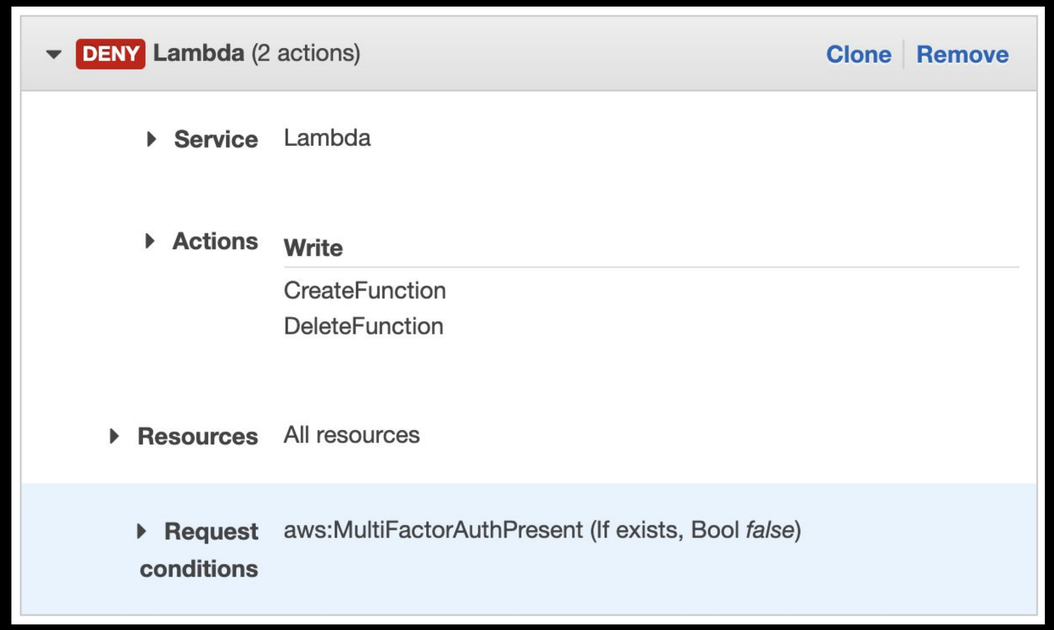
```
{
    "Version": "2012-10-17",
    "Statement": {
        "Sid": "tutorialsdojo",
        "Effect": "Deny",
        "Action": [
            "lambda:CreateFunction",
            "lambda:DeleteFunction"
        ],
        "Resource": "*",
        "Condition": {
            "BoolIfExists": {
                "aws:MultiFactorAuthPresent": "false"
            }
        }
    }
}
```

MFA Condition

JSON EDITOR

```
{
    "Version": "2012-10-17",
    "Statement": {
        "Sid": "tutorialsdojo",
        "Effect": "Deny",
        "Action": [
            "lambda:CreateFunction",
            "lambda:DeleteFunction"
            ],
        "Resource": "*",
        "Condition": {
            "BoolIfExists": {
                "aws:MultiFactorAuthPresent": "false"
            }
        }
    }
}
```

VISUAL EDITOR



Standalone Policy

Inline Policy

- **Remains unchanged** even if you delete its associated IAM identity
- It doesn’t have a strict one-to-one relationship to its associated IAM identity

- **Will be automatically be deleted** if you delete its associated identity
- Has a strict one-to-one relationship to its associated IAM identity

IAM Policy Types

- Identity-based Policies
- Resource-based Policies
- Permissions Boundaries
- AWS Organizations SCPs
- S3 Access Control Lists (ACLs)
- Session Policies

Identity-Based Policy

- A policy that you **attach to an IAM Identity**
- Two Types:

Managed Policies

- A type of a standalone policy
- Can either be AWS managed or Customer-managed

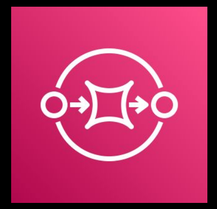
Inline Policies

- Maintains a strict one-to-one relationship between a policy and an IAM identity.
- Tightly-coupled with its associated IAM Identity

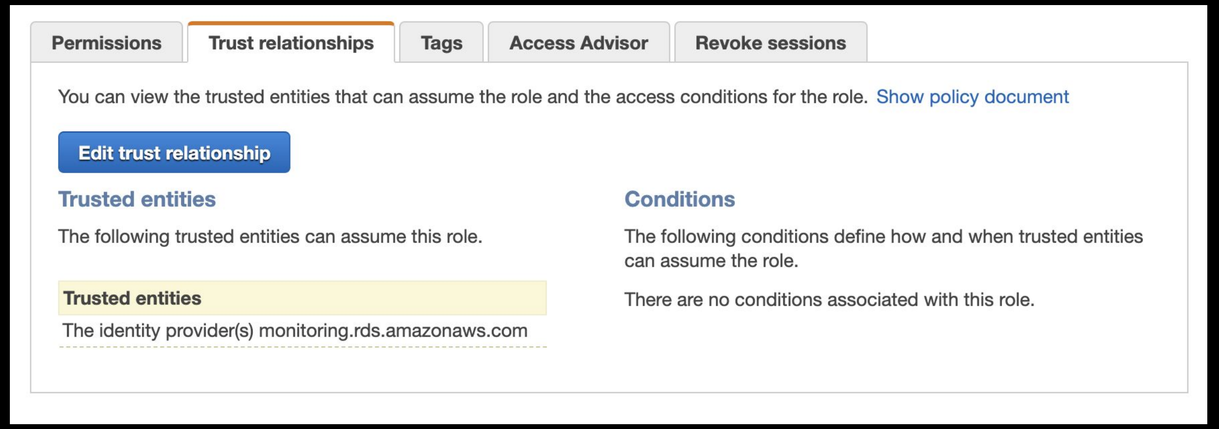
Resource-Based Policy

- Attaches an inline policy to a **specific AWS Resource**
- Types:

S3 Bucket Policy



SQS Access Policy

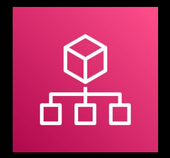


Trust Policy

Permissions Boundaries

- Defines the **maximum permissions** that an identity-based policy can grant to an IAM entity
- Does not explicitly grant permissions
- **Sets a clear boundary** to ensure that a given IAM policy will not over-provision the permissions to your AWS resources

Service Control Policies
(SCPs)

- Primarily used in:  **AWS Organizations**
- Defines the **maximum permissions for account members** of an organization or organizational unit.
- Limits the permissions that identity-based policies or resource-based policies grant to the IAM users or roles within the AWS account
- IAM policies can't restrict the AWS account root user. In the contrary, the specified actions from an attached SCP can affect all IAM identities, including the root user, of the member account

Access Control List
(ACL)

- Primarily used in: **Amazon S3**
- Controls which principals in other AWS accounts can access a particular bucket
- These are **cross-account permission policies** that grant certain permissions to a specified principal that you define
- ACLs cannot grant permissions to entities within the same account

Sessions Policies

- Limits the permissions that an identity-based policy grants to a **particular session**
- Works like **Permissions Boundaries**
- **Sets a limit of what kind of permission a session has**, without granting any permissions.
- Aside from an identity-based policy, the permissions of a session policy can also come from a resource-based policy
- If there’s an explicit deny in any of the policies, then it will effectively override any allowed permissions

IAM Policy Basics

Policy-wide Information

Statements

```
{
    "Id": "TutorialsDojoPolicy1",
    "Version": "2012-10-17",

    "Statement": [
        {
            "Sid": "AllowAllActionsOnBooksTable",
            "Effect": "Allow",
            "Action": "dynamodb:*",
            "Resource": "arn:aws:dynamodb:us-west-2:123456789012:table/Books"
        },
                    Logical OR
        {
            "Sid": "ListObjectsInBucket",
            "Effect": "Allow",
            "Action": ["s3:ListBucket", "s3:DeleteObject"],
            "Resource": ["arn:aws:s3:::tutorialsdojo-manila"]
        }
    ]
}
```

IAM Statement Elements

```
{
    "Sid"       : "AllowActionsOnBooksTable",
    "Effect"    : "Allow",
    "Principal" : { "AWS": "arn:aws:iam::123456789012:root" }
    "Action"    : [
                      "dynamodb:*",
                      "s3:*",
    "Resource"  :     arn:aws:s3:::tutorialsdojo/*
    "Condition" :  {
        "IpAddress": {
            "aws:SourceIp": "220.110.16.0/20"
    }
}
```

Statement ID

ALLOW or DENY

CONDITION ELEMENT

CONDITION ELEMENT

- String
- Numeric
- Date
- Boolean
- Binary
- ARN
- IfExists
- IpAddress
- …and many more!

```
"Condition" : { "{condition-operator}" : { "{condition-key}" : "{condition-value}" }}
```

StringEquals
StringNotEquals
StringEqualsIgnoreCase
StringNotEqualsIgnoreCase
StringLike
StringNotLike

NumericEquals
NumericNotEquals
NumericLessThan
NumericLessThanEquals
NumericGreaterThan
NumericGreaterThanEquals

DateEquals
DateNotEquals
DateLessThan
DateLessThanEquals
DateGreaterThan
DateGreaterThanEquals

ArnEquals, ArnLike
ArnNotEquals, ArnNotLike

IpAddress
NotIpAddress

CONDITION ELEMENT

IfExists

- StringEqualsIfExists
- NumericEqualsIfExists
- BoolIfExists
- IpAddressIfExists
- etc…

Shares the Amazon S3 bucket named **`tutorialsdojo-manila`** with an external vendor while ensuring that the **bucket owner** is still be able to access all objects

```
. . .
        "Action": [
            "s3:PutObject"
        ],
        "Resource": “arn:aws:s3:::tutorialsdojo-manila/*”,
        "Condition": {
            "StringEquals": {
                "s3:x-amz-acl": "bucket-owner-full-control"
            }
        }
. . .
```

Users will be denied of all API actions *(except for the **s3:PutObject** action)* if their **multi-factor authentication (MFA)** is not enabled

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "DenyAllTDojoUsersNotUsingMFA",
        "Effect": "Deny",
        "NotAction": “s3:PutObject",
        "Resource": "*",
        "Condition": {
            "BoolIfExists": {
                "aws:MultiFactorAuthPresent": "false"}
            }
    }]
}
```

IAM Policy Evaluation Logic

IAM POLICY

```
{
    "Id": "TutorialsDojoPolicy1",
    "Version": "2012-10-17",

    "Statement": [
        {
            "Effect": "Allow",
            "Action": "lambda:*",
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": ["lambda:CreateFunction", "lambda:DeleteFunction"],
            "Resource": "*"
        }
    ]
}
```

Allows the API Action

Denies the API Action

Logical **OR**

1. **Authentication**
2. Process the **request context**
3. **Evaluate all policies** within a single account

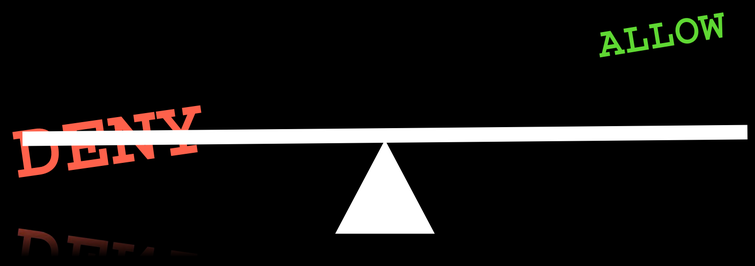
If the IAM policies are within a **single** account…

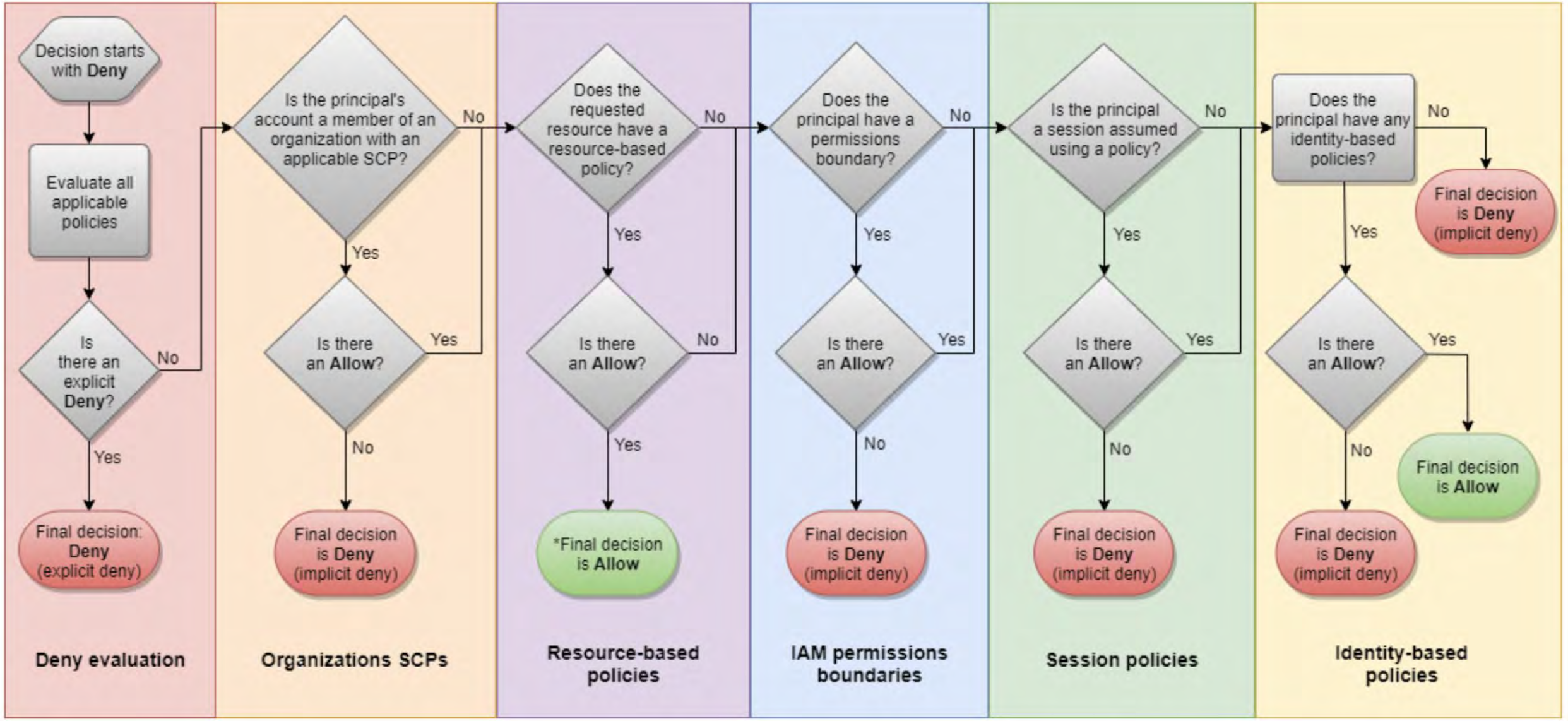
All requests will be implicitly denied

Process the explicit `ALLOW` statements for identity-based or resource-based policy

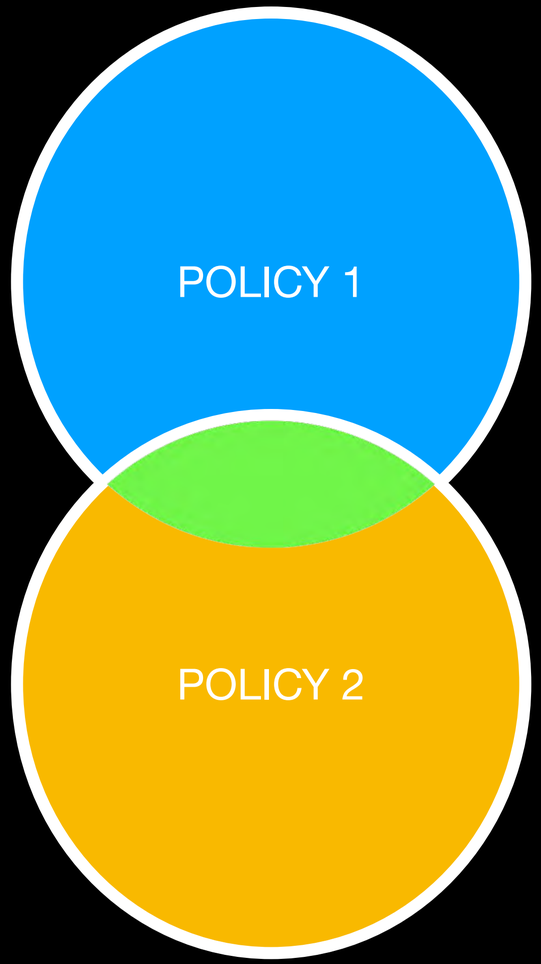
Except for the AWS account root user

- **Permissions Boundaries**
- **Sessions Policies**
- **Service Control Policies** (SCPs)





```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:TerminateInstances",
            "Resource": "*",
            "Condition": {
                "IpAddress": {
                    "aws:SourceIp": "49.147.194.0/24"
                }
            }
        },
        {
            "Effect": "Deny",
            "Action": "ec2:*",
            "Resource": "*",
            "Condition": {
                "StringNotEquals": {
                    "ec2:Region": "us-west-1"
                }
            }
        }
    ]
}
```



This policy will allow you to terminate an Amazon EC2 instance in the us-west-1 region as long as your source IP is within the `49.147.194.0/24` CIDR block.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:*",
                "ds:*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": "ds:Delete*",
            "Resource": "*"
        }
    ]
}
```

This policy provides full access to Amazon EC2.

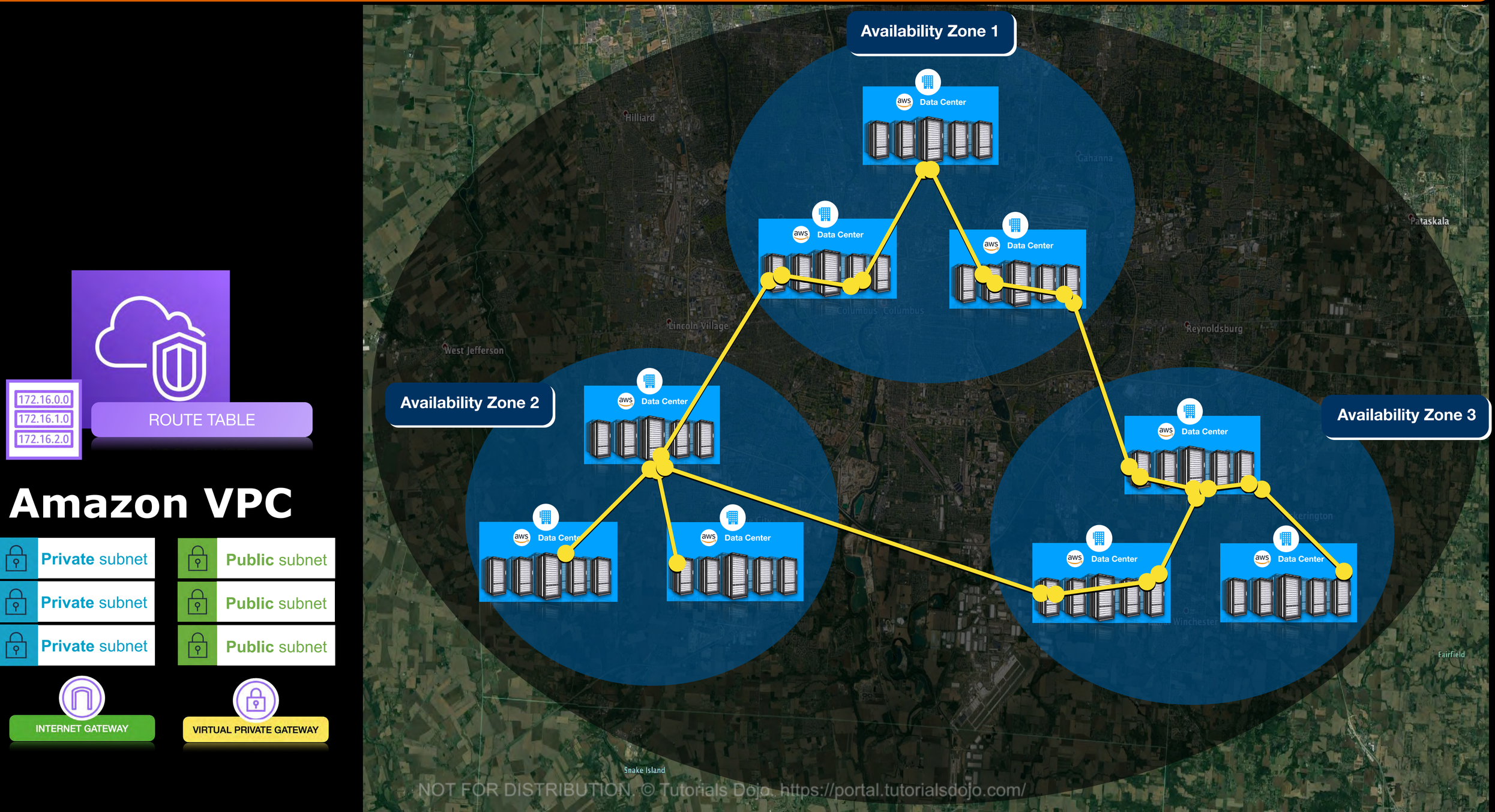
It also allows creating, reading and updating the AWS Directory Service (DS) directories but not delete them.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "lambda:*",
            "Resource": "*"
        },
        {
            "Effect": "Deny",
            "Action": [
               "lambda:CreateFunction",
               "lambda:DeleteFunction"
            ]
            "Resource": "*",
               "Condition": {
                 "IpAddress": {
                     "aws:SourceIp": “220.200.16.0/24"
            }
        }

        }
    ]
}
```

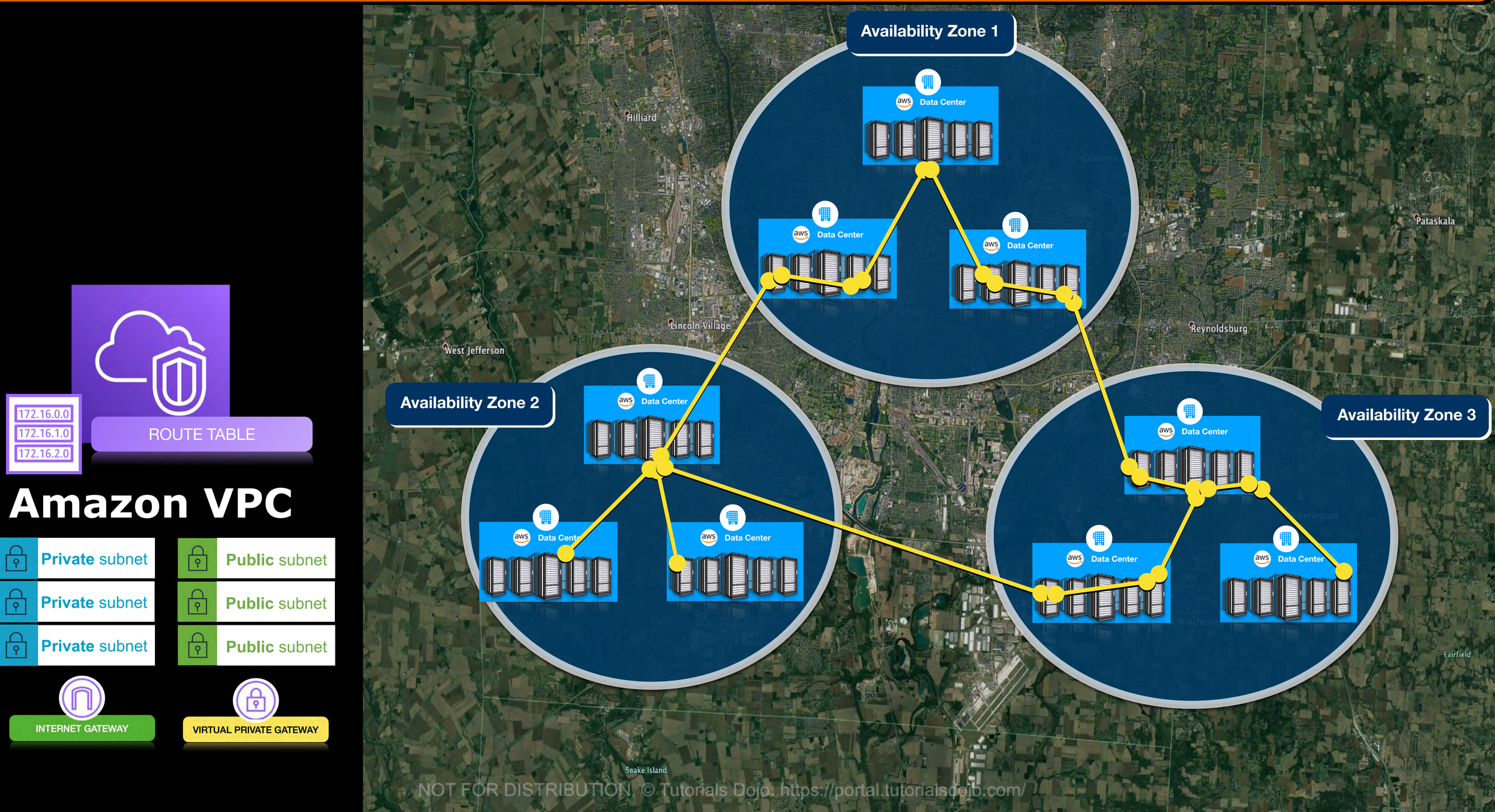
This allows an AWS Lambda function to be **created or deleted** as long as the IP address of the request does NOT fall under the `220.200.16.0/24` IP range.

Amazon VPC Overview

US East (Ohio) us-east-2



US East (Ohio) us-east-2



aws CLOUD

REGION

Amazon VPC

IPv4 CIDR Range: 10.0.0.0/16

IPv6 CIDR Range: 2001:db8:1234:1a00::/56

172.16.0.0
172.16.1.0
172.16.2.0

ROUTE TABLE

Private subnet

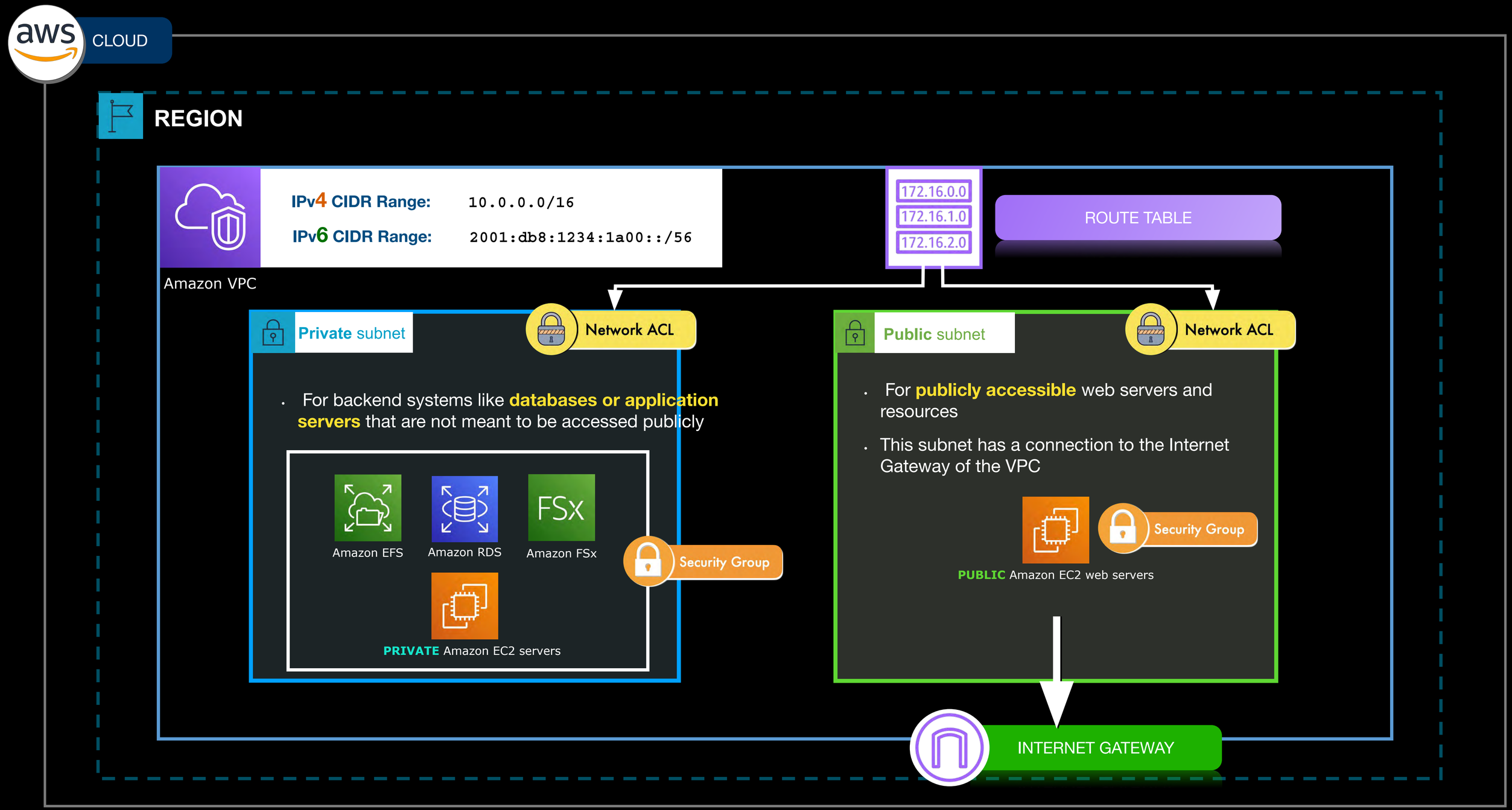
10.0.0.0/24

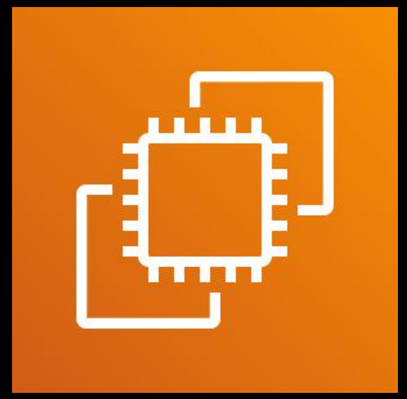
- A subnet **must reside entirely within one Availability Zone** only
- One subnet **cannot span to two or more AZs**.

Public subnet

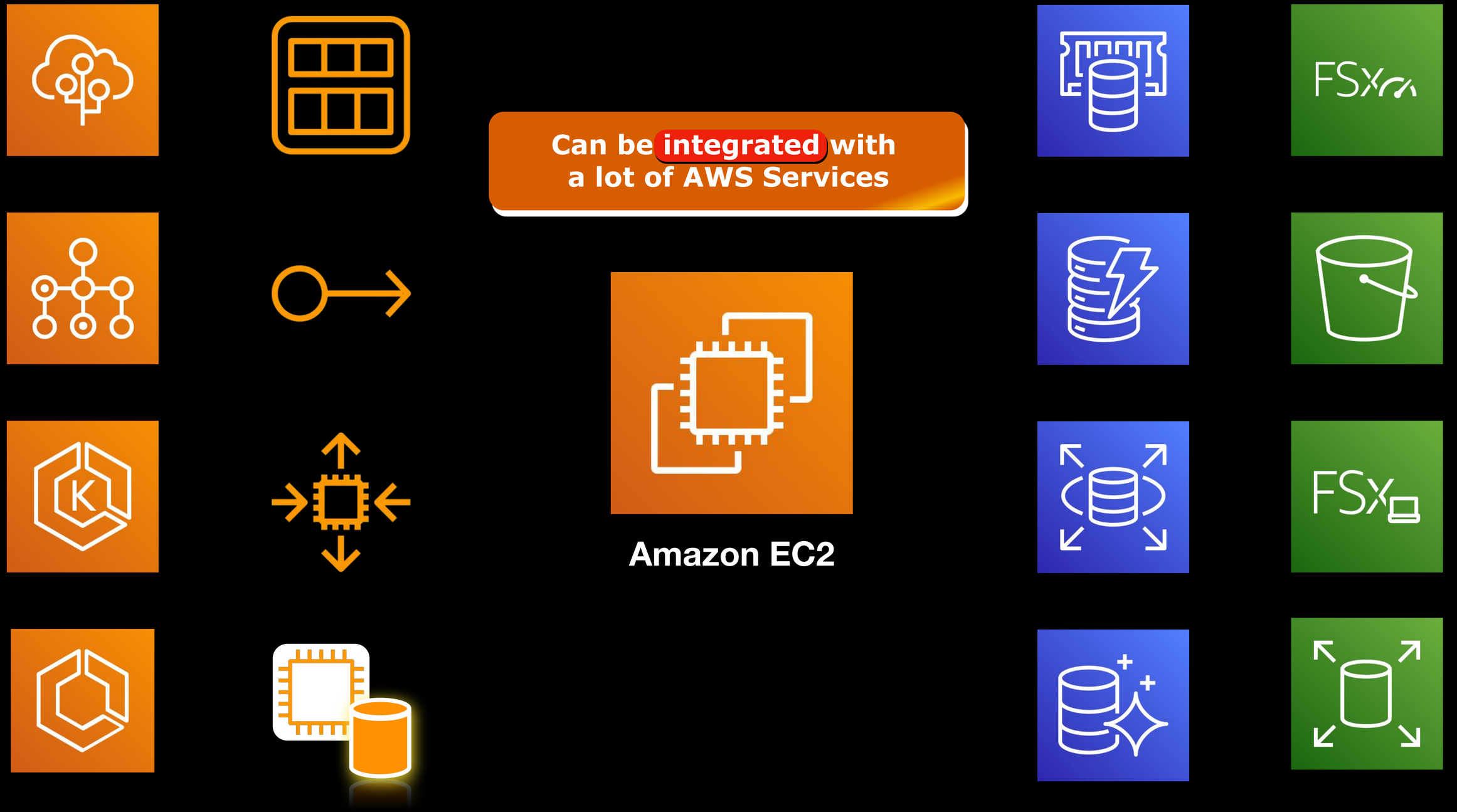
10.0.1.0/24

- You can **have multiple subnets in the same** Availability Zone.





Amazon EC2 Overview



Amazon VPC

Elastic IP Address

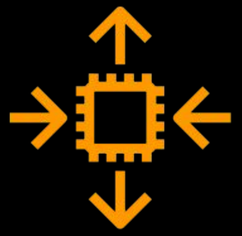
Elastic Network Interface (ENI)

Elastic Network Adapter (ENA)

Placement Groups

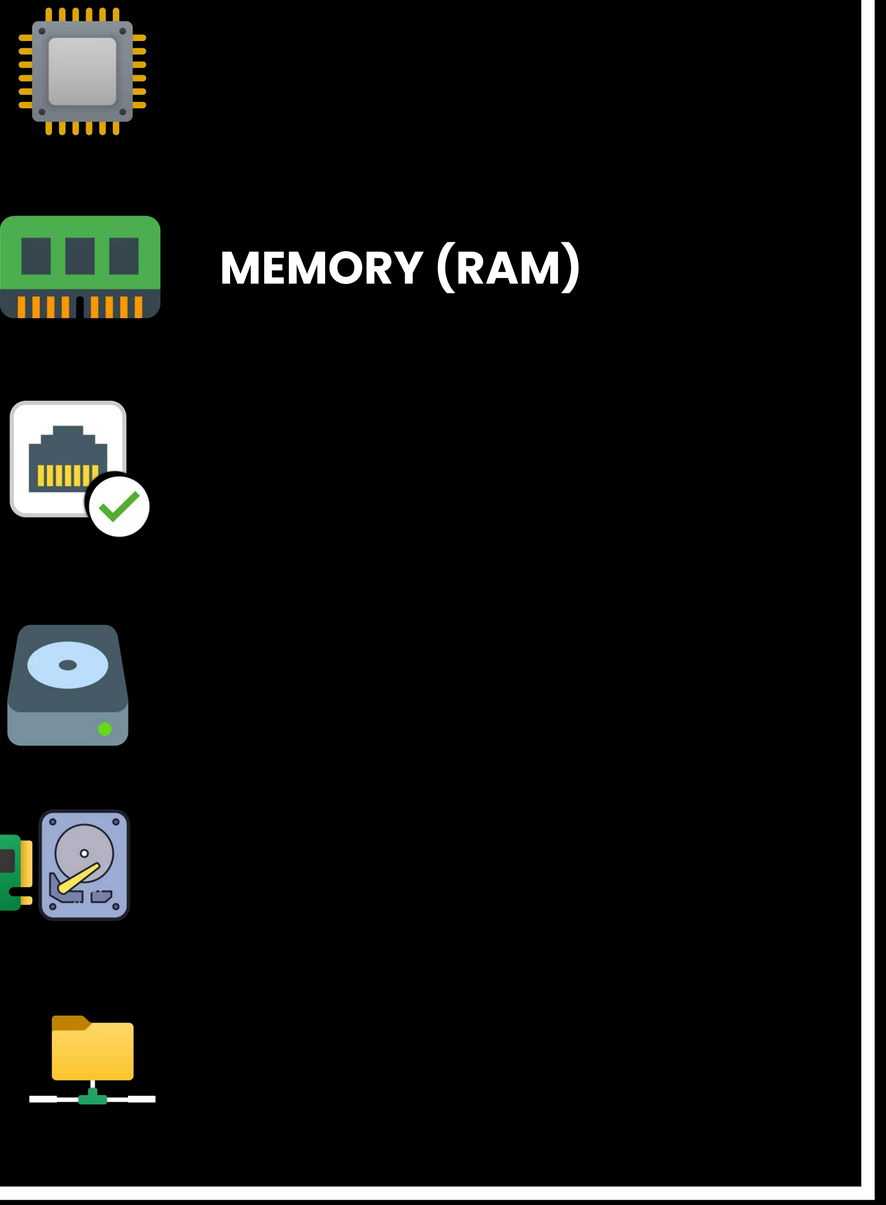
Elastic Fabric Adapter (EFA)

Your Computer

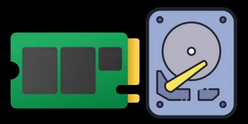


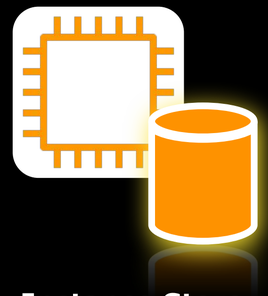
Amazon EC2

both have



MEMORY (RAM)





Instance Store

Amazon EBS

Amazon EFS

Amazon FSx for Lustre

Amazon FSx for Windows File Server

Amazon S3

OBJECT STORAGE

NETWORK

Amazon VPC

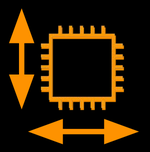
Elastic IP Address

Elastic Network Interface (ENI)

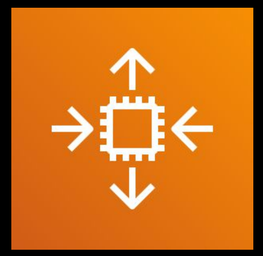
Placement Groups

Elastic Network Adapter (ENA)

Elastic Fabric Adapter (EFA)



AUTO SCALING

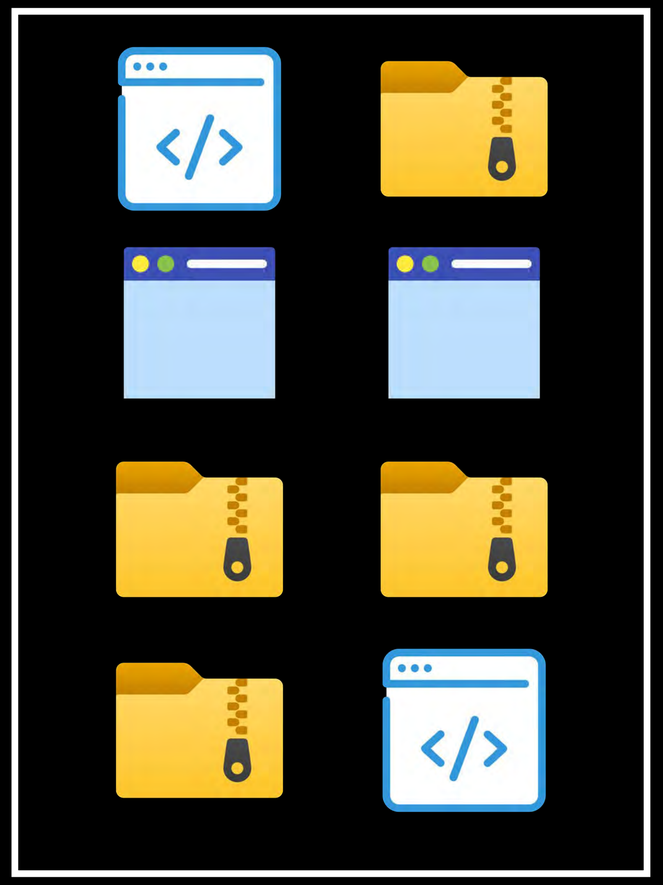


Amazon EC2 Auto Scaling

Amazon Machine Image (AMI)

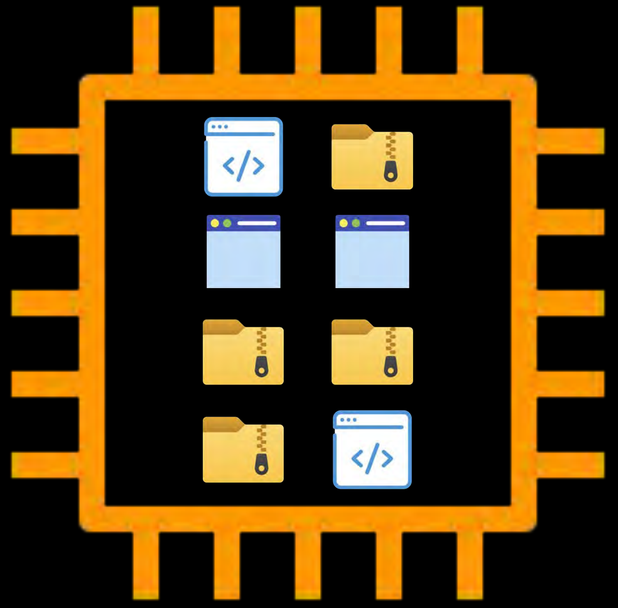
Amazon Machine Image (AMI)

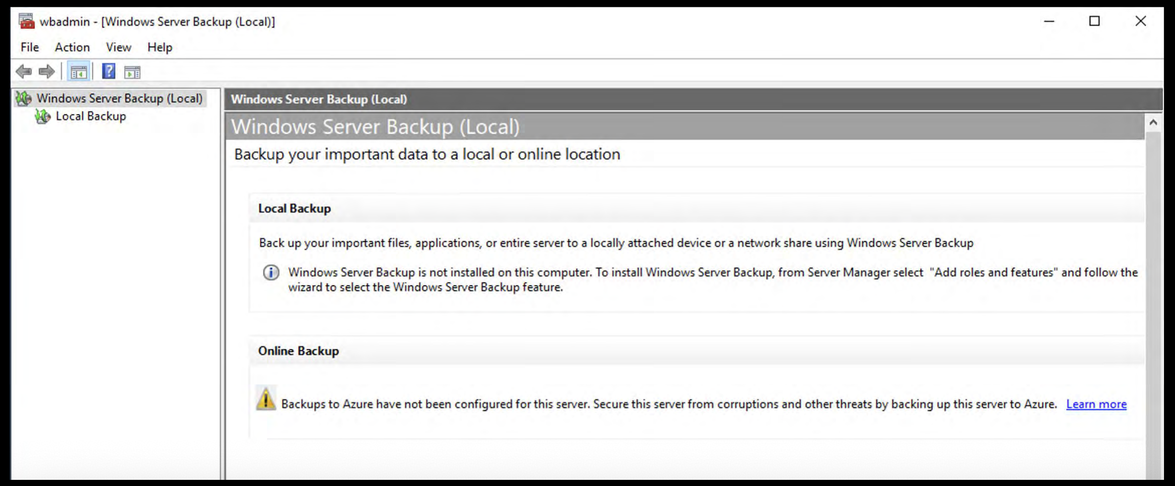
apps & configurations

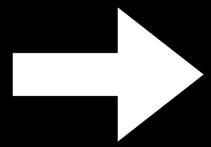


AMI

EC2 Instance







Amazon Machine Image (AMI)

Amazon Machine Image (AMI)

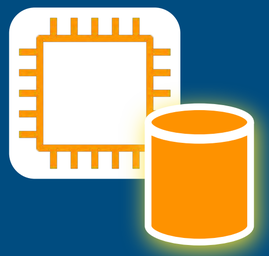
Volume Snapshots

Block Device Mapping

Launch Permissions

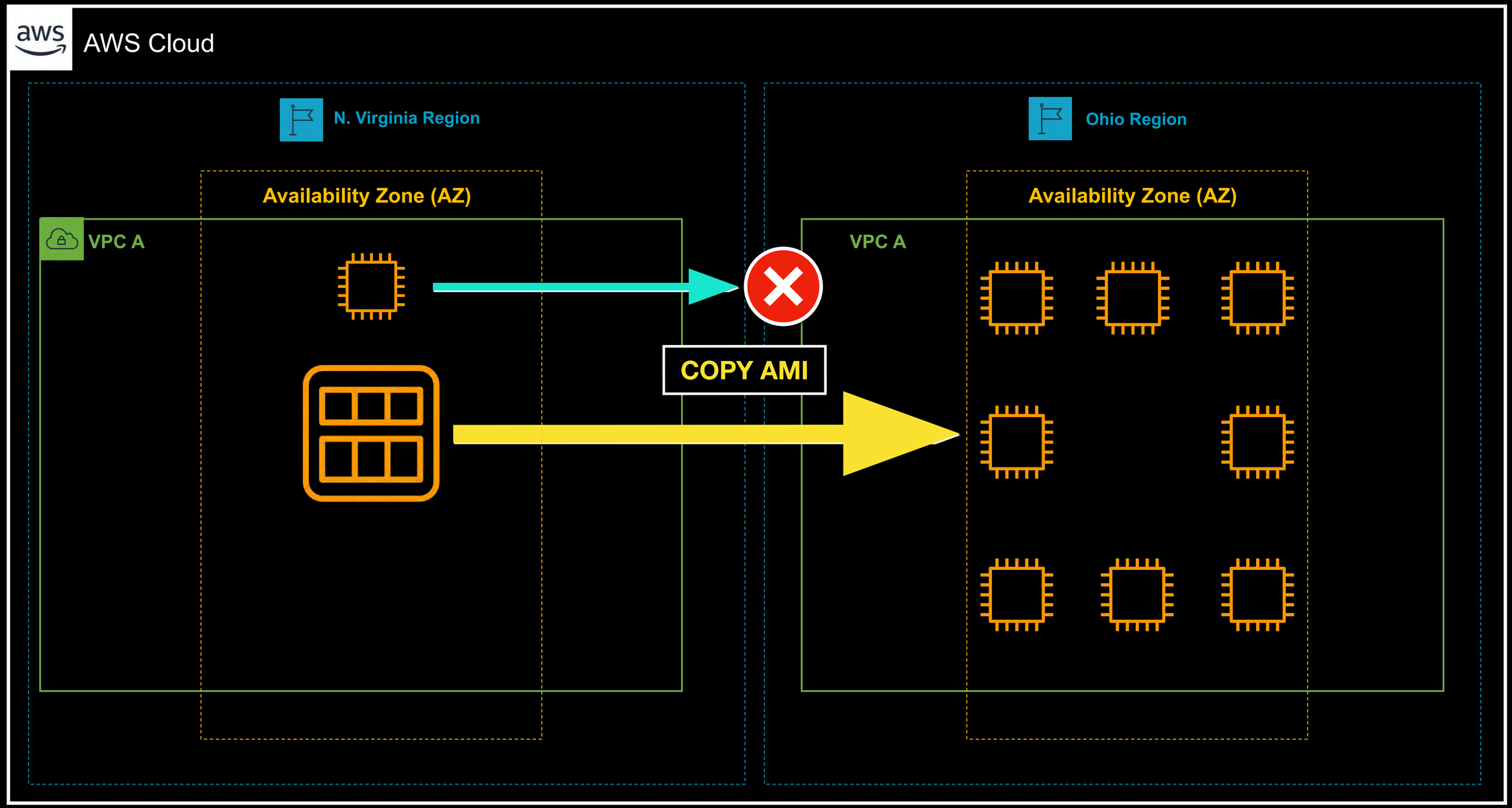
BLOCK STORE TYPE

Amazon Machine Image (AMI)

	Volume Snapshots	Block Device Mapping	Launch Permissions
 **Amazon EBS**	 EBS Snapshots	 Amazon EBS Volumes mapping	• Public • Explicit • Implicit
 **Amazon EC2 Instance Store**	 Template for the root volume	N/A	

Amazon Machine Image (AMI)

- **Regional** in scope
- You can **copy your AMI to another AWS Region**
- You can also copy your AMI to **another AWS account**



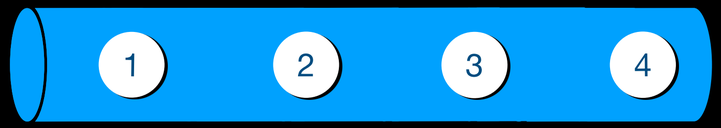
AWS Marketplace

Amazon Machine Image (AMI)

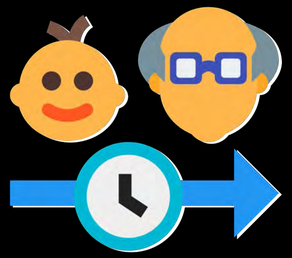
VIRTUALIZATION TYPE

VIRTUALIZATION TYPE	BOOT UP PROCESS	SUPPORT FOR SPECIAL HARDWARE EXTENSIONS
PV Paravirtual	Uses special boot loader called **PV-GRUB**	**N/A**
HVM Hardware Virtual Machine	Executes the master boot record of the root block device of your image	Uses several special hardware extensions such as **enhanced networking** or **GPU processing**

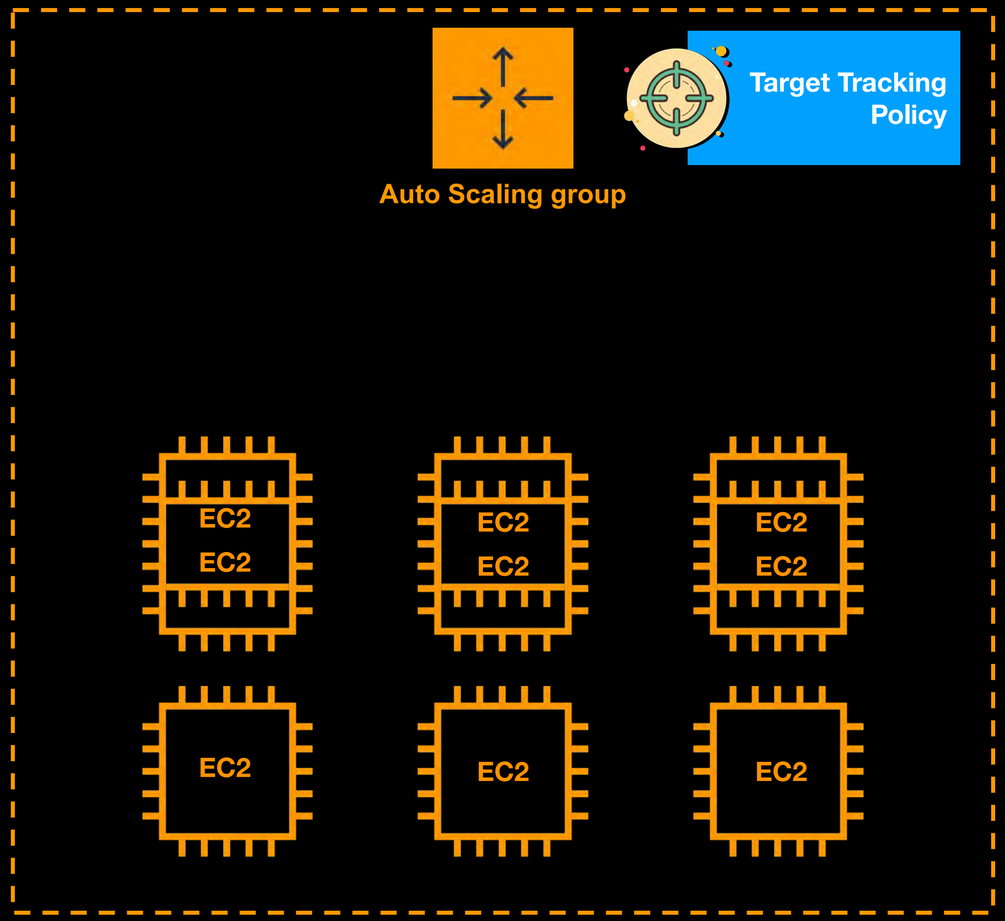
Amazon SQS

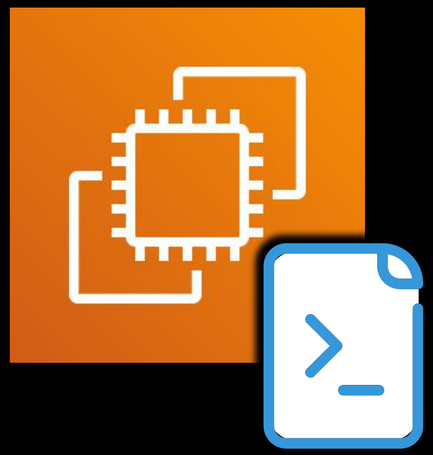


- **Age** of the Oldest Message



Amazon Machine Image (AMI)



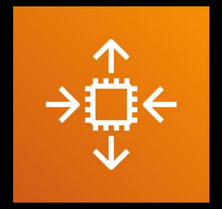


Instance User Data

```
#!/bin/bash
yum update -y
mkdir tdojologs
systemctl start httpd
echo “tutorialsdojo OK!”
```

User Data

Amazon EFS

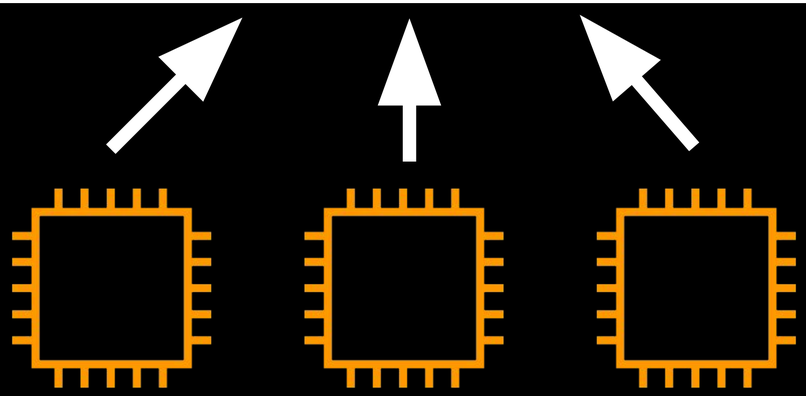


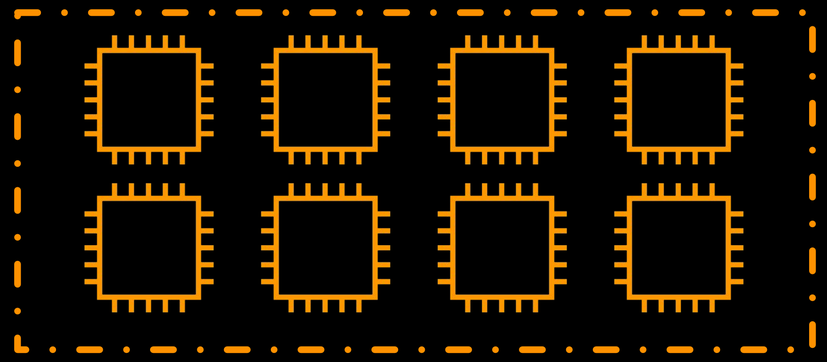
Auto Scaling Group

```
mkdir ~/tutorialsdojo-efs

sudo mount -t nfs -o nfsvers=4.1,\
rsize=1048576,wsize=1048576,hard,\
timeo=600,retrans=2,noresvport \
awsjonbonsoefs:/    ~/tutorialsdojo-efs
```

```
#!/bin/bash
curl
https://s3.amazonaws.com/aws-cloudwatch/dow
nloads/latest/awslogs-agent-setup.py -O
chmod +x ./awslogs-agent-setup.py
./awslogs-agent-setup.py -n -r us-east-1 -c
s3://tutorialsdojo
```

User Data



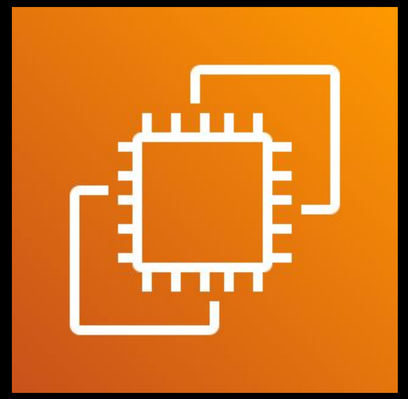


User Data

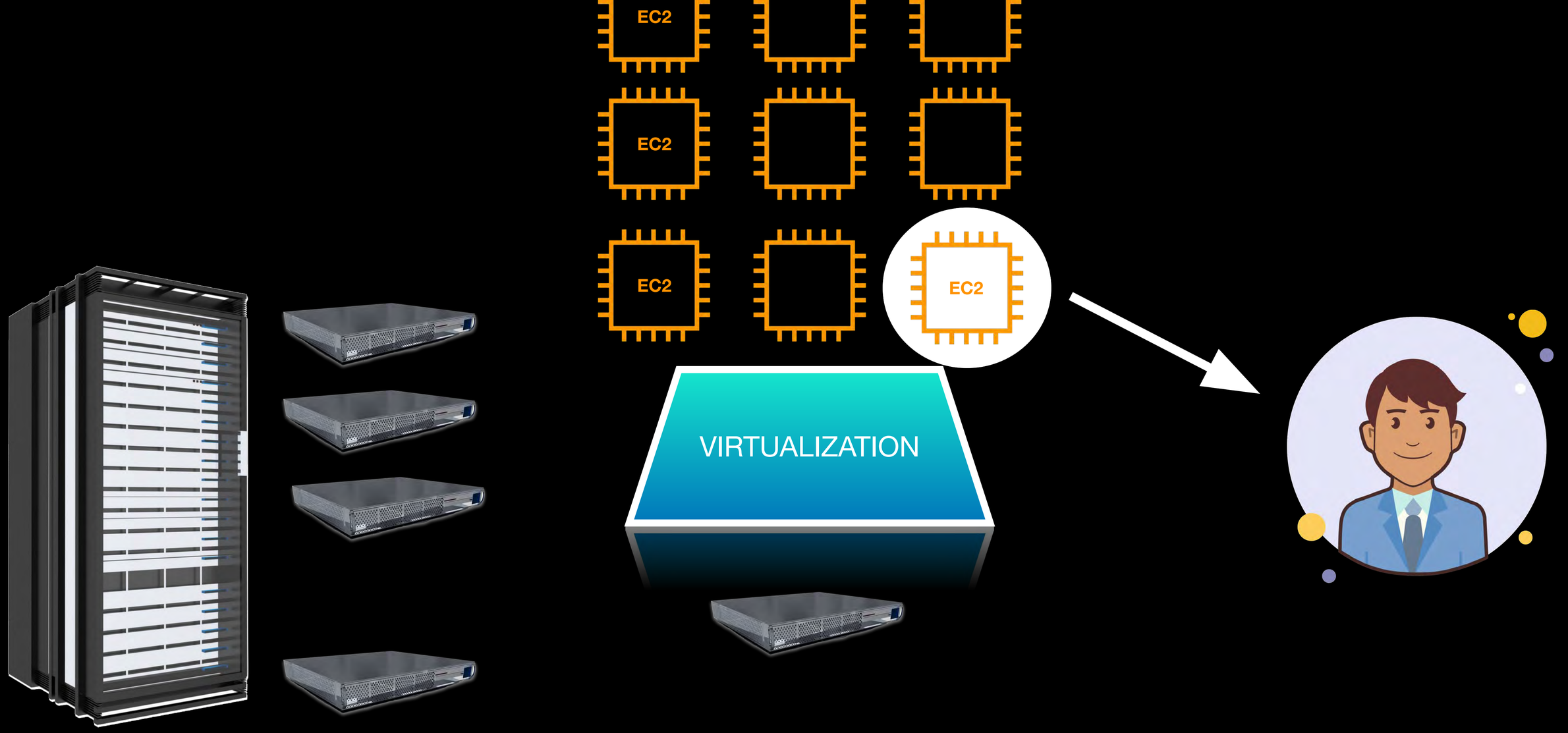
- Must be in a **base64-encoded format**
- Limited to **16 KB only when in raw form**
- Accessible from the Instance Metadata using this URI:

```
http://169.254.169.254/latest/user-data
```

- **Only run once** upon the first EC2 Instance Launch
- Modifying the User Data and restarting the instance won’t affect the initial User Data

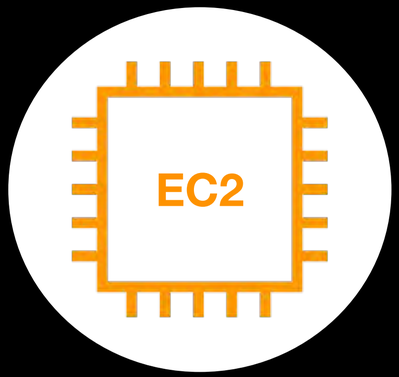


Instance Metadata



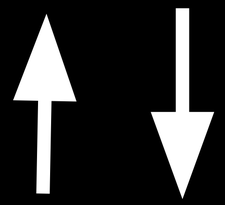
MANIFEST

METADATA



- AMI
- Hostname
- Public IP address
- Private IP address
- Instance type
- MAC address
- Security groups
- Security credentials
- IAM Roles of your instance
- . . . and many more!

INSTANCE METADATA



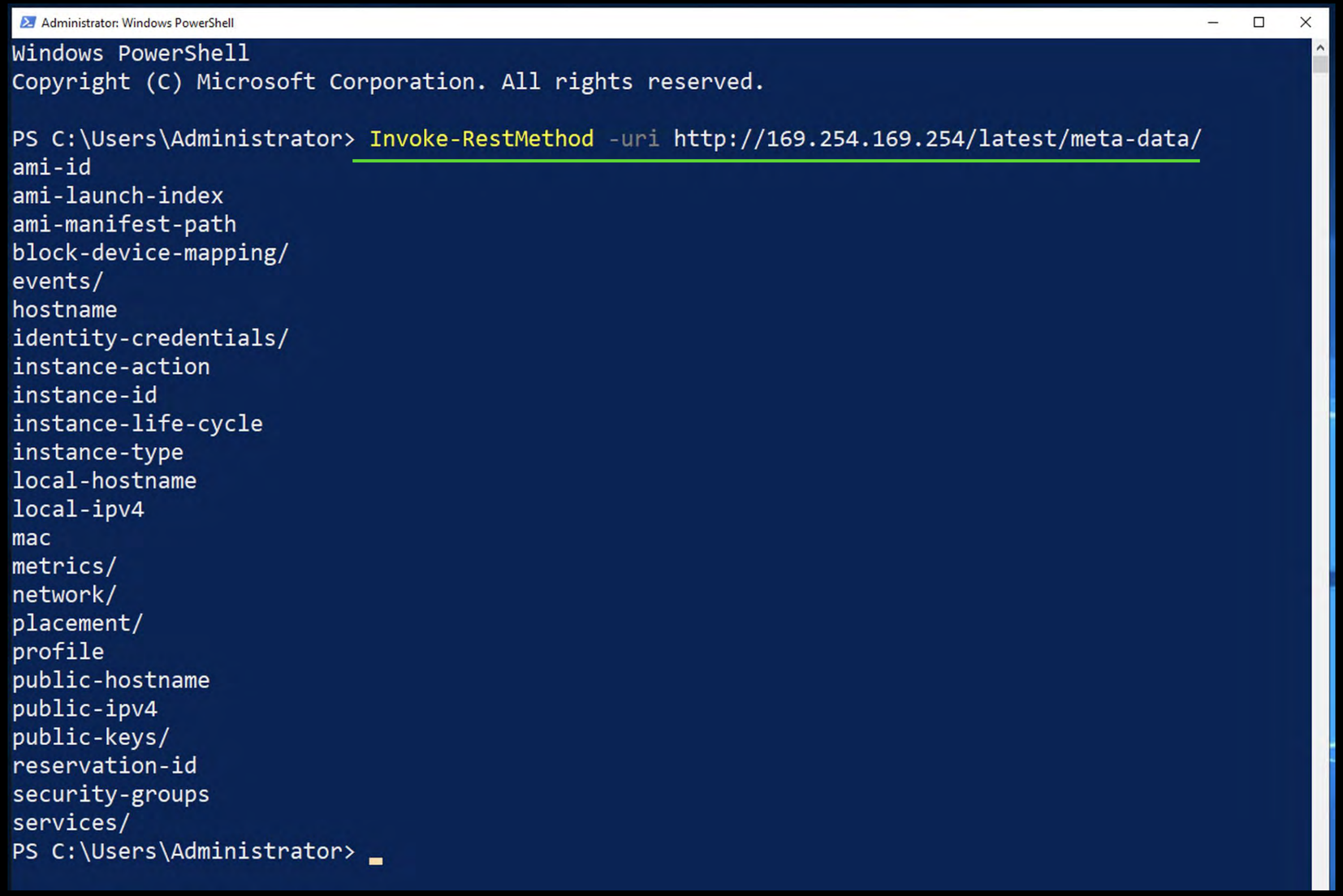
Link-local Address

INSTANCE METADATA SERVICE

http://169.254.169.254/latest/meta-data/

```
jonbonso — Philippines - Amazon EC2 Instance - Cebu, Davao, Palawan, Cagayan De Oro, Tarlac, Pangasinan — Tutorials Dojo Manila Server — ssh -i ~/

jonbonso@tutorialsdojo >curl http://169.254.169.254/latest/meta-data
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
events/
hostname
iam/
identity-credentials/
instance-action
instance-id
instance-life-cycle
instance-type
local-hostname
local-ipv4
mac
metrics/
network/
placement/
profile
public-hostname
```

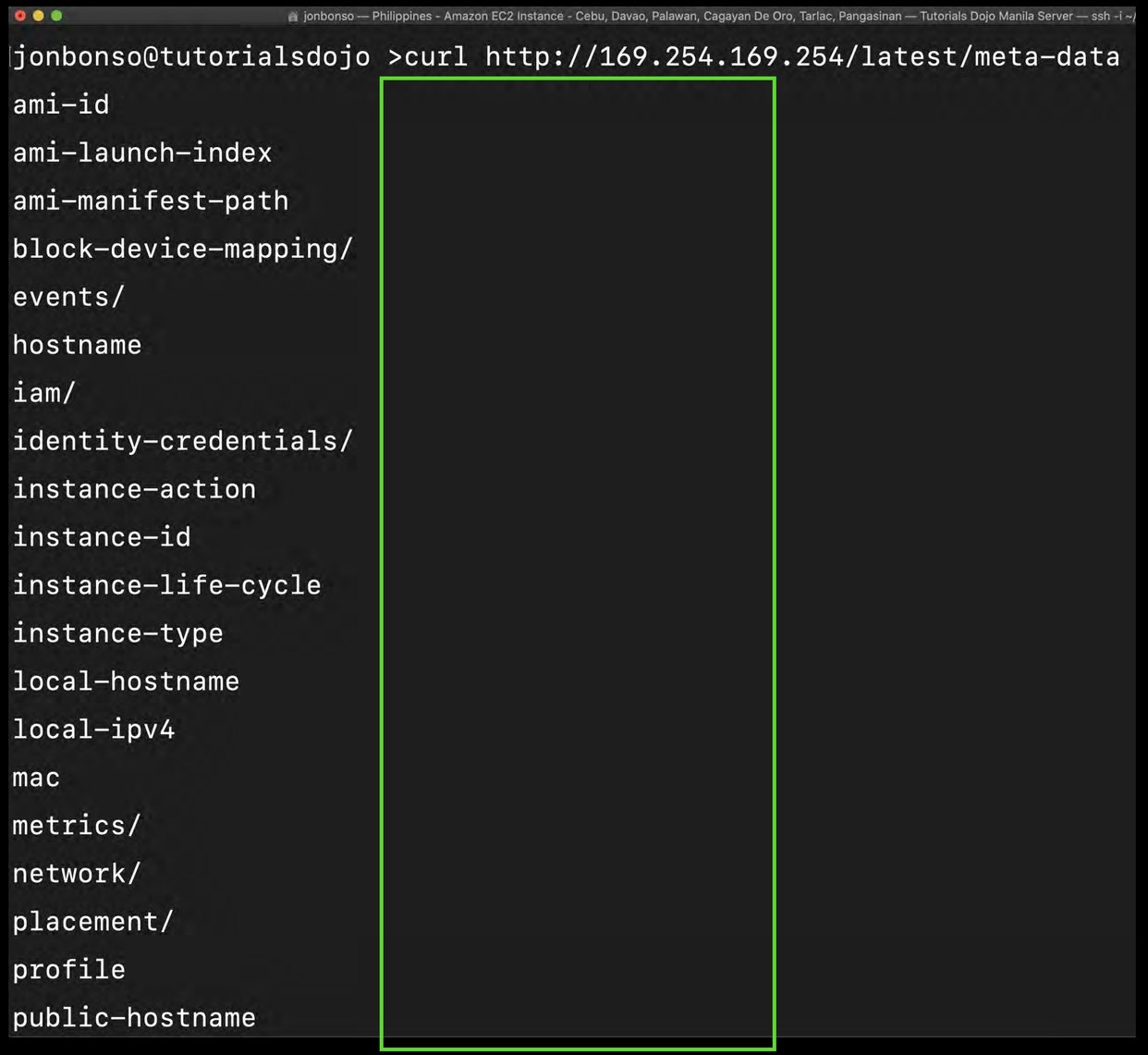
Windows



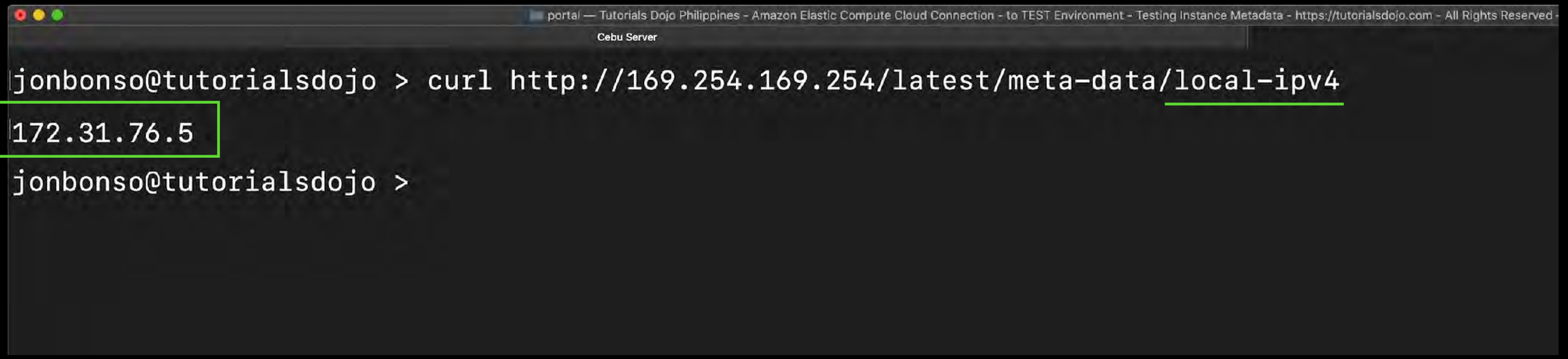
INSTANCE METADATA SERVICE version 2

Session Oriented

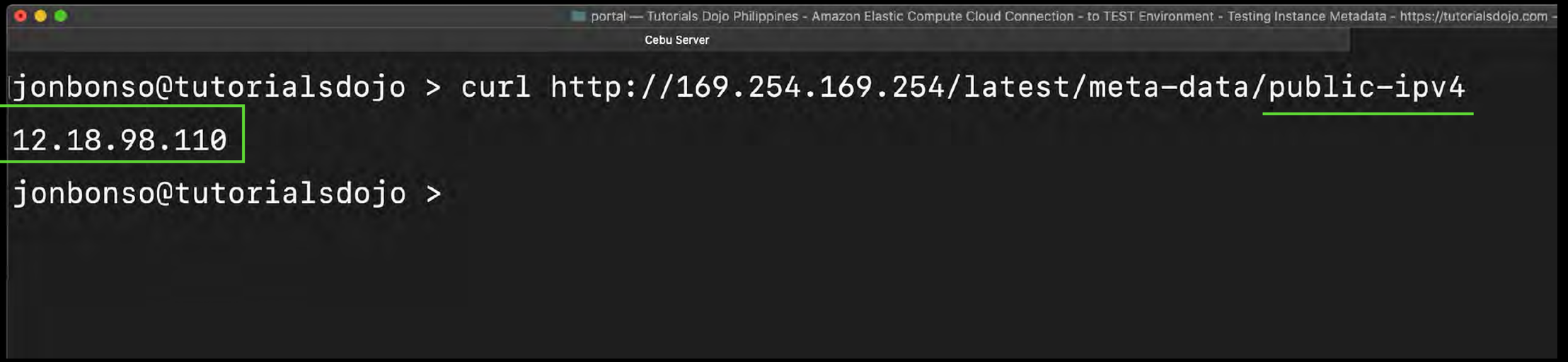
CATEGORIES



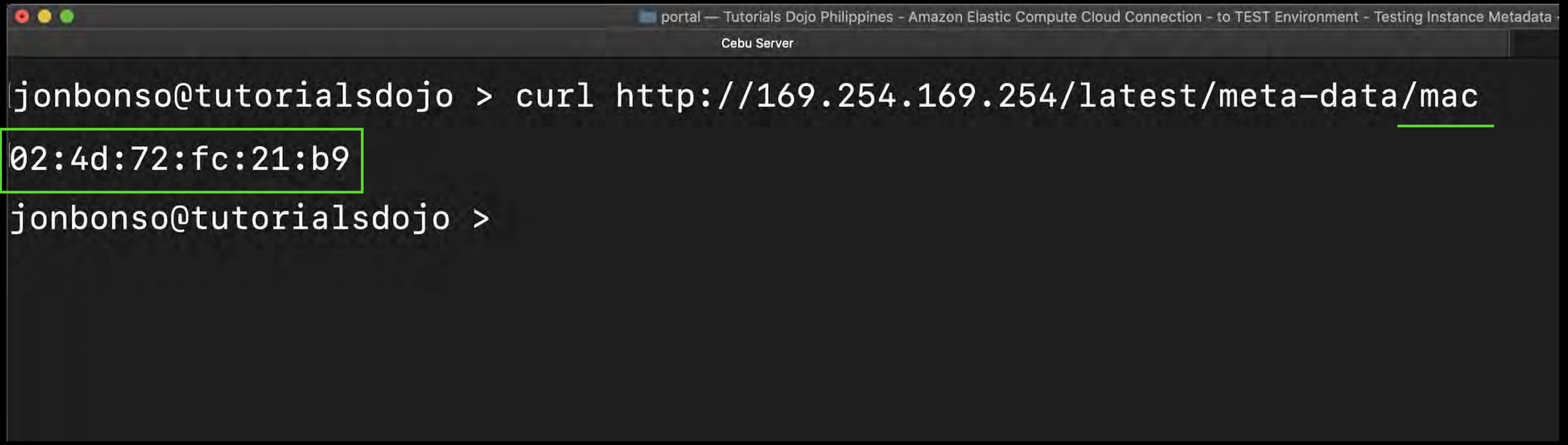
Private IP Address



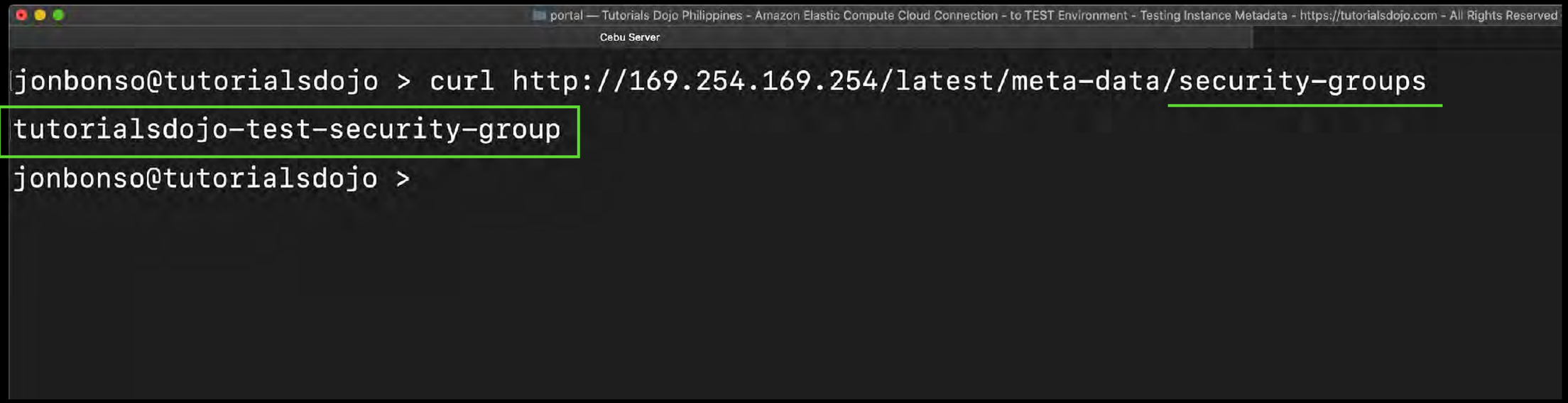
Public IP or Elastic IP Address



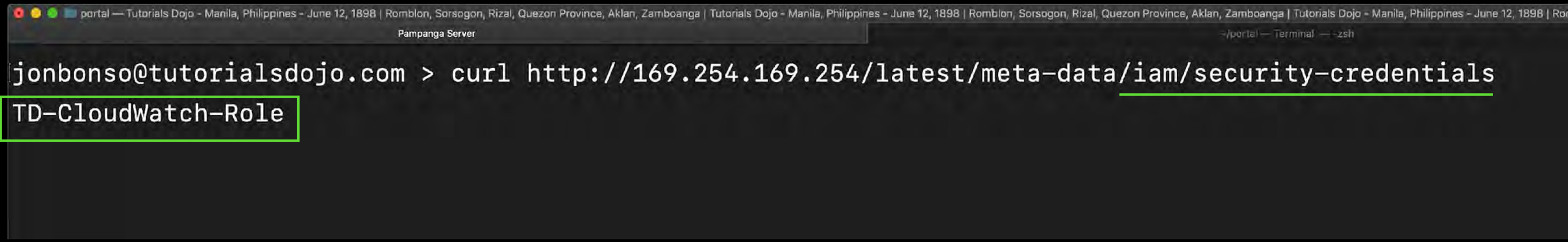
Media Access Control (MAC) Address



Security Groups



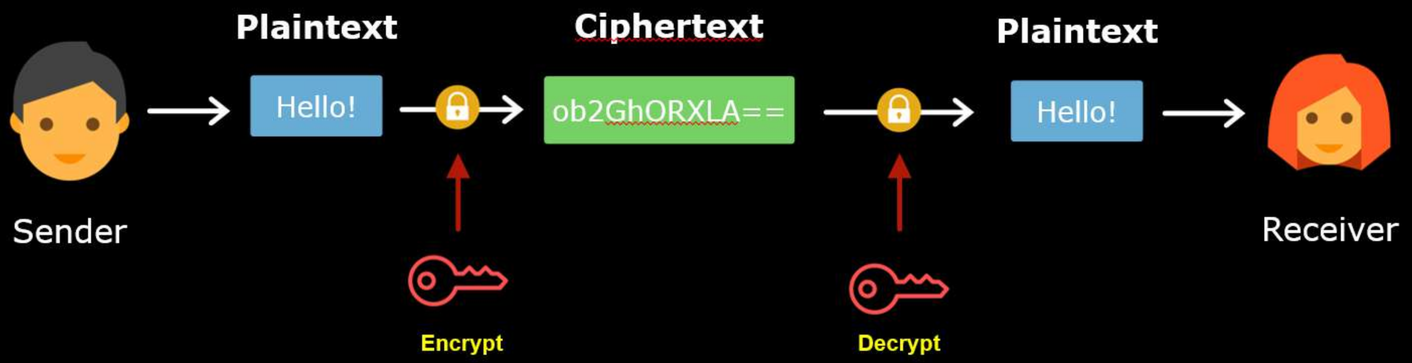
Instance Profile



AWS Key Management Service

AWS KMS

- A manage service that allows you to create, rotate, enable and disable encryption keys
- When you think of data encryption in AWS, think of AWS KMS
- Encryption - the process of converting plaintext data to ciphertext



AWS KMS

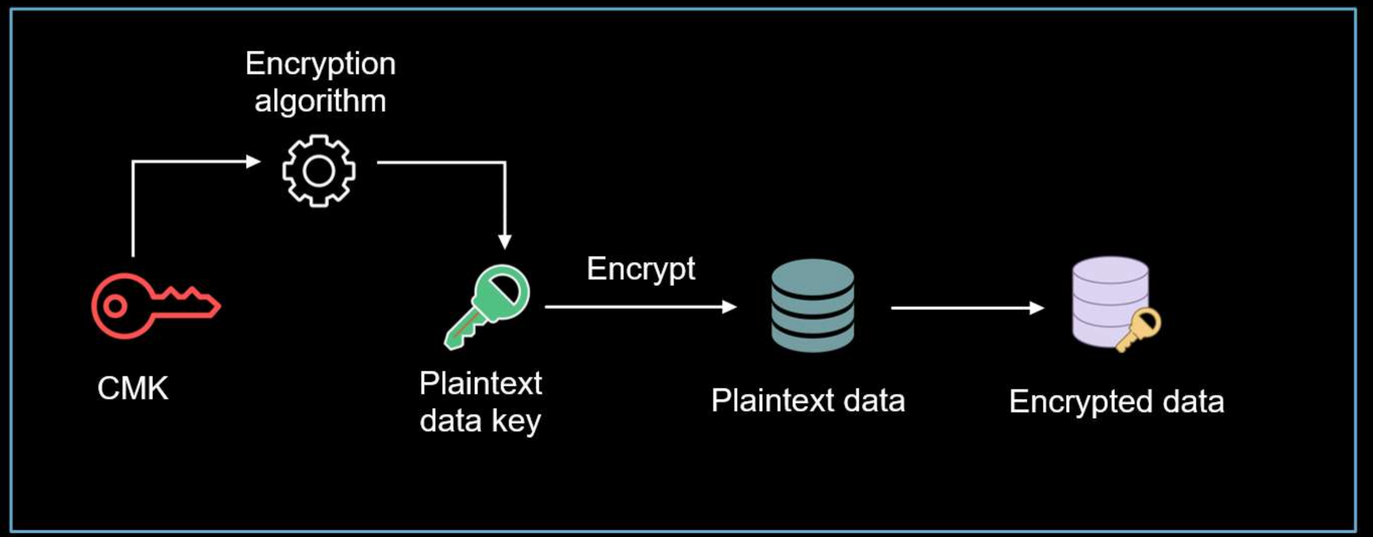
- Creating an encryption key is as easy as clicking a button or calling an API
- KMS uses a Key policy to control access to a KMS key
- You can use CloudTrail for auditing key usage

AWS KMS

- Customer Master Key (CMK)
  - a KMS resource that serves as your master or root key
- KMS supports symmetric and asymmetric encryption
- Symmetric encryption
  - uses a single key for encrypting and decrypting data
  - a symmetric CMK is a 256-bit AES key
  - AWS Services with KMS integration use symmetric CMKs
- Asymmetric encryption uses a related key pair:
  - Public (encrypt) and Private (decrypt) key
  - RSA and Elliptic curve (ECC) key pair
  - usually used for signing and verification

AWS KMS

- A CMK can encrypt data up to 4KB
- Use envelope encryption to encrypt data > 4KB
- Envelope encryption is the process of encrypting plaintext data using a CMK-generated data key



AWS KMS

Three types of CMK:

- Customer Managed CMK
  - CMKs that you create, own, and manage in your AWS account
  - KMS uses a managed *key material* to generate a CMK
  - You can import your own *key material*
  - You can enable automatic rotation (every 1 year) for CMKs generated using a KMS key material
  - You're charged monthly for each CMK + KMS API requests
  - You can't export CMKs generated by AWS KMS

AWS KMS

Three types of CMK:

- AWS-managed CMK
  - CMKs that are created, managed, and used on your behalf by an AWS Service
  - Free
  - automatically rotates every 3 years

AWS KMS

Three types of CMK:

- AWS-owned CMK

  - a collection of CMKs that AWS owns and manages for use in multiple AWS accounts

  - you cannot view, manage, or use AWS-owned CMKs, or audit their use

AWS KMS

- KMS uses a FIPS 140-2 level 2 validated-hardware security module to generate and store KMS keys
- Federal Information Processing Standard Publication (FIPS) is a standard that governs the approval of cryptographic modules
- HSM is a dedicated tamper-resistant computing device that uses physical processes to generate strong encryption keys

image by www.sefira.cz

- You share tenancy access to HSM with other customers
- Use AWS CloudHSM if you wish to procure single-tenant HSM

AWS KMS API Commands

AWS KMS

Encrypt

- converts plaintext data to ciphertext data using a CMK
- suitable for encrypting data under 4 KB

Decrypt

- converts ciphertext data to plaintext data using a CMK
- can only be used on data or data keys encrypted by a CMK

GenerateDataKey

- generates a data key and returns the plaintext and ciphertext version of it
- You can't use the data key generated by this command in an Encrypt API operation

GenerateDataKeyWithoutPlaintext

- generates a data key and returns only the ciphertext version of it
- use the Decrypt API to convert the ciphertext data key to plaintext in order to use it

Amazon S3 Overview

Amazon S3

- An **object storage** service
- S3 stands for **"Simple Storage Service"**
- **Highly durable, available & scalable** storage service
- Primarily **used to store static data** that does not change frequently
- Allows your files to be publicly available via the Internet

a set of name-value pairs

OBJECT

Highly scalable and allows you to store virtually unlimited amounts of files

BUCKET

BUCKET NAMING GUIDELINES

- The S3 bucket name is **globally unique**
- The namespace is **shared by all AWS accounts** around the world
- Example:
  - If you created an S3 bucket named "`tutorialsdojo`", then no other AWS user can **create a bucket with that same name**
  - If someone tries to create a new bucket called "`tutorialsdojo`", then that request will fail

Amazon S3 Folders and Prefixes

- Helps you organize or group your objects
- S3 has a **flat structure**
- The concept of a "folder" is not hierarchical unlike Amazon EFS
- Example:

Object key name

`tutorialsdojo/aws.jpeg`

Prefix | Filename

- Amazon S3 does NOT support POSIX, including:
  - Concurrent file modification
  - File system access semantics
  - File locking

AWS Cloud

N. Virginia Region

Automatically replicates your objects to **all Availability Zones** of the AWS region by default

Availability Zone (AZ) 2

Availability Zone (AZ) 3

YOUR VPC

AVAILABILITY 99.99%

DURABILITY 99.999999999%

- The probability that an object remains intact and accessible after a period of one year

DURABILITY

100%	Absolutely no data loss per year
99%	1% chance of data loss per year
99.99%	**0.01%** chance of data loss per year
99.999999999%	0.000000001% chance of data loss per year or one lost data every 10 million years

Amazon S3 Storage Classes

S3 Standard

For **frequently accessed** data

S3 Intelligent-Tiering

For changing or
unknown access patterns

S3 Standard-IA
(Infrequent Access)

S3 **One** Zone-IA
(Infrequent Access)

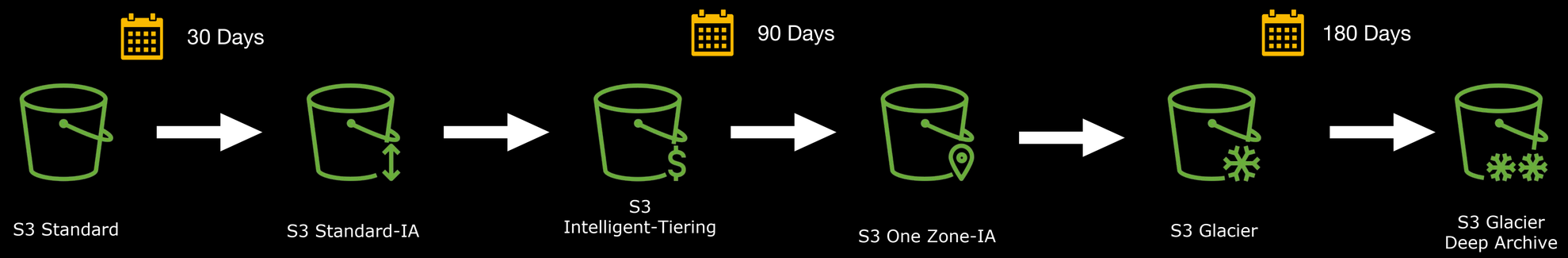
For storing long-lived,
yet **less frequently accessed** data

S3 Glacier

S3 Glacier **Deep** Archive

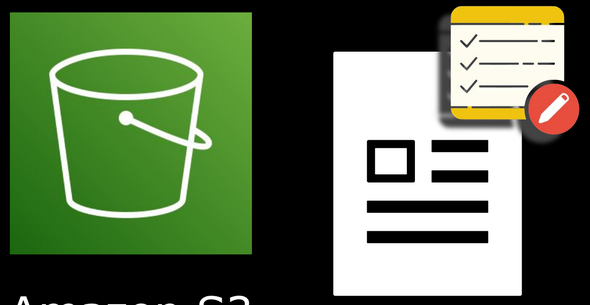
For **low-cost long-term storage**
and data archiving

Lifecycle Policy

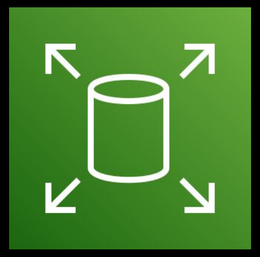


Static Website Hosting

- Launch a **static website** with HTML pages, downloadable packages, images, media files, or other **client-side scripts**
- **Cost-effective solution** for hosting your static websites with no server management required (serverless)
- **Cannot be used for running server-side scripts** such as PHP, JSP, ASP.NET etc…

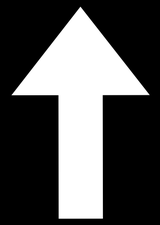


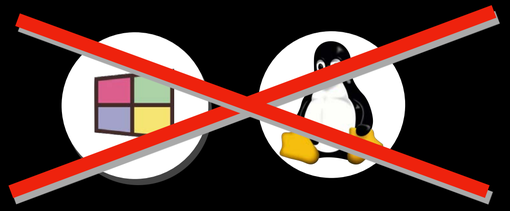
Amazon S3

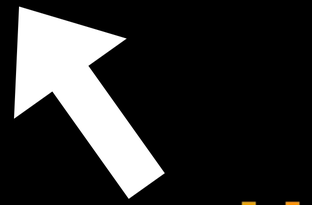


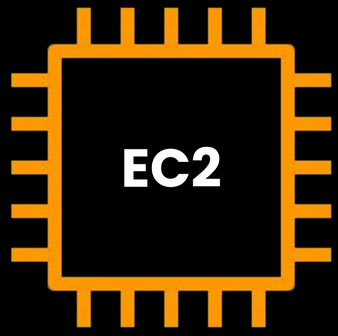
Amazon EBS

Amazon EFS









- Invoked via a **REST API** request call

- **Attached/Mounted** to the Amazon EC2 instance

S3 Versioning

Multi-Factor Authentication
(MFA)

- Prevent accidental data deletion in Amazon S3

Access Control List
(ACL)

- Secure access to your S3 buckets and objects

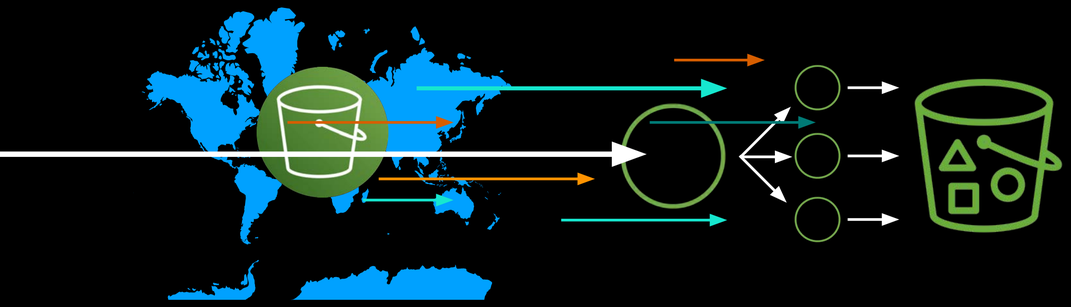
Bucket Policy

- Control external access to your Amazon S3 bucket

Cross Region Replication (CRR)

- Automatically replicate objects to a different AWS Region for backup purposes

---



Transfer Acceleration

Multipart Upload

- Accelerate or expedite the data transfer (upload/download) of S3 objects

...and many other S3 features!

Amazon S3 Storage Classes

Amazon S3 Storage Classes

S3 Standard

S3 Intelligent-Tiering

S3 Standard-Infrequent Access (Standard-IA)

S3 One Zone-Infrequent Access (One Zone-IA)

S3 Glacier

S3 Glacier Deep Archive

S3 Standard

- Primarily used for storing your data that are **frequently accessed**
- Highly durable, highly available, and high performance object storage
- Replicates your data to **3 or more** Availability Zones
- **99.99%** Availability
- **No minimum storage duration charge**
- **No data retrieval fee**

S3 Standard

USE CASES

- For setting up a highly available and durable **static web hosting**
- As a **temporary storage service** for storing the nightly log processing of your application, where the logs are meant to be stored for 1 day (24 hours) only. It is a cost-effective option for this case since it has **no minimum storage duration charge**

S3 Standard

LIMITATIONS

- **Not cost-effective** as this storage class is the **most expensive** among all other classes
- **Not recommended for data archiving, for infrequently access files** or for any workloads that require a cost-effective storage

S3 Standard-IA

- Primarily used for storing **infrequently accessed data** but provides a way to **rapidly retrieve the stored files**
- Replicates your data to **3 or more** Availability Zones
- **99.99%** Availability
- **30-day** minimum storage duration charge
- Has a data retrieval fee that is measured per gigabyte (GB)

S3 Standard-IA

USE CASES

- As a long-term storage for **long-lived, but infrequently accessed data**
- For data backups
- As a data store for your Disaster Recovery (DR) files
- For storing the **primary backup copies** of your on-premises dataset

S3 One Zone-IA

- For storing **less frequently accessed and easily reproducible data** that requires **immediate retrieval** when needed
- **30-day** minimum storage duration charge
- Cheaper than:  **S3 Standard-IA**
- Only uses **1 Availability Zone**
- **99.95% Availability** (*the **lowest** among all other Amazon S3 storage classes*)

S3 One Zone-IA

USE CASES

- If you require a **cost-effective option to store infrequently accessed data**
- For **workloads that do not require the availability and resilience** of the Amazon S3 Standard or S3 Infrequent Access class
- For storing **secondary backup copies of rarely-accessed on-premises dataset**
- For storing easily **recreatable data**

S3 One Zone-IA

LIMITATIONS

- The data is replicated in a **single AZ** only
- **Not recommended for storing your company's primary backup copies or any critical business data** that is difficult to reproduce

S3 Intelligent-Tiering

- Delivers **automatic cost savings**
- **Automatically moves your objects between different access tiers** whenever your access pattern changes
- **30-day** minimum storage duration charge
- **No data retrieval fee**
- Moves data to the most cost-effective access tier **without any operational overhead**
- Stores the objects in four access tiers:
  - 2 **low-latency** access tiers
  - 2 optional **archive** access tiers

S3 Intelligent-Tiering

USE CASES

- Suitable if your data has an **unpredictable access pattern**
- For buckets with a mix of frequent and infrequent accessed data
- If the access patterns to **your data vary all the time**
- If some of your files are accessed frequently while the **others are rarely accessed** (move to Glacier)
- If some of your data are **accessed less frequently** than others (move to IA tier)
- If **you are unsure** of how frequently your data will be accessed

S3 Intelligent-Tiering

USE CASES

- If you want to **keep costs low by automatically moving your data** to the appropriate S3 storage class
- If your data will be **accessed by users over variable periods of time**
- If you need storage with **no management overhead**
- If you want to **avoid lifecycle policies that are not consistently implemented** or are partially implemented

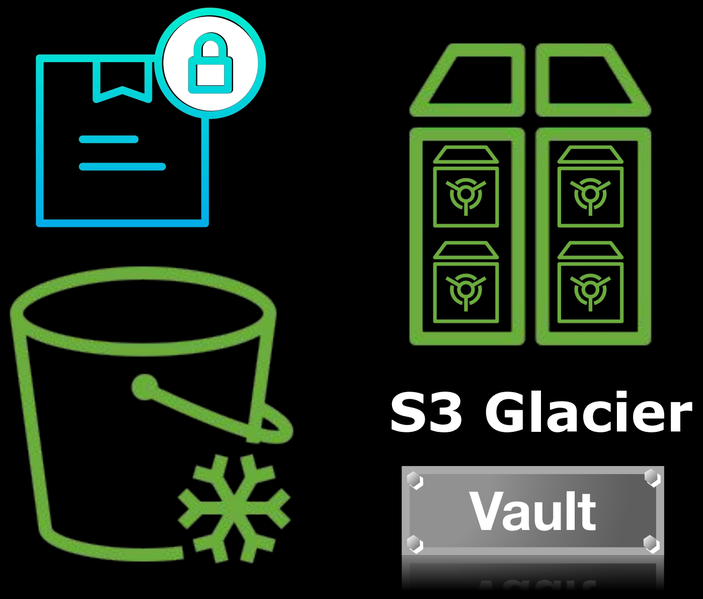
S3 Glacier

- A secure, durable, and low-cost storage
- Suitable for **data archiving**
- A cost-effective storage solution **for rarely accessed data and does not require a fast retrieval time**
- Replicates your data to **3 or more** Availability Zones
- **99.99%** Availability
- **90 day**-minimum storage duration charge
- **High data retrieval fee** (expensive)

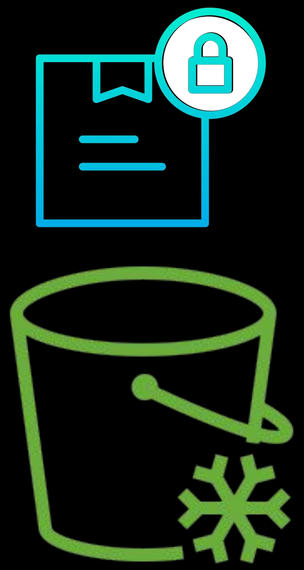
S3 Glacier

- **Has its own management console** apart from the regular Amazon S3 console
- 2 Ways to store your data:
  - Using the Amazon S3 console
  - Using the Amazon Glacier console
- Automatically move your data from S3 Standard or S3 Standard-IA to Amazon S3 Glacier by using a **lifecycle policy**

- Has a resource called: **Vault**
- A vault is a **container for storing your data archives**
- Base unit of storage in S3 Glacier, containing a **unique ID** and **an optional description**
- Can only be created in the Amazon S3 Glacier console
- You must provide the vault name and its corresponding AWS Region



- Use a **Vault Lock** to ensure data integrity and access control to your Amazon S3 Glacier Vaults
- A Vault Lock is an access policy that helps you **enforce regulatory and compliance requirements**
- You can specify a **"Write Once Read Many" (WORM)** control to lock your Glacier vault policy from future edits
- A Glacier vault access policy **can no longer be changed** when the vault lock process has been completed after 24 hours



S3 Glacier

USE CASES

- Applicable if your company wants to **retain its archives for a specific number of years** before the files can be deleted

- If you want to **deny users from modifying or deleting an archive** until after 1 year, 3 years, 7 years et cetera

S3 Glacier Archival Retrieval Options

EXPEDITED

- Quickly access a **subset of your data archives**
- Allows you to access your archived data within **1 - 5 minutes** (file size should **NOT** exceed **250 MB**)
- Ensure sufficient retrieval capacity for your *Expedited* retrieval operations by purchasing **provisioned capacity**

STANDARD

- **Default option** for retrieval requests
- Allows you to access any of your glacier archives within **3 – 5 hours**

BULK

- **Lowest-cost** retrieval option
- Retrieves large amounts of data archive in less than half a day
- Typically completes the process within **5 – 12 hours**

S3 Glacier Deep Archive

- The **lowest-cost** storage class in Amazon S3.
- Supports **long-term retention and digital preservation** for your data
- Primarily used to retain your data sets for **7 to 10 years or longer to meet regulatory compliance requirements**
- Replicates your data to **3 or more** Availability Zones
- **99.99%** Availability

S3 Glacier Deep Archive

- **180-day** minimum storage duration charge (*roughly* ***6 months***)
- Should be used for data archiving only
- The data stored here should be **rarely accessed with no strict retrieval time**

S3 Glacier Deep Archive - Retrieval Options

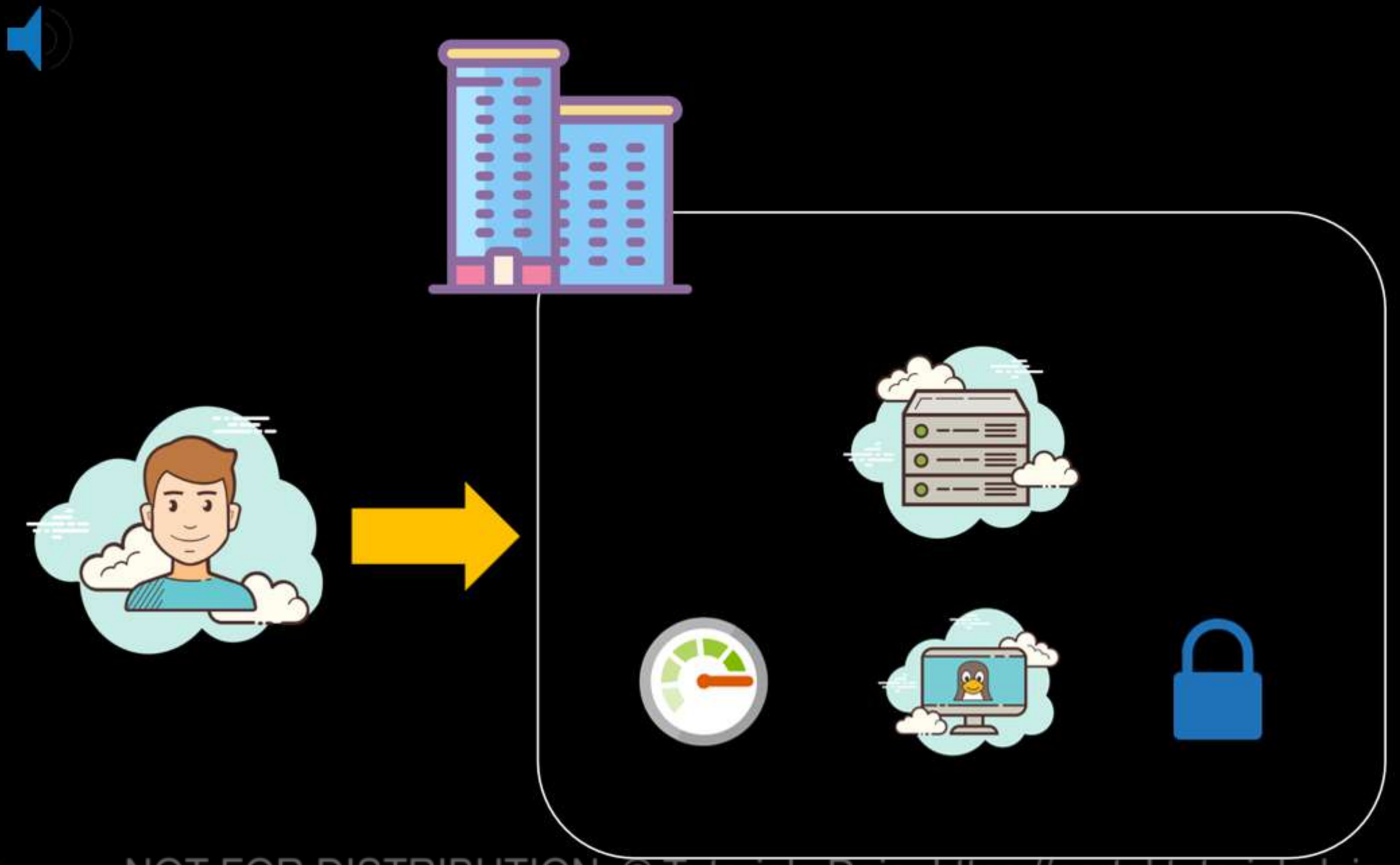
STANDARD

- **Default option** for retrieval requests
- Data will be restored within **12 hours**

BULK

- **Costs lower** than the Standard retrieval option
- Data will be restored within **48 hours**

AWS Lambda

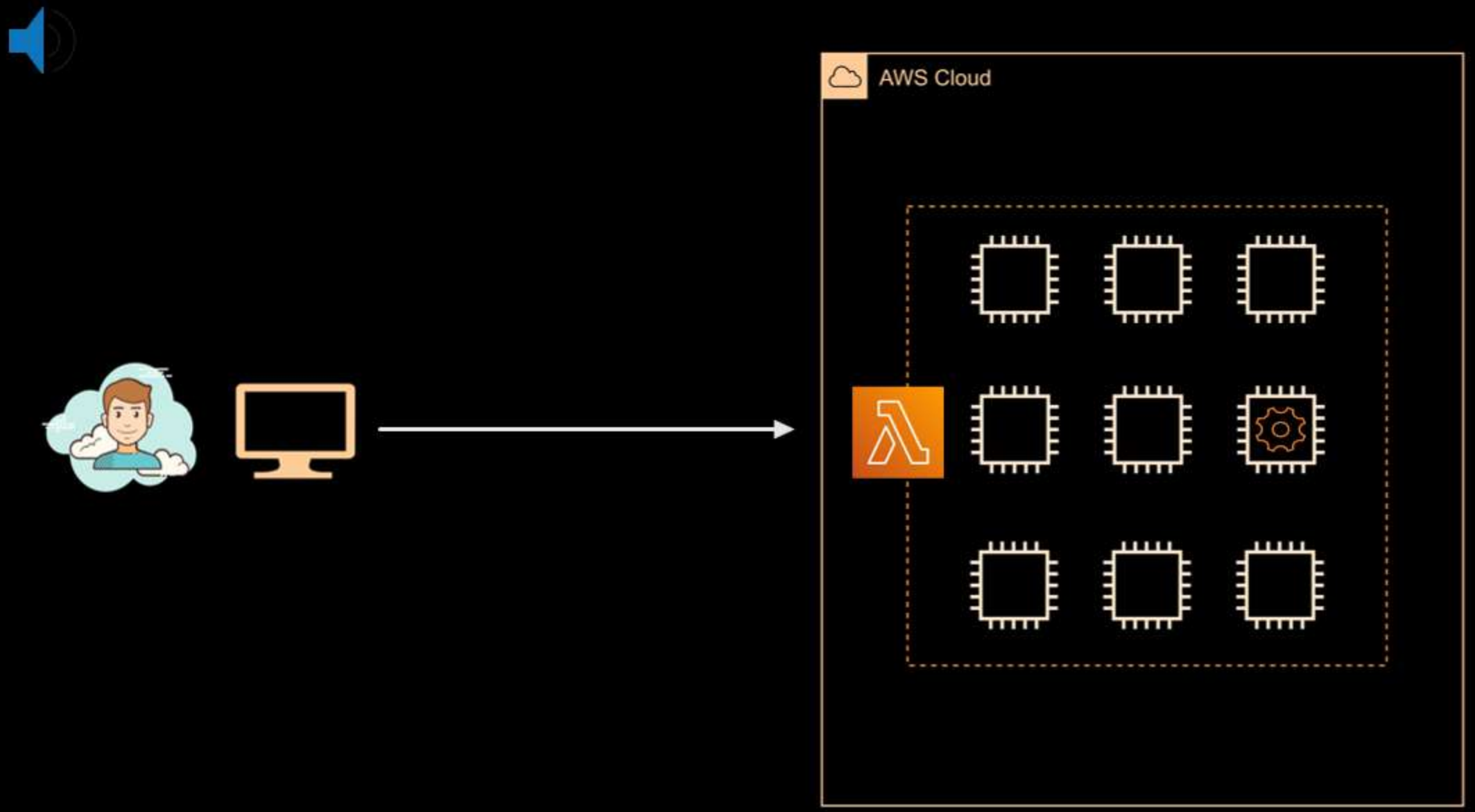


No servers to manage!

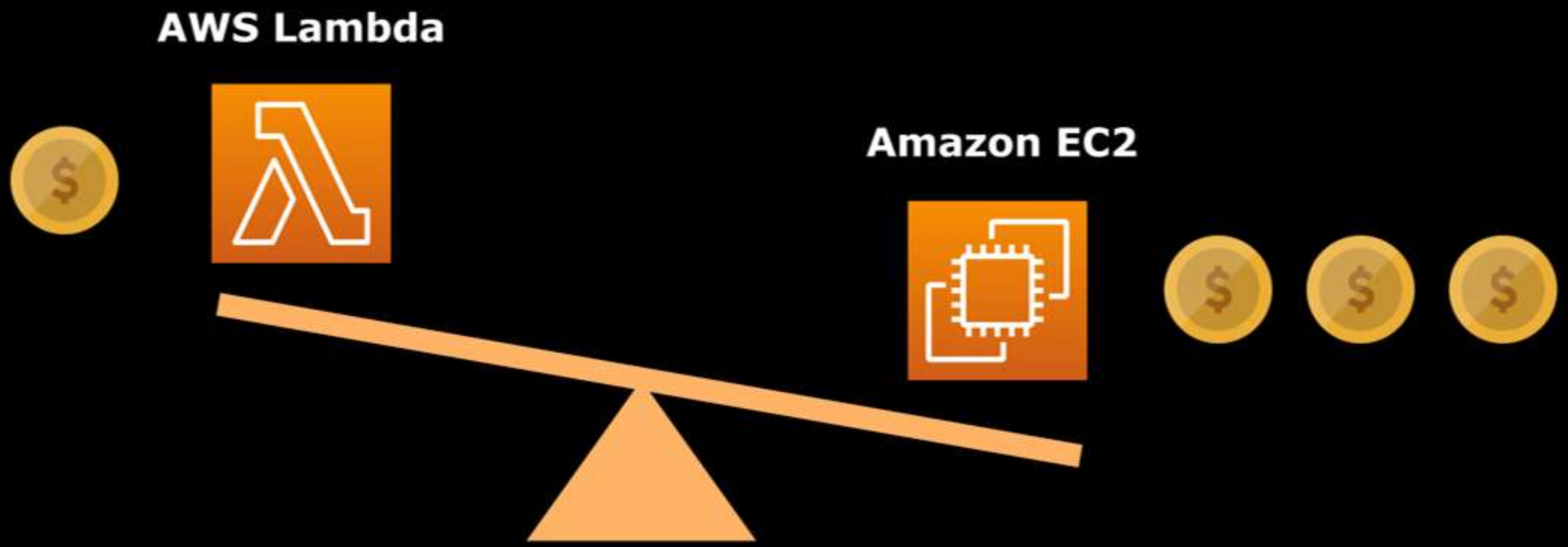
Create a Lambda function

Deploy your code

Run



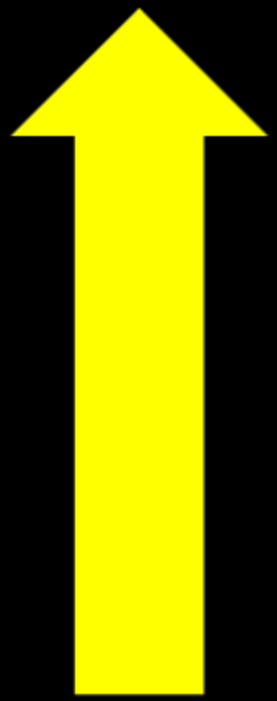
Faster development time



AWS Lambda is a lot cheaper than Amazon EC2

You only pay for invocation requests and the time your function executes

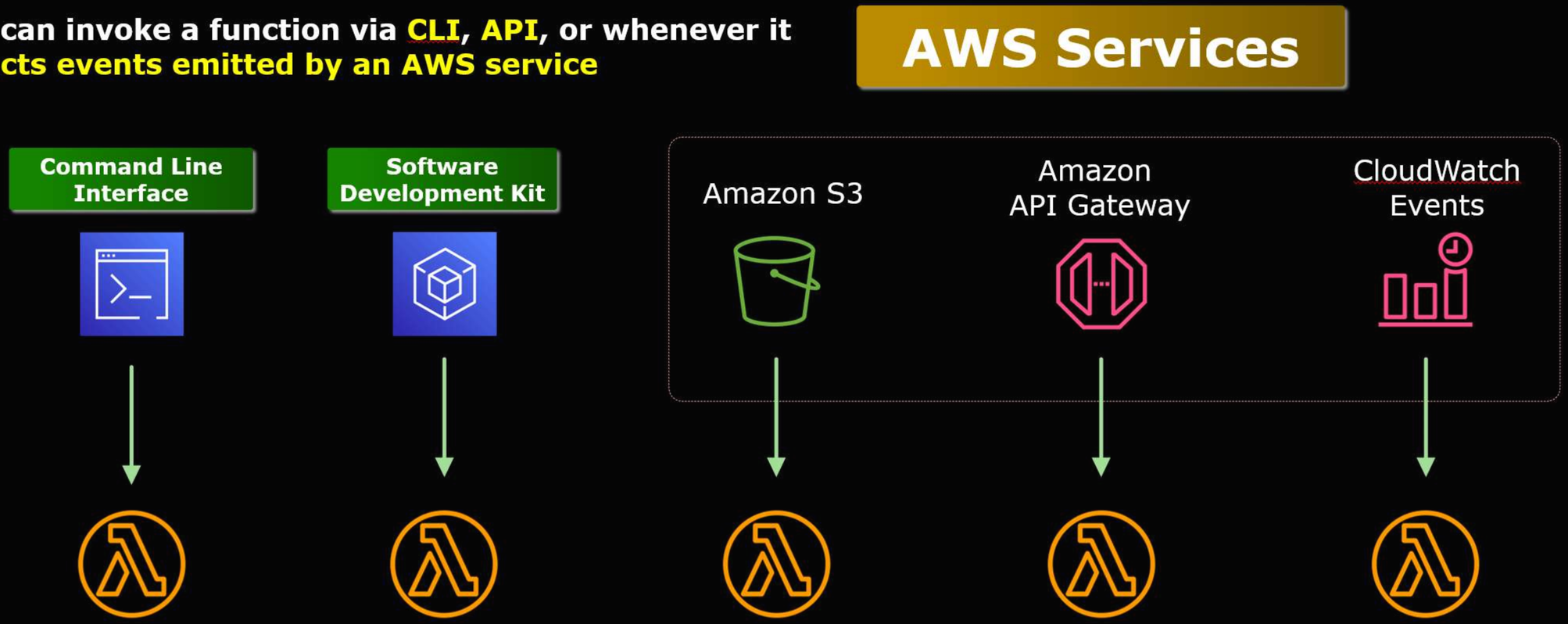
Instance	vCPU*	Mem (GiB)	Storage	Dedicated EBS Bandwidth (Mbps)	Network Performance
m4.large	2	8	EBS-only	450	Moderate
m4.xlarge	4	16	EBS-only	750	High
m4.2xlarge	8	32	EBS-only	1,000	High
m4.4xlarge	16	64	EBS-only	2,000	High
m4.10xlarge	40	160	EBS-only	4,000	10 Gigabit
m4.16xlarge	64	256	EBS-only	10,000	25 Gigabit

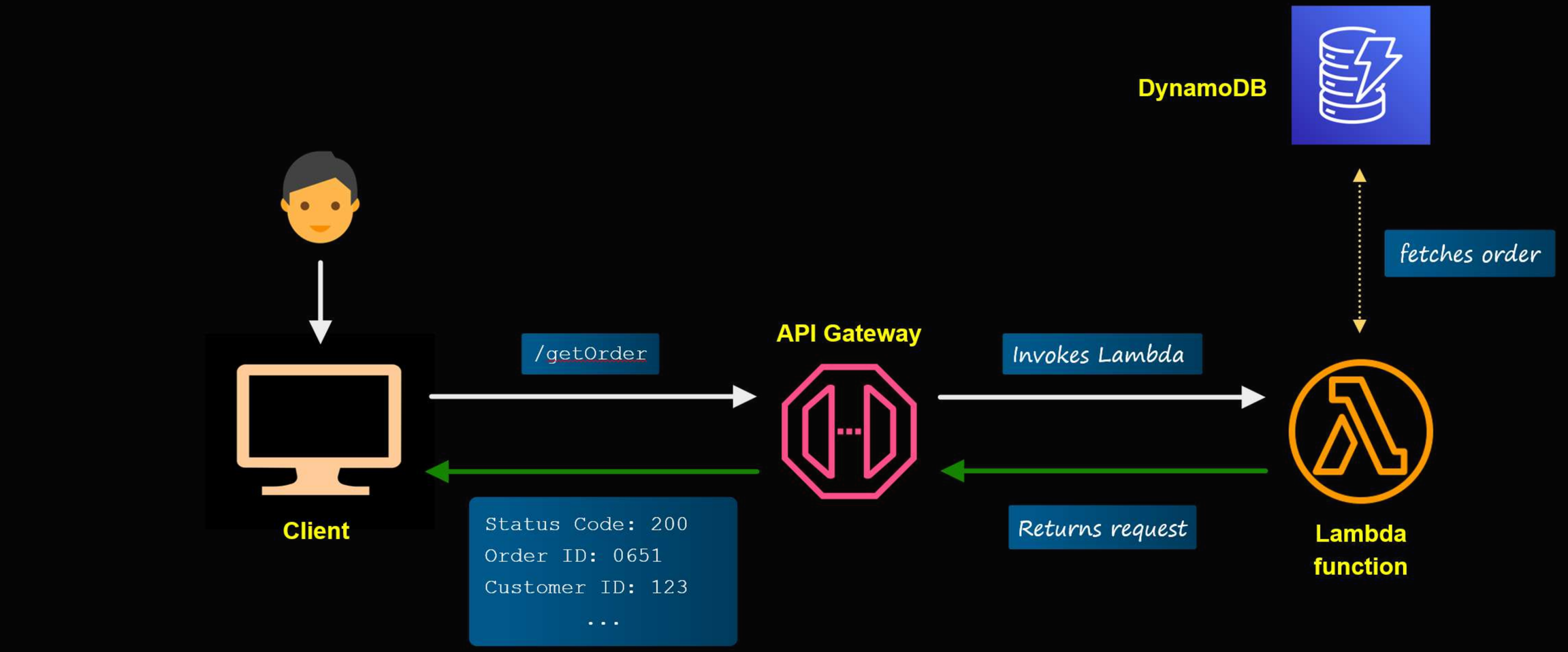


The higher the memory

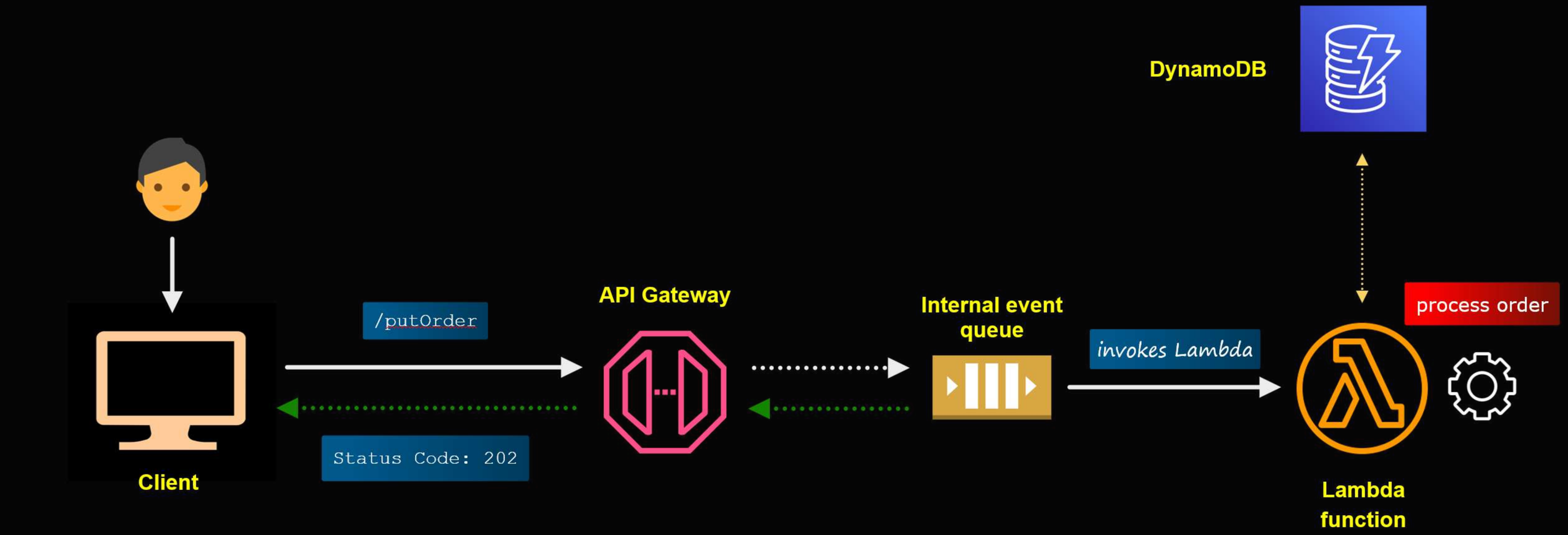
The shorter the execution time of the Lambda function

- You can invoke a function via CLI, API, or whenever it detects events emitted by an AWS service



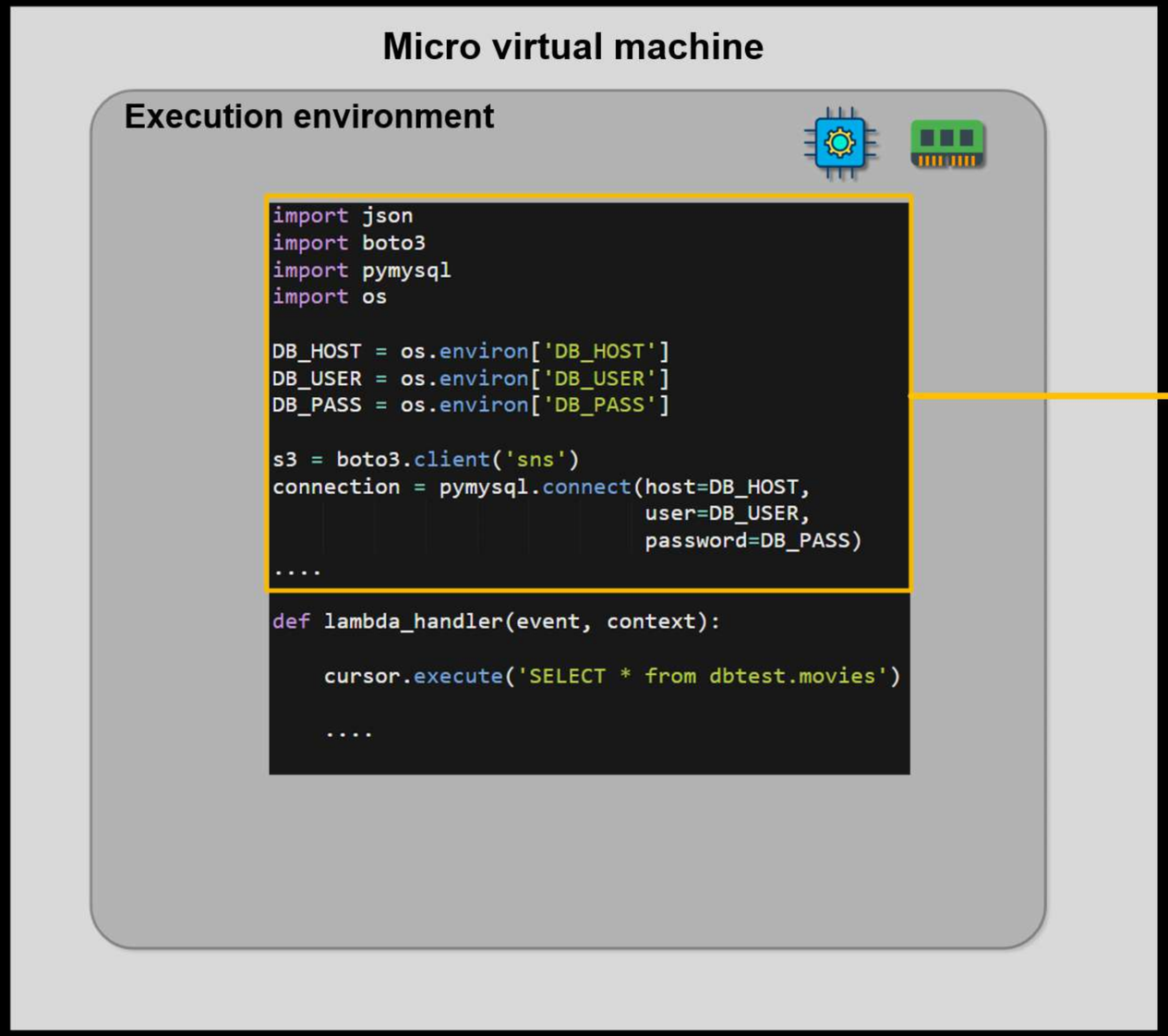


- Use for operations that don't need to wait and return a response from the Lambda function
- Suitable for batch operations, video encoding, order processing, etc.
- Uses an internal event queue to invoke the function



INIT

- Occurs when a Lambda function is invoked for the first time or after a long period of inactivity
- The time it takes to bootstrap the execution environment is called the **cold start**
- In most cases, you don't pay for the **cold start**
- The INIT phase has a time limit of 10 seconds
- You pay for cold starts longer than 10 seconds
- Only import the dependencies that your function needs to reduce cold start

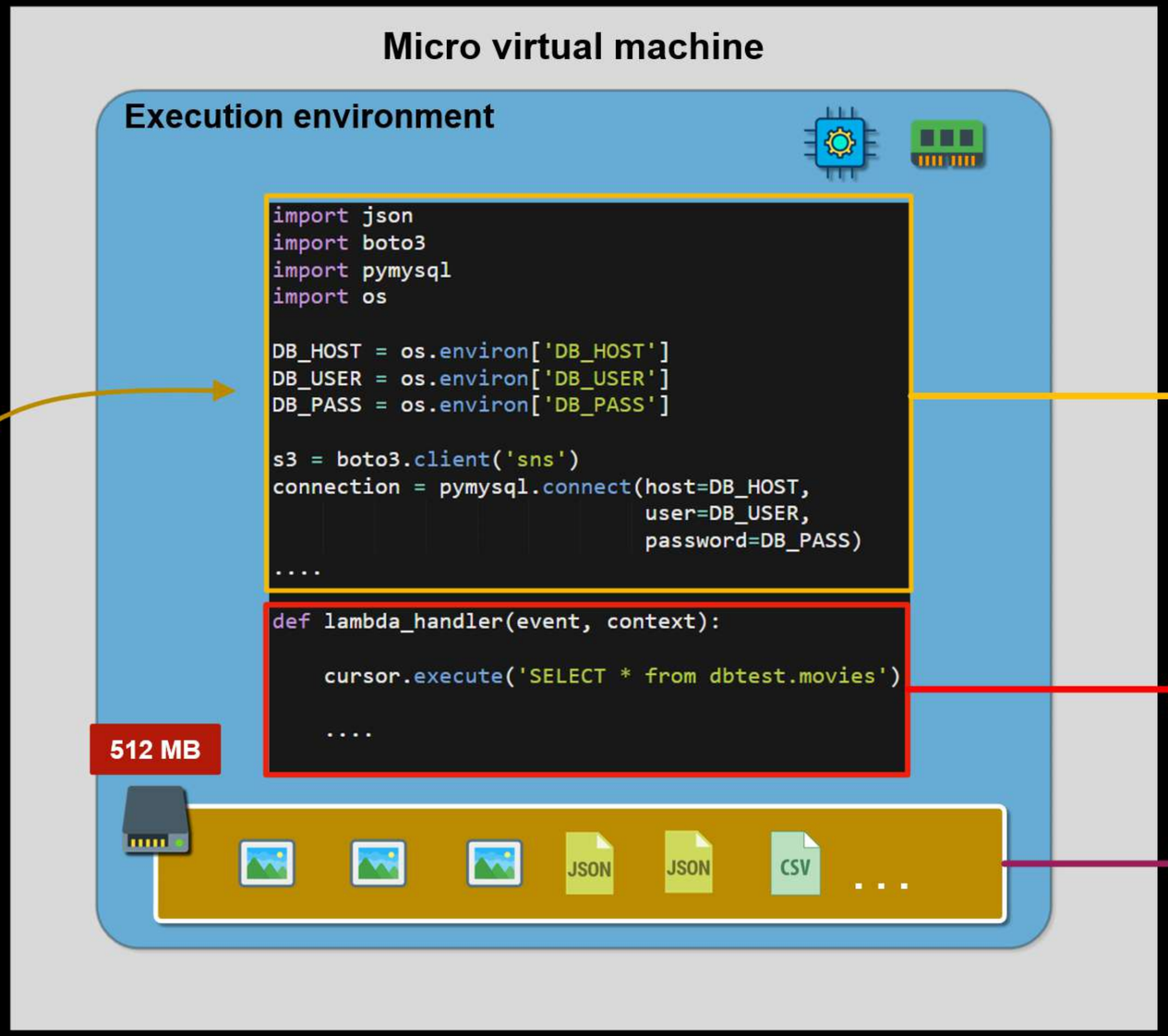


INVOKE

- The Lambda function handler is executed
- Only the function handler is executed during the INVOKE phase

Execution environment / Context Reuse

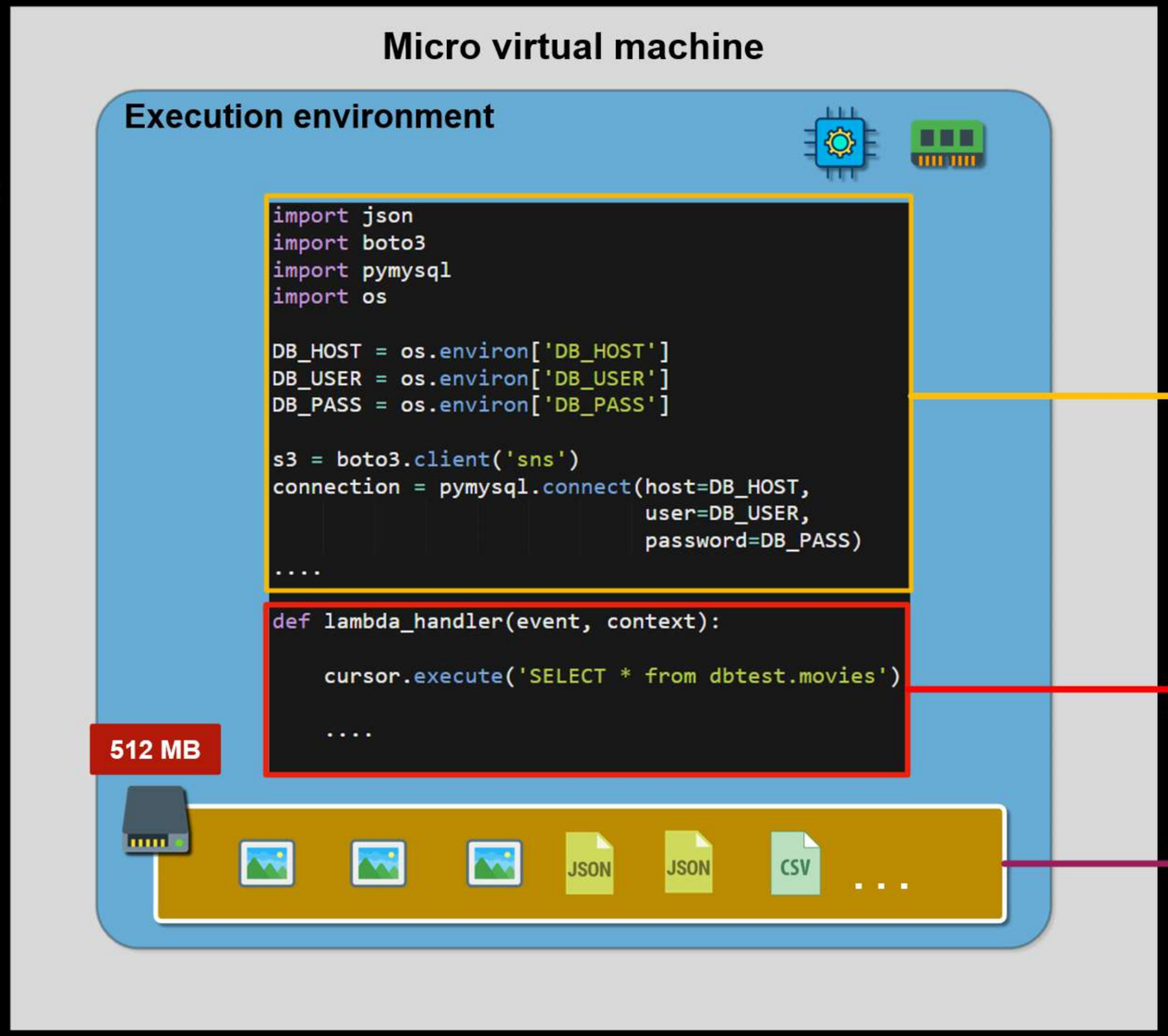
- Reusing the resources that were initialized during the INIT Phase
- Reusing the initialized resources will help lower the total execution time
- The execution context is temporary. Avoid storing state data in it



SHUTDOWN

- AWS Lambda terminates the execution environment after a period of inactivity

- AWS Lambda wipes all initialized resources

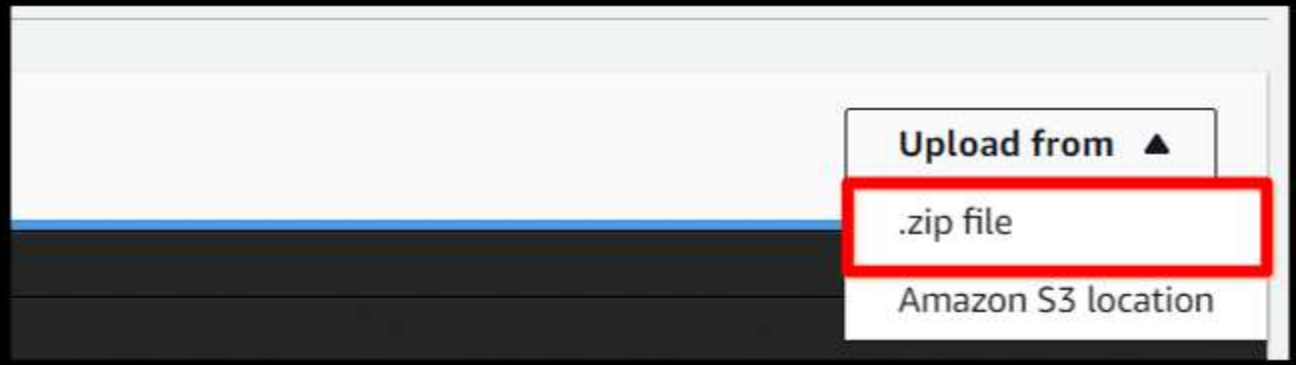


Deploying codes with external dependencies to AWS Lambda

STEP 1 Install all external dependencies in the same folder where your code is stored

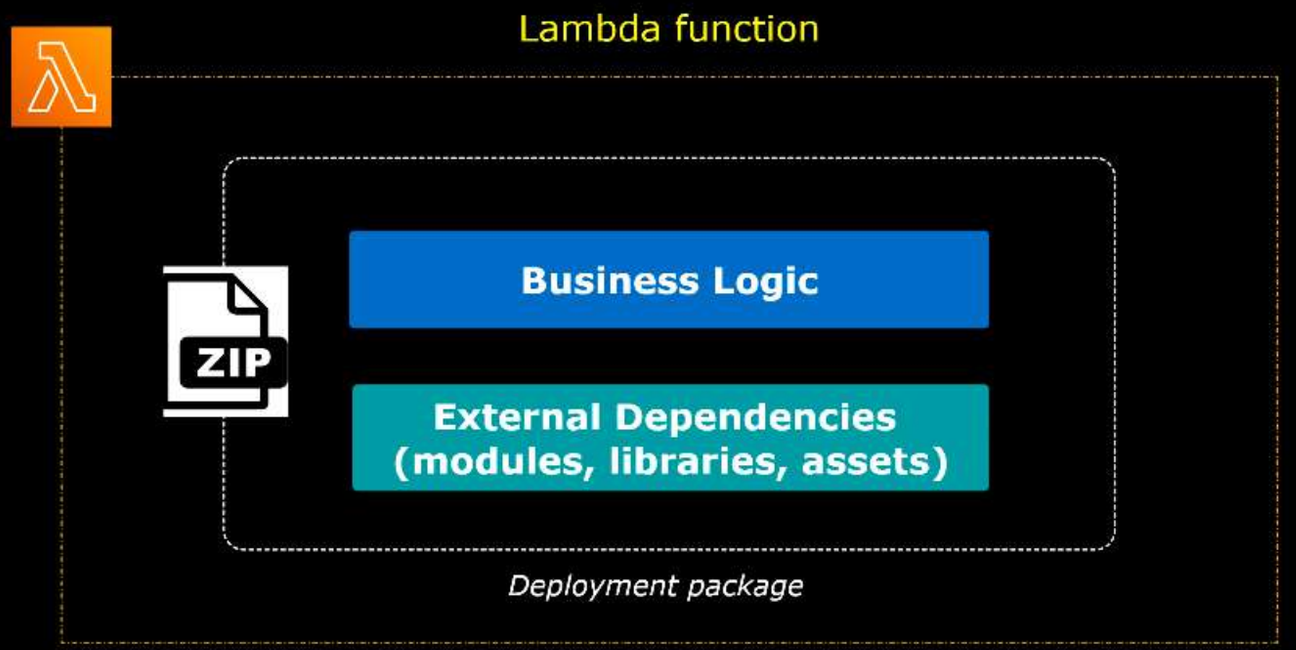
STEP 2 Create a deployment package by zipping up the whole folder

STEP 3 Upload the deployment package to your Lambda function



Lambda Layers

Structure of a typical Lambda function

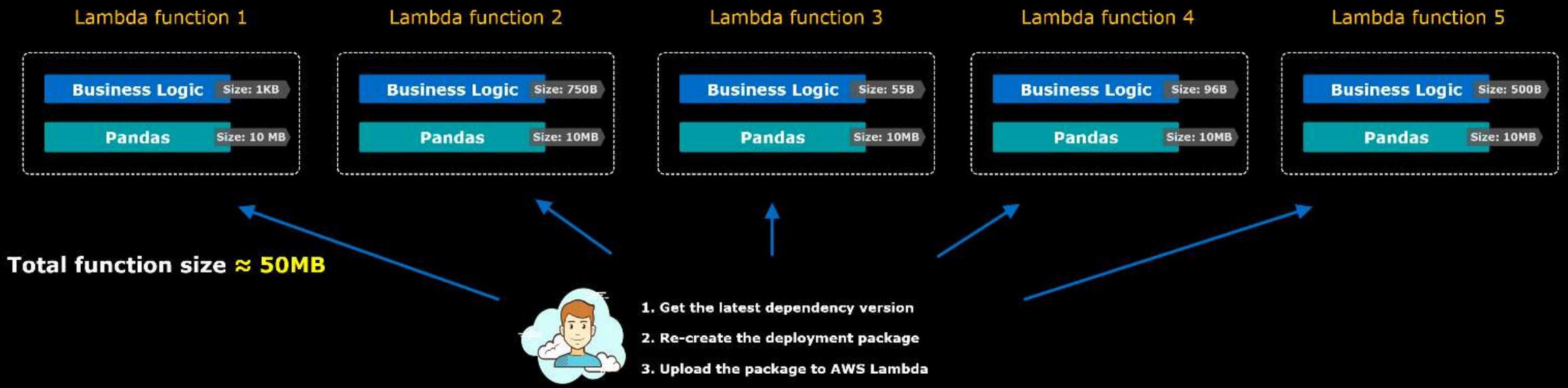


This set up is not good for multiple functions sharing the same external dependencies

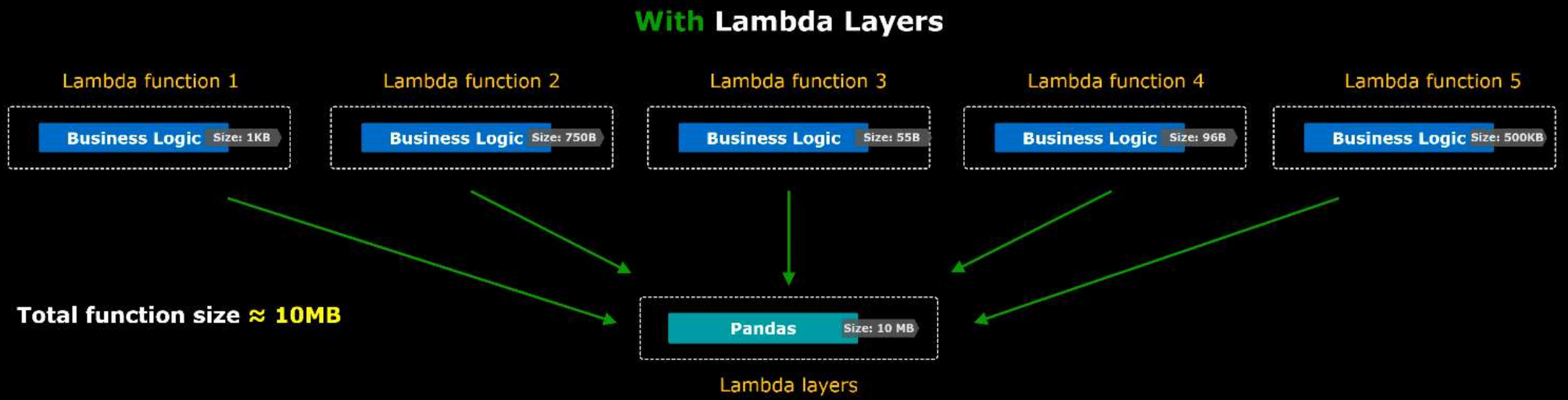
- The code size for each function is large
- Updating the dependencies will consume a lot of time and effort

Lambda Layers

Without Lambda Layers

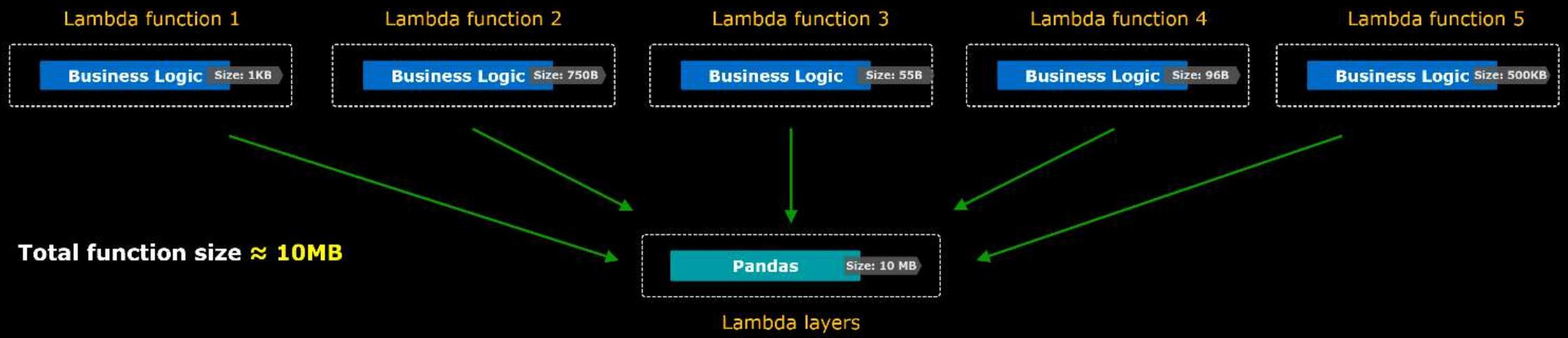


Lambda Layers



- Use Lambda Layers to store external dependencies that will be used by multiple functions
- External dependencies are deployed to Lambda Layers as zip files

Lambda Layers

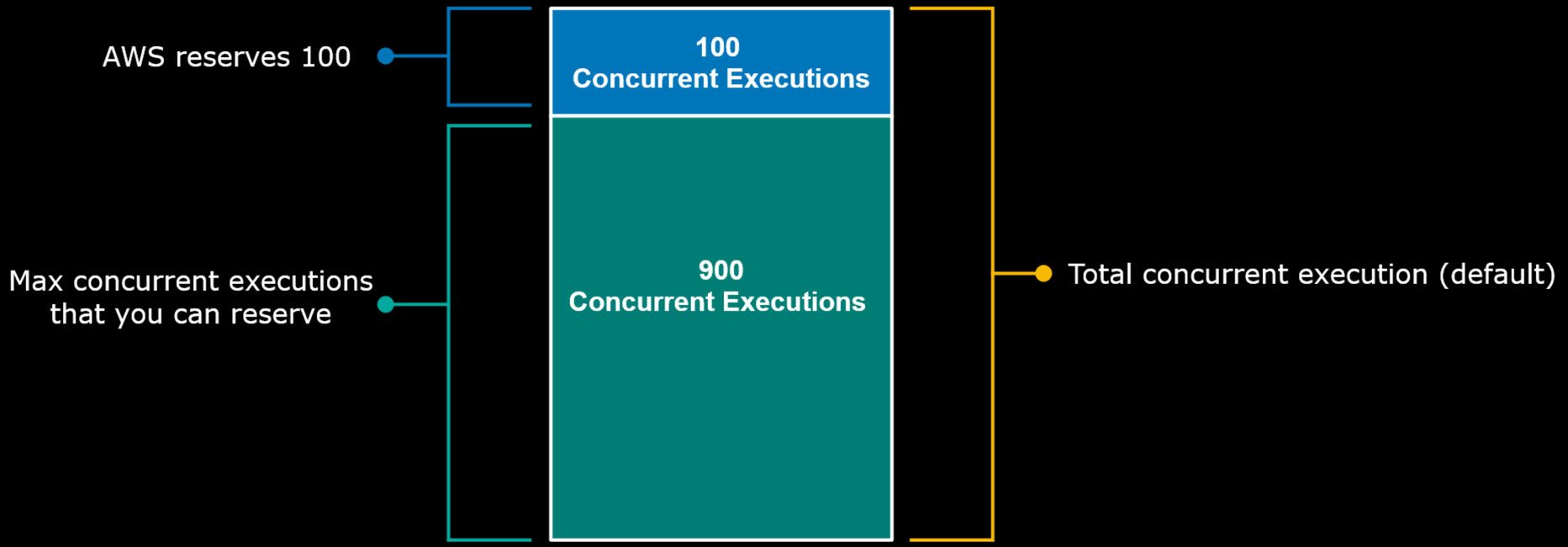


Advantages

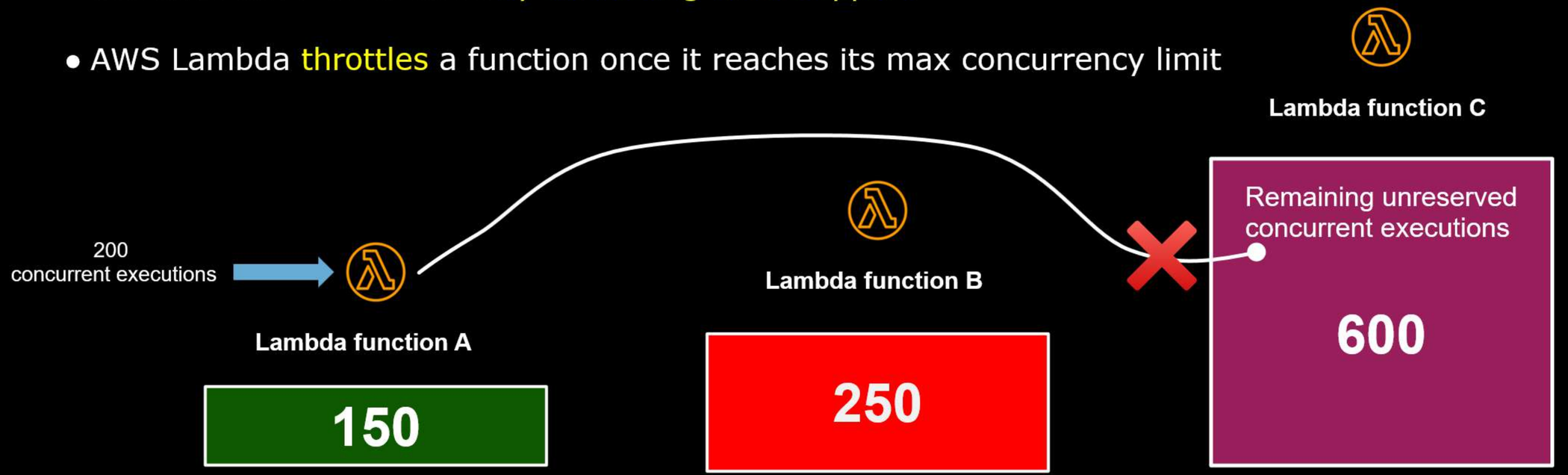
- External dependencies are shareable
- Lambda Layers reduces the size of your deployment package and the size of all your functions
- Results in faster deployments
- Lambda Layers give you a separation of concern. You can now develop your business logic and manage dependencies independently

Concurrency in AWS Lambda

- The number of Lambda functions that can run simultaneously at a given period of time
- Each AWS region has a default limit of 1,000 unreserved concurrent executions
- The limit can be increased by contacting AWS Support



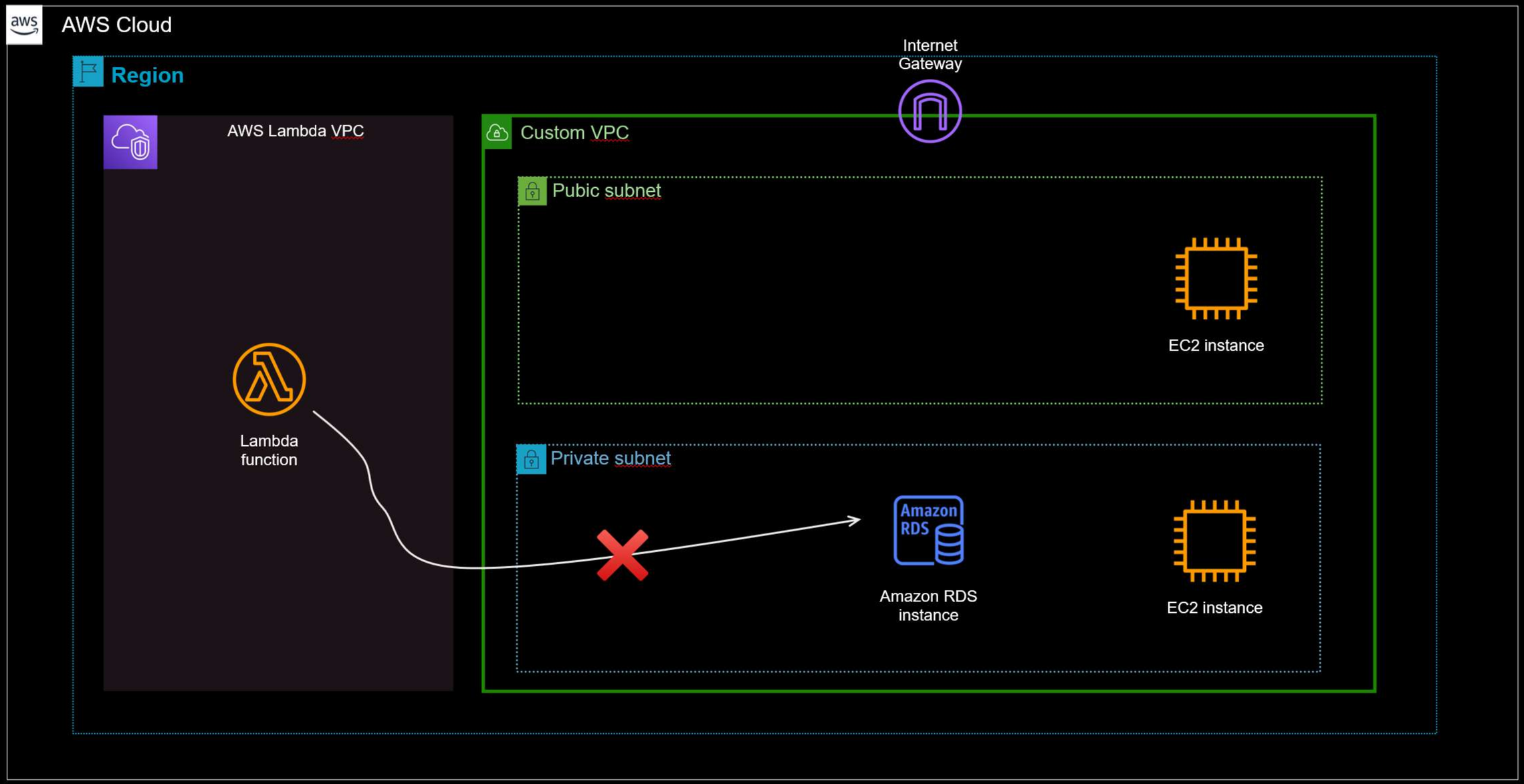
- The number of Lambda functions that can run simultaneously at a given period of time
- Each AWS region has a default limit of 1,000 unreserved concurrent executions
- The limit can be increased by contacting AWS Support
- AWS Lambda throttles a function once it reaches its max concurrency limit

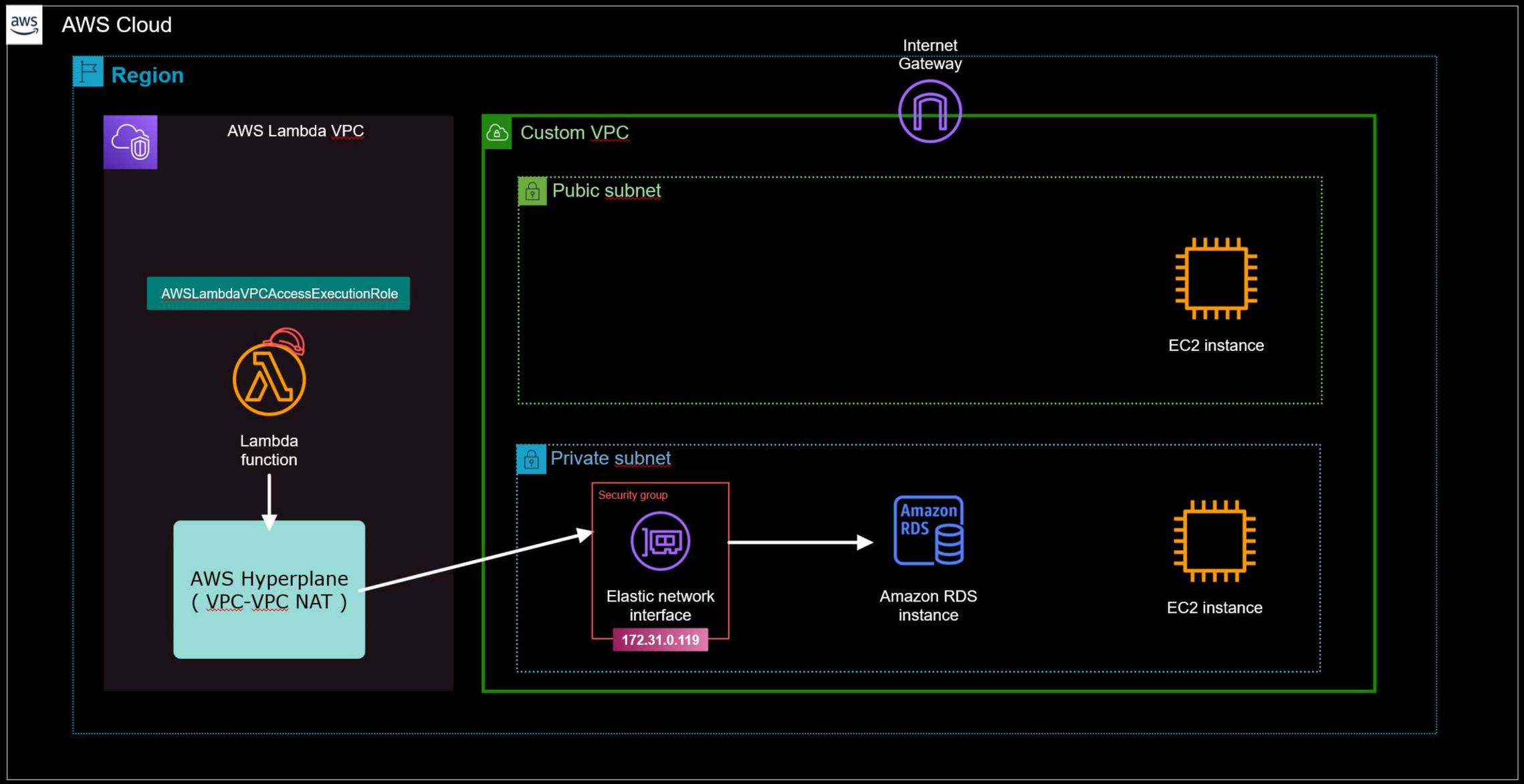


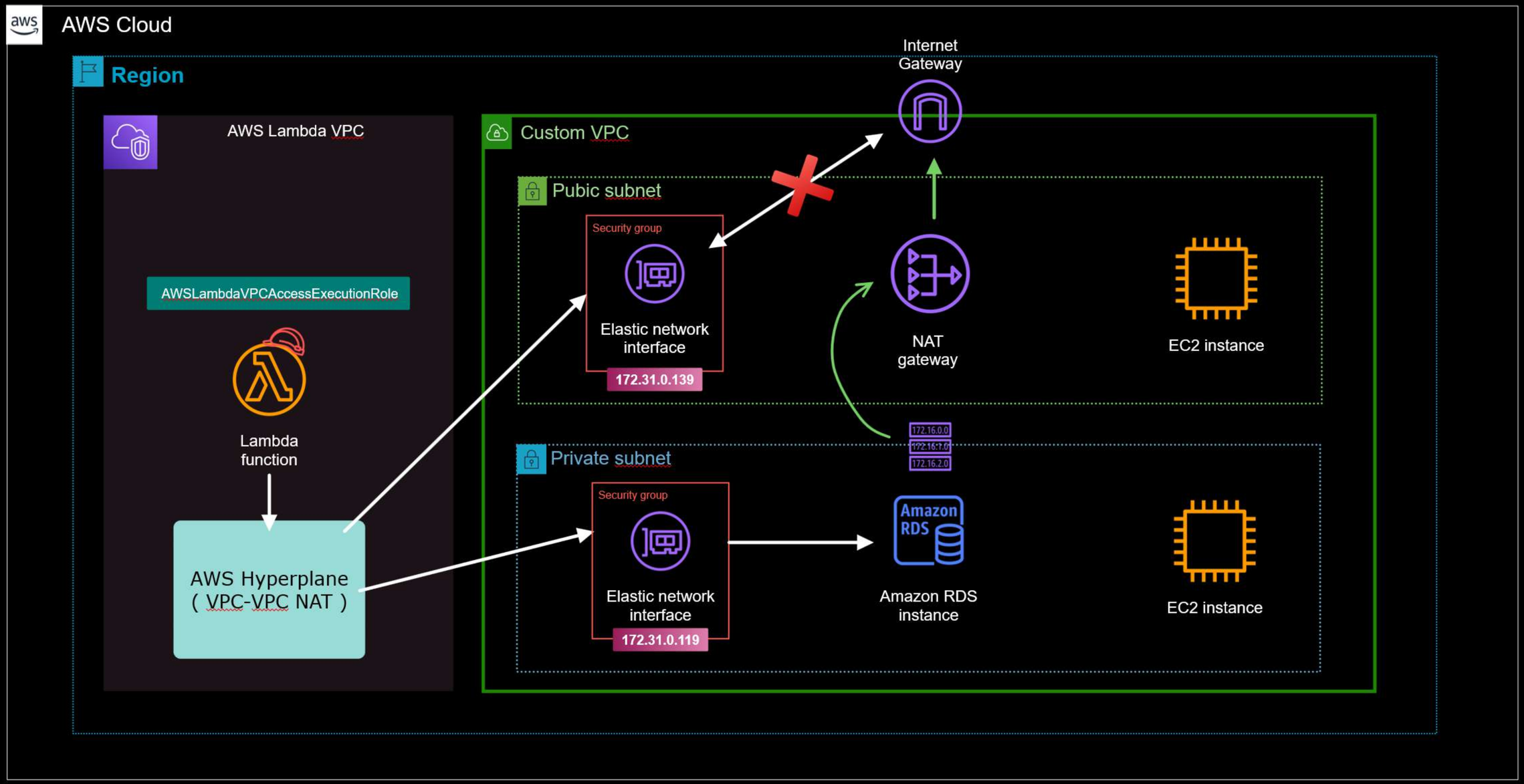
- The number of Lambda functions that can run simultaneously at a given period of time
- Each AWS region has a default limit of 1,000 unreserved concurrent executions
- The limit can be increased by contacting AWS Support
- AWS Lambda throttles a function once it reaches its max concurrency limit
- You can increase the number of reserved concurrent executions to a function to reduce throttling

Advantages of reserving concurrency:

- It prevents your Lambda function from scaling out of control
- It prevents functions from blocking other functions from scaling







Amazon DynamoDB

- A **serverless** and **non-relational database**

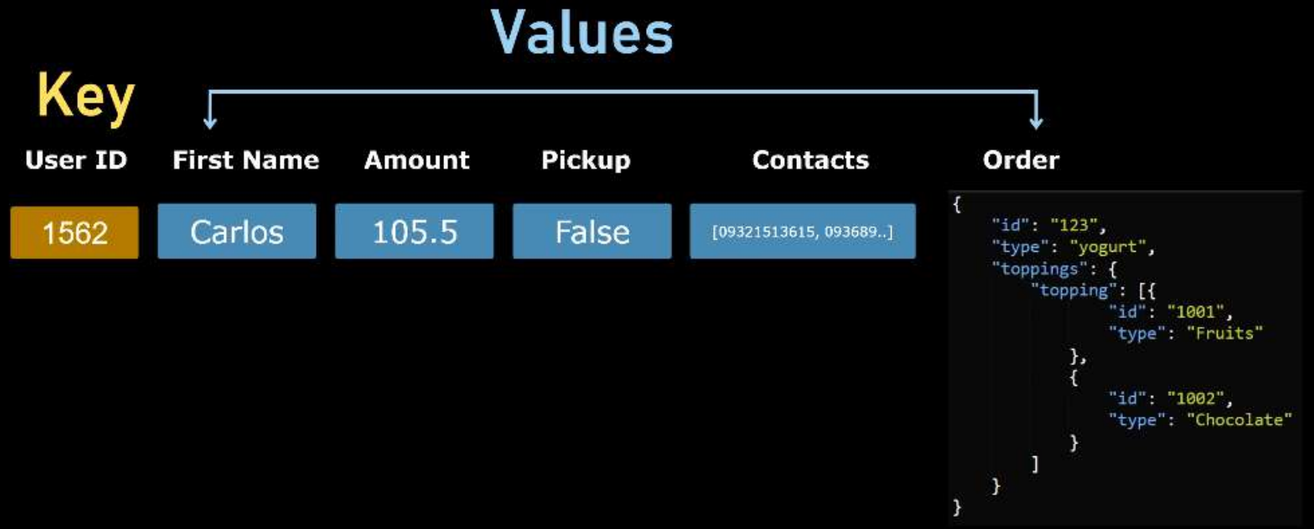
Amazon DynamoDB

- A **serverless** and **non-relational database**
- A mix of **key-value** and **document** database model

Key	Value
User ID	**First Name**
1562	Carlos
1553	John

Amazon DynamoDB

- A **serverless** and **non-relational database**
- A mix of **key-value** and **document** database model



Amazon DynamoDB

- A **serverless** and **non-relational database**
- A mix of **key-value** and **document** database model
- Provides **high-throughput** and **single-digit latency** performance
- It **scales automatically**
- Requires **zero database administration**
- Replicates data across **multiple AZs** within a region
- Built-in **fault tolerance**


```
connection = pymysql.connect(host = 'tdojo.cs2bzkifrtow.us-east-2.rds.amazonaws.com',
                             user= 'tdojo',
                             port = 3306,
                             password= Tdojo12345678)
```

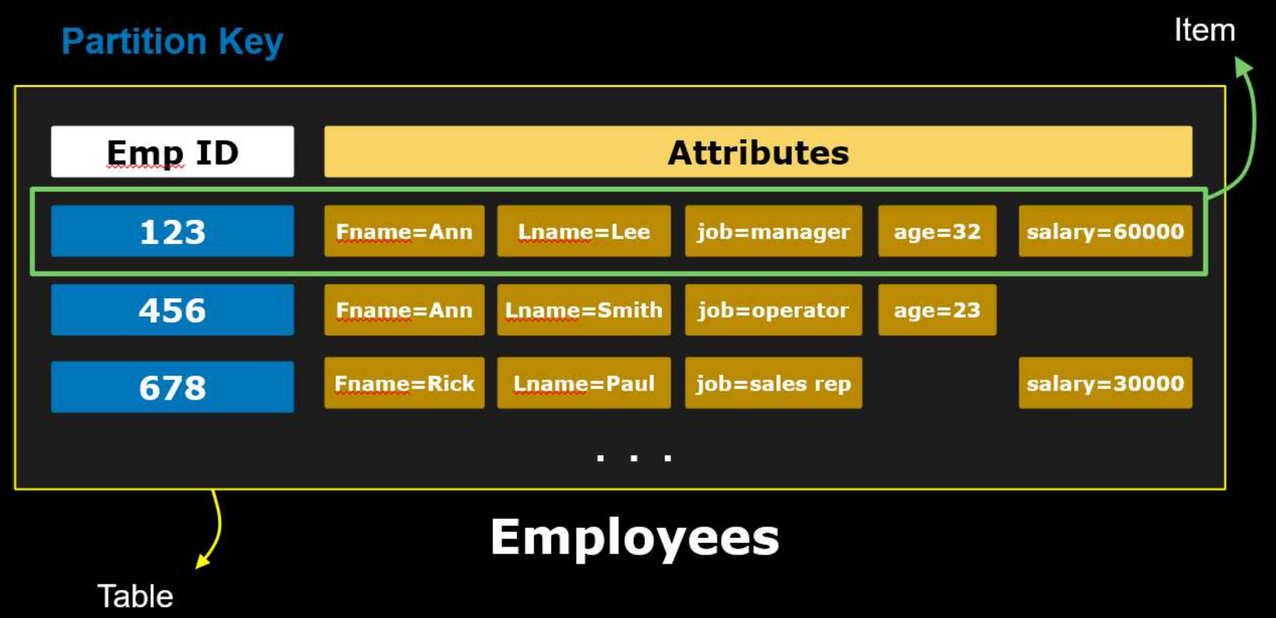
```
import boto3

dynamo = boto3.resource('dynamodb')
table = dynamo.Table('Users')
table.put_item( Item = eventItem)
```

Amazon DynamoDB

- A **serverless** and **non-relational database**
- A mix of **key-value** and **document** database model
- Provides **high-throughput** and **single-digit latency** performance
- It **scales automatically**
- Requires **zero database administration**
- Replicates data across **multiple AZs** within a region
- Built-in **fault tolerance**
- Can also be accessed using SQL queries via **PartiQL**

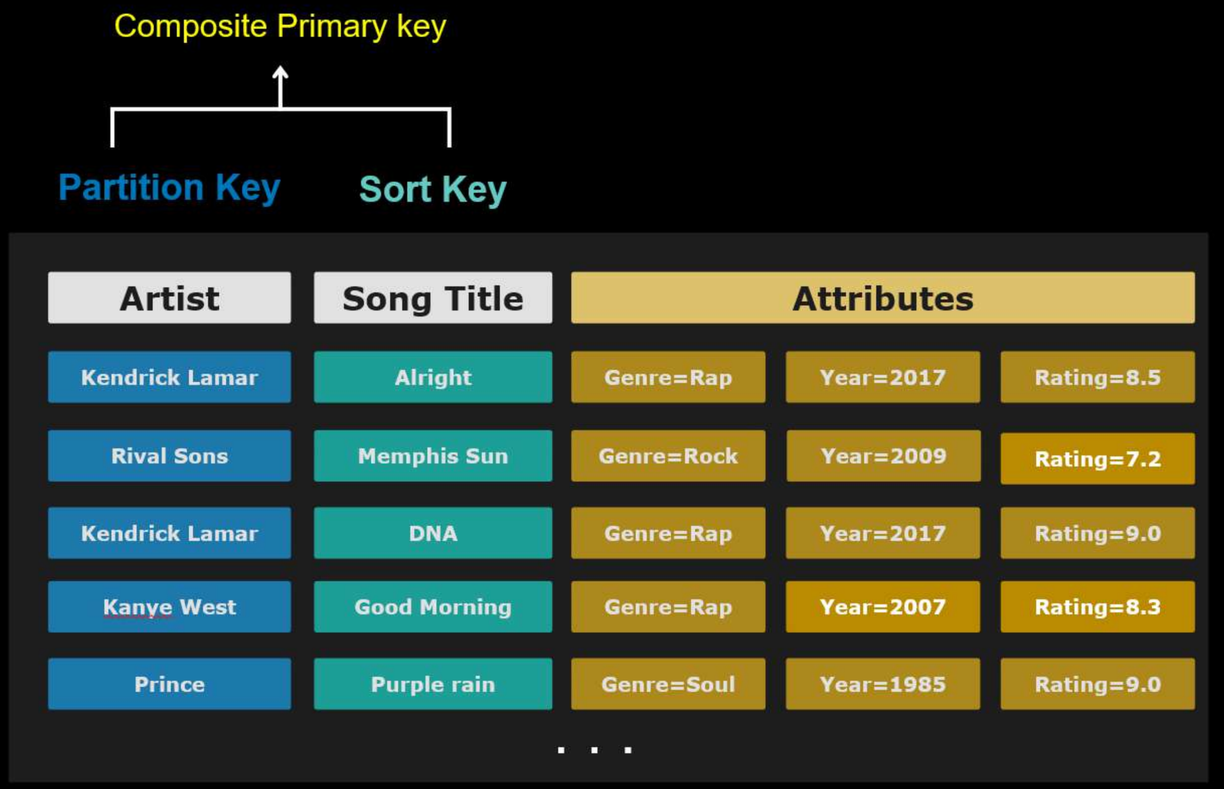
DynamoDB Core Components

- Table
  - a collection of related items or records
- Item
  - represents a record
  - can have one or more attributes
- Attribute
  - fundamental data element of an item
- Primary keys
  - uniquely identifies each item in a table
  - can be Partition key or Partition key and Sort key (optional)
- Partition key (Required)
  - used in a hash function to identify an item's location in a partition



DynamoDB Core Components

- Table
  - a collection of related items or records
- Item
  - represents a record
  - can have one or more attributes
- Attribute
  - fundamental data element of an item
- Primary keys
  - uniquely identifies each item in a table
  - can be Partition key or Partition key and Sort key (optional)
- Partition key (Required)
  - used in a hash function to identify an item's location in a partition
- Sort key (Optional)
  - gives additional flexibility when querying data
  - can be used to sort the order of items that share the same partition key



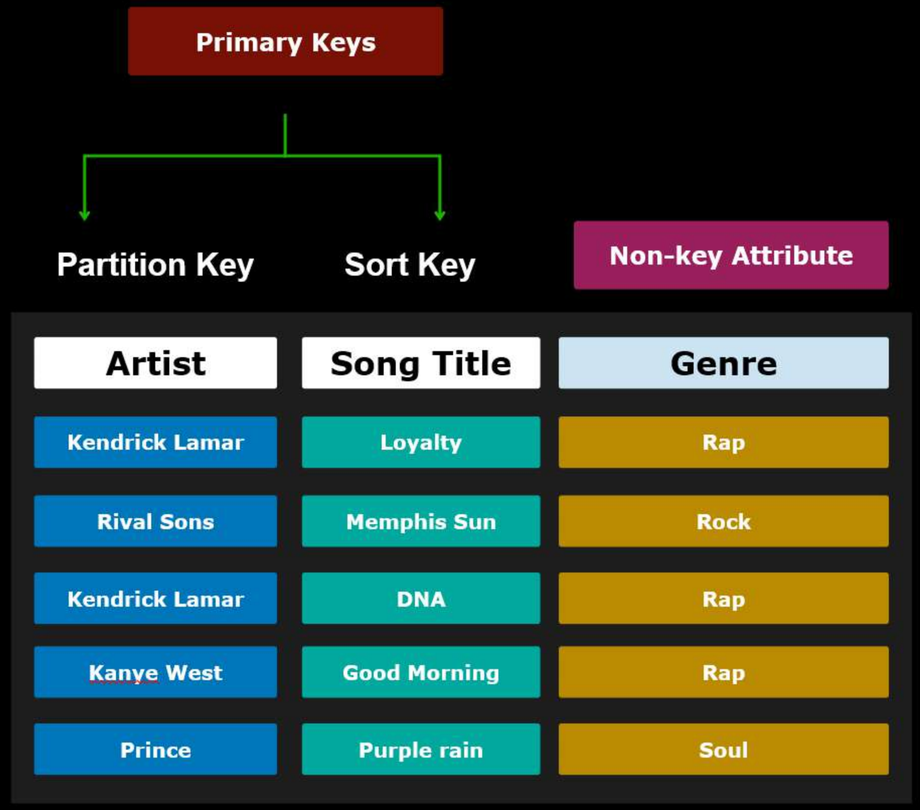
Music

Performance Considerations for Scans

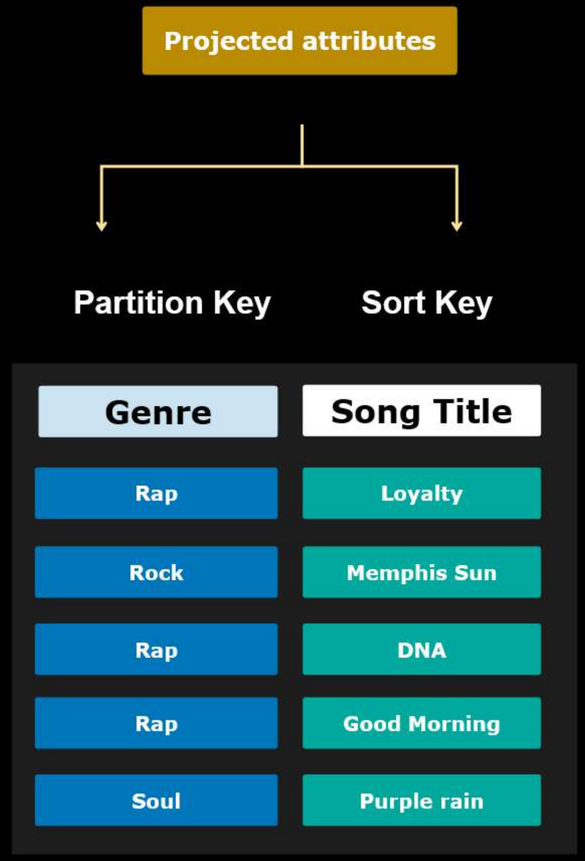
- Scan operations are slow and less efficient than other operations in DynamoDB
- As your table grows, the Scan operations becomes slower
- Every iteration uses Read Capacity Unit
- If possible, avoid using a Scan operation on a large table

Secondary Indexes

- A data structure that contains a **subset of attributes from a table**, along with an **alternate key** to support Query operations.



Base Table



Secondary Index

Attribute Projections

ONLY KEYS - only the **Partition Key** and **Sort key** (***if there's any***)

INCLUDE - attributes **described in ONLY KEYS + other non-key attributes**

ALL - **all attributes** of the base table are projected

Global Secondary Index

Global Secondary Index

Search can span all items in a base table, across all partitions

- You can create a GSI **anytime** (during or after the creation of a table)

- A table  can have up to **20 Global Secondary Indexes** (default limit)

Primary keys (Base table + GSI) are always projected

(Base Table)

Partition Key | Sort Key | Attributes

Artist	Song Title	Genre	Rating
Kendrick Lamar	Loyalty	Rap	8.5
Rival Sons	Memphis Sun	Rock	7.3
Kendrick Lamar	DNA	Rap	9.0
Kanye West	Good Morning	Rap	8.2
Prince	Purple rain	Soul	8.8

(Global Secondary Index)

Partition Key | Attributes | Projected Attributes

Genre	Song Title	Artist
Rap	Loyalty	Kendrick Lamar
Rock	Memphis Sun	Rival Sons
Rap	DNA	Kendrick Lamar
Rap	Good Morning	Kanye West
Soul	Purple rain	Prince

Sample Query Get all songs by Kendrick Lamar:

Artist	Song Title	Genre	Rating
Kendrick Lamar	DNA	Rap	8.5
Kendrick Lamar	Loyalty	Rap	9.0

Sample Query Get all rap songs:

No Rating attribute since it's **not included** in the **projected attribute**

Genre	Song Title	Artist
Rap	Loyalty	Kendrick Lamar
Rap	DNA	Kendrick Lamar
Rap	Good Morning	Kanye West

(Base Table)

Partition Key | Sort Key | Attributes

Artist	Song Title	Genre	Rating
Kendrick Lamar	Loyalty	Rap	8.5
Rival Sons	Memphis Sun	Rock	7.3
Kendrick Lamar	DNA	Rap	9.0
Kanye West	Good Morning	Rap	8.2
Prince	Purple rain	Soul	8.8

(Global Secondary Index)

Partition Key | Attributes

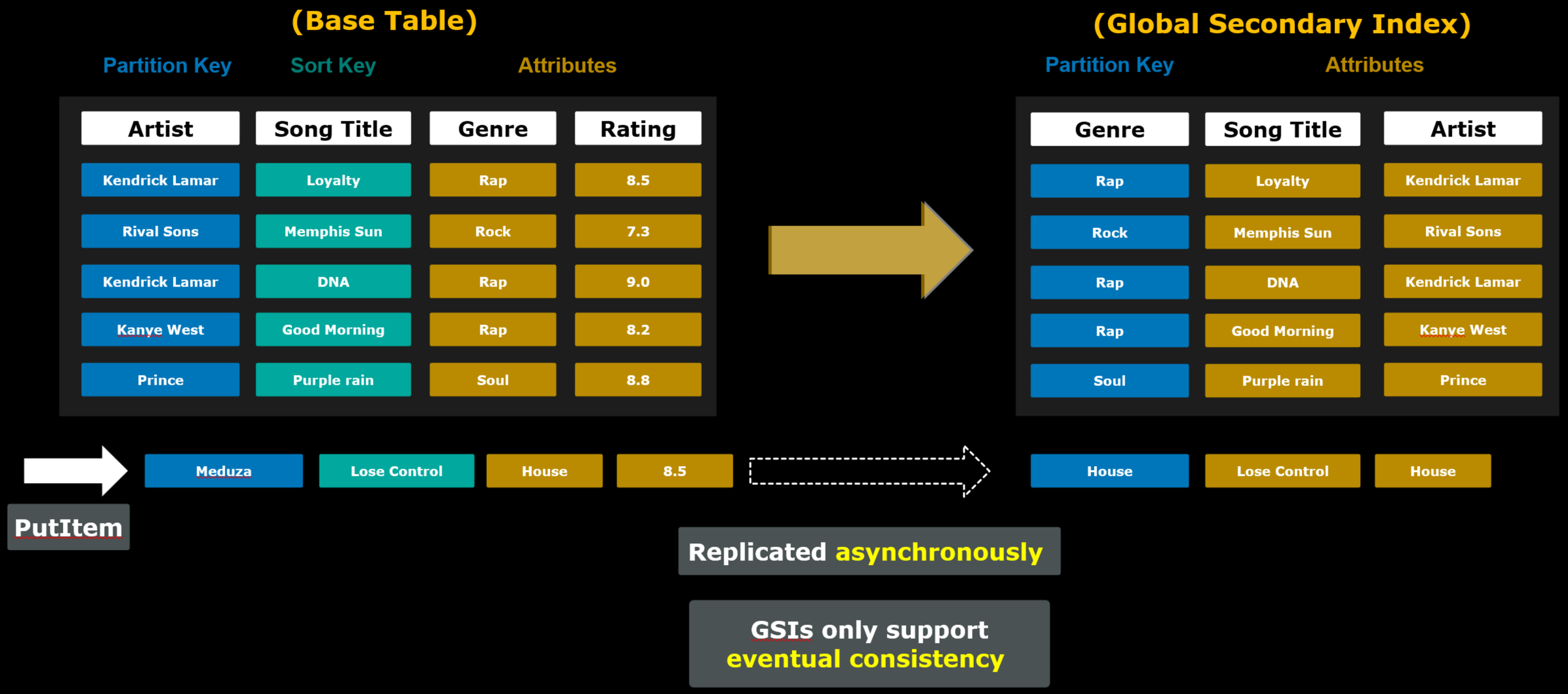
Genre	Song Title	Artist
Rap	Loyalty	Kendrick Lamar
Rock	Memphis Sun	Rival Sons
Rap	DNA	Kendrick Lamar
Rap	Good Morning	Kanye West
Soul	Purple rain	Prince

Provisioned RCU: 10

Provisioned WCU: 10

Provisioned RCU: 10

Provisioned WCU: 10



(Base Table)

Partition Key | Sort Key | Attributes

Artist	Song Title	Genre	Rating
Kendrick Lamar	Loyalty	Rap	8.5
Rival Sons	Memphis Sun	Rock	7.3
Kendrick Lamar	DNA	Rap	9.0
Kanye West	Good Morning	Rap	8.2
Prince	Purple rain	Soul	8.8

(Global Secondary Index)

Partition Key | Attributes

Genre	Song Title	Artist
Rap	Loyalty	Kendrick Lamar
Rock	Memphis Sun	Rival Sons
Rap	DNA	Kendrick Lamar
Rap	Good Morning	Kanye West
Soul	Purple rain	Prince

1 UPDATE = 2 Writes

(Base Table)

Partition Key | Sort Key | Attributes

Artist	Song Title	Genre	Rating
Kendrick Lamar	Loyalty	Rap	8.5
Rival Sons	Memphis Sun	Rock	7.3
Kendrick Lamar	DNA	Rap	9.0
Kanye West	Good Morning	Rap	8.2
Prince	Purple rain	Soul	8.8

(Global Secondary Index)

Partition Key | Attributes

Genre	Song Title	Artist
Rap	Loyalty	Kendrick Lamar
Rock	Memphis Sun	Rival Sons
Rap	DNA	Kendrick Lamar
Rap	Good Morning	Kanye West
Soul	Purple rain	Prince

Provisioned WCU: 10

Allocate

Provisioned WCU: 10

OR

Provisioned WCU: 20

Global Secondary Index

Partition Key	Sort Key	Attributes	
Director	Title	Rating	Release Date
Christopher Nolan	The Dark Knight	9.0	2008-07-14
Martin Scorsese	The Irishman	7.8	2019-11-27
Christopher Nolan	Inception	8.8	2010-07-08

(Base Table: Movies)

Possible Partition keys: Director, Title, Rating, Release Date

Possible Sort keys: Director, Title, Rating, Release Date

Local Secondary Index

Partition Key	Sort Key	Attributes	
Director	Title	Rating	Release date
Christopher Nolan	The Dark Knight	9.0	2008-07-14
Martin Scorsese	The Irishman	7.8	2019-11-27
Christopher Nolan	Inception	8.8	2010-08-2010

(Base Table: Movies)

Partition key is fixed

Possible Sort keys: Rating, Release date

Local Secondary Index

Partition Key	Sort Key	Attributes	
Director	Title	Rating	Release Date
Christopher Nolan	The Dark Knight	9.0	2008-07-14
Martin Scorsese	The Irishman	7.8	2019-11-27
Christopher Nolan	Inception	8.8	2010-07-08

(Base Table: Movies)

Partition Key	Sort Key	Attributes
Director	Release Date	Title
Christopher Nolan	2008-07-14	The Dark Knight
Martin Scorsese	2019-11-27	The Irishman
Christopher Nolan	2010-07-08	Inception

(Local Secondary Index: ReleaseDateLSI)

Sample Query: "*Fetch all movies of Christopher Nolan, ordered by its release date*"

You can only create an LSI during the creation of a table

Partition Key	Sort Key	Attributes
Director	Release date	Title
Christopher Nolan	2008-07-14	The Dark Knight
Christopher Nolan	2010-07-08	Inception

Local Secondary Index

Partition Key	Sort Key	Attributes	
Director	Title	Rating	Release Date
Christopher Nolan	The Dark Knight	9.0	2008-07-14
Martin Scorsese	The Irishman	7.8	2019-11-27
Christopher Nolan	Inception	8.8	2010-07-08

(Base Table: Movies)

Partition Key	Sort Key	Attributes
Director	Release Date	Title
Christopher Nolan	2008-07-14	The Dark Knight
Martin Scorsese	2019-11-27	The Irishman
Christopher Nolan	2010-07-08	Inception

(Local Secondary Index: ReleaseDateLSI)

Provisioned RCU: 10

Provisioned WCU: 10

LSI shares throughput with its base table

Unlike GSI, LSI supports eventually and strongly consistent reads

Local Secondary Index

Partition Key	Sort Key	Attributes	
Director	Title	Rating	Release Date
Christopher Nolan	The Dark Knight	9.0	2008-07-14
Martin Scorsese	The Irishman	7.8	2019-11-27
Christopher Nolan	Inception	8.8	2010-07-08

(Base Table: Movies)

Partition Key	Sort Key	Attributes
Director	Release date	Title
Christopher Nolan	2008-07-14	The Dark Knight
Martin Scorsese	2019-11-27	The Irishman
Christopher Nolan	2010-07-08	Inception

(Local Secondary Index: ReleaseDateLSI)

The **Rating** attribute will show in the result even if it's not projected into the **LSI** as it will be fetched from the **base table**

Query

```
response = client.query(
    TableName='Movies',
    IndexName='ReleaseDateLSI',
    ProjectionExpression = "Title, Rating",
    KeyConditionExpression = "Director = :v",
    ExpressionAttributeValues = {
        ":v": {"S": "Christopher Nolan"}
    })
```

Result

```
{'Count': 2,
 'Items': [{'Rating': {'N': '9'}, 'Title': {'S': 'The Dark Knight'}},
           {'Rating': {'N': '8.8'}, 'Title': {'S': 'Inception'}}],
 'ResponseMetadata': {'HTTPHeaders': {'connection': 'keep-alive',
                                      'content-length': '140',
                                      'content-type': 'application/x-amz-
                                      'date': 'Mon, 05 Jul 2021 14:19:09
                                      'server': 'Server',
                                      'x-amz-crc32': '3043785351',
                                      'x-amzn-requestid': '3KGUP2DA0I5RIQ
                      'HTTPStatusCode': 200,
                      'RequestId': '3KGUP2DA0I5RIQOP23URI6ITGJVV4KQNSO5AE
                      'RetryAttempts': 0},
 'ScannedCount': 2}
```

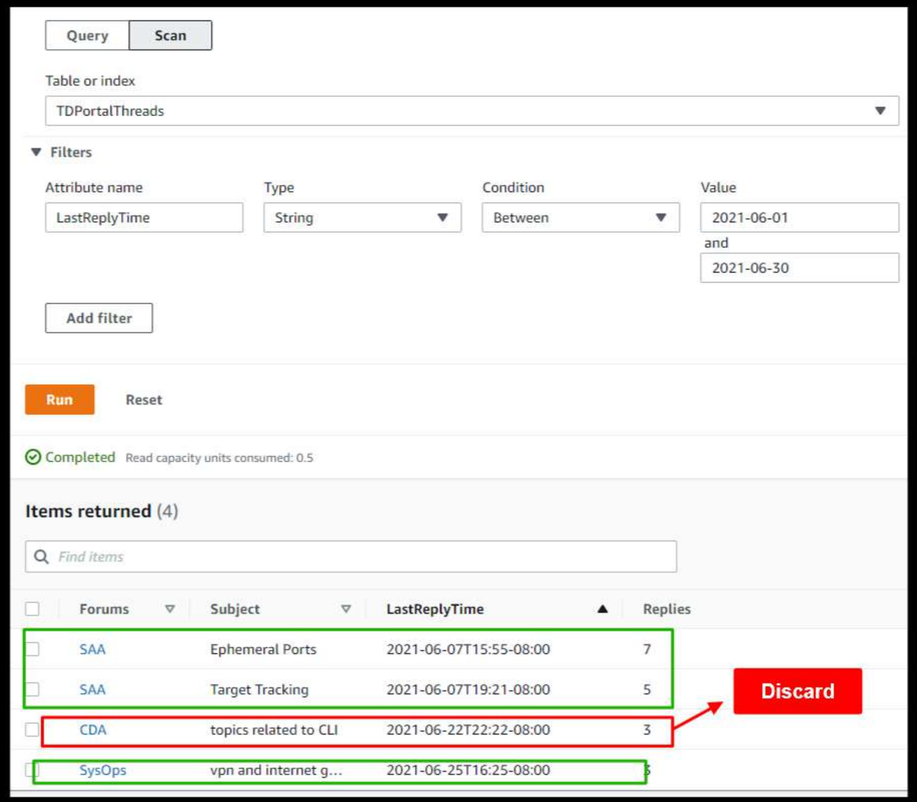
Local Secondary Index

"Get all questions that has been posted in SAA and SysOps during the month of June"

Partition Key	Sort Key	Attributes	
Forums	**Subject**	**LastReplyTime**	**Replies**
SAA	Ephemeral Ports	2021-06-07T15:55-08:00	7
SAA	Elastic IP question	2021-05-028T08:33-08:00	2
CDA	AWS SAM service?	2021-07-02T21:04-08:00	2
SAA	Target Tracking	2021-06-07T19:21-08:00	5
SysOps	vpn and internet gateway	2021-06-25T16:25-08:00	3
CDA	topics related to CLI	2021-06-22T22:22-08:00	3

TD Portal Threads

Scan



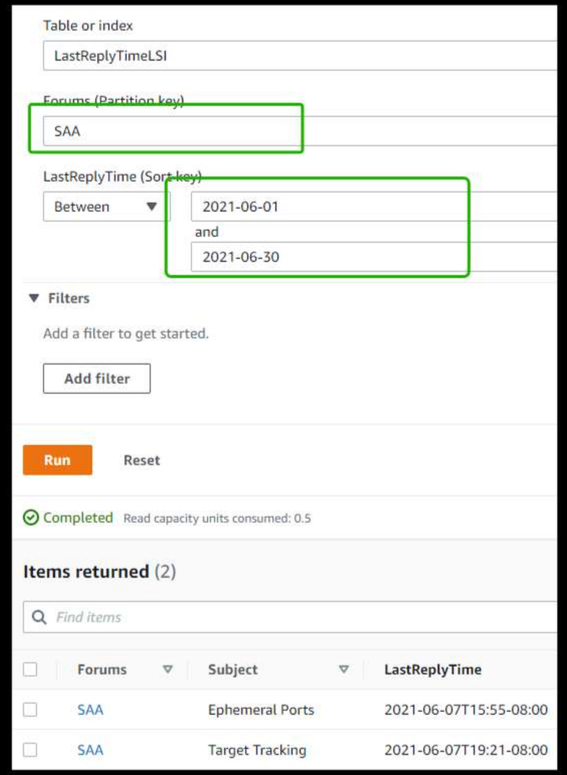
Local Secondary Index

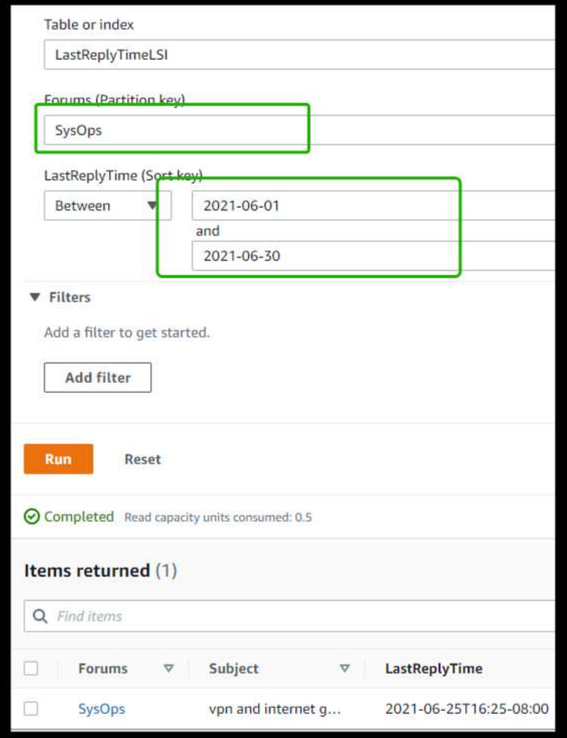
"Get all questions that has been posted in SAA and SysOps during the month of June"

2 Queries

Partition Key	Sort Key	Attributes
Forums	LastReplyTime	Subject
SAA	2021-06-07T15:55-08:00	Ephemeral Ports
SAA	2021-05-028T08:33-08:00	Elastic IP question
CDA	2021-07-02T21:04-08:00	AWS SAM service?
SAA	2021-06-07T19:21-08:00	Target Tracking
SysOps	2021-06-25T16:25-08:00	vpn and internet gateway
CDA	2021-06-22T22:22-08:00	topics related to CLI

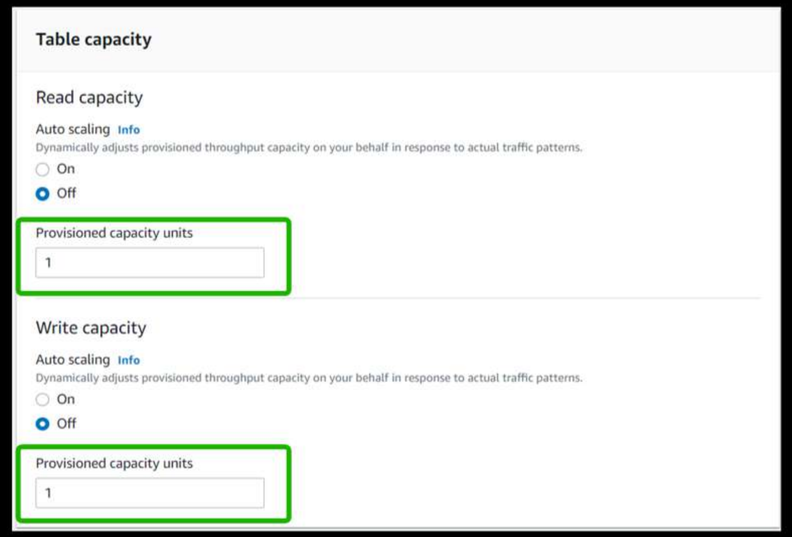
LastReplyTimeLSI





Provisioned Capacity Mode

- You set a **fixed** amount of read and write throughput



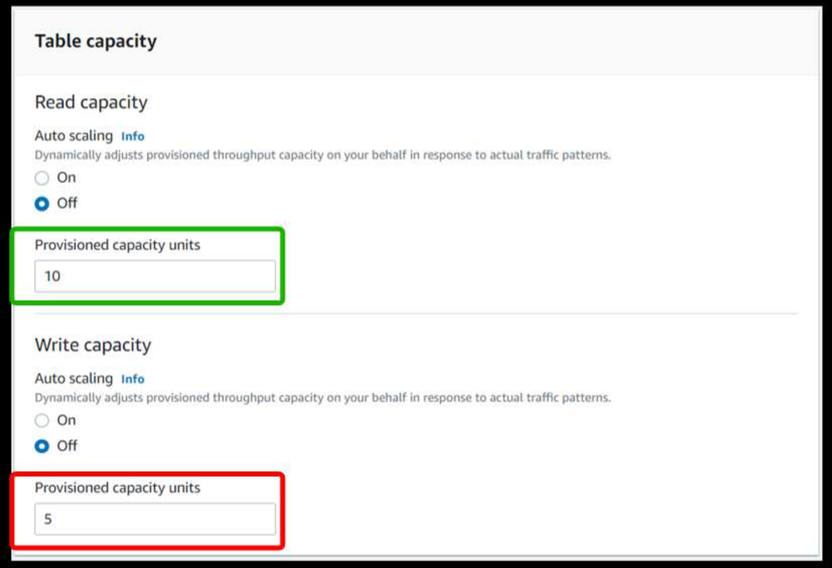
On-demand Capacity Mode

Provisioned Capacity Mode

- You set a **fixed** amount of read and write throughput
- Throughput is specified in terms of **capacity units:**

Read Capacity Unit (RCU)

Write Capacity Unit (WCU)



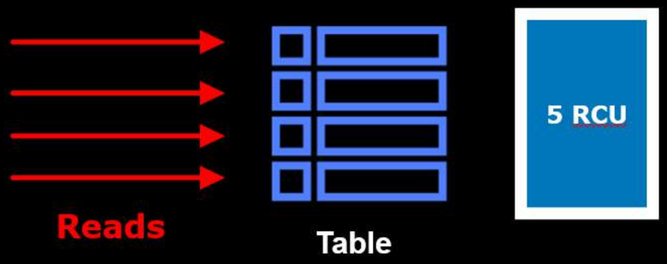
On-demand Capacity Mode

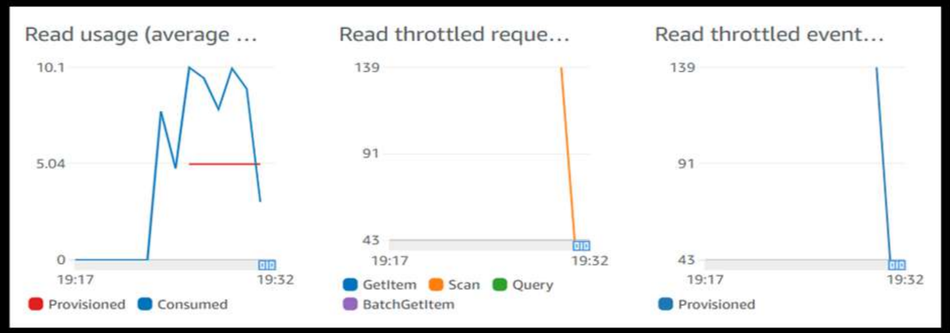
Provisioned Capacity Mode

- You set a **fixed** amount of read and write throughput
- Throughput is specified in terms of **capacity units:**

Read Capacity Unit (RCU)

Write Capacity Unit (WCU)





On-demand Capacity Mode

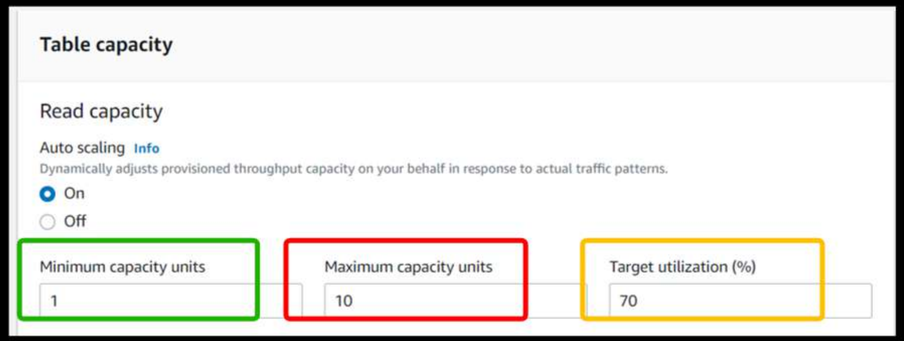
Provisioned Capacity Mode

- You set a **fixed** amount of read and write throughput
- Throughput is specified in terms of **capacity units:**

Read Capacity Unit (RCU)

Write Capacity Unit (WCU)

- **You can enable DynamoDB auto-scaling**
  - Uses AWS Application Auto Scaling Service to scale capacity



On-demand Capacity Mode

Provisioned Capacity Mode

- You set a **fixed** amount of read and write throughput
- Throughput is specified in terms of **capacity units:**

Read Capacity Unit (RCU)

Write Capacity Unit (WCU)

- **You can enable DynamoDB auto-scaling**
  - Uses  AWS Application Auto Scaling Service to scale capacity
  - Capacity is allocated dynamically up to the maximum provisioned capacity
  - If the capacity demand > max provisioned capacity, requests will still be throttled
- **Use Cases:**
  - web applications with predictable or consistent traffic patterns
  - great for controlling costs
  - you pay a fixed amount based on the provisioned capacity (no auto scaling)
  - you always pay for the minimum provisioned capacity (with auto scaling)

On-demand Capacity Mode

Provisioned Capacity Mode

- You set a **fixed** amount of read and write throughput
- Throughput is specified in terms of **capacity units:**

Read Capacity Unit (RCU) | Write Capacity Unit (WCU)

- **You can enable DynamoDB auto-scaling**
  - Uses  AWS Application Auto Scaling Service to scale capacity
  - Capacity is allocated dynamically up to the maximum provisioned capacity
  - If the capacity demand > max provisioned capacity, requests will still be throttled
- **Use Cases:**
  - web applications with predictable or consistent traffic patterns
  - great for controlling costs
  - you pay a fixed amount based on the provisioned capacity (no auto scaling)
  - you always pay for the minimum provisioned capacity (with auto scaling)

On-demand Capacity Mode

- Go wild and **scale out as needed**
- No capacity planning required
- Throughput is measured in **request units:**

Read Request Unit (RRU) | Write Request Unit (WRU)

Provisioned Capacity Mode

- You set a **fixed** amount of read and write throughput
- Throughput is specified in terms of **capacity units:**

Read Capacity Unit (RCU) | Write Capacity Unit (WCU)

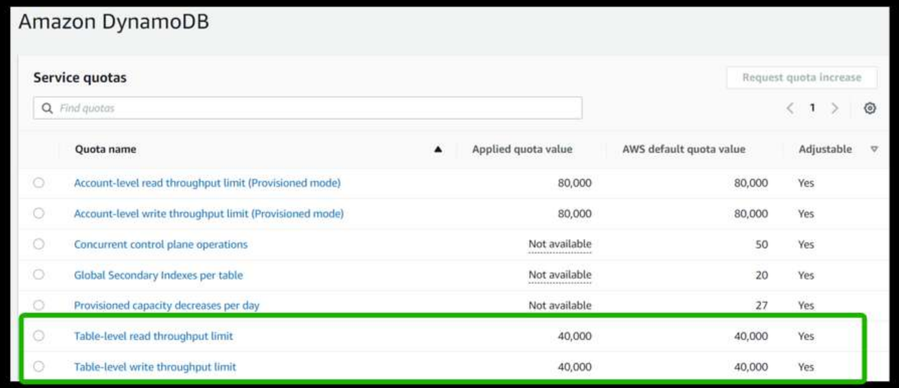
- **You can enable DynamoDB auto-scaling**
  - Uses AWS Application Auto Scaling Service to scale capacity
  - Capacity is allocated dynamically up to the maximum provisioned capacity
  - If the capacity demand > max provisioned capacity, requests will still be throttled
- **Use Cases:**
  - web applications with predictable or consistent traffic patterns
  - great for controlling costs
  - you pay a fixed amount based on the provisioned capacity (no auto scaling)
  - you always pay for the minimum provisioned capacity (with auto scaling)

On-demand Capacity Mode

- Go wild and **scale out as needed**
- No capacity planning required
- Throughput is measured in **request units:**

Read Request Unit (RRU) | Write Request Unit (WRU)

- DynamoDB will add capacity as your application demands **until it reaches your table's limit**



Provisioned Capacity Mode

- You set a **fixed** amount of read and write throughput
- Throughput is specified in terms of **capacity units:**

Read Capacity Unit (RCU) | Write Capacity Unit (WCU)

- **You can enable DynamoDB auto-scaling**
  - Uses  AWS Application Auto Scaling Service to scale capacity
  - Capacity is allocated dynamically up to the maximum provisioned capacity
  - If the capacity demand > max provisioned capacity, requests will still be throttled
- **Use Cases:**
  - web applications with predictable or consistent traffic patterns
  - great for controlling costs
  - you pay a fixed amount based on the provisioned capacity (no auto scaling)
  - you always pay for the minimum provisioned capacity (with auto scaling)

On-demand Capacity Mode

- Go wild and **scale out as needed**
- No capacity planning required
- Throughput is measured in **request units:**

Read Request Unit (RRU) | Write Request Unit (WRU)

- DynamoDB will add capacity as your application demands **until it reaches your table's limit**
- **Use Cases:**
  - if you want a true pay-per-use billing model
  - suitable for applications with unknown workloads and unpredictable traffic patterns
  - no cost control
  - can be more expensive than the provisioned capacity mode

Read Capacity Unit (RCU)

1. 1 RCU = 1 strongly consistent read request per second for an item up to 4KB = 2 eventually consistent read request per second

2. 2 RCUs = 1 transactional read of up to 4KB item

3. For items > 4KB, follow the formula:

RCU = (total item size)/ 4KB, rounded up

Write Capacity Unit (WCU)

1. 1 WCU = 1 standrad write request per second for an item up to 1KB

2. 2 WCUs = 1 transactional write for an item up to 1KB

3. For items > 1KB, follow the formula:

WCU = (total item size)/ 1KB, rounded up

Given

Average item size: 5 KB

Item reads each second: 10 eventually consistent reads/s

Item writes each second: 5 standard writes/s

Find the RCU and WCU?

Solution

RCU = 5 KB/4KB = 1.25 ≈ **2**

WCU = 5 KB * 5 = **25 WCUs**

RCU(10 eventually consistent reads/s) = (2/2) = 1 * 10 = **10 RCUs**

RCU (10 strongly consistent reads/s) = 2 * 10 RCUs = **20 RCUs**

Given

Average item size: 8 KB

Item reads each second: 20 strongly consistent reads/s

Item writes each second: 2 transactional writes/s

Find the RCU and WCU?

Solution

1 RCU = Read 4KB/s

Hence, it takes 2 RCUs to read 8 KB/s

RCU = 2 * 20 = 40 RCUs

1 WCU = Write 1KB/s (Standard)

Hence, it takes 8 WCU to read 8 KB/s

1 transactional write = 2*standard write

1 transactional write = 2*8 = 16 WCUs

WCU = 16 * 2 = 32 WCUs

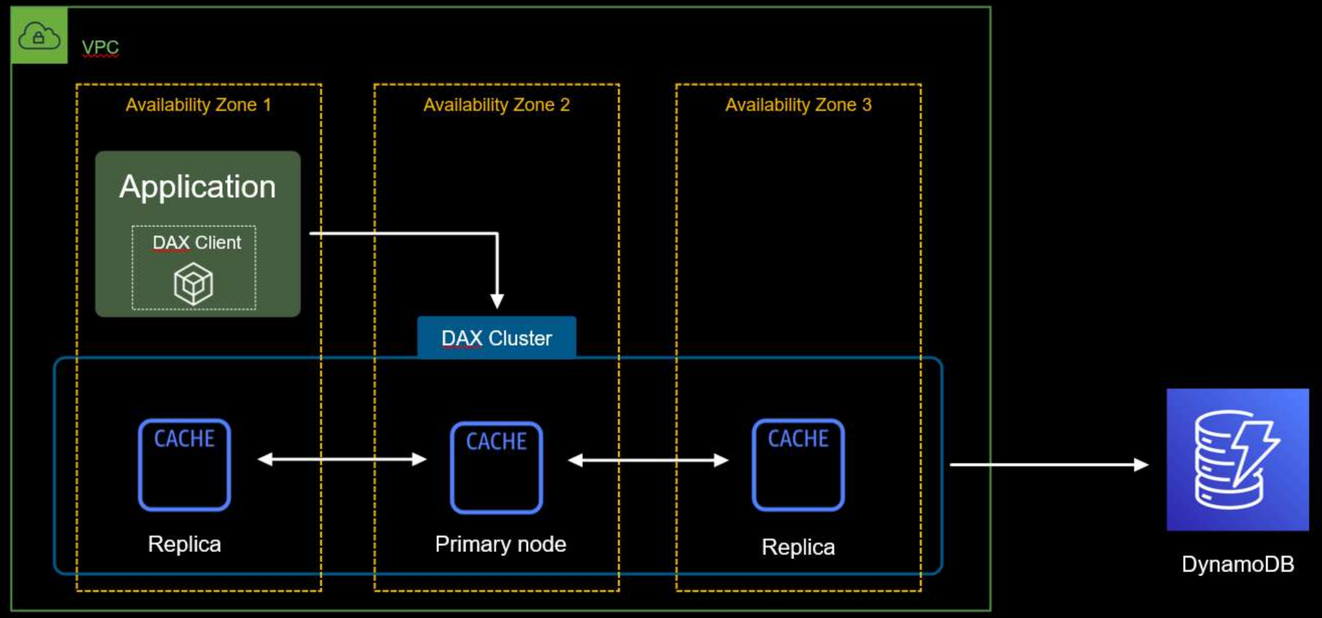
DynamoDB Accelerator (DAX)

DynamoDB Accelerator (DAX)

- A highly available, in-memory cache that is purposely built for DynamoDB
- Can deliver response times in microseconds for millions of requests per second

Overview of a DAX cluster

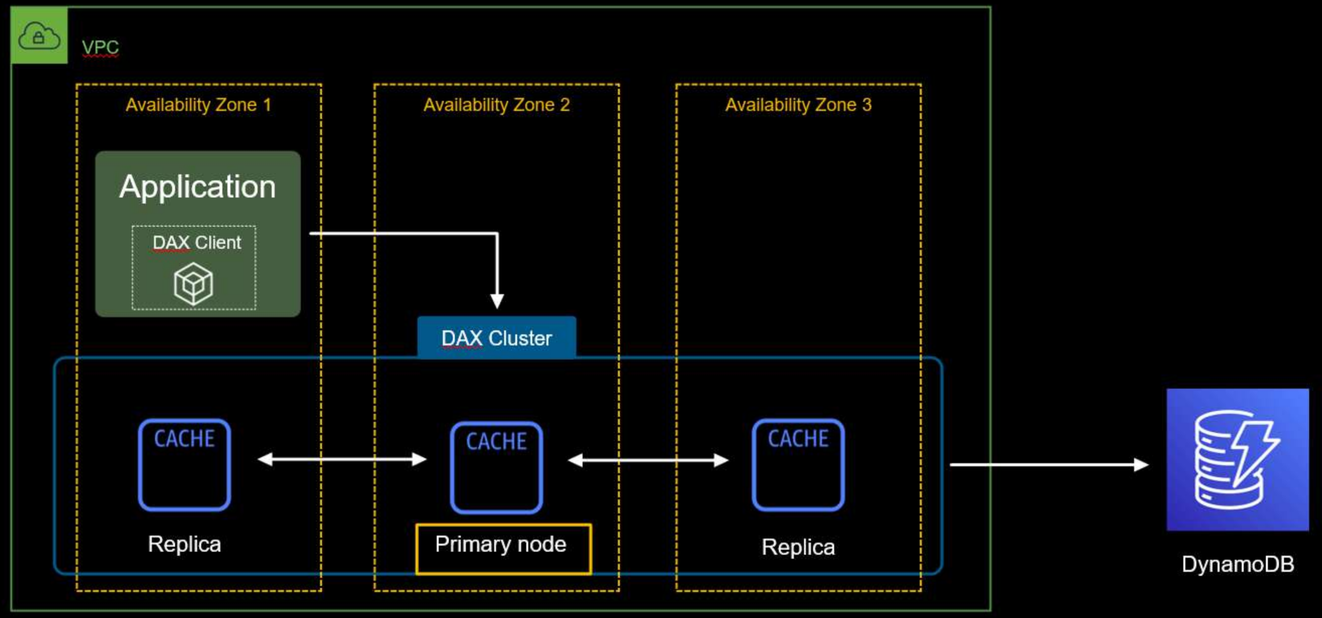
DynamoDB Accelerator (DAX)



- A DAX cluster is made up of nodes
- A node contains an instance of the DAX software and a single replica of the cached data
- When you create a DAX cluster, you get to configure its size by selecting the node type and the number of nodes

DynamoDB Accelerator (DAX)

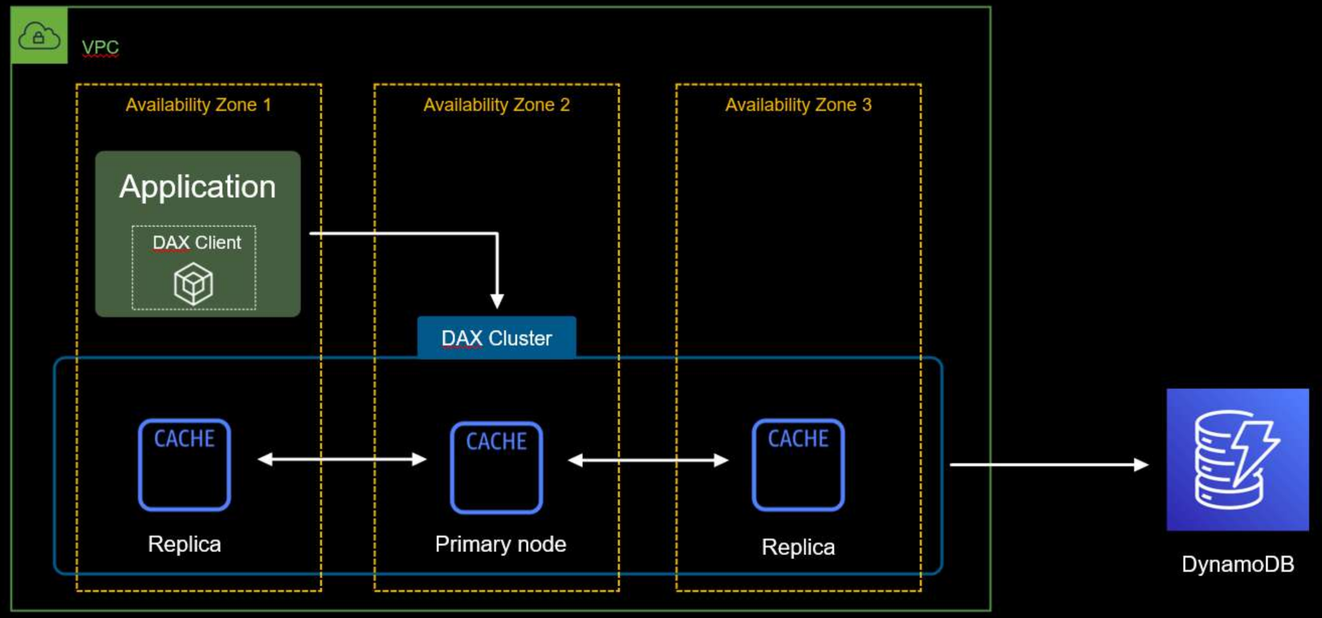
Overview of a DAX cluster



- For high availability, it is recommended that at least three nodes be distributed across AZs
- DAX automatically assigns which node is the primary node
- The read-throughput depends on the cluster size

DynamoDB Accelerator (DAX)

Overview of a DAX cluster



- A DAX cluster can be scaled vertically and horizontally
- DAX runs in a VPC, it is not accessible over the public internet

DynamoDB Accelerator (DAX)

```python
import boto3
import amazondax

#DynamoDB Client
dynamodb_client = boto3.client('dynamodb')

#Dax Client
dax_client = amazondax.AmazonDaxClient(endpoint_url='dax://tdojo.pgt7y1.dax-clusters.us-east-2.amazonaws.com')


response = dax_client.query(
    TableName='Movies',
    IndexName='ReleaseDateLSI',
    ProjectionExpression = "Title, Rating",
    KeyConditionExpression = "Director = :v",
    ExpressionAttributeValues = {
        ":v": {"S": "Christopher Nolan"}
    })
```

DAX Client

- After creating the DAX cluster, point your existing DynamoDB API calls to the DAX cluster endpoint

DynamoDB Accelerator (DAX)

Use DAX...

- for any read-intensive applications that require response times in microseconds
- to mitigate the effects of hot partitions by caching items that are read more frequently than others
- to cache large data sets that are repeatedly read by an application to save RCU

DynamoDB Accelerator (DAX)

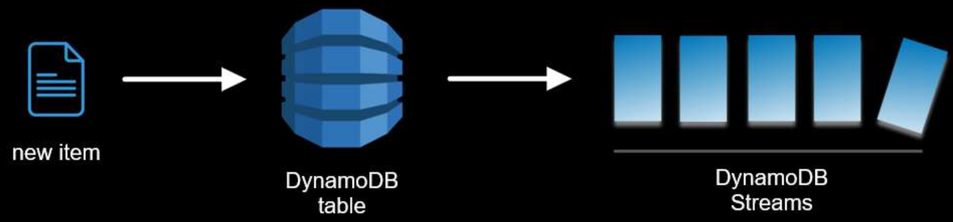
DAX is not ideal...

- for applications that require strongly-consistent reads. DAX only serves eventually consistent reads
- for write-heavy applications
- if you have an existing caching layer for DynamoDB and you wish to use your own caching logic for your application

DynamoDB Streams

DynamoDB Streams

- A DynamoDB feature that captures item-level changes that occurs in a DynamoDB table
- Write operations (PUT, UPDATE, DELETE) are detected as events



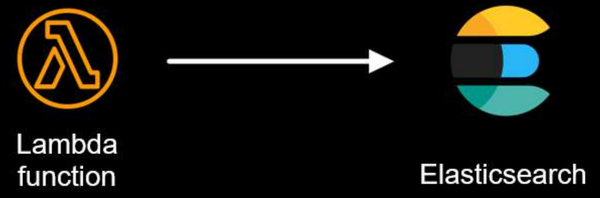
- UPDATE or PUT operations that do not change any data in an item are ignored

DynamoDB Streams

Sample Use Case:

Data replication for text searches

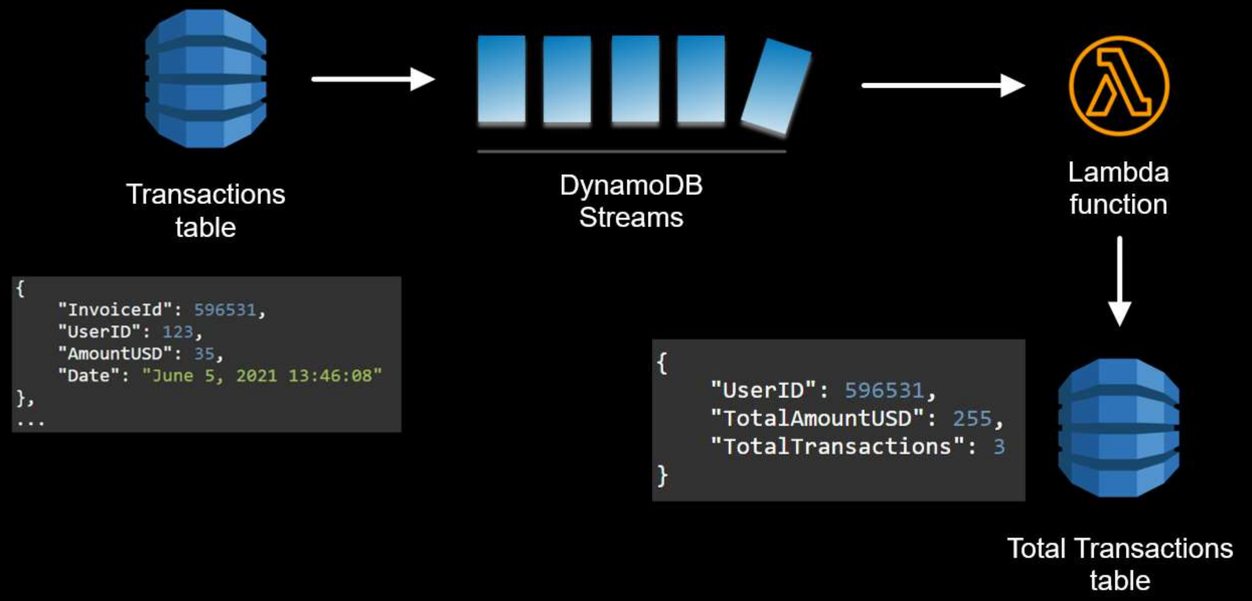
Process stream
records in batches



Sample Use Case:

Data aggregation

DynamoDB Streams



DynamoDB Streams

- a *StreamViewType* determines what kind of data you are sending to a stream

Four SteamViewTypes

1. KEYS_ONLY - only the key attributes (Partition Key & Sort key) of the modified item are written to the stream

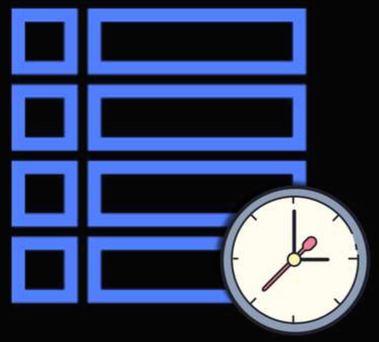
2. NEW_IMAGE - the modified version of the entire item is written to the stream

3. OLD_IMAGE - the entire item as it appeared prior to the update is written to the stream

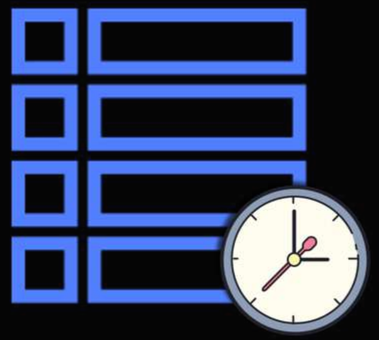
3. NEW_AND_OLD_IMAGES - both the old and modified version of the entire item is written to the stream

DynamoDB Streams

- Stream records older than 24 hours are removed
- DynamoDB writes stream records in near-real time without impacting the performance of the source table
- Stream records are ordered based on the sequence of item-level modifications
- Each record appears exactly once in the stream



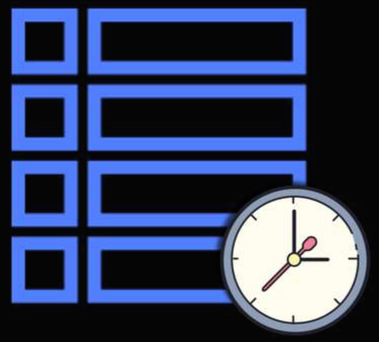
DynamoDB Time-To-Live(TTL)



DynamoDB TTL

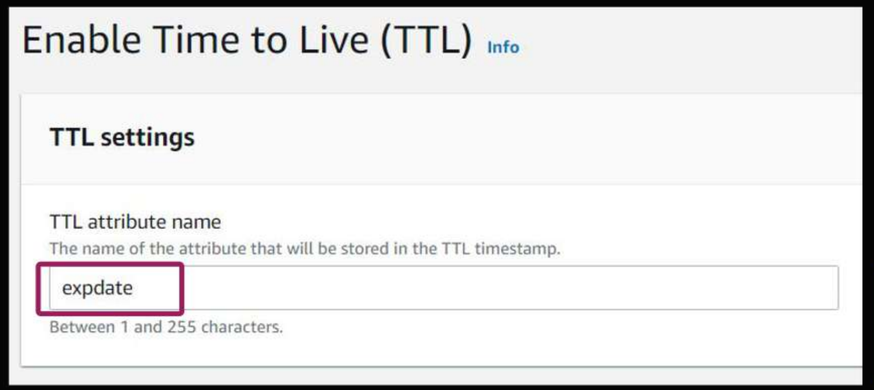
- TTL allows you to automatically expire an item based on a defined timestamp

Setting a TTL



DynamoDB TTL

1. Enable TTL and assign an TTL attribute name



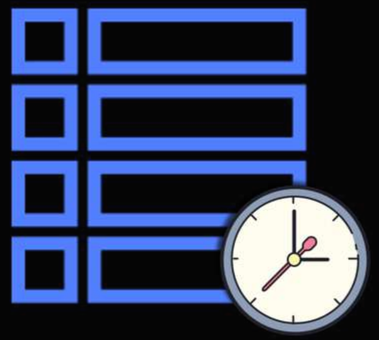
Only one TTL attribute per table

2. Use a Number data type with value in Epoch time format

TTL attribute name

Epoch time format = **Thursday, June 3, 2021 5:14:56 PM**

Data Type

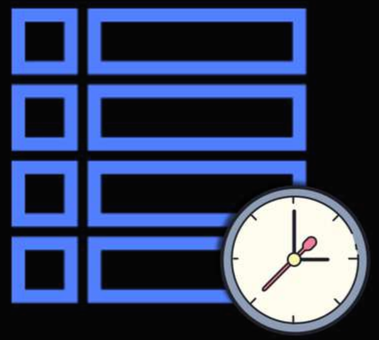


DynamoDB TTL

- Enabling TTL is free of charge
- Deleting of items does not consume write capacity units

Example TTL use cases

1. Cleaning up old records (*ex. session data*)
2. Reducing table size to save on storage costs
3. Expiring user subscription to services



DynamoDB TTL

- Items are not deleted immediately. Expired TTL can still appear in queries
- The exact duration within which an item gets deleted after expiration depends on the table size
- DynamoDB deletes expired items within 48 hours of expiration

DynamoDB Transactions

DynamoDB Transactions

- DynamoDB Transactions lets you perform ACID operations across one or more tables in an AWS region

ACID → assures the safety and integrity of a transaction

Atomicity
- no stale data

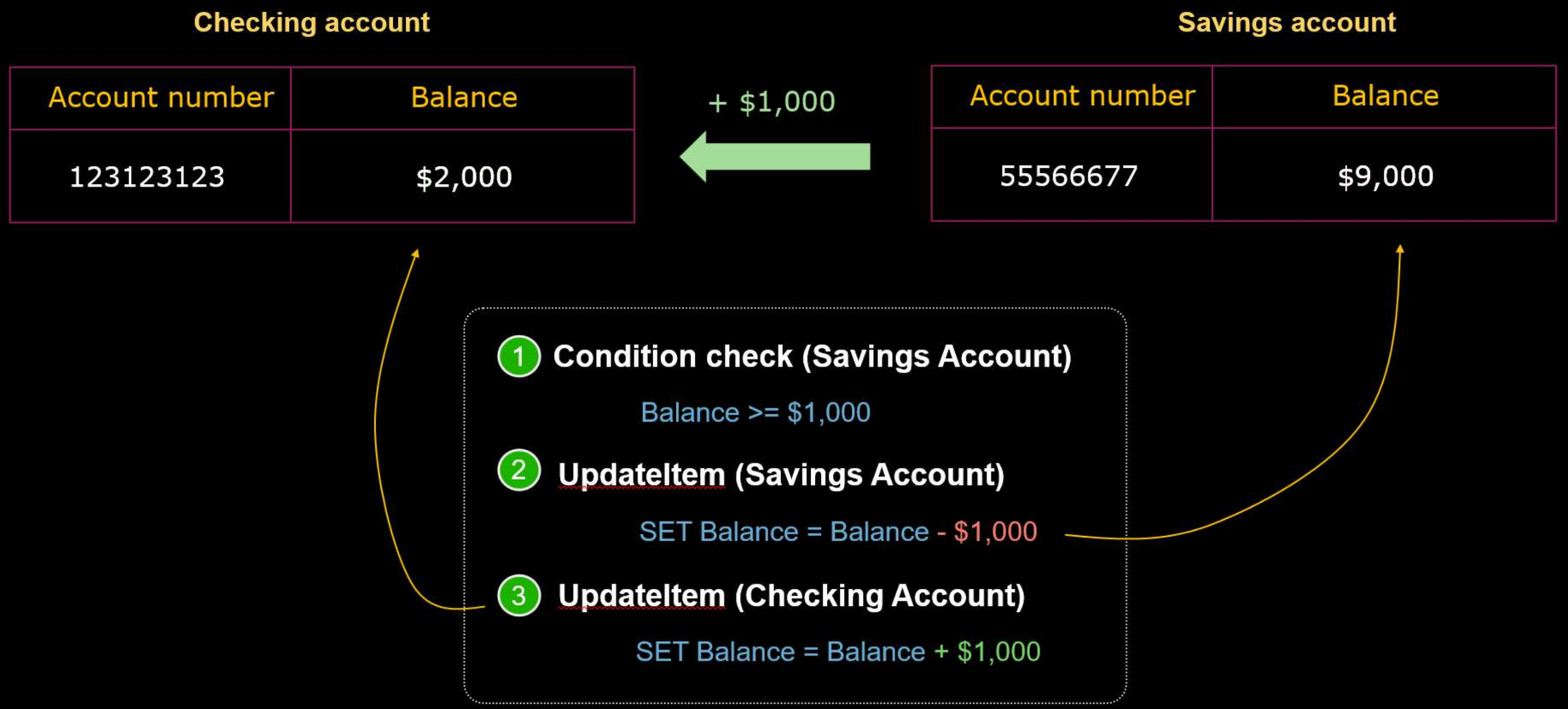
Consistency
- avoids data corruption

Isolation
- overall accuracy and reliability of a database

Durability

What is a transaction?

- a collection of queries that effectively acts as a single unit of work



What is a transaction?

- a collection of queries that effectively acts as a single unit of work

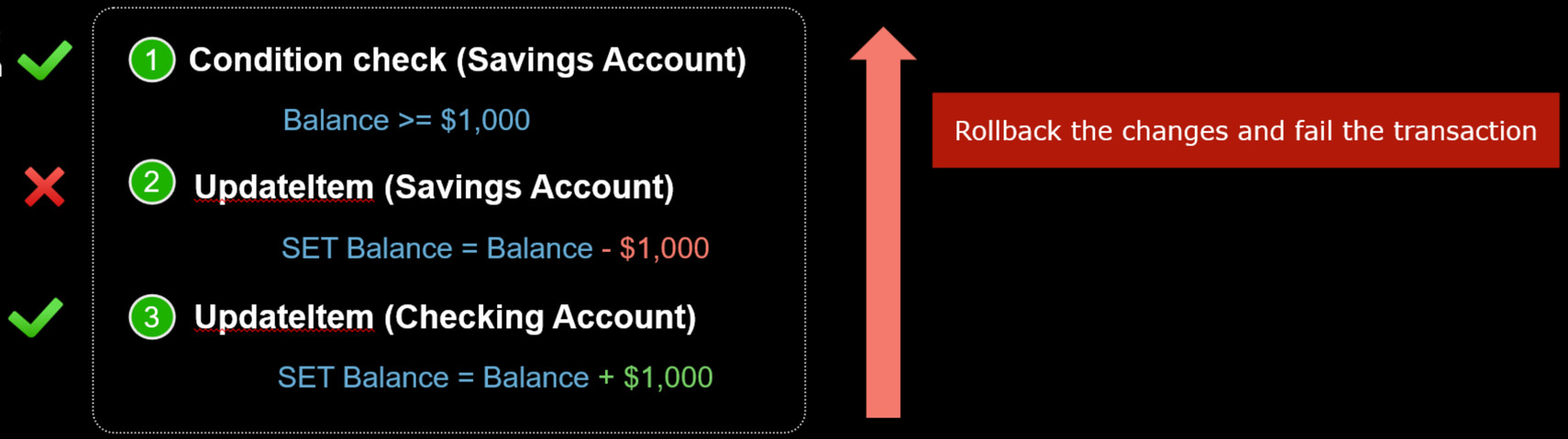
Checking account

Account number	Balance
123123123	$2,000

Savings account

Account number	Balance
55566677	$9,000

- Atomicity requires the transaction as a whole to be successfully executed or if a part of the transaction fails, then the entire transaction is invalidated



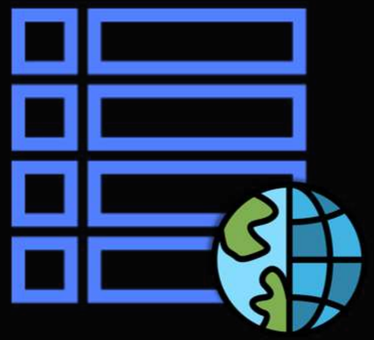
DynamoDB Transactions

Two transaction operations

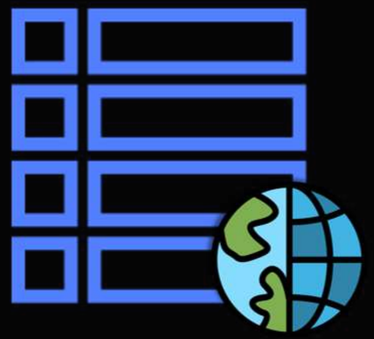
1. TransactWriteItems
   - group up to 25 write actions in a single all-or-nothing operation
   - you can add PutItem, UpdateItem, DeleteItem, and ConditionCheck operations
2. TransactGetItems
   - group up to 25 Get items and condition check (should not exceed 4MB)

DynamoDB Transactions

- Transactions are 2x more expensive than Standard reads and writes
- DynamoDB performs two underlying reads or writes of every item in the transaction

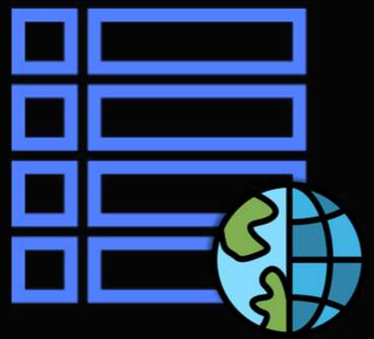


DynamoDB Global Tables



DynamoDB
Global Tables

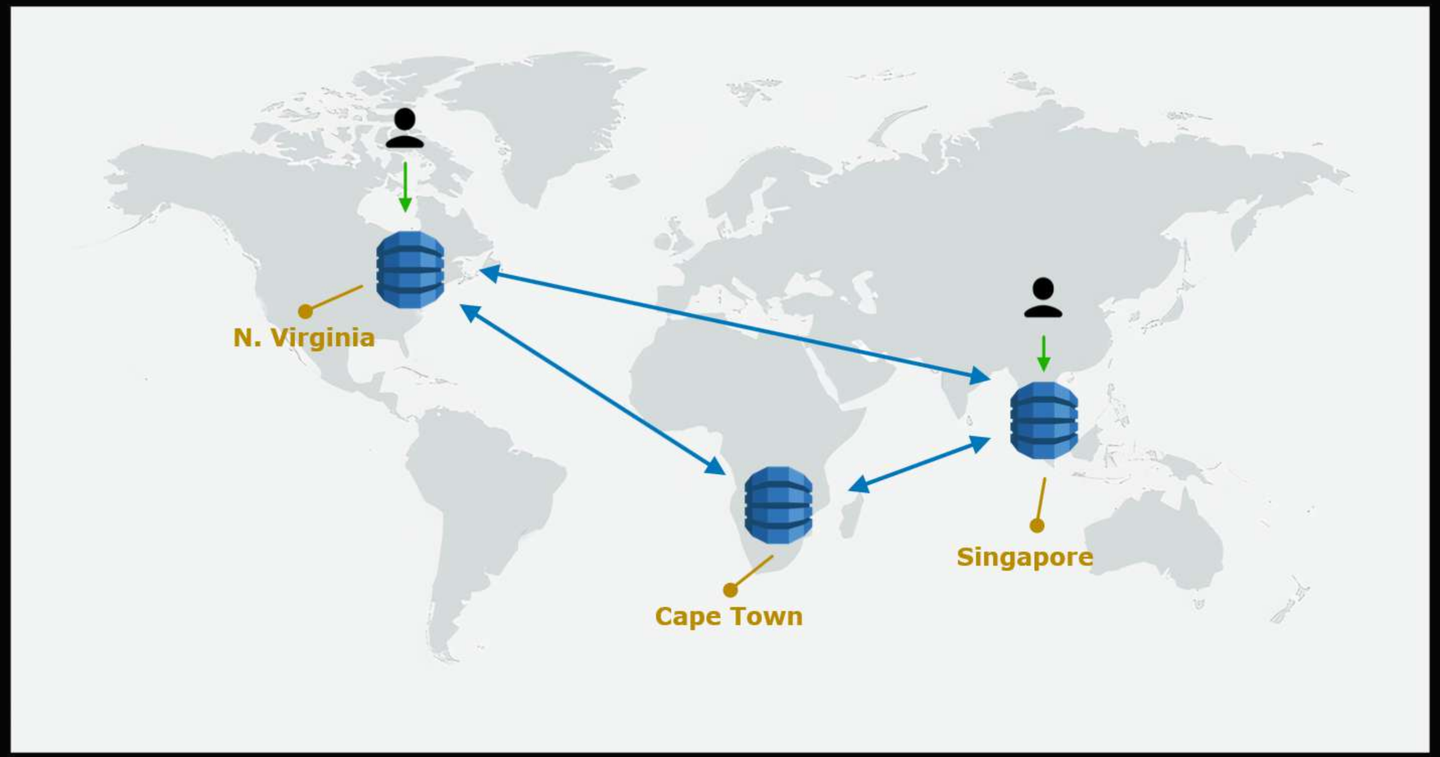
- DynamoDB Global Tables allows you to automatically replicate tables across AWS Regions



DynamoDB
Global Tables

- DynamoDB Global Tables allows you to automatically replicate tables across AWS Regions
- DynamoDB fully manages the synchronization of tables via DynamoDB Streams
- To use create a global table, enable DynamoDB Streams with the NEW_AND_OLD_IMAGES StreamViewType

How Global Tables Work?

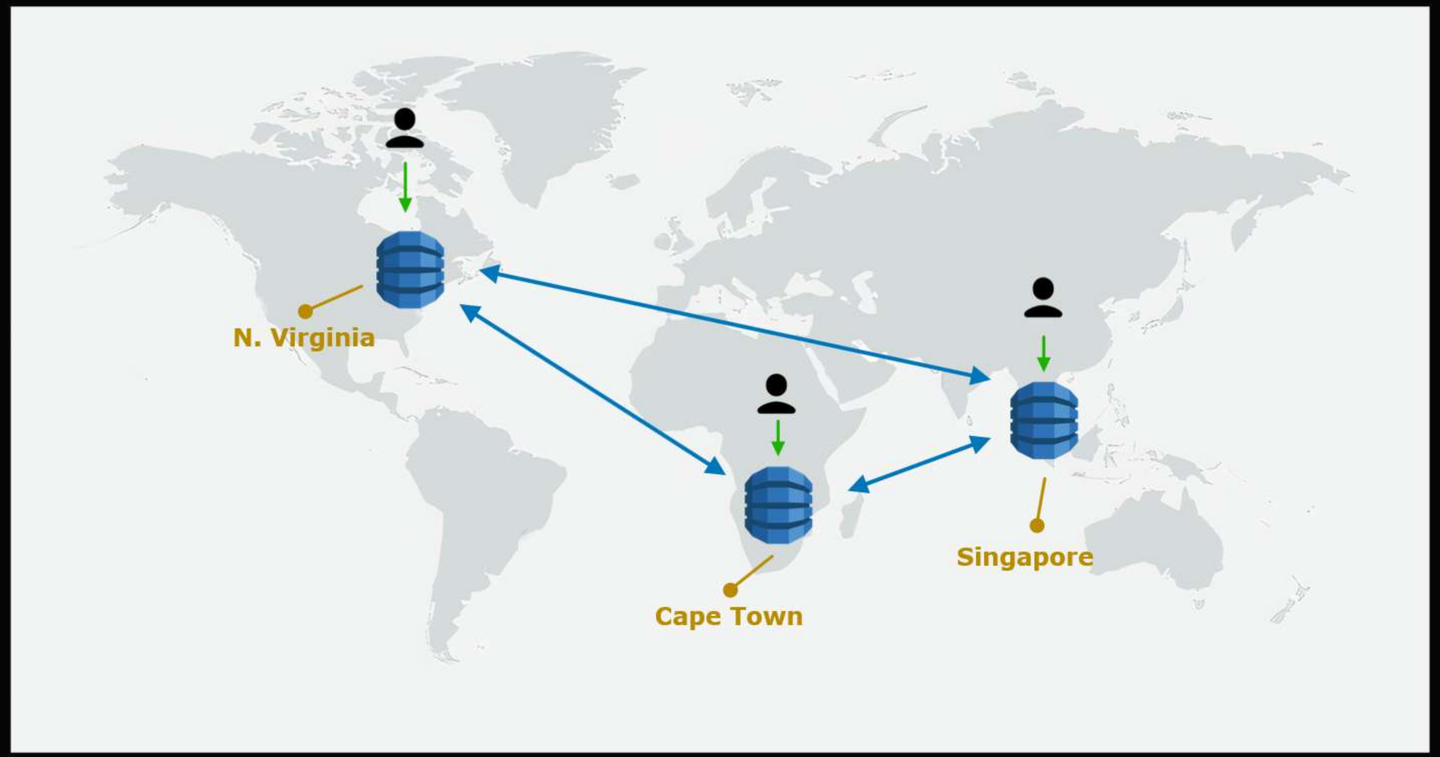


Subjects

Subject	Schedule	Instructor code
Math107	8AM - 10AM	521

- Change is propagated asynchronously across the tables

How Global Tables Work?



Subjects

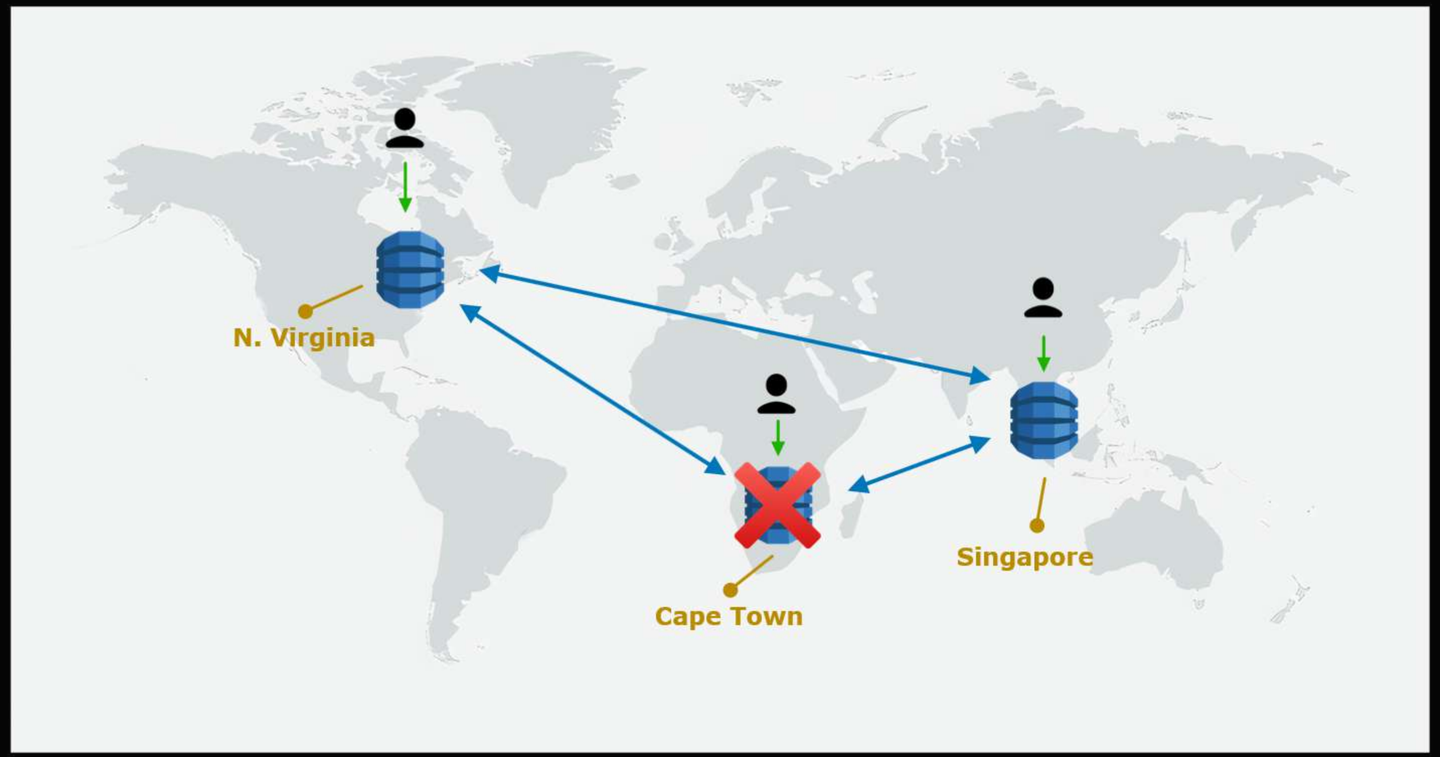
Subject	Schedule	Instructor code
Math107	10AM - 12PM	302

CASE 1: ✗ ✓

CASE 2: ✓ ✗

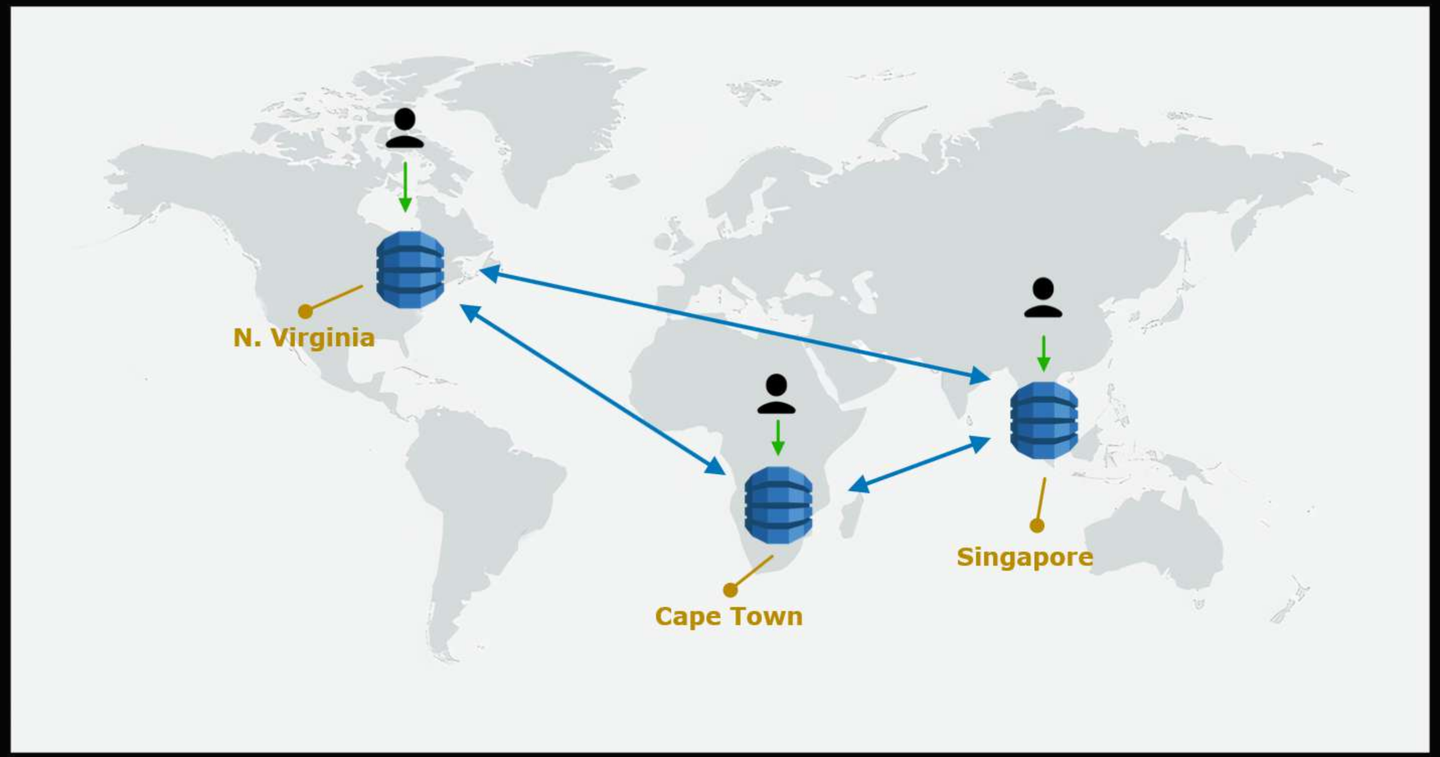
- Change is propagated asynchronously across the tables
- DynamoDB uses a last writer wins reconciliation to resolve conflicts between concurrent updates

Why use Global Tables?



- Protection from regional failures

Why use Global Tables?

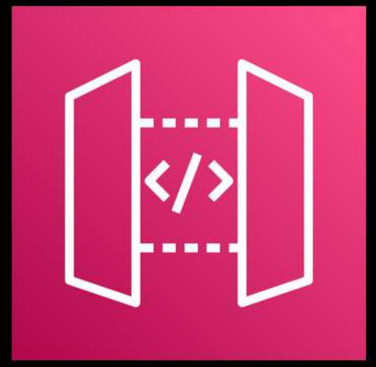


- Protection from regional failures
- Provides low-latency to applications with globally dispersed users

Your location is: PH

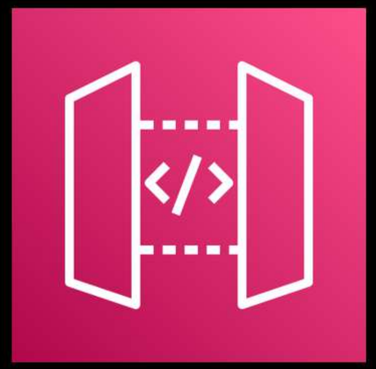
Latency median (ms)

Region code	Region description	Latency (ms)
us-east-1	US East (N. Virginia)	3002
us-west-1	US West (N. California)	3005
ca-central-1	Canada (Central)	3002
eu-west-1	EU (Ireland)	3010
eu-central-1	EU (Frankfurt)	3005
ap-south-1	Asia Pacific (Mumbai)	3008
ap-southeast-1	Asia Pacific (Singapore)	211

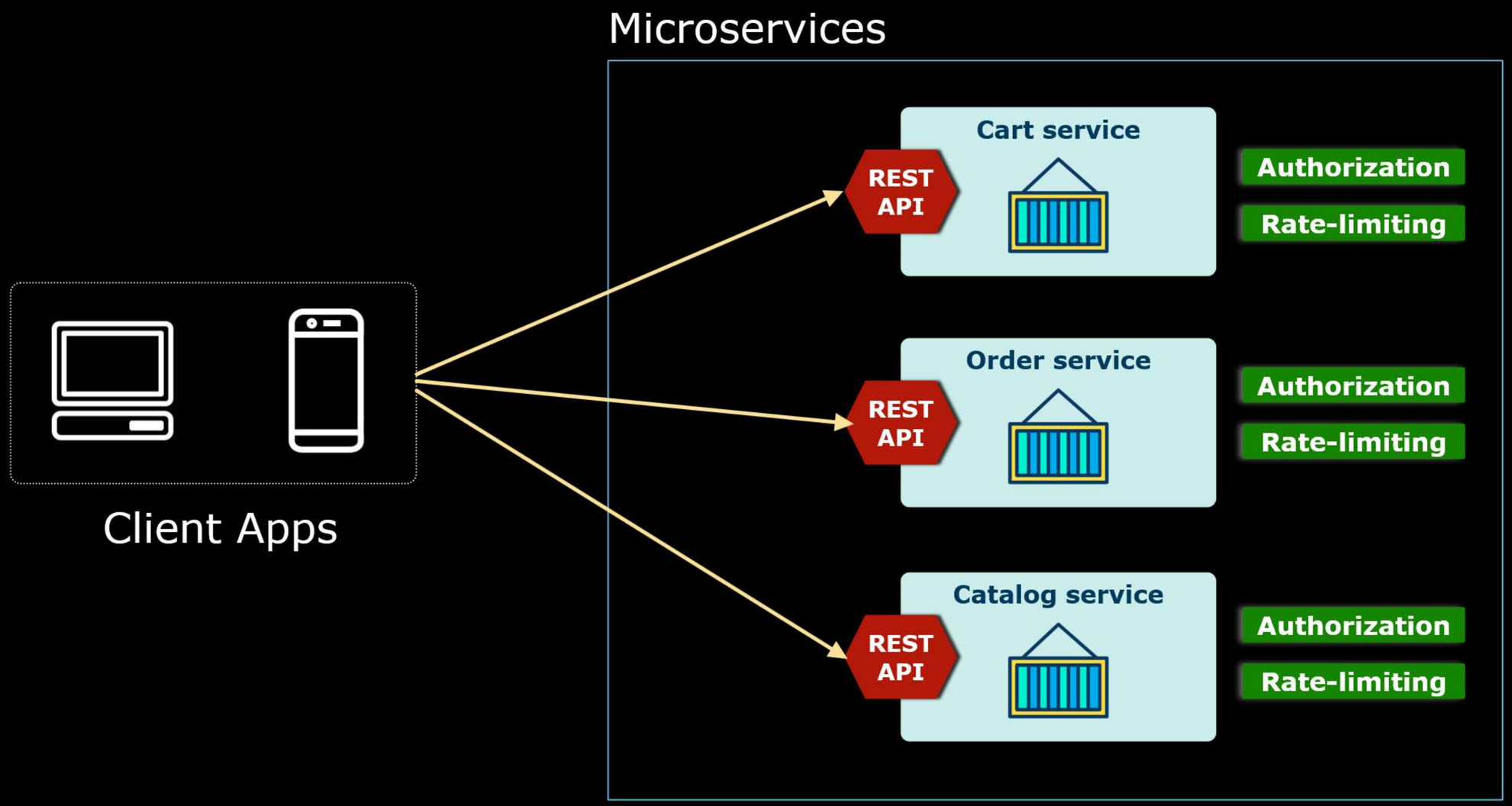


Amazon API Gateway

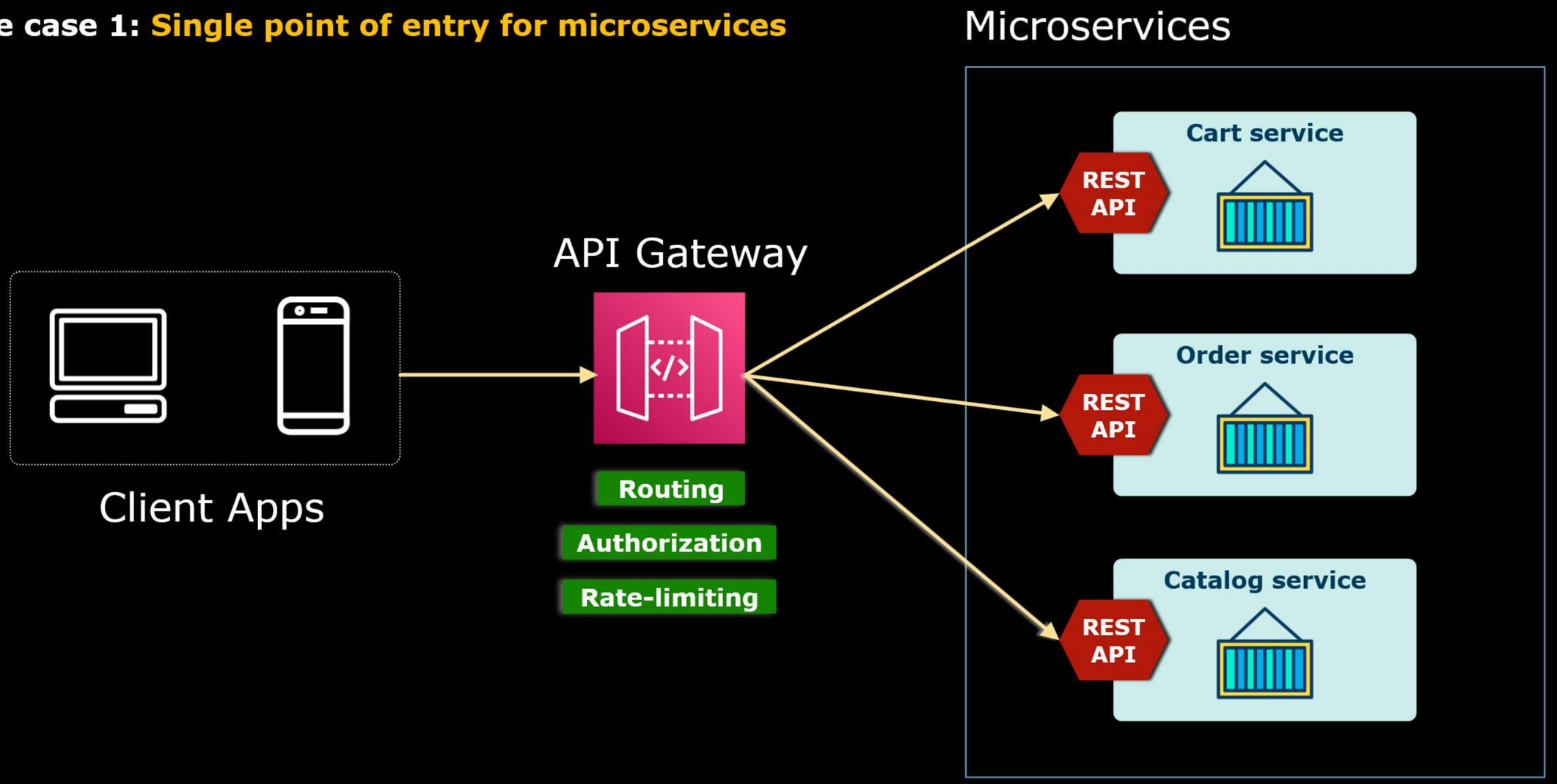
- API management service that lets you create, publish, maintain, monitor, and secure APIs at any scale



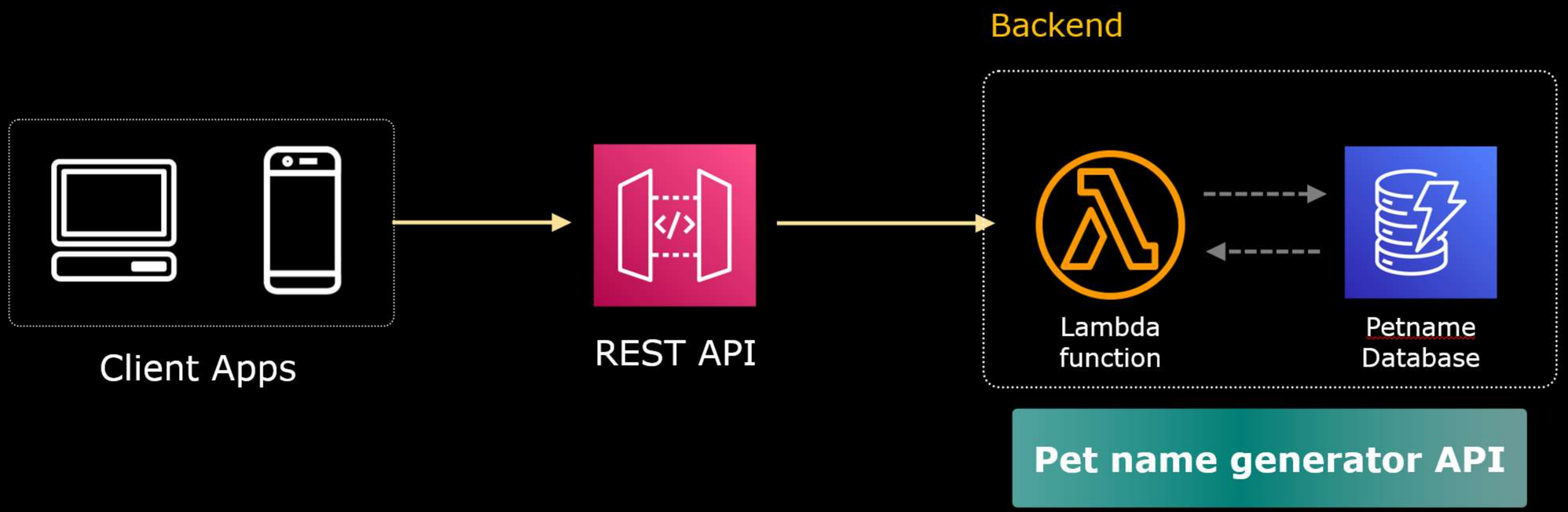
Amazon API Gateway



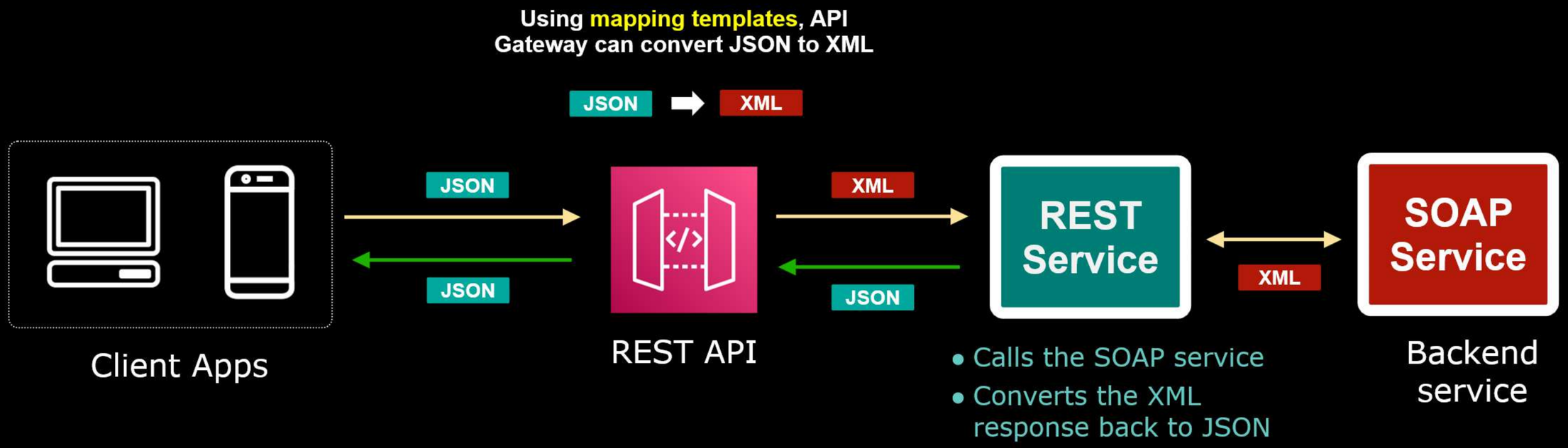
Use case 1: Single point of entry for microservices

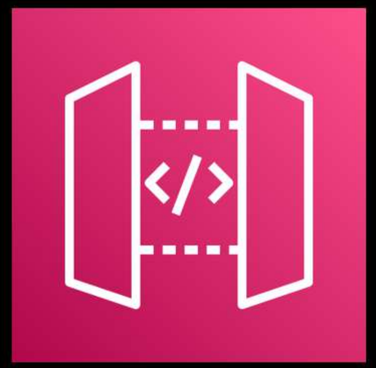


Use case 2: Widely used in serverless applications



Use case 3: Modernize legacy applications





Amazon API Gateway

- You can choose between a REST API, HTTP API, and a WebSocket API endpoint

1. **REST API**

   - gives you full access to API Gateway features like caching, creating API keys, and usage plans

   - integration with other AWS services

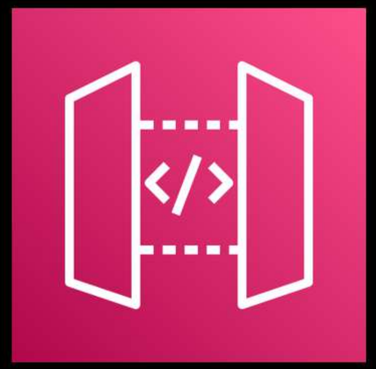
2. **HTTP API**

   - cheap and designed for low-latency applications

   - lacks other API Gateway features

3. **WebSocket API**

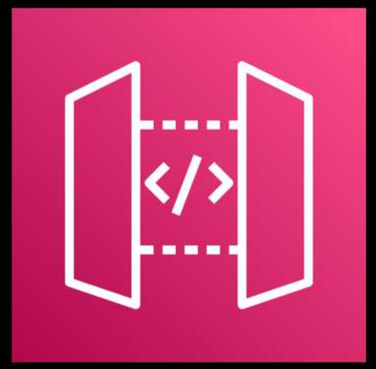
   - suitable for real-time applications such as chat applications



Amazon API Gateway

- Endpoint Types:

- API Gateway does not support unencrypted endpoints
- HTTPS for REST and HTTP API
- WSS for WebSocket API



Amazon API Gateway

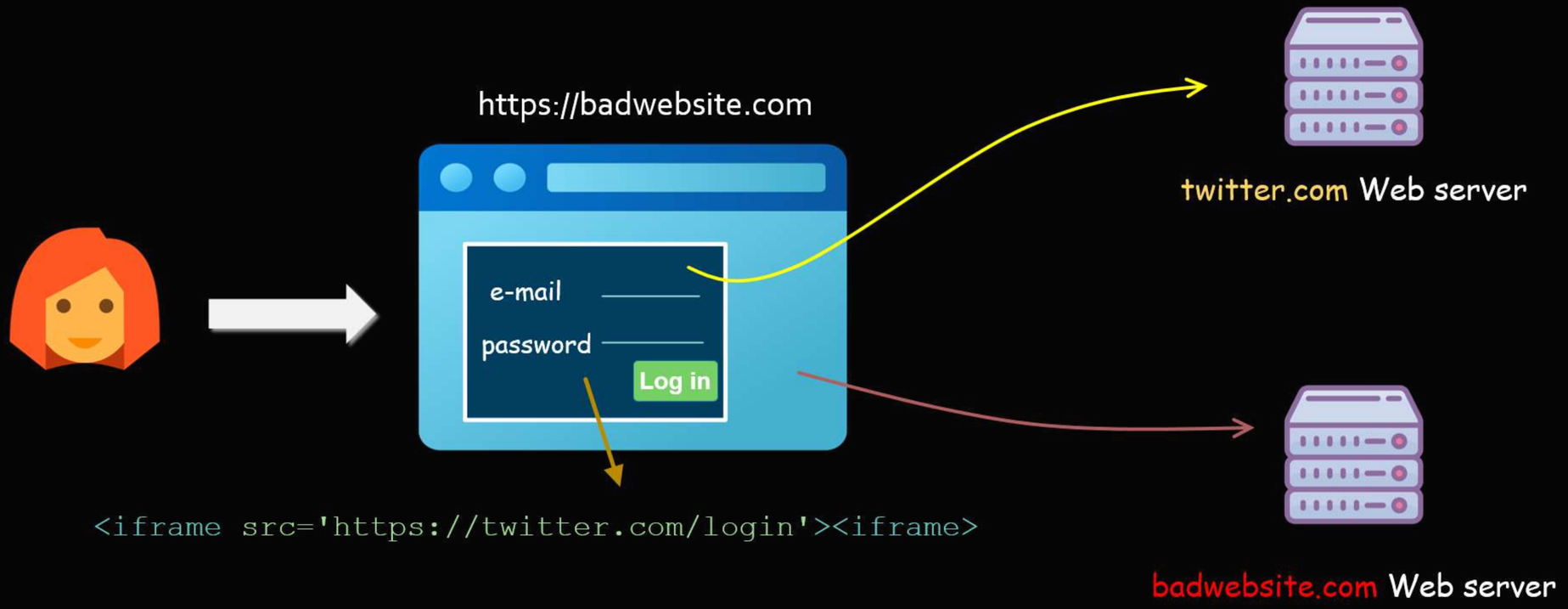
- Endpoint Types:
  1. **Edge-optimized**
     - requests are routed to the nearest CloudFront Point of Presence
     - best for geographically distributed clients
  2. **Regional**
     - requests are routed in the same region where the API is deployed
     - intended for clients in the same region
  3. **Private**
     - APIs can only be accessed from a VPC using a VPC endpoint

Cross-Origin Resource Sharing

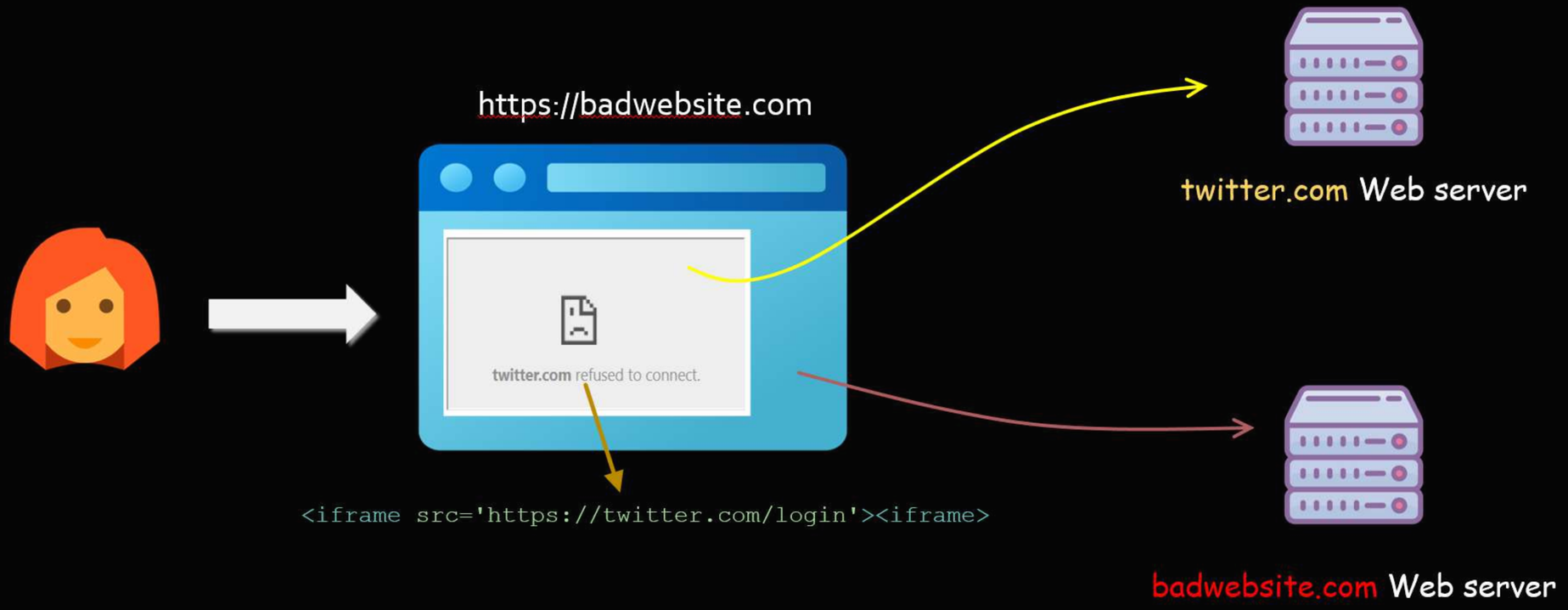
Cross-Origin Resource Sharing

- HTTP-header based mechanism that allows a server to indicate any other origin aside its own from which a browser should authorize loading of resources (from developer.mozilla.org)
- CORS is NOT an API Gateway feature, but a feature most web browsers have
- CORS aims to relax the restrictions of the Same-Origin Policy
- To understand CORS, one must understand the Same-Origin Policy

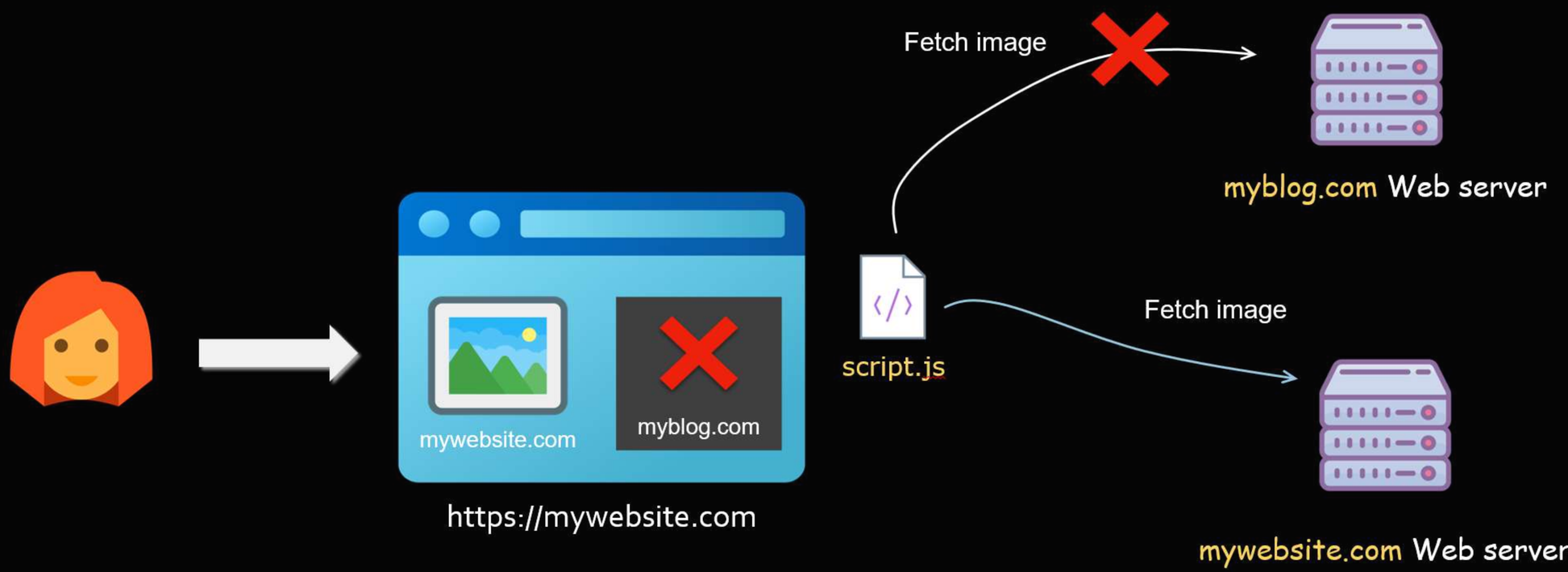
Without Same-Origin Policy



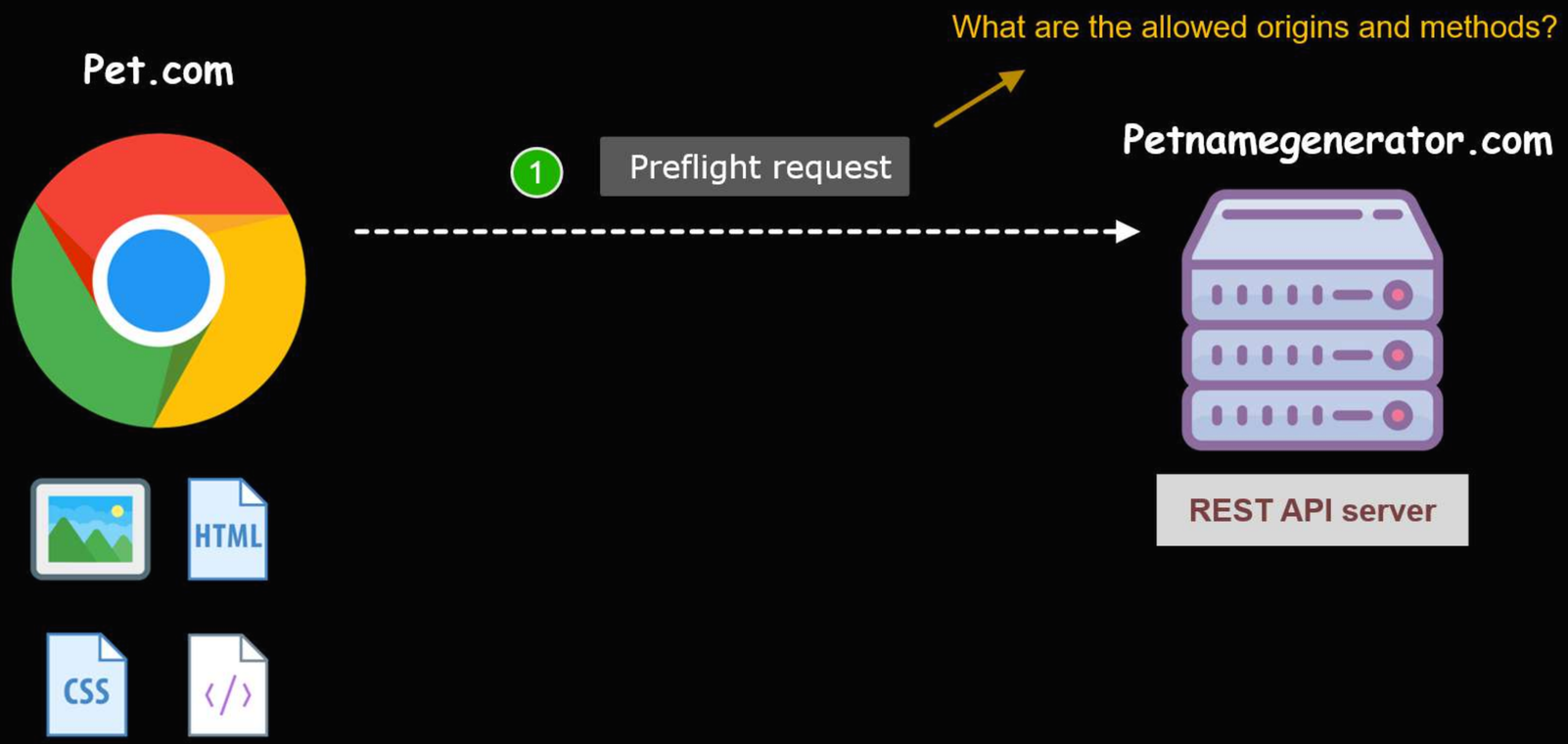
With Same-Origin Policy



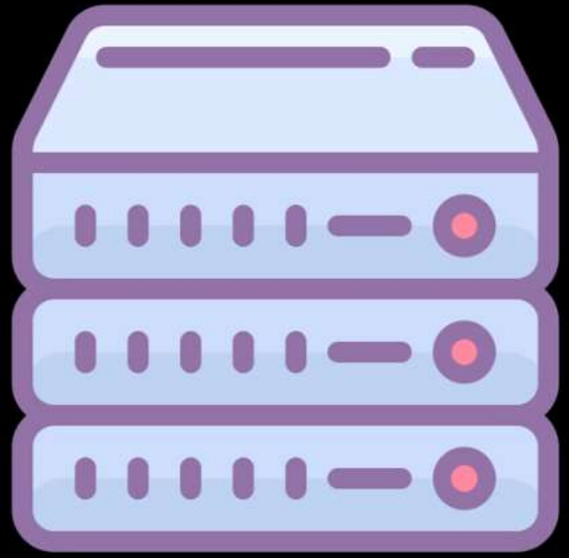
With Same-Origin Policy



Cross-Origin Resource Sharing



Petnamegenerator.com



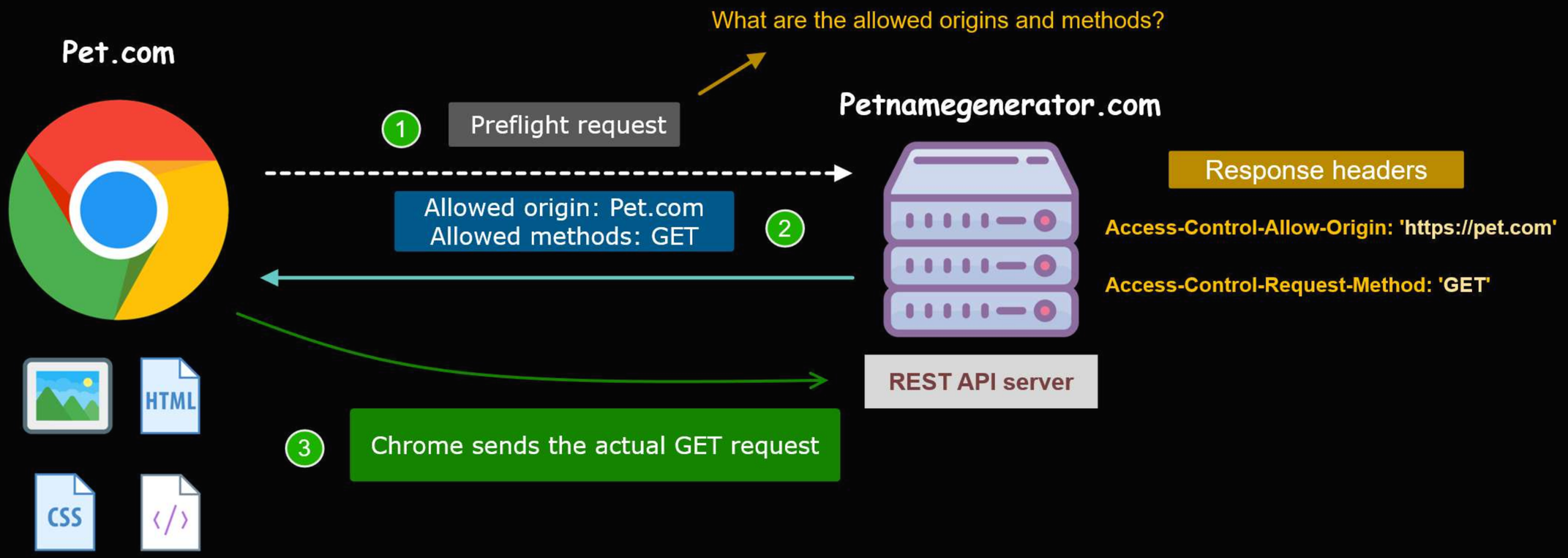
REST API server

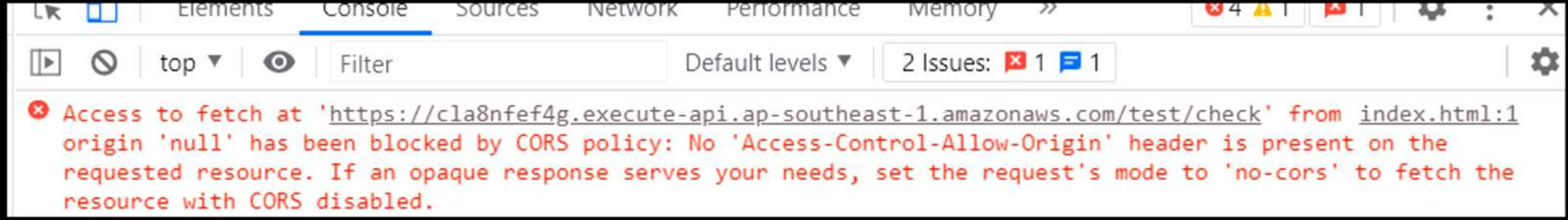
Response headers

Access-Control-Allow-Origin: 'https://pet.com'

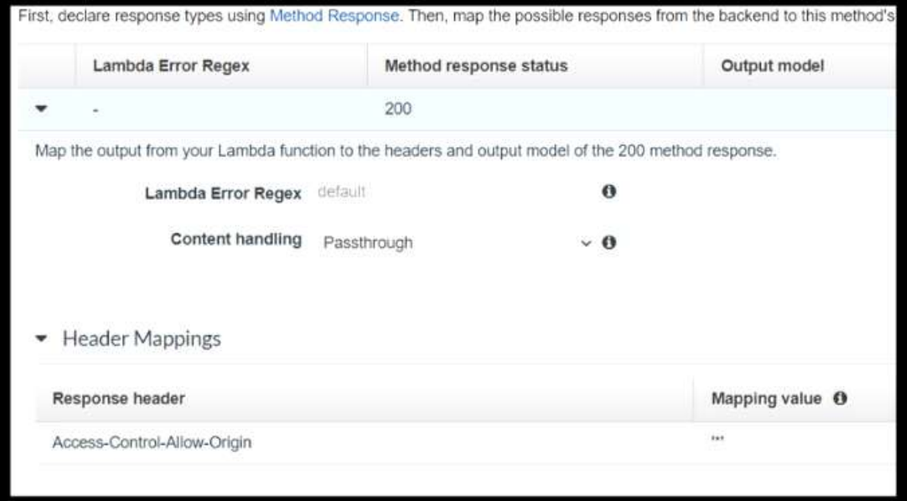
Access-Control-Request-Method: 'GET'

Cross-Origin Resource Sharing

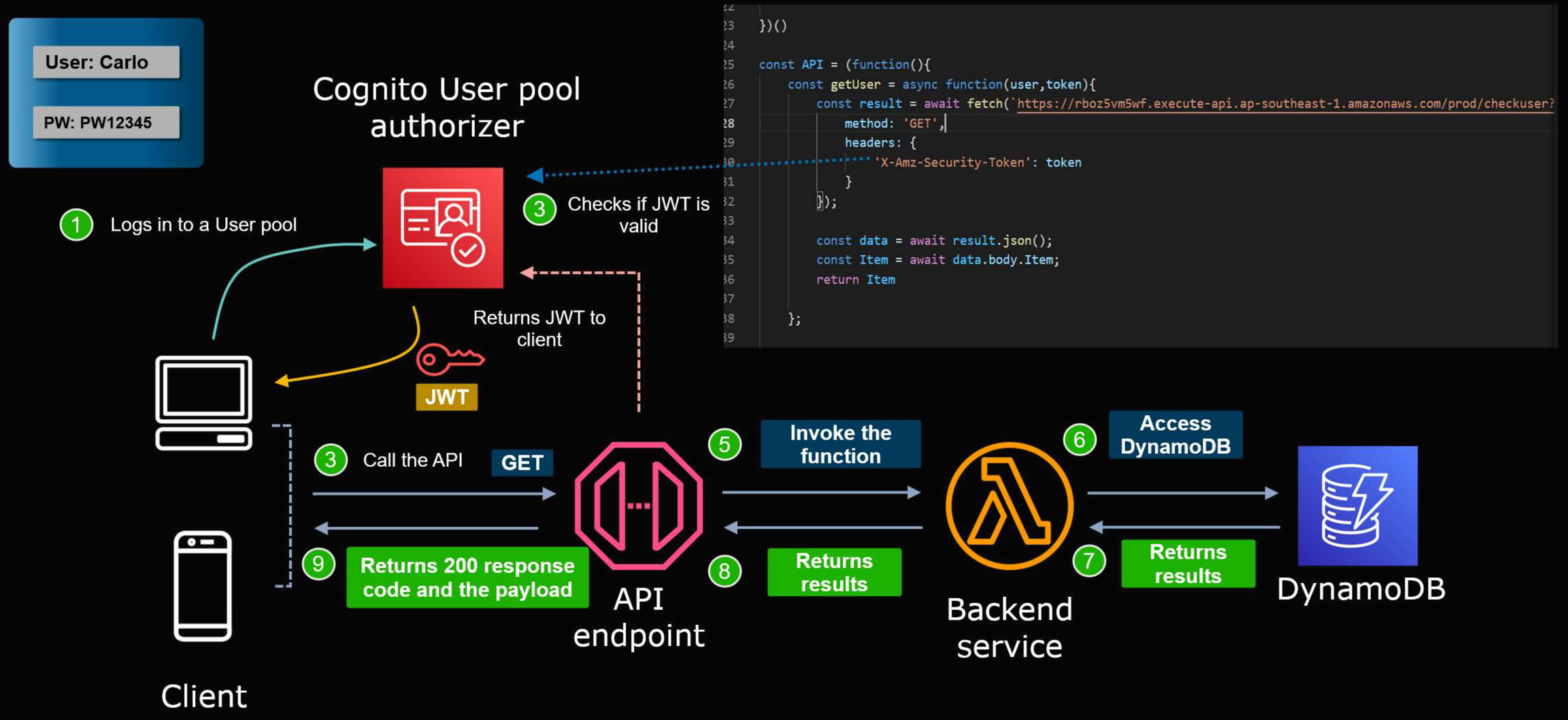


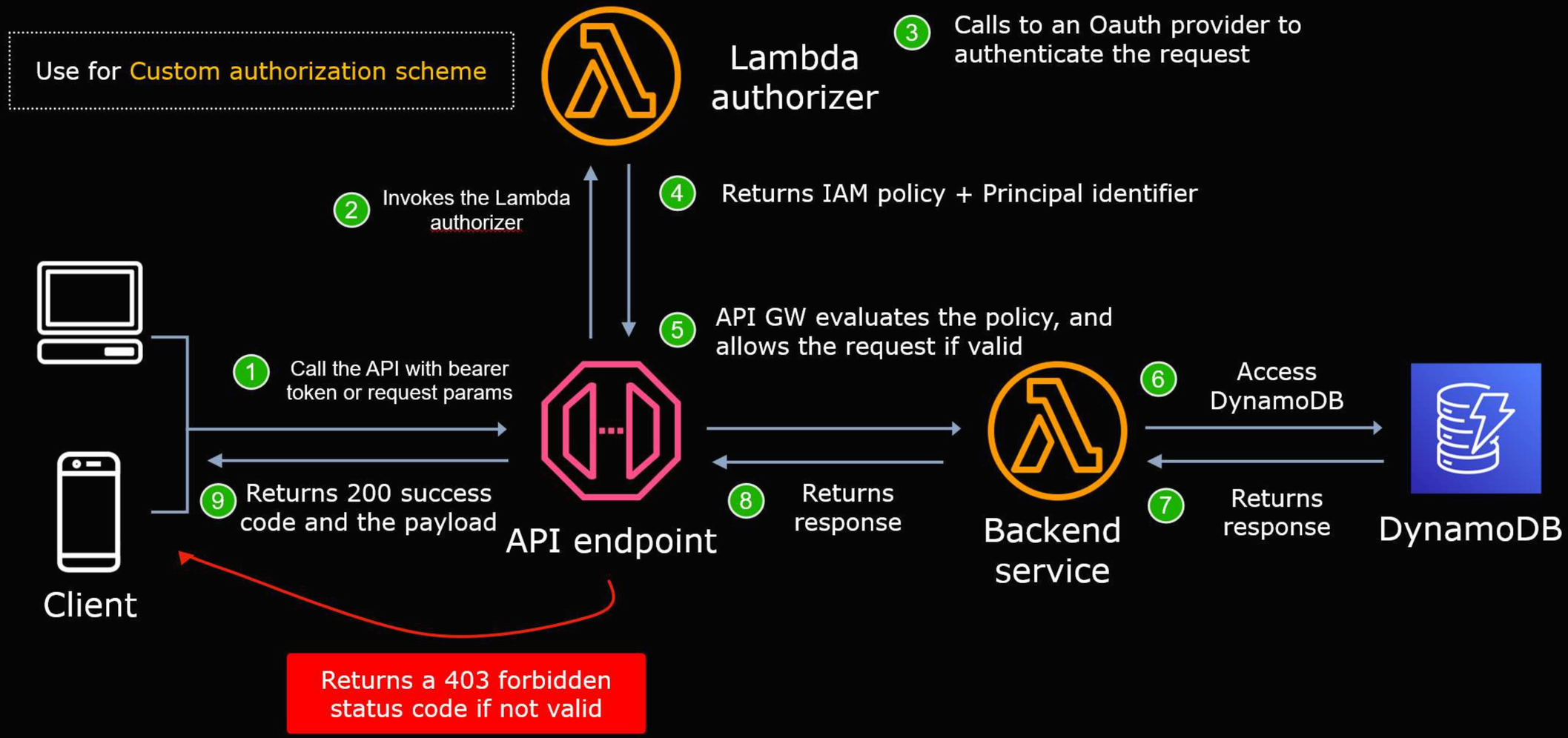


CORS is disabled by default in API Gateway



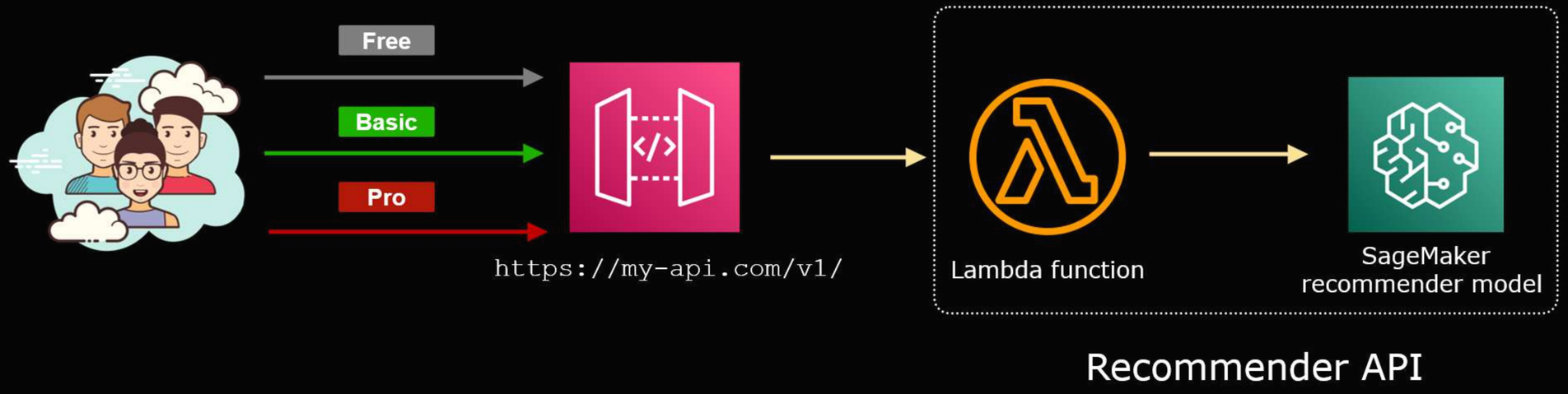
API Gateway **Authorizer**





Usage Plans

- can help you control different levels of access to an API



Sample Usage plan

Basic
$5

Quota

10,00 API calls/day

Rate limit

10 requests per minute

Prem
$10

Quota

50,000 API calls/day

Rate limit

50 requests per minute

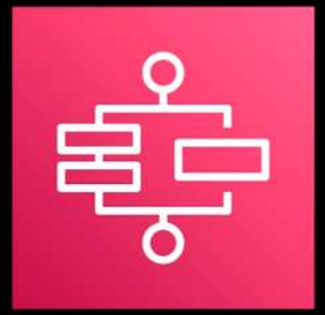
Ultra
$15

Quota

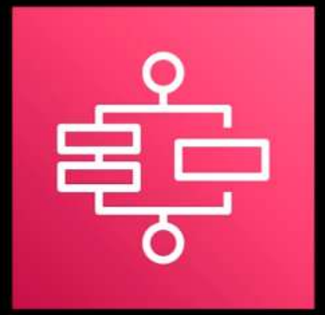
100,000 requests/day

Rate limit

100 requests per minute



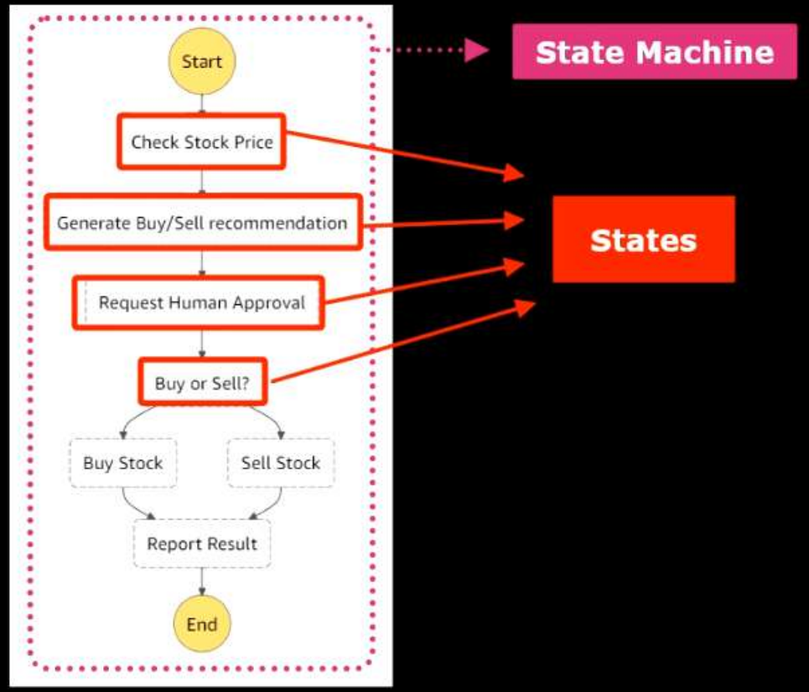
AWS Step Functions

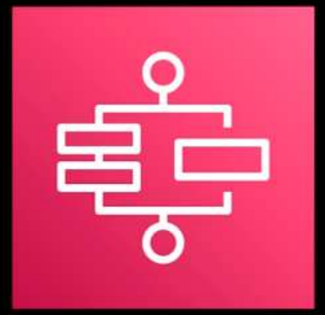


AWS Step Functions

- Orchestrate the components of a microservices application or any processes into a **serverless workflow**

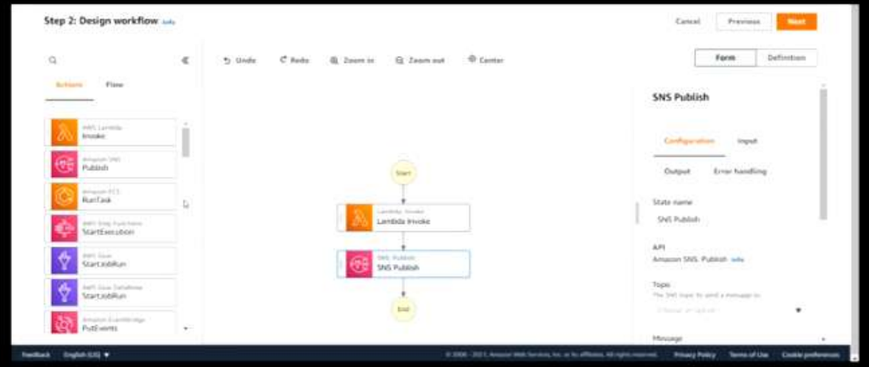
Example of a workflow





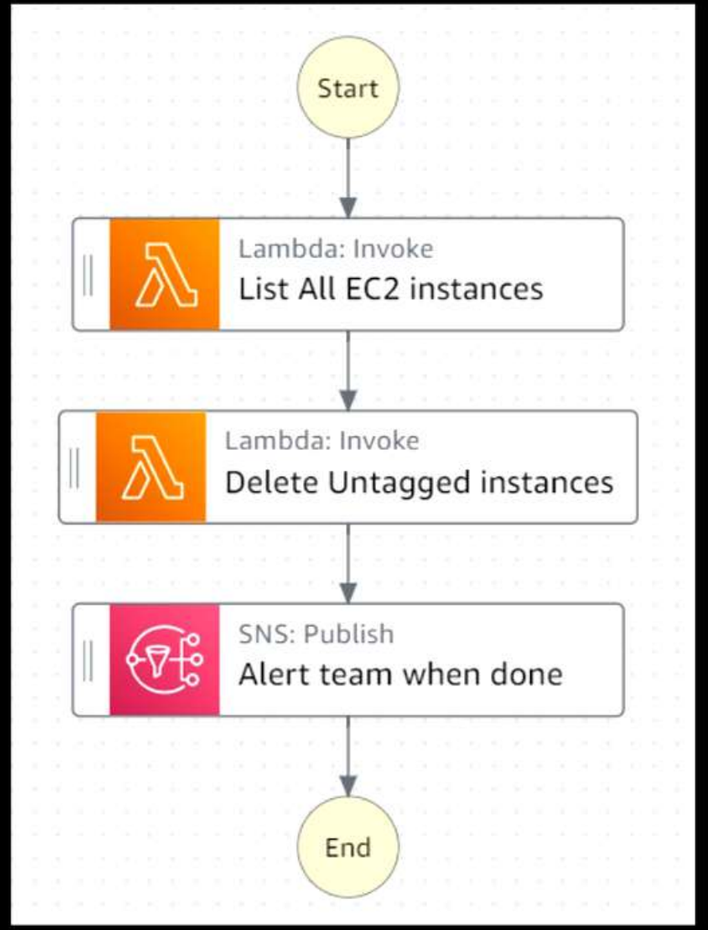
AWS Step Functions

- Orchestrate the components of a microservices application or any processes into a **serverless workflow**
- Review your state machine **visually**
- A State Machine is defined using the **Amazon States Language** (ASL)
- Step Functions can write ASL for you with its **drag-and-drop interface**

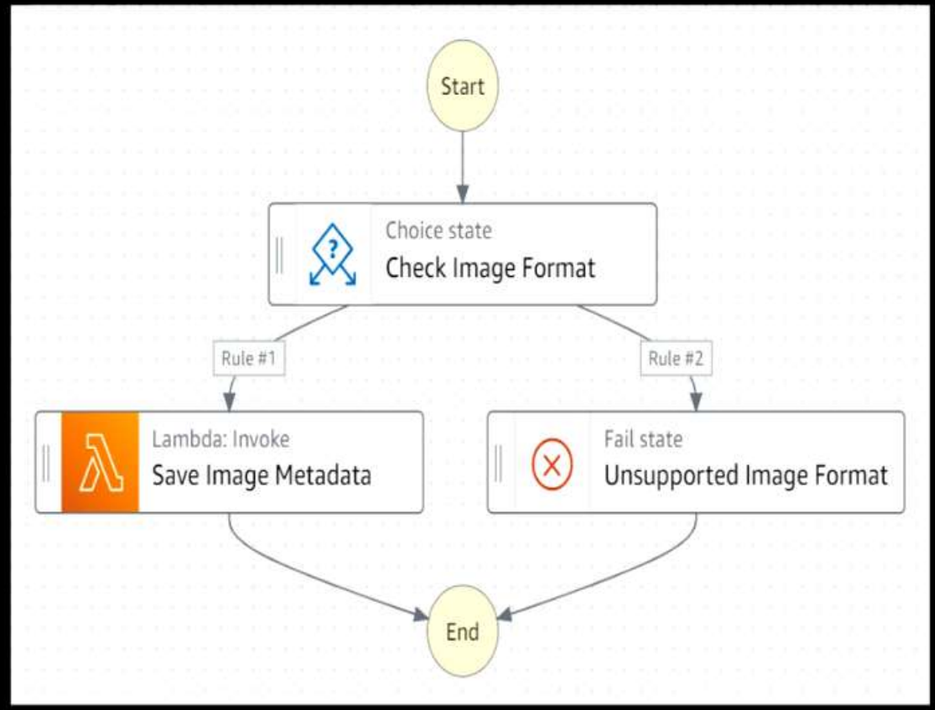


Task	Single unit of work
Choice	Add if-then-else logic
Parallel	Begin parallel branches of execution
Wait	Delays for a specified time
Fail	Stop execution and marks as failure
Succeed	Stop execution and marks as success
Pass	Passes input to its output
Map	Adds a for-each loop

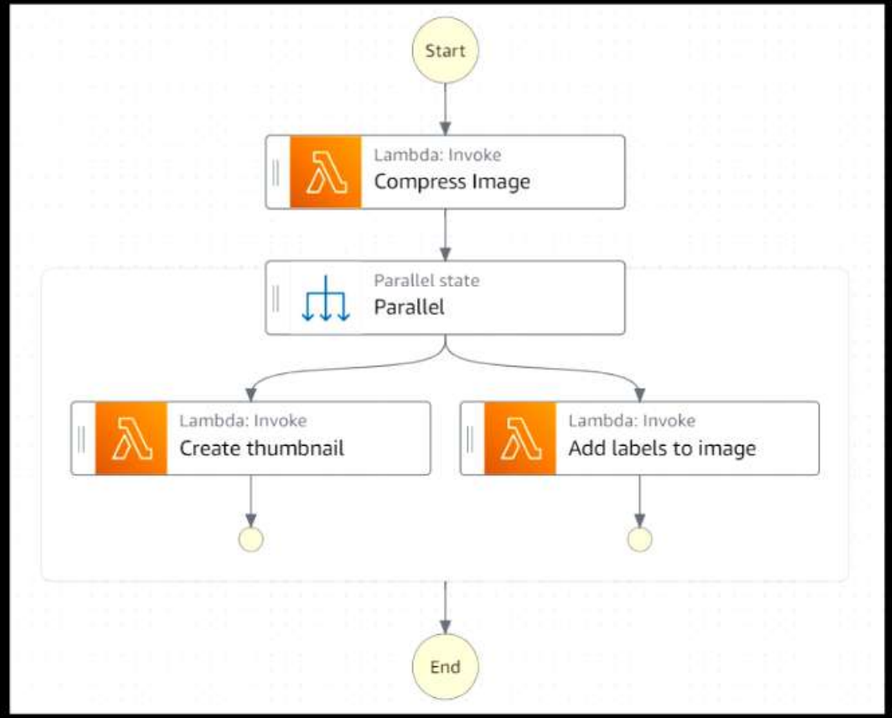
Task	Single unit of work
Choice	Add if-then-else logic
Parallel	Begin parallel branches of execution
Wait	Delays for a specified time
Fail	Stop execution and marks as failure
Succeed	Stop execution and marks as success
Pass	Passes input to its output
Map	Adds a for-each loop



Task	Single unit of work
Choice	Add if-then-else logic
Parallel	Begin parallel branches of execution
Wait	Delays for a specified time
Fail	Stop execution and marks as failure
Succeed	Stop execution and marks as success
Pass	Passes input to its output
Map	Adds a for-each loop

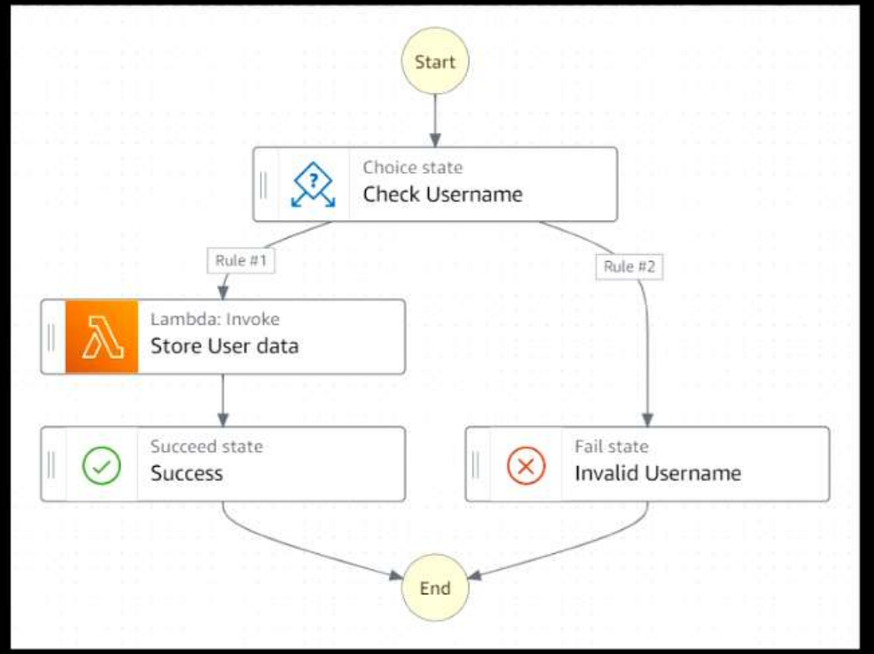


Task	Single unit of work
Choice	Add if-then-else logic
Parallel	Begin parallel branches of execution
Wait	Delays for a specified time
Fail	Stop execution and marks as failure
Succeed	Stop execution and marks as success
Pass	Passes input to its output
Map	Adds a for-each loop

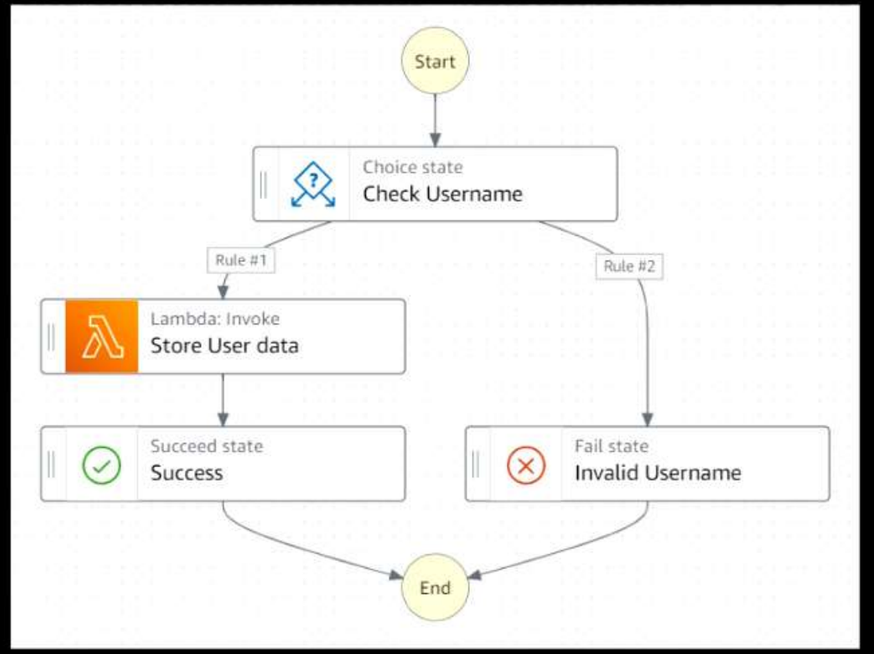


Task	Single unit of work
Choice	Add if-then-else logic
Parallel	Begin parallel branches of execution
Wait	Delays for a specified time
Fail	Stop execution and marks as failure
Succeed	Stop execution and marks as success
Pass	Passes input to its output
Map	Adds a for-each loop

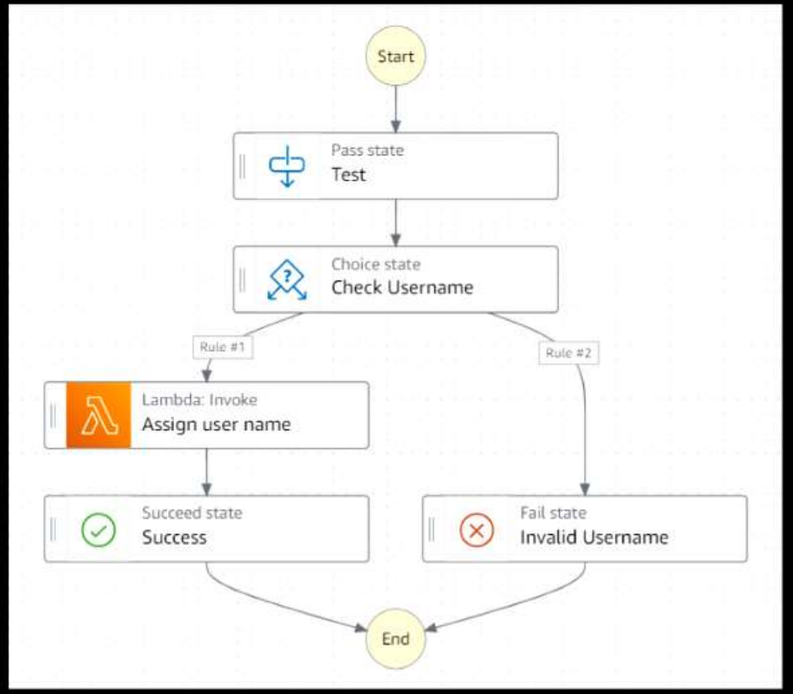
Task	Single unit of work
Choice	Add if-then-else logic
Parallel	Begin parallel branches of execution
Wait	Delays for a specified time
Fail	Stop execution and marks as failure
Succeed	Stop execution and marks as success
Pass	Passes input to its output
Map	Adds a for-each loop



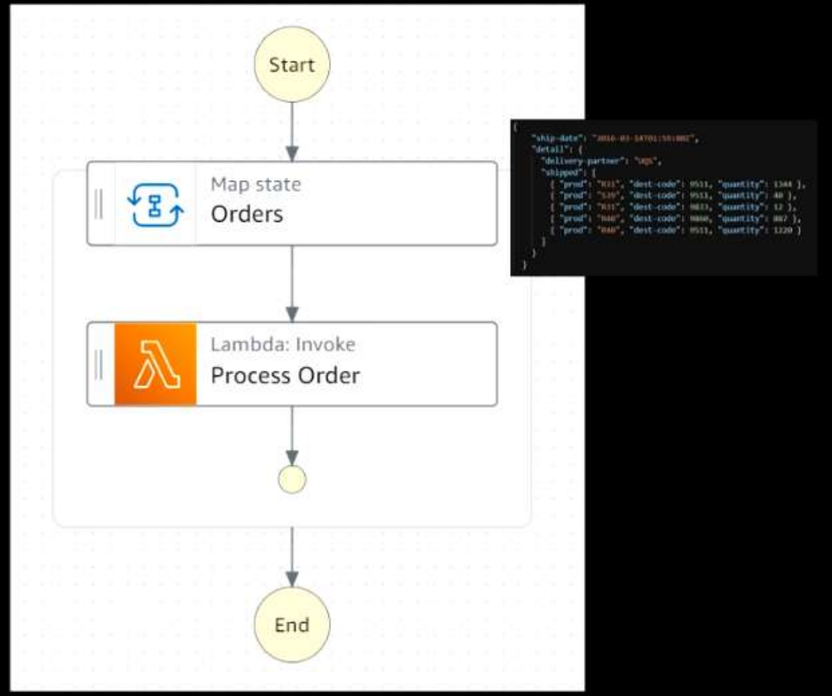
Task	Single unit of work
Choice	Add if-then-else logic
Parallel	Begin parallel branches of execution
Wait	Delays for a specified time
Fail	Stop execution and marks as failure
Succeed	Stop execution and marks as success
Pass	Passes input to its output
Map	Adds a for-each loop



Task	Single unit of work
Choice	Add if-then-else logic
Parallel	Begin parallel branches of execution
Wait	Delays for a specified time
Fail	Stop execution and marks as failure
Succeed	Stop execution and marks as success
Pass	Passes input to its output
Map	Adds a for-each loop



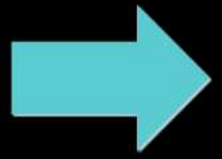
Task	Single unit of work
Choice	Add if-then-else logic
Parallel	Begin parallel branches of execution
Wait	Delays for a specified time
Fail	Stop execution and marks as failure
Succeed	Stop execution and marks as success
Pass	Passes input to its output
Map	Adds a for-each loop

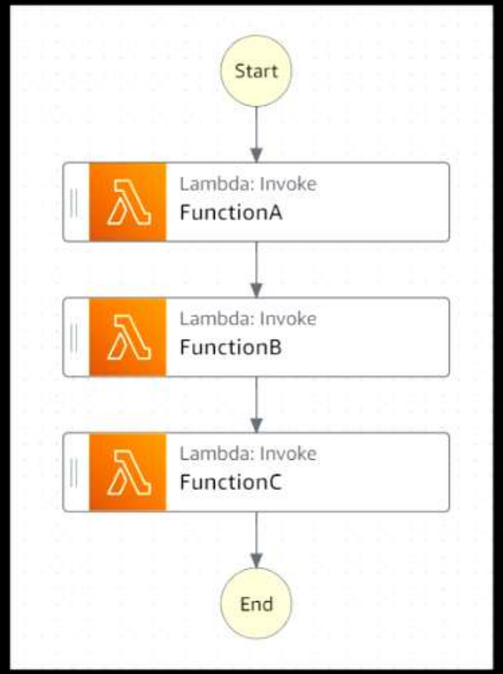


Built-in retry and error handling capabilities

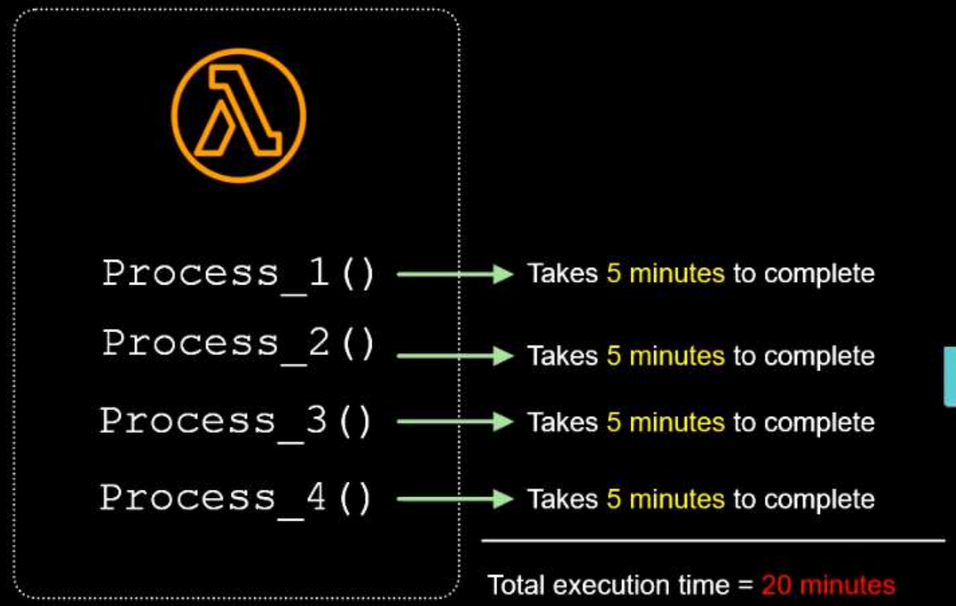
lambda_function.py | Execution results

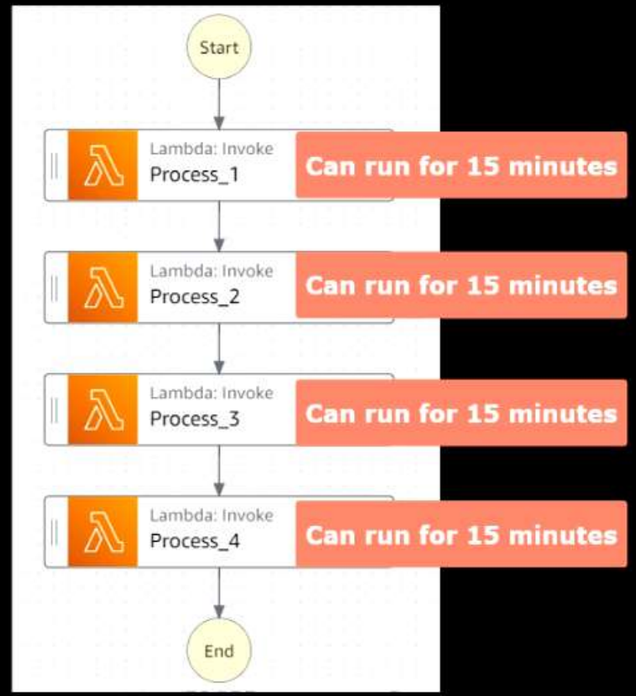
```python
import random

def lambda_handler(event, context):

    resultA = functionA()

    resultB = functionB(resultA)

    resultC = functionC(resultB)

    return {
        'statusCode': 200,
        'body': resultC
    }

def functionA():
  number = random.randint(0,5)
  return number

def functionB(number):
  return number*number

def functionC(result):
  return {
      "Output": result
  }
```

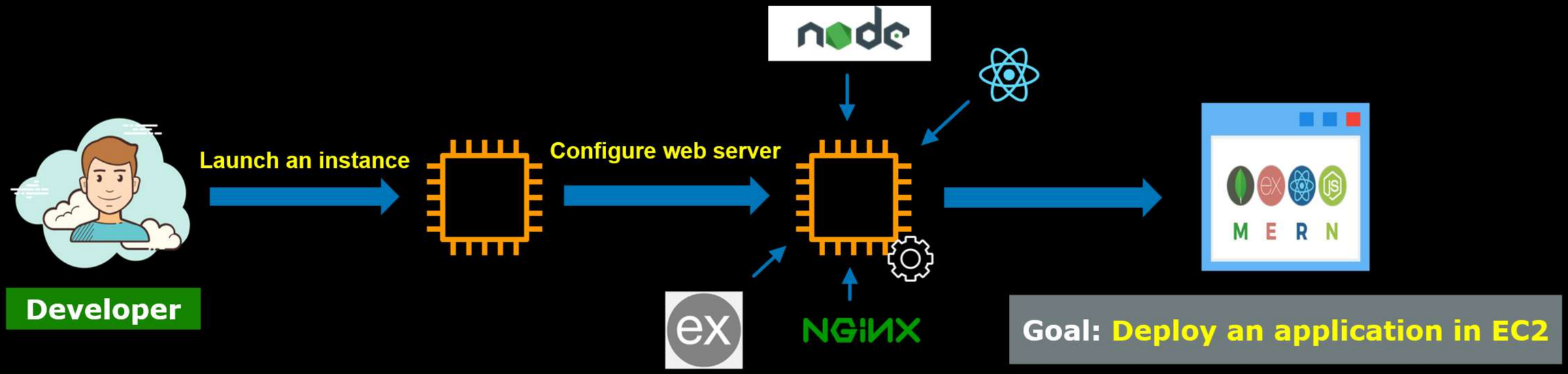


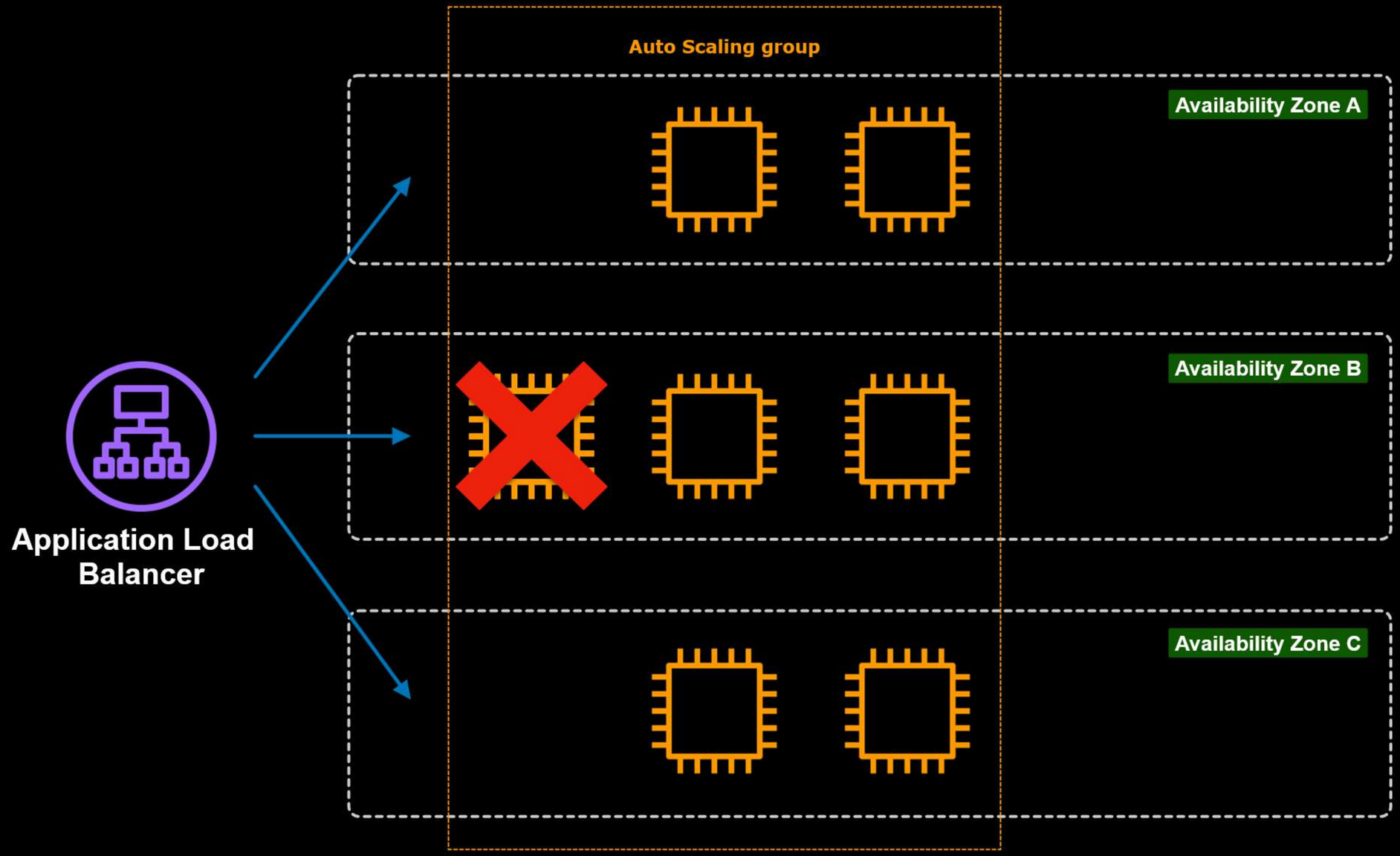


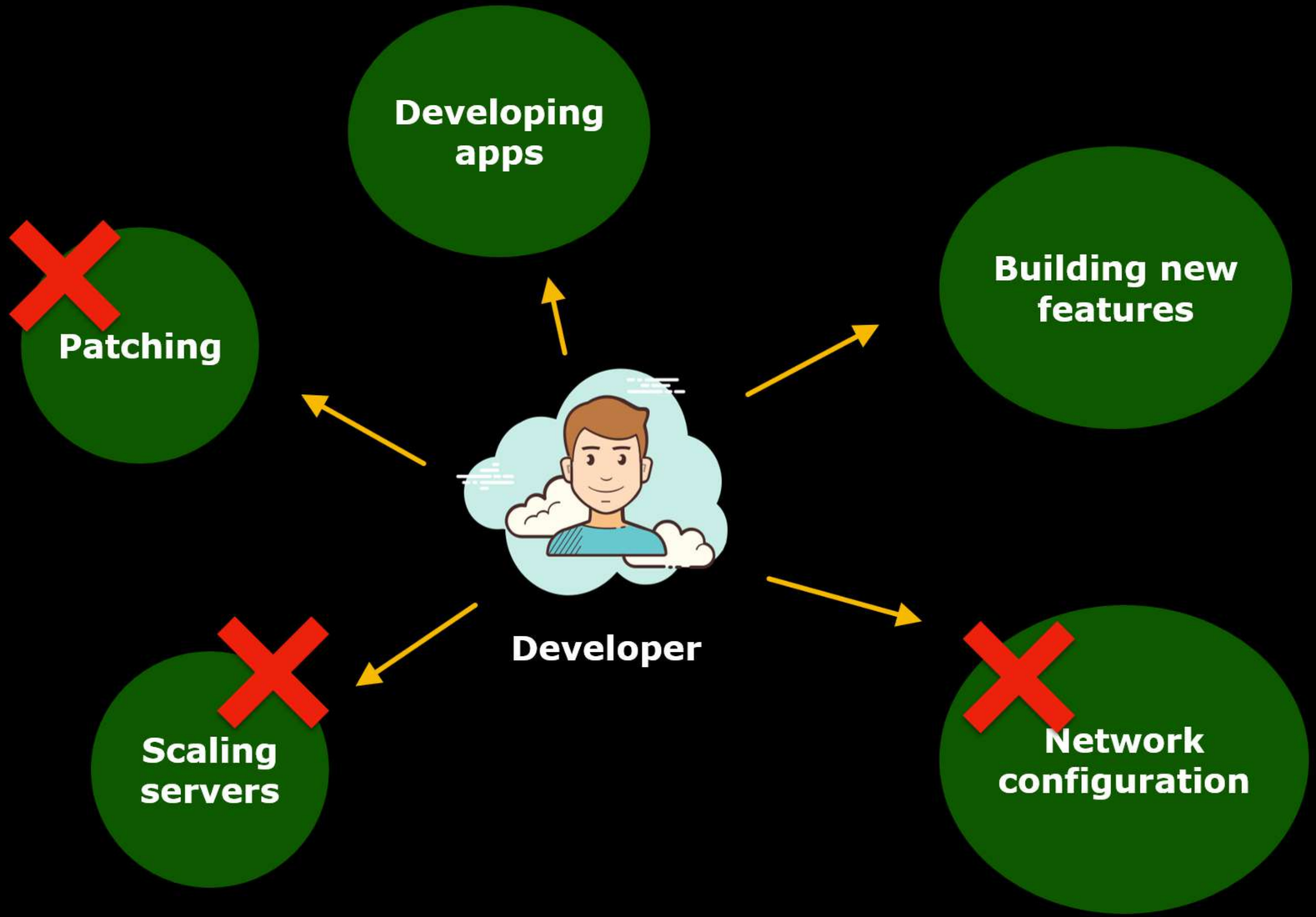
Suitable for handling long-running jobs











AWS Elastic Beanstalk

- AWS Elastic Beanstalk is a **Platform as a Service (PaaS)** product
- **PaaS removes** the need for organizations to **manage the underlying infrastructure** and allows you to **focus on the deployment and management of your applications**
- **As a developer, you can just upload your code**
- **Elastic Beanstalk takes care of:**
  - **resource provisioning**
  - **load balancing**
  - **monitoring**
  - **database management**
  - **patching, and many more..**

AWS Elastic Beanstalk

- **Configuration settings in Elastic Beanstalk:**
  - **Auto scaling group**
  - **Modify the root volume of your instance**
  - **Configure database (RDS only)**
  - **If you have a database outside the Elastic Beanstalk environment , ensure that your instance's security group and db connection settings are properly set**
  - **AWS X-ray (debugging)**
  - **Amazon CloudWatch Logs (monitoring)**
  - **Amazon SNS (notifications)**

AWS Elastic Beanstalk

- Elastic Beanstalk supports applications written in:
  - Java
  - .Net
  - Nodejs
  - PHP
  - Ruby
  - Python
  - Go
- It also supports Dockerized applications

AWS Elastic Beanstalk

- **Elastic Beanstalk is free**
- **You only pay for the created resources (ex: instances, ALB)**
- **You still have full control over your resources**

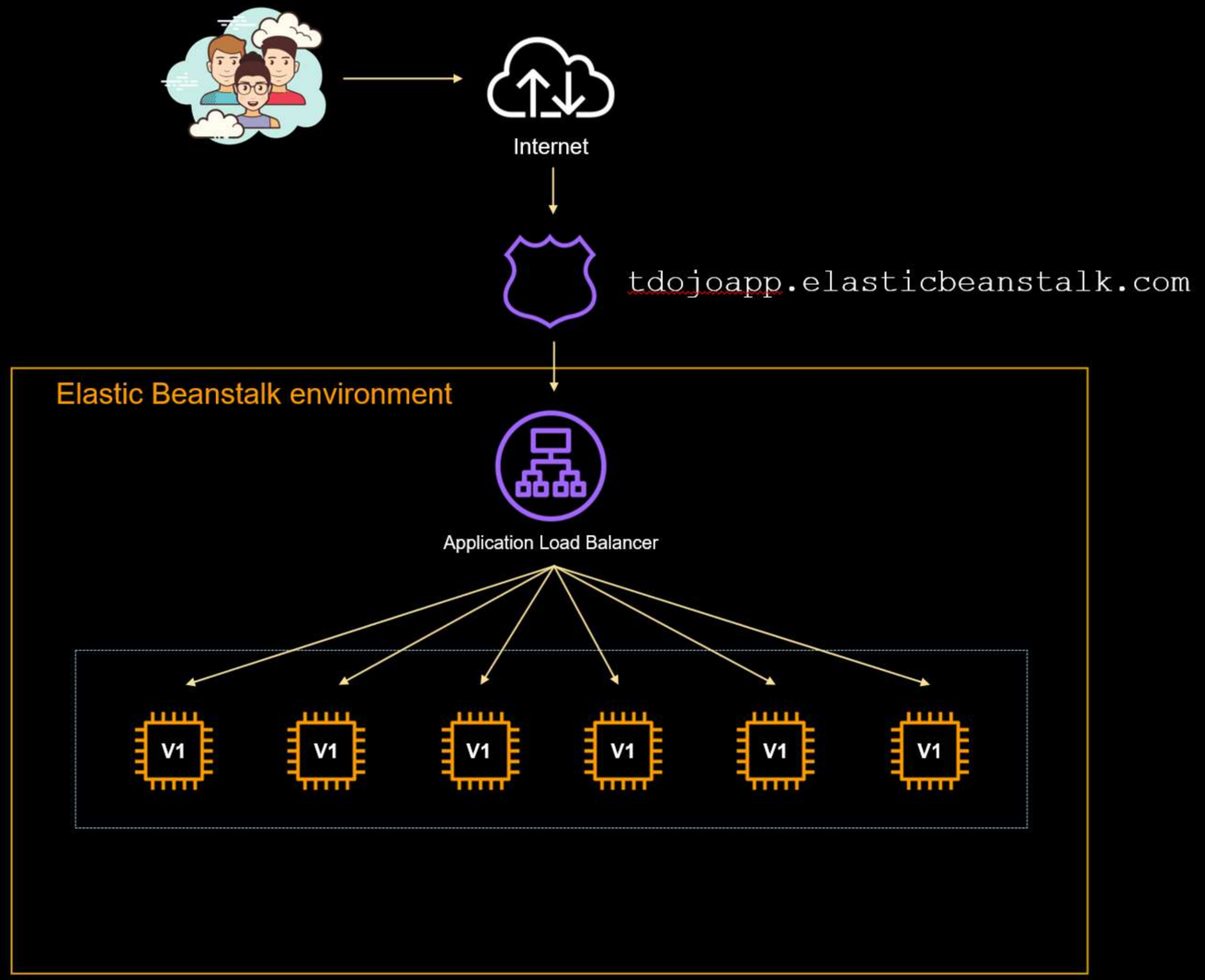
AWS Elastic Beanstalk

- **Elastic Beanstalk is free**
- **You only pay for the created resources (ex: instances, ALB)**
- **You still have full control over your resources**

Elastic Beanstalk Deployment Policies

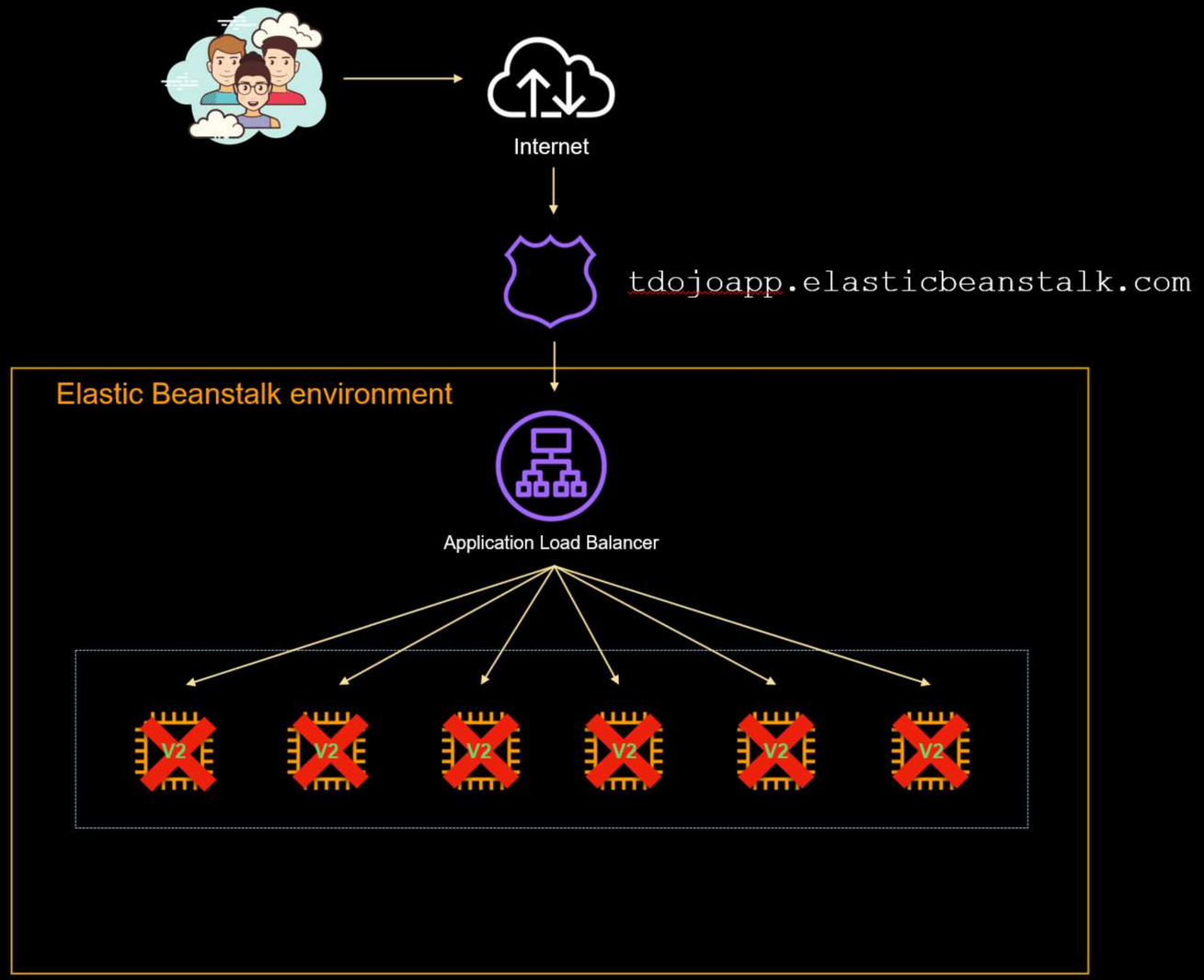
All at once

- **New code version is deployed to all instances at the same time**



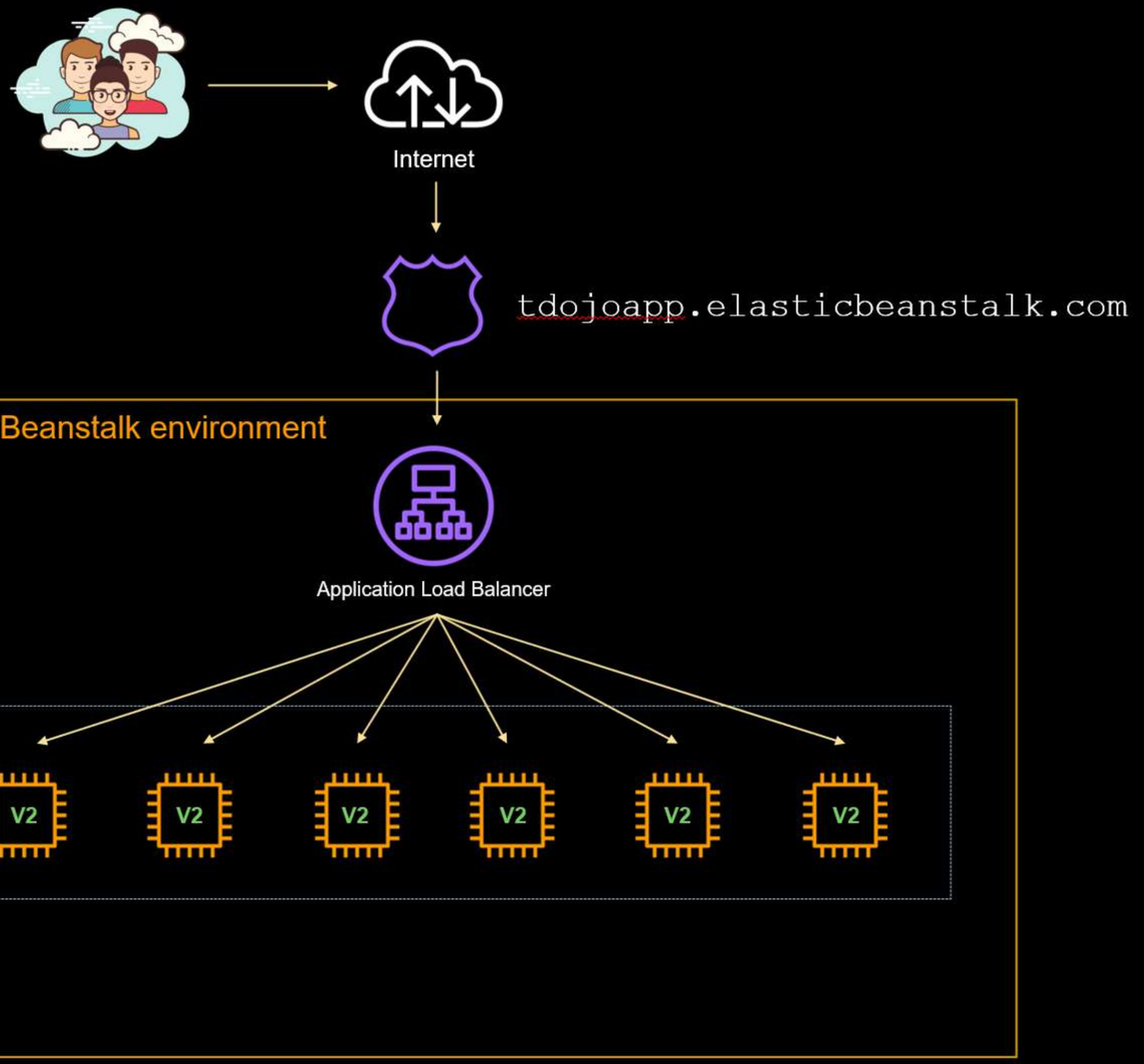
All at once

- **New code version is deployed to all instances at the same time**



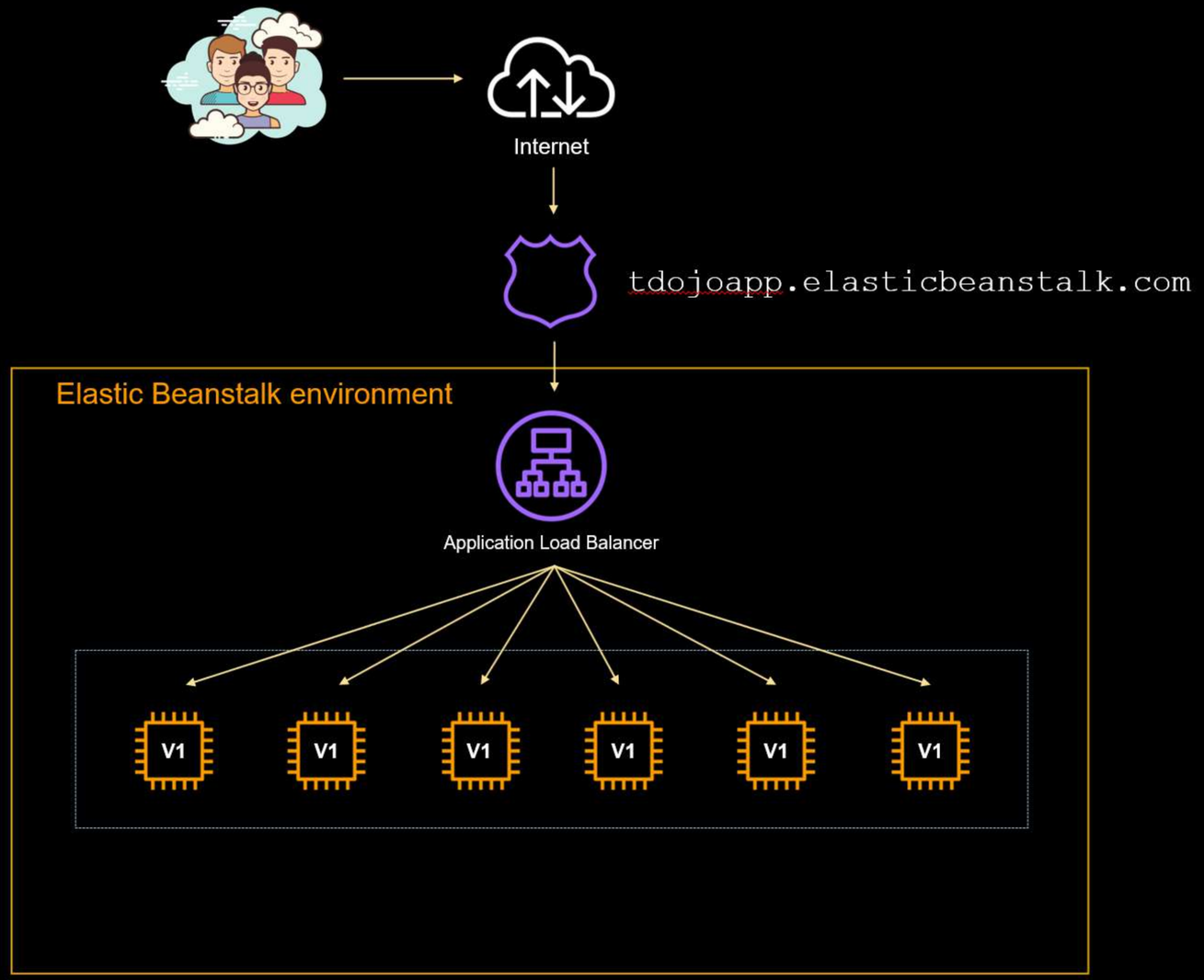
All at once

- New code version is deployed to all instances at the same time
- Deployment is fast
- Suitable for quickly releasing changes in a development environment
- Your application is unavailable during deployment
- It is not recommended in a production setting



Rolling

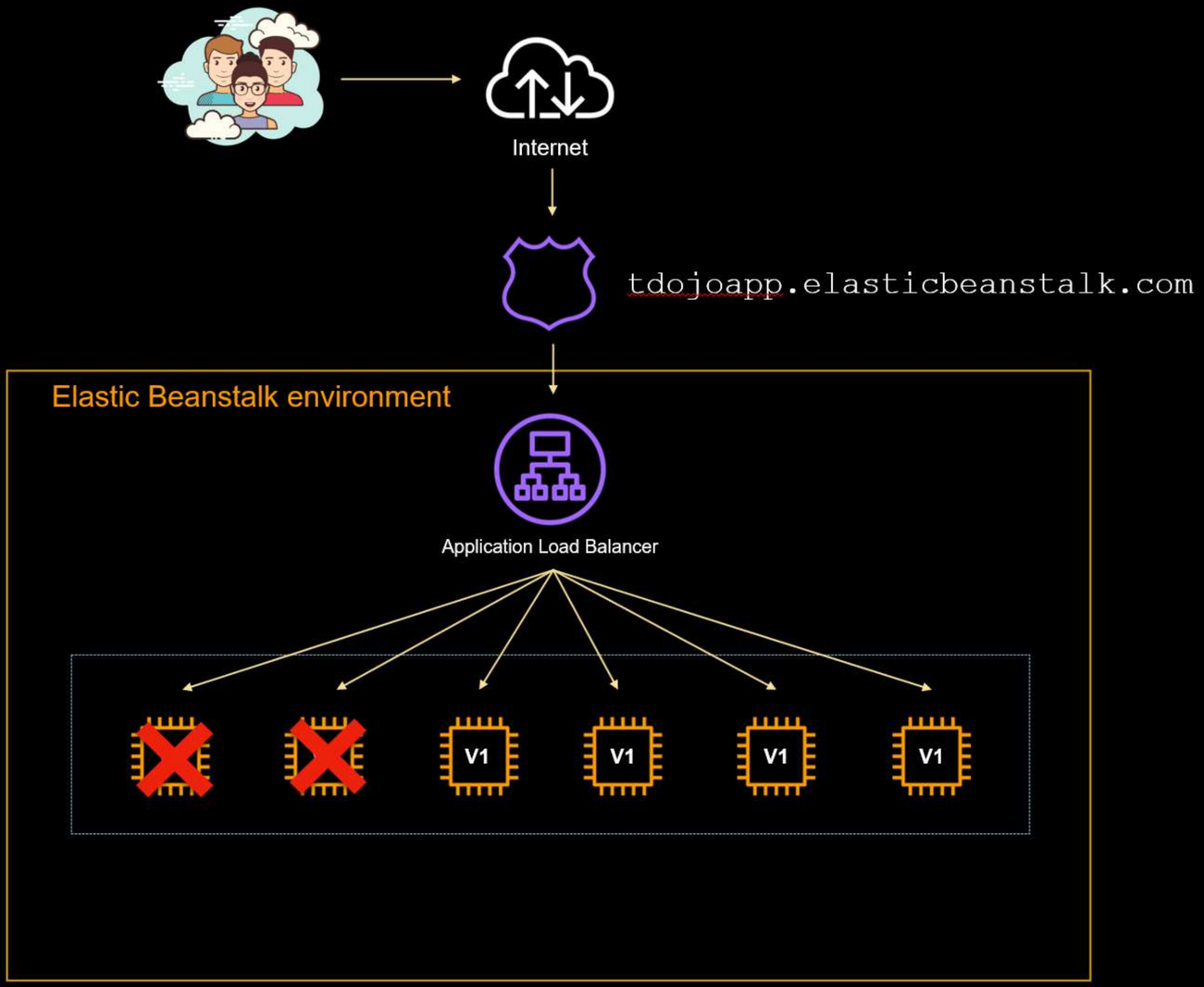
- New code version is deployed in batches
- You choose the batch size
- Batch size is the number of instances to deploy in each batch



Batch size = 2

Rolling

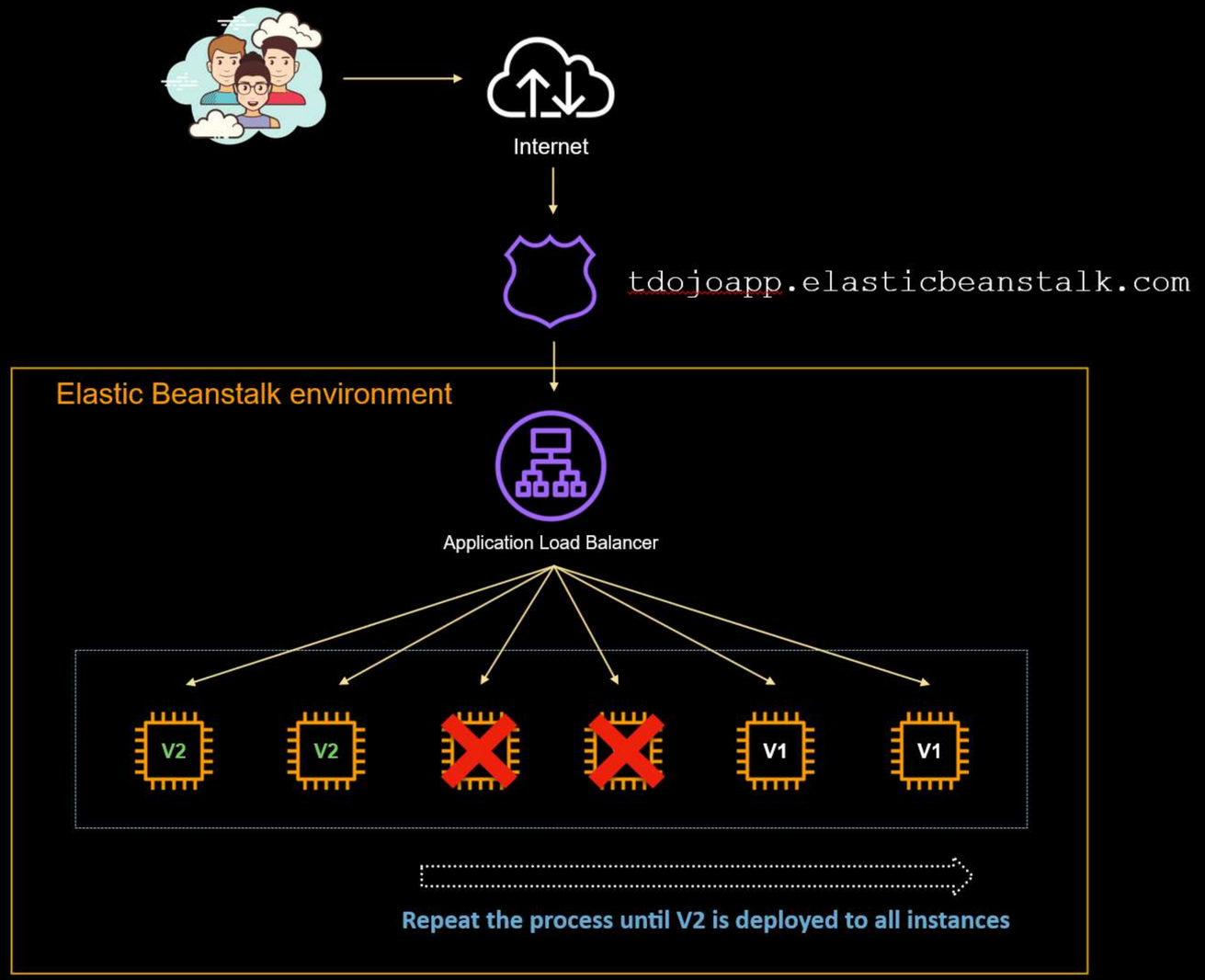
- New code version is deployed in batches
- You choose the batch size
- Batch size is the number of instances to deploy in each batch



Batch size = 2

Rolling

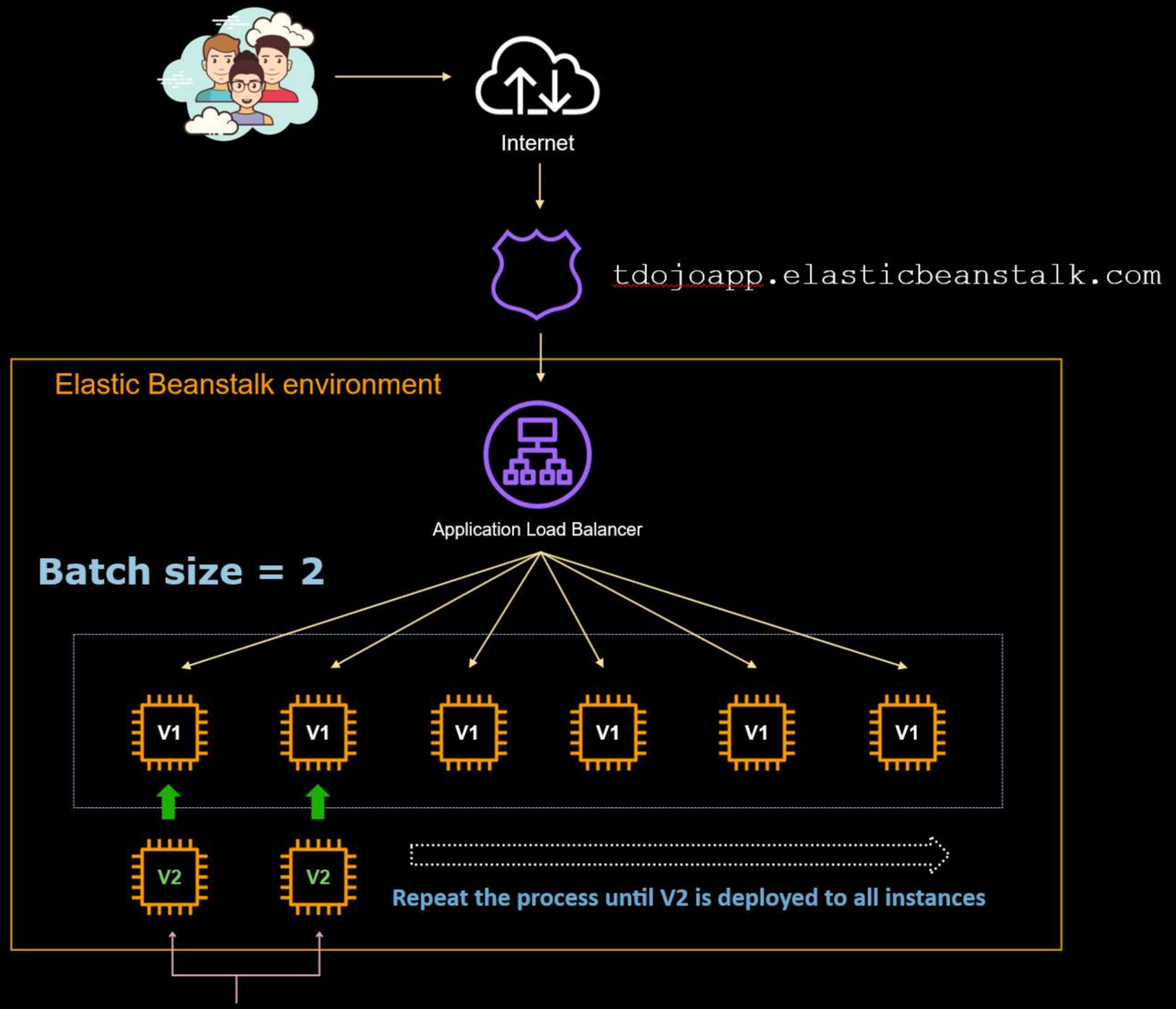
- New code version is deployed in batches
- You choose the batch size
- Batch size is the number of instances to deploy in each batch
- The impact of failed deployment is lower than All at once
- The application is available at a reduced capacity during deployment



Batch size = 2

Rolling with an additional batch

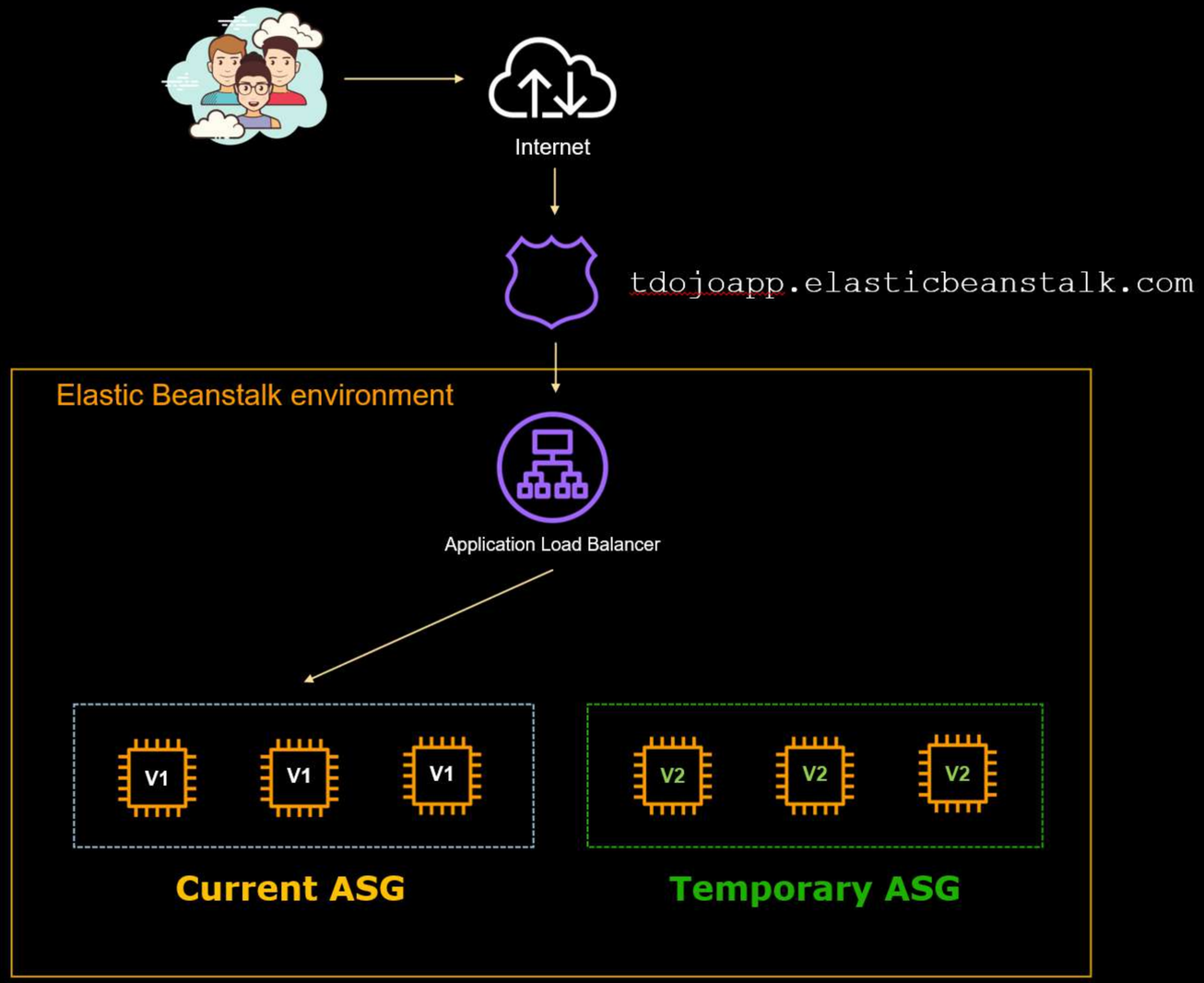
- A **variation** of the Rolling deployment
- Your application will run at **full capacity** during deployment
- If the deployment of first batch fails, the created instances are **terminated**
- If the succeeding batches fail, you need to **redeploy the old version** to revert back the changes
- Elastic Beanstalk **won't** be able to create new instances once you reach the max instance limit



ELB will deploy V2 to new instances and replace the old instances

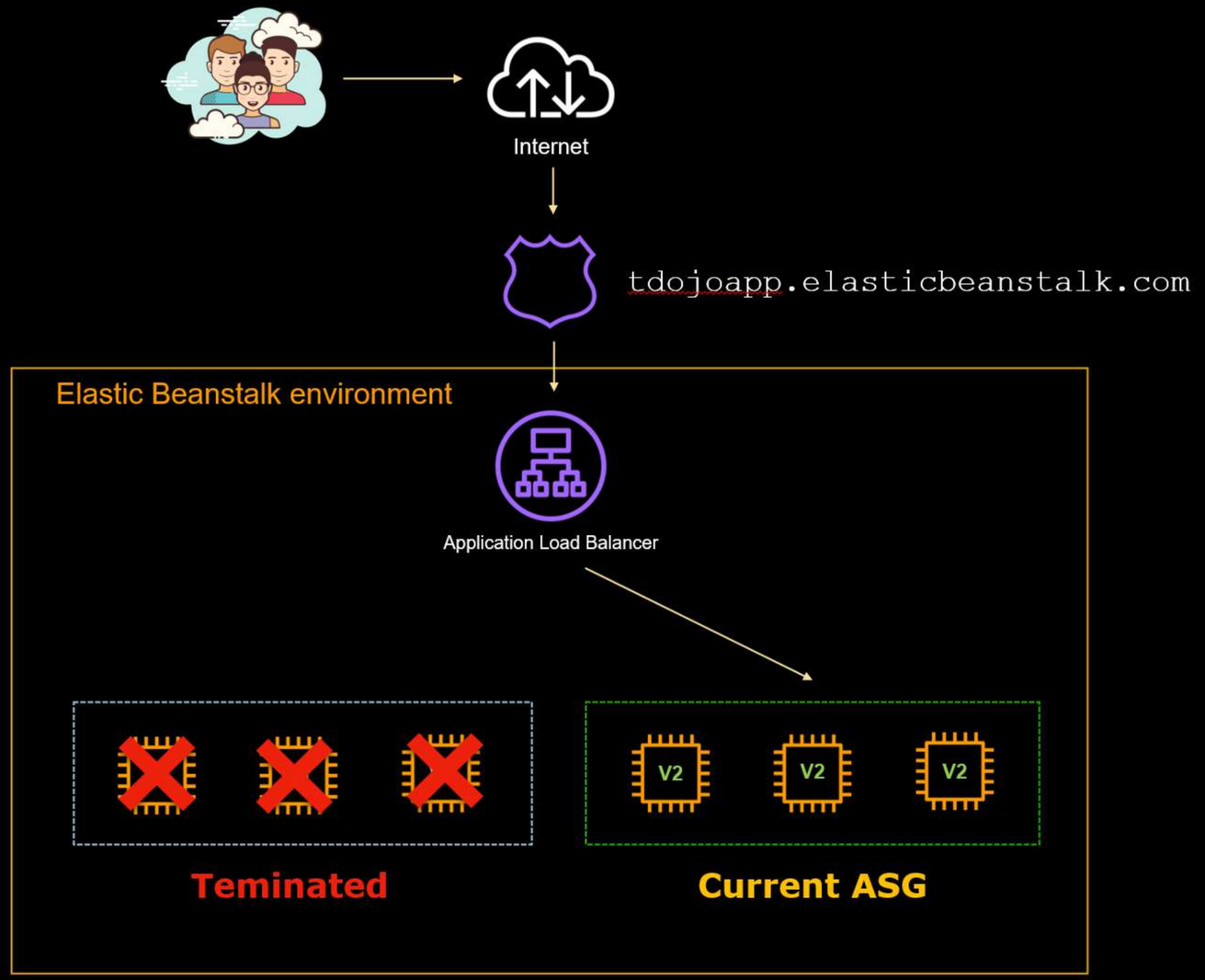
Immutable

- Existing instances are not updated
- ELB duplicates the current instances in a separate Auto Scaling group



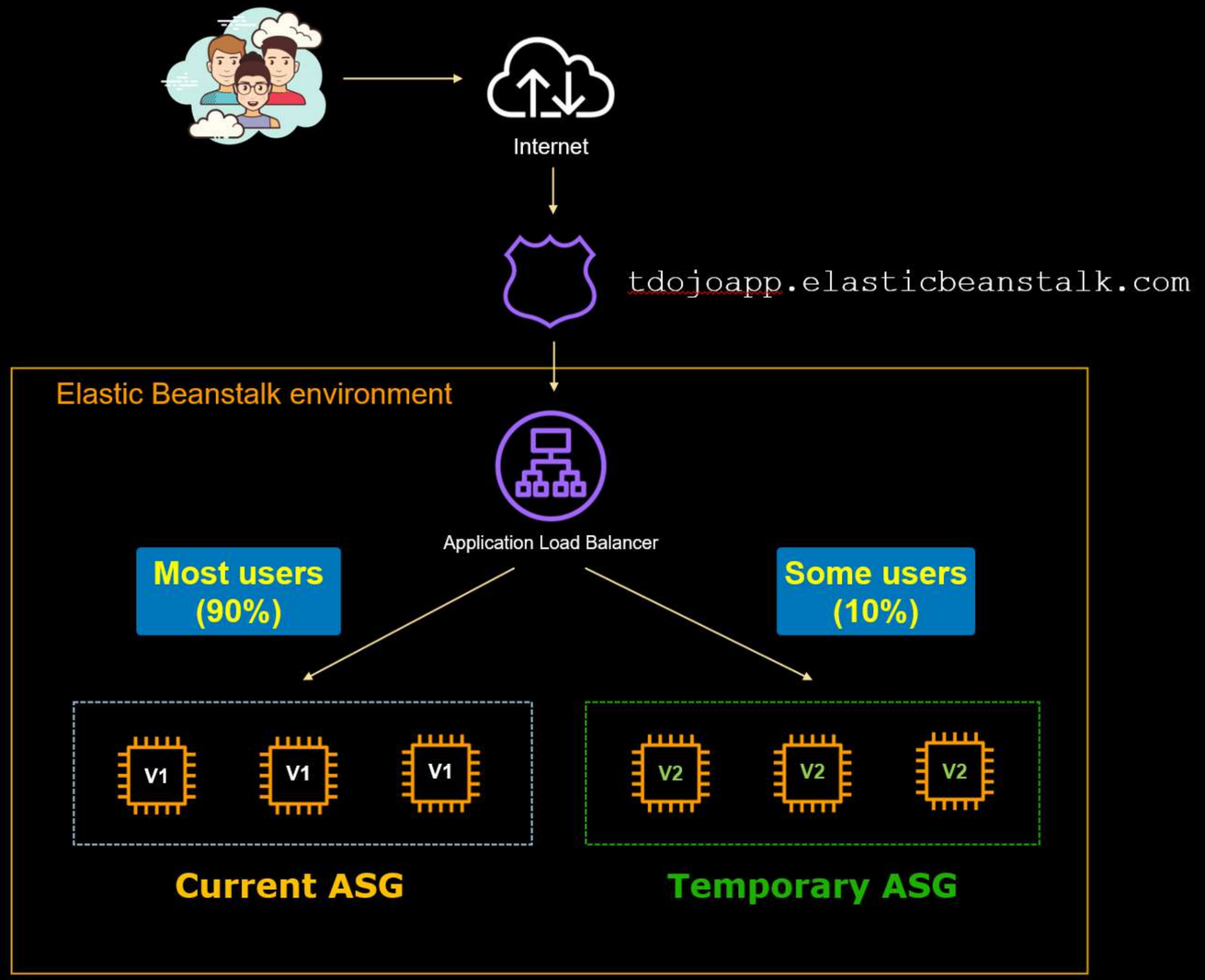
Immutable

- Existing instances **are not updated**
- ELB **duplicates** the current instances in a separate Auto Scaling group
- Rollbacks **don't involve downtime**
- Rollbacks are **faster**
- Your application runs at **full capacity** during deployment
- It's **costly** to implement



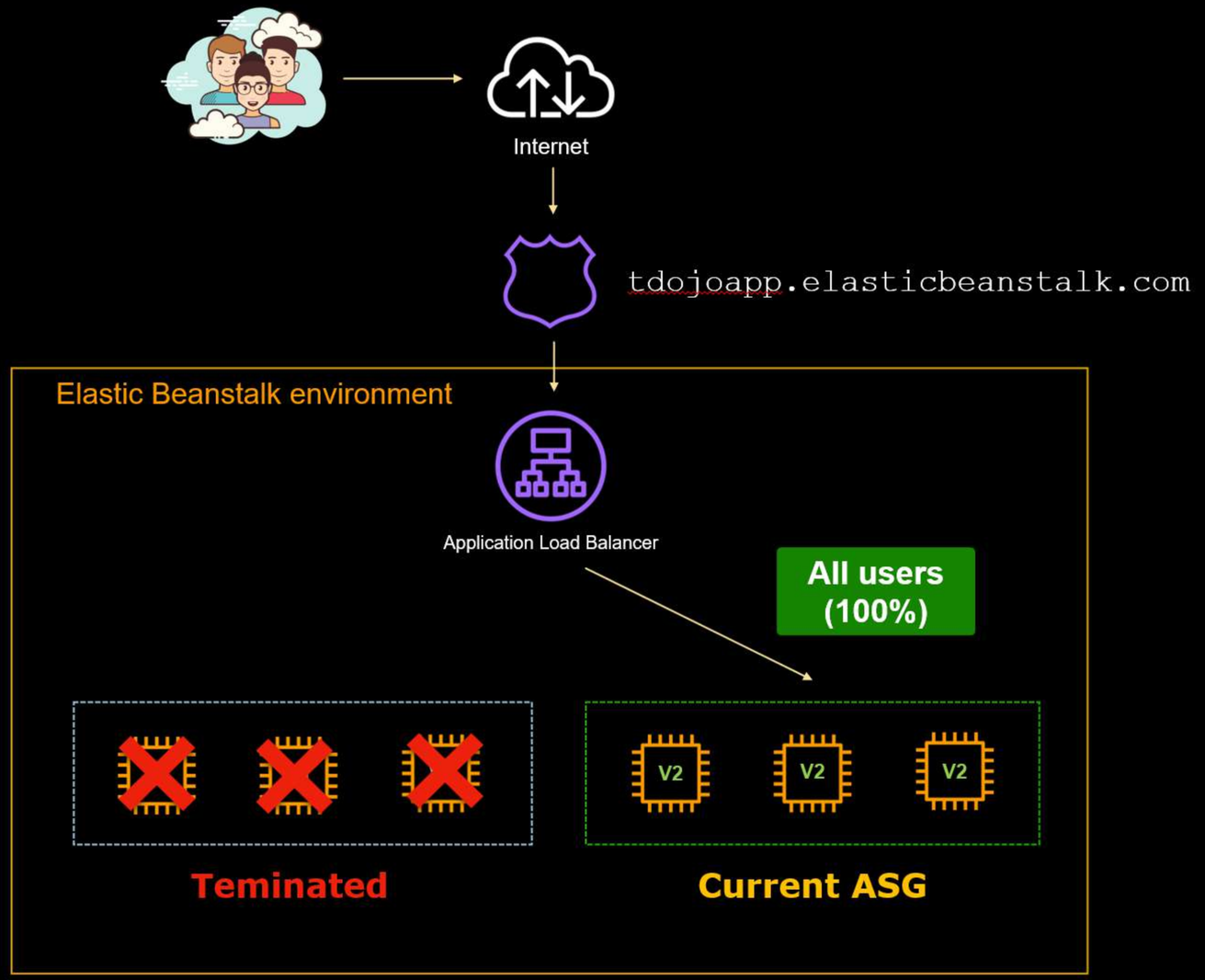
Traffic Splitting

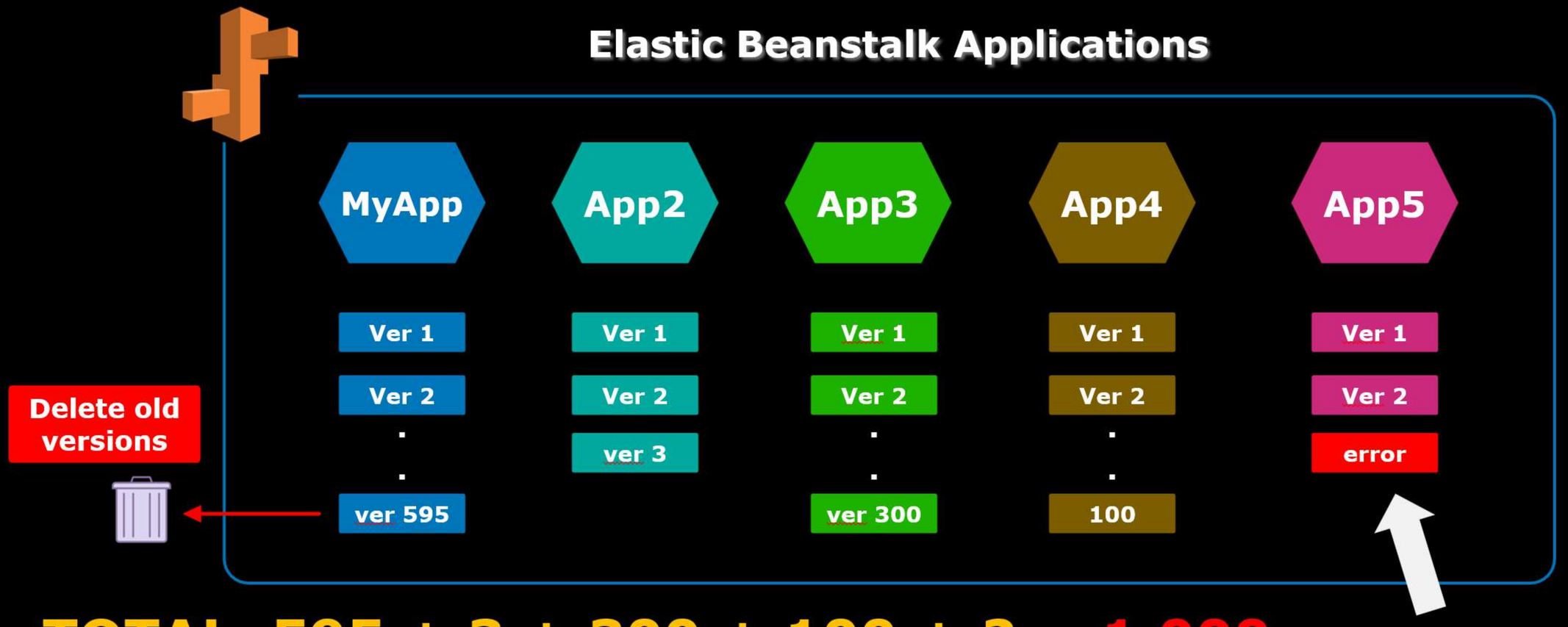
- Suitable for **A/B or Canary testing**
- A Canary test allows you to **gradually release** changes to a small percentage of your users
- ELB deploys updates to a separate ASG and **shifts a percentage of traffic** towards it
- Deployment is successful if the new instances **pass the health check** within the defined evaluation period



Traffic Splitting

- Suitable for **A/B or Canary testing**
- A Canary test allows you to **gradually release** changes to a small percentage of your users
- ELB deploys updates to a separate ASG and **shifts a percentage of traffic** towards it
- Deployment is successful if the new instances **pass the health check** within the defined evaluation period
- 100% of the traffic is **shifted the new instances** once the deployment is successful
- In case a deployment fails, traffic are **rerouted** to the old instances



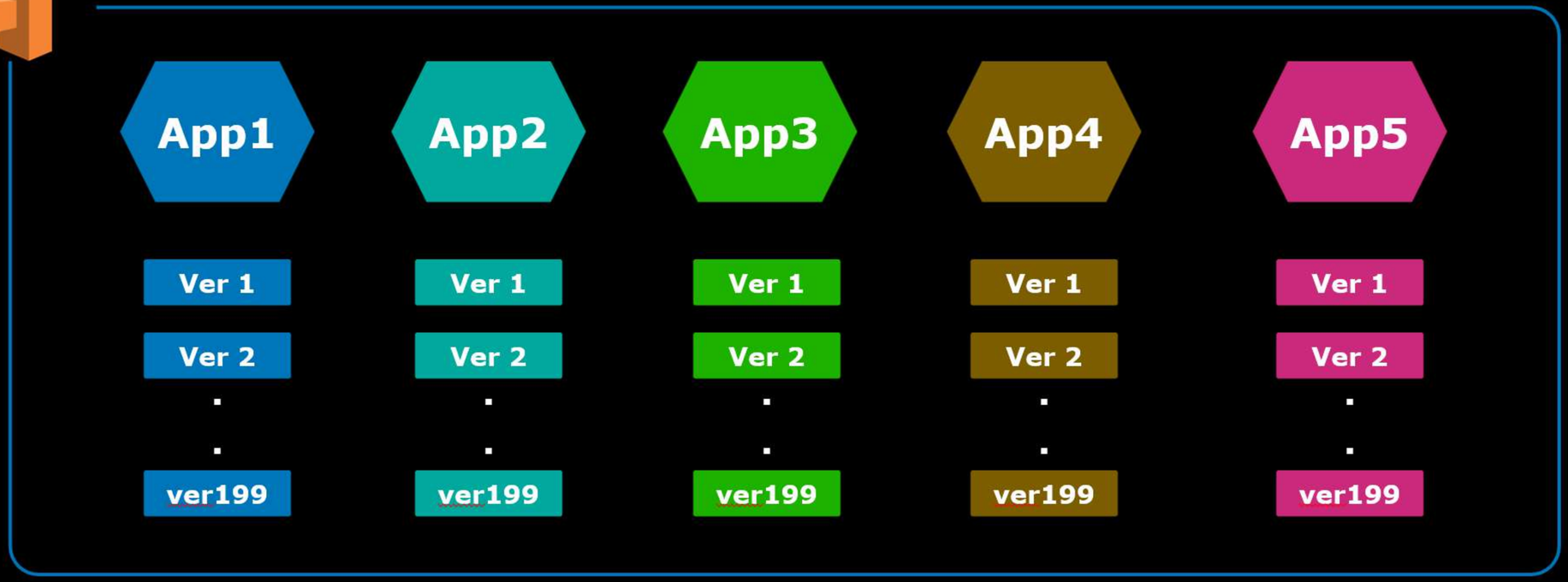


Elastic Beanstalk Lifecycle Policy

- Determines how old versions should stay before Elastic Beanstalk removes them from your account

- Tells Elastic Beanstalk to delete versions if the total number of application versions exceeds a specified number

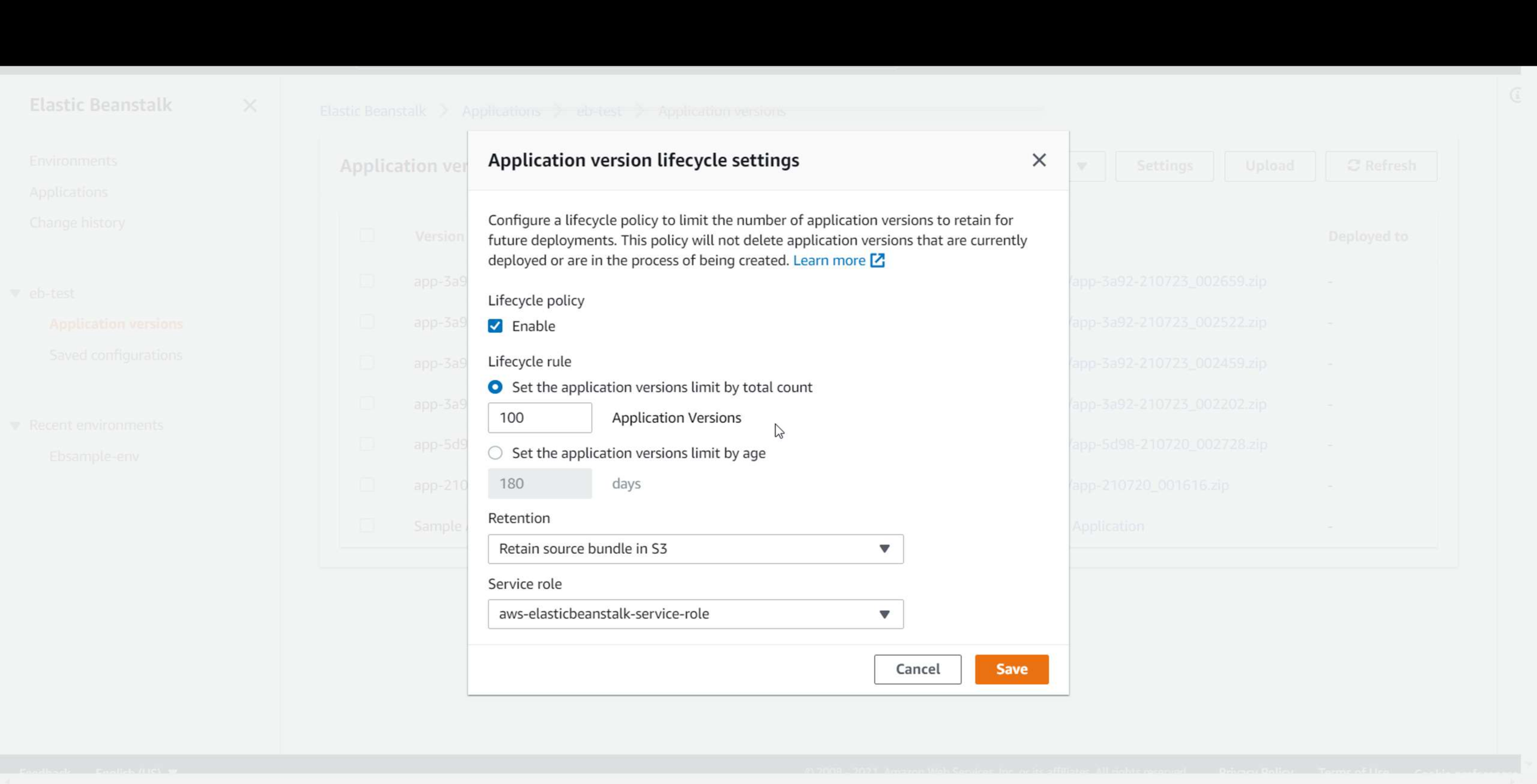
Elastic Beanstalk Applications

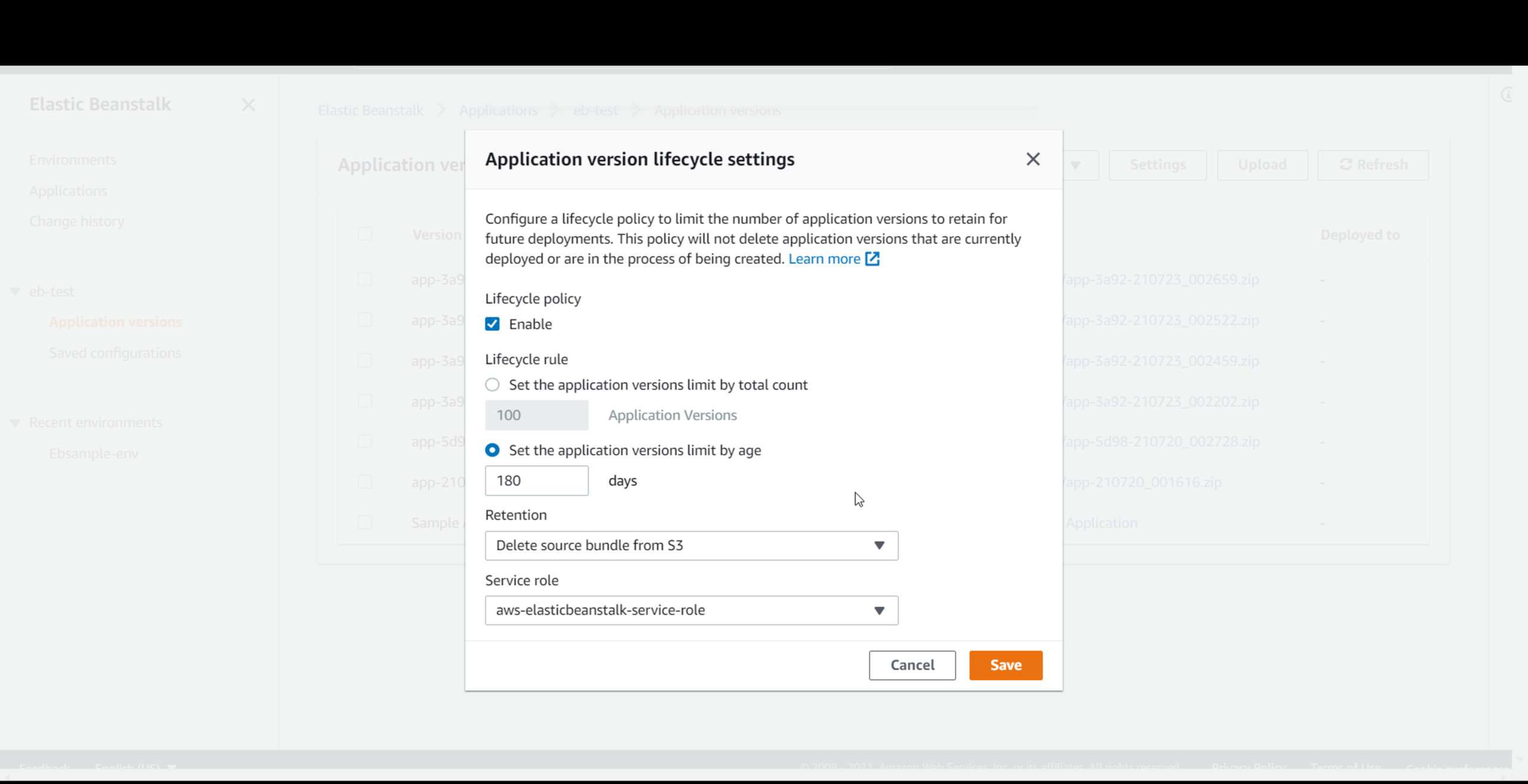


Version quota: 1,000 > Total versions = 995

Elastic Beanstalk Lifecycle Policy

- Determines how old versions should stay before Elastic Beanstalk removes them from your account
- Tells Elastic Beanstalk to delete versions if the total number of application versions exceeds a specified number
- The policy will only apply if the application version creation succeeds
- You must manually delete versions if you've reached the version quota





EB CLI

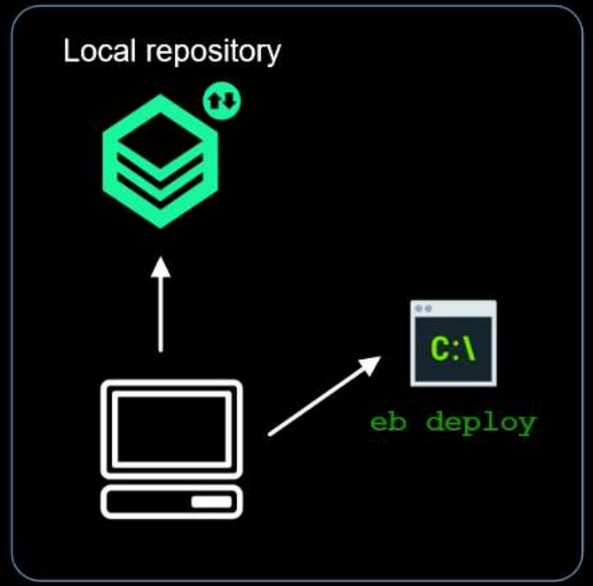
- Elastic Beanstalk's own set of commands that you can use to manage your application environments from a terminal
- EB CLI is different from AWS CLI

EB CLI

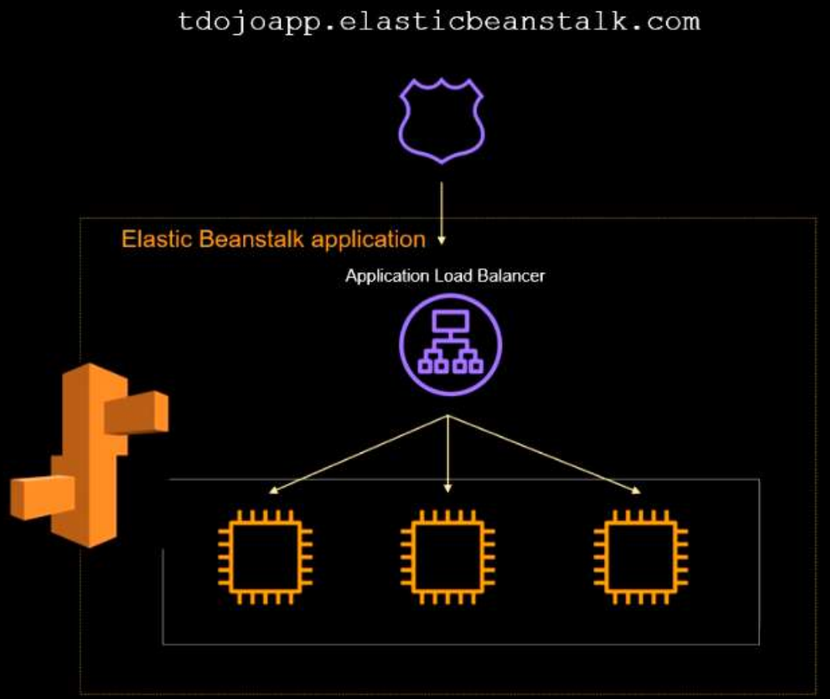
- Elastic Beanstalk's own set of commands that you can use to manage your application environments from a terminal
- EB CLI is different from AWS CLI

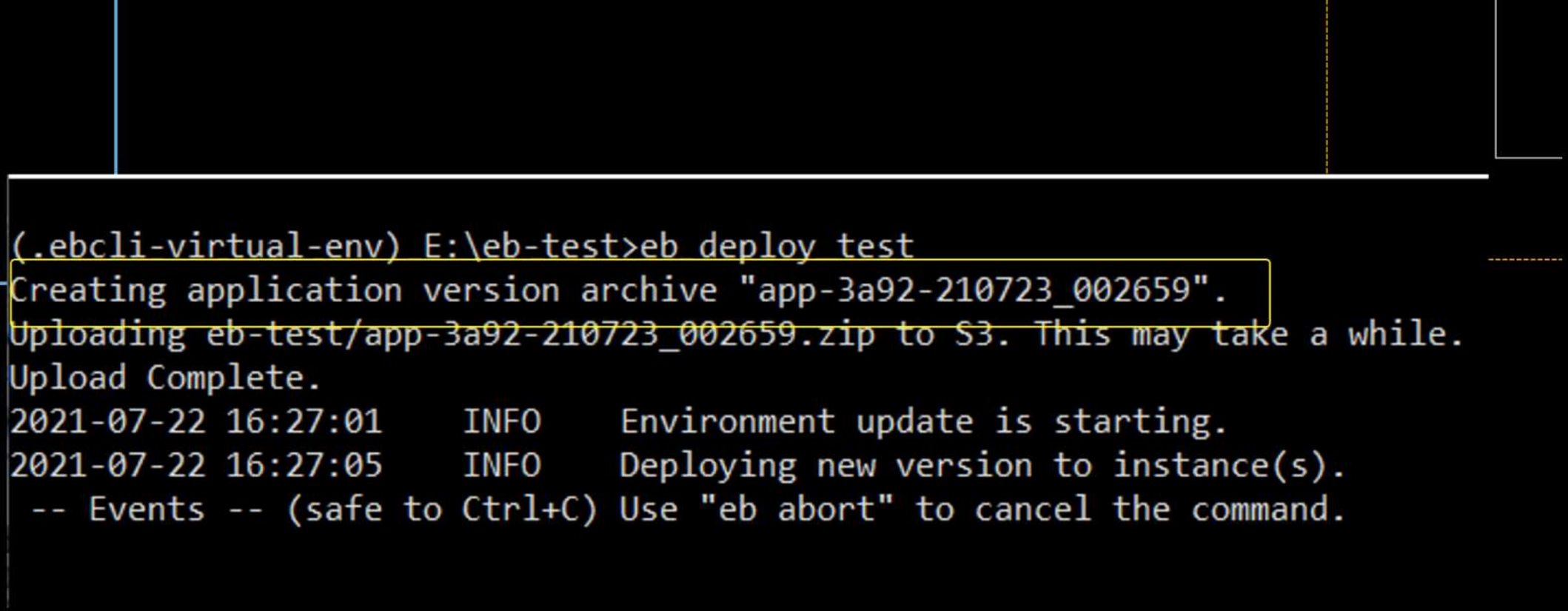
Why use EB CLI over the Elastic Beanstalk Console?

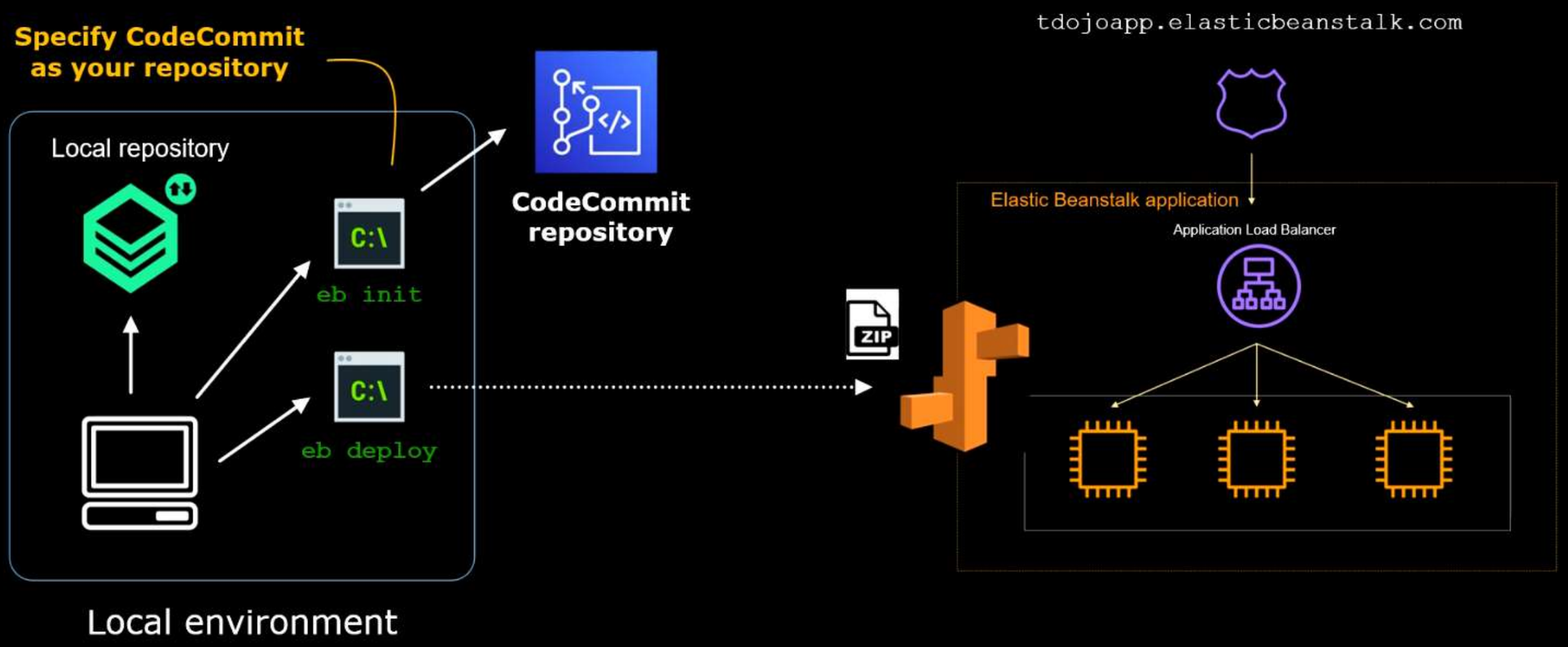
- Simplify creating, updating, and monitoring EB environments from your local environment
- With the EB Console, you'd have to archive and redeploy the entire project



Local environment





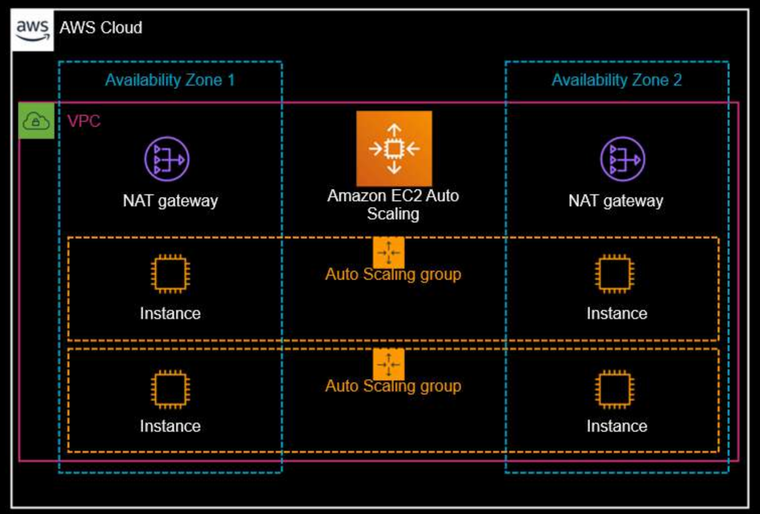


AWS CloudFormation

- Allows you to manage your whole infrastructure using **code**
- Provision AWS resources in an **automated**, **orderly**, and **predictable** fashion

AWS CloudFormation

Manual processes are slow and prone to errors



Tasks

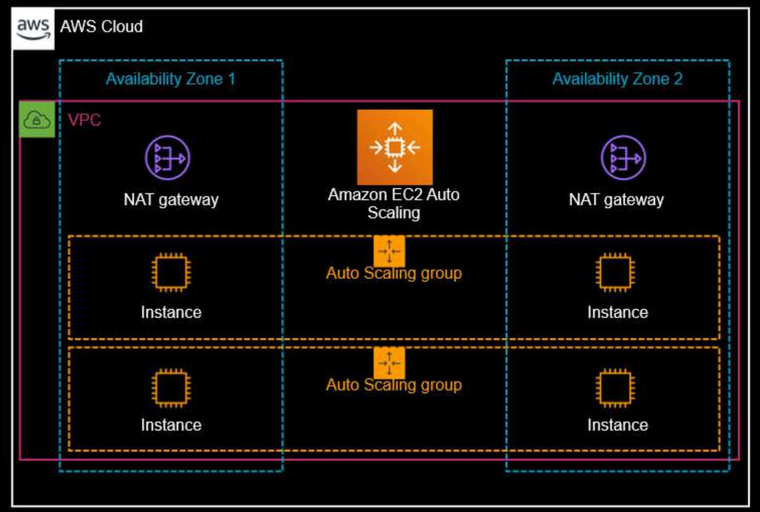
1. Change the instance type
2. Modify the Security groups' inbound rule

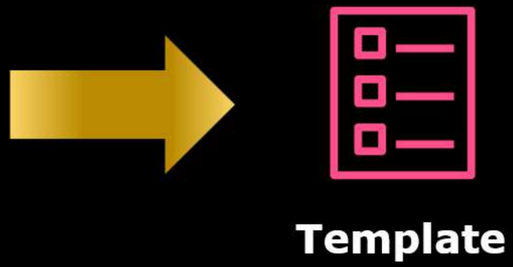
...

10. Update ASG's capacity

AWS CloudFormation is an infrastructure-as-code (IaC) service that helps you:

Simplify infrastructure management

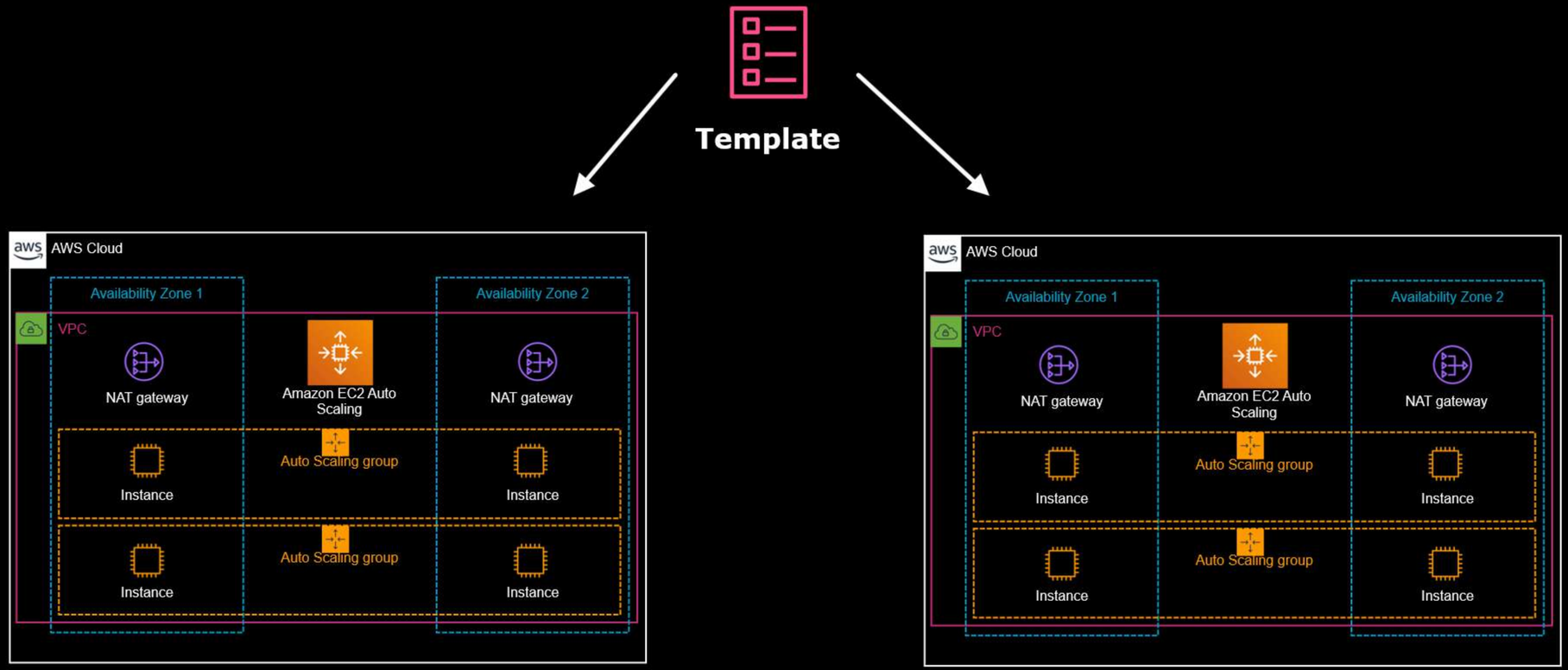




Template

AWS CloudFormation is an infrastructure-as-code (IaC) service that helps you:

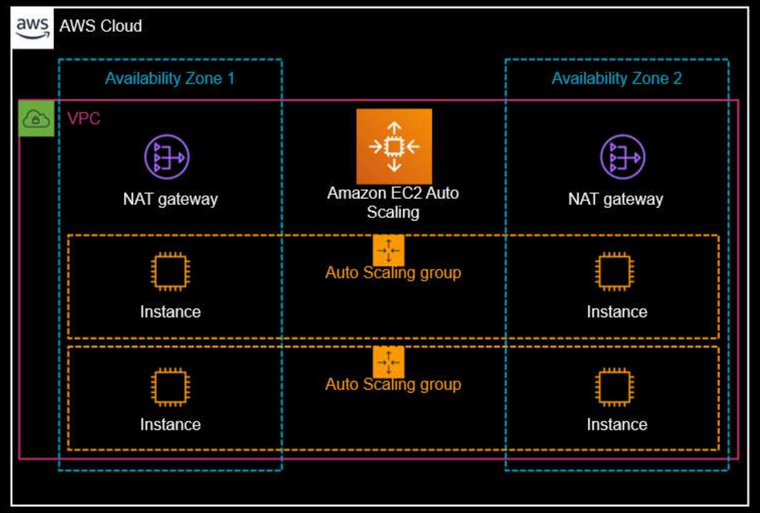
Replicate your infrastructure

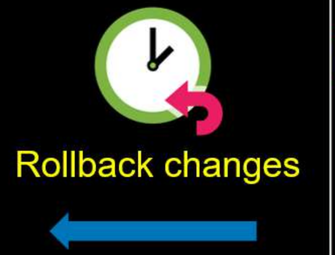


AWS CloudFormation is an infrastructure-as-code (IaC) service that helps you:

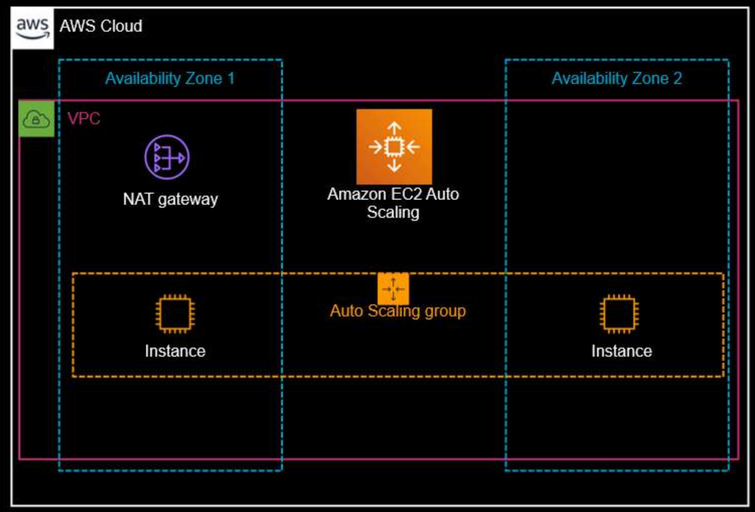
Control and track changes to your infrastructure

Before Update





After Update



AWS Serverless Application Model (SAM)

AWS SAM

- An open-source framework for building serverless applications
- SAM has two parts: SAM template and the SAM CLI

SAM Template

- A SAM template is an extension of a CloudFormation template
- It provides a shorthand syntax for defining serverless resources

CloudFormation template

```yaml
AWSTemplateFormatVersion: 2010-09-09
Resources:
  HelloFunction:
    Type: AWS::Lambda::Function
    Properties:
      ...
  LambdaPermission:
    Type: AWS::Lambda::Permission
    Properties:
      ...
  LambdaExecRole:
    Type: AWS::IAM::Role
    Properties:
      ...
  HelloRestAPI:
    Type: AWS::ApiGateway::RestApi
    Properties:
      ...
  GetMessage:
    Type: AWS::ApiGateway::Method
    Properties:
      ....
  DefaultStage:
    Type: AWS::ApiGateway::Stage
    Properties:
      ...
```

CloudFormation template

```yaml
AWSTemplateFormatVersion: 2010-09-09
Resources:
  HelloFunction:
    Type: AWS::Lambda::Function
    Properties:
      ...
  LambdaPermission:
    Type: AWS::Lambda::Permission
    Properties:
      ...
  LambdaExecRole:
    Type: AWS::IAM::Role
    Properties:
      ...
  HelloRestAPI:
    Type: AWS::ApiGateway::RestApi
    Properties:
      ...
  GetMessage:
    Type: AWS::ApiGateway::Method
    Properties:
      ....
  DefaultStage:
    Type: AWS::ApiGateway::Stage
    Properties:
      ...
```

The number of lines of code required to declare the resources is **greatly reduced!**

SAM template

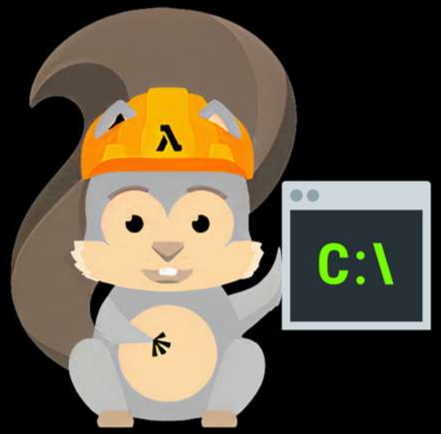
```yaml
AWSTemplateFormatVersion: 2010-09-09
Transform: AWS::Serverless-2016-10-31
Resources:
  HelloWorld:
    Type: AWS::Serverless::Function
    Properties:
      Handler: index.handler
      Runtime: python3.8
      CodeUri: hello_world/
      Events:
        ApiEvent:
          Type: Api
          Properties:
            Path: /path
            Method: get
```

SAM template

```yaml
AWSTemplateFormatVersion: 2010-09-09
Transform: AWS::Serverless-2016-10-31
Resources:
  HelloWorld:
    Type: AWS::Serverless::Function
    Properties:
      Handler: index.handler
      Runtime: python3.8
      CodeUri: hello_world/
      Events:
        ApiEvent:
          Type: Api
          Properties:
            Path: /path
            Method: get
```

AWS::Serverless transform macro

- You must declare the AWS::Serverless transform macro in the Transform section to use SAM resources
- A macro is a command that automatically expands to multiple simpler commands
- Before deploying, a SAM template is transformed and expanded to a native CloudFormation template
- SAM is specifically designed for serverless applications
- You can use the native CloudFormation resources, intrinsic functions, and pseudo parameters in a SAM template



SAM CLI

- A set of CLI commands that you can use to build, test, and debug serverless applications

AWS CodeCommit

AWS CodeCommit

- A fully-managed source control service that hosts secure Git-based repositories, similar to GitHub.

Other code repositories...

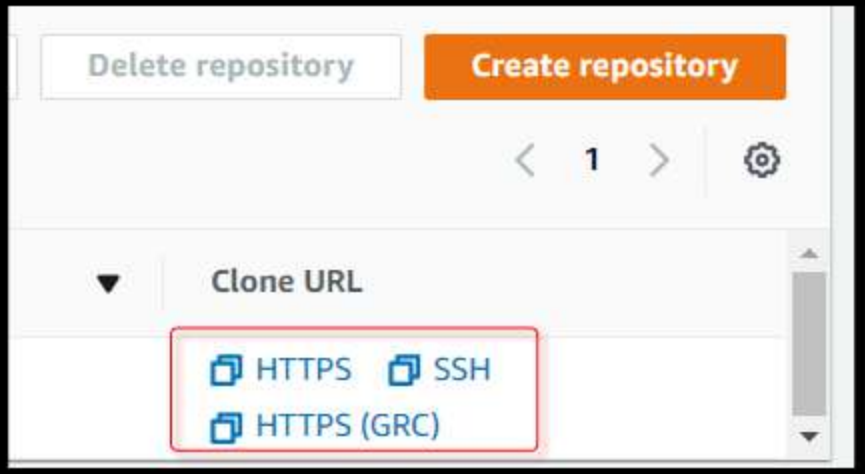
GitHub

Bitbucket

GitLab

AWS CodeCommit

- A fully-managed source control service that hosts secure Git-based repositories, similar to GitHub.
- You can transfer your files to and from AWS CodeCommit using HTTPS or SSH



AWS CodeCommit

- A fully-managed source control service that hosts secure Git-based repositories, similar to GitHub.
- You can transfer your files to and from AWS CodeCommit using HTTPS or SSH
- CodeCommit offers unlimited repositories

Under the hood, CodeCommit uses Amazon S3 and DynamoDB

AWS CodeCommit

- A fully-managed source control service that hosts secure Git-based repositories, similar to GitHub.
- You can transfer your files to and from AWS CodeCommit using HTTPS or SSH
- CodeCommit offers unlimited repositories
- Repository data are highly available and durable

Integrations with other AWS services will be a lot easier and more convenient for you as a developer.

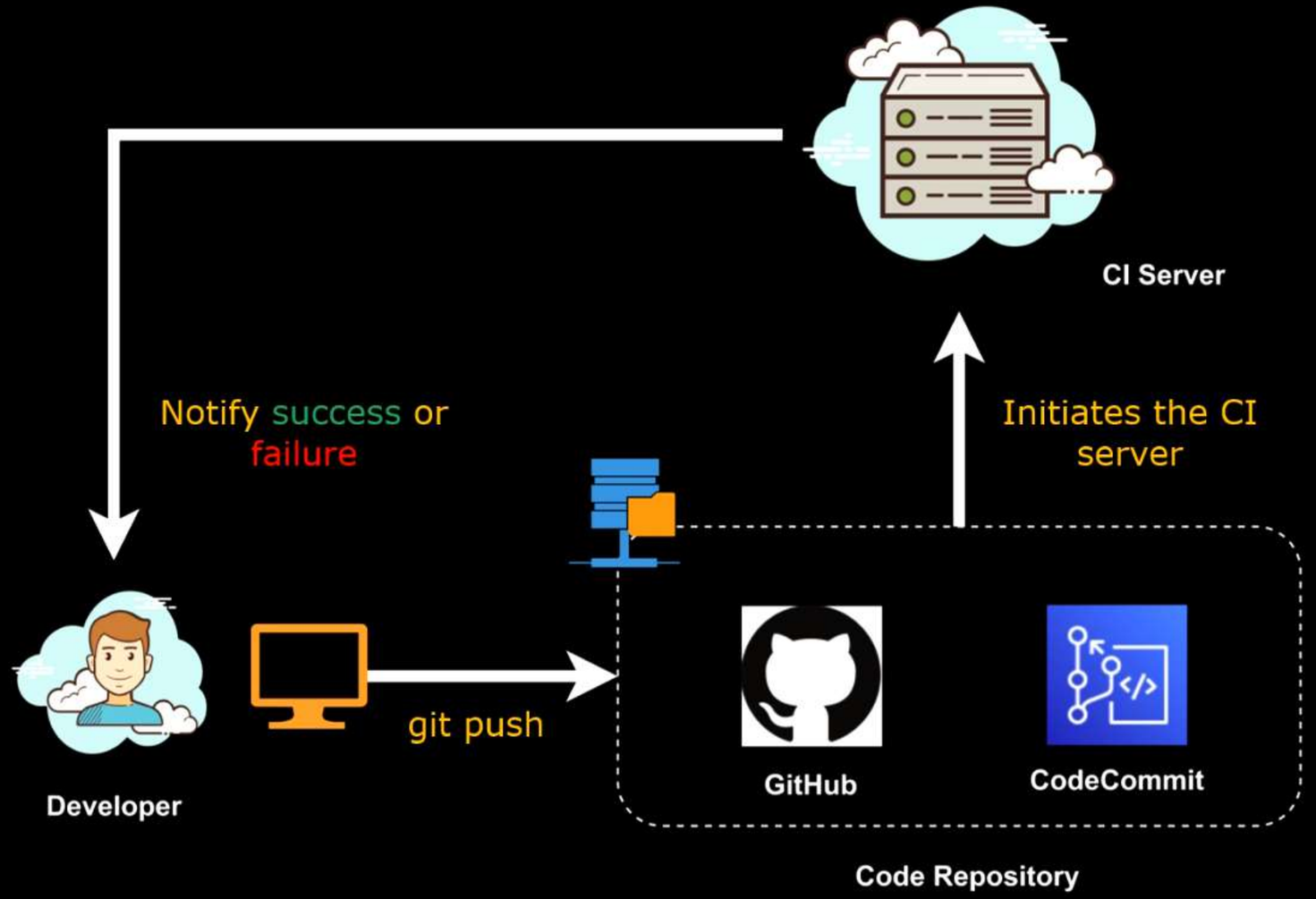
AWS CodeBuild

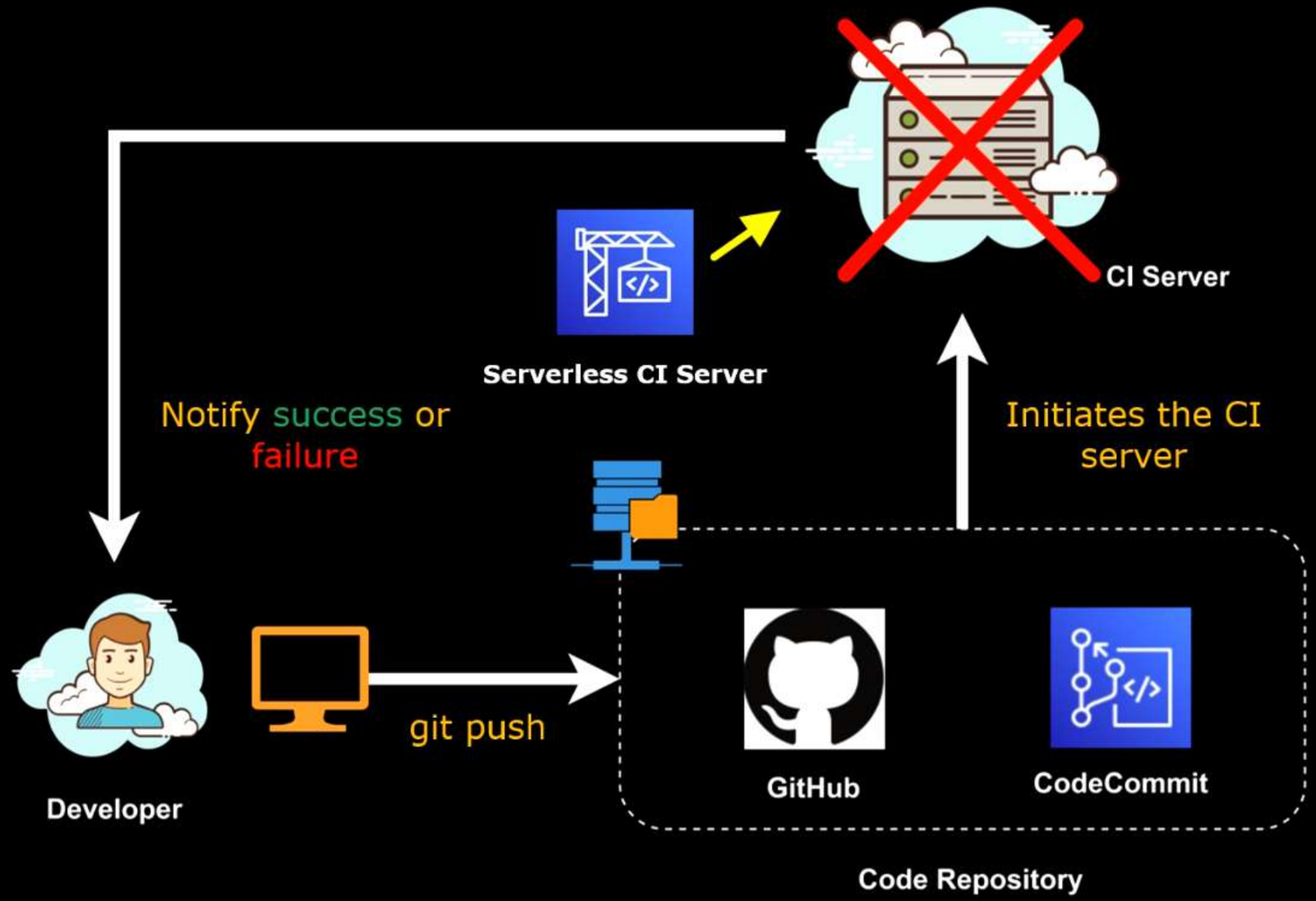
AWS CodeBuild

- A fully managed continuous integration service
- Compiles source code, runs tests, and produces a build artifact of that code that is ready to deploy

- Continuous integration is a practice in software development that allows multiple developers to integrate frequent code changes into a shared repository without running into "merge hell"

- Continuous integration allows you to detect those errors before they are merged with other codes.





AWS CodeBuild

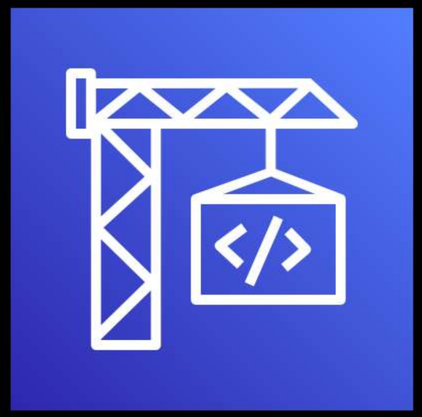
- CodeBuild eliminates the need to set up, patch, update, and manage your own build servers and software
- It provides prepackaged build environments as Docker images
- A build environment is the combination of the operating system, programming language, and tools used by CodeBuild to run a build

Operating system: Linux

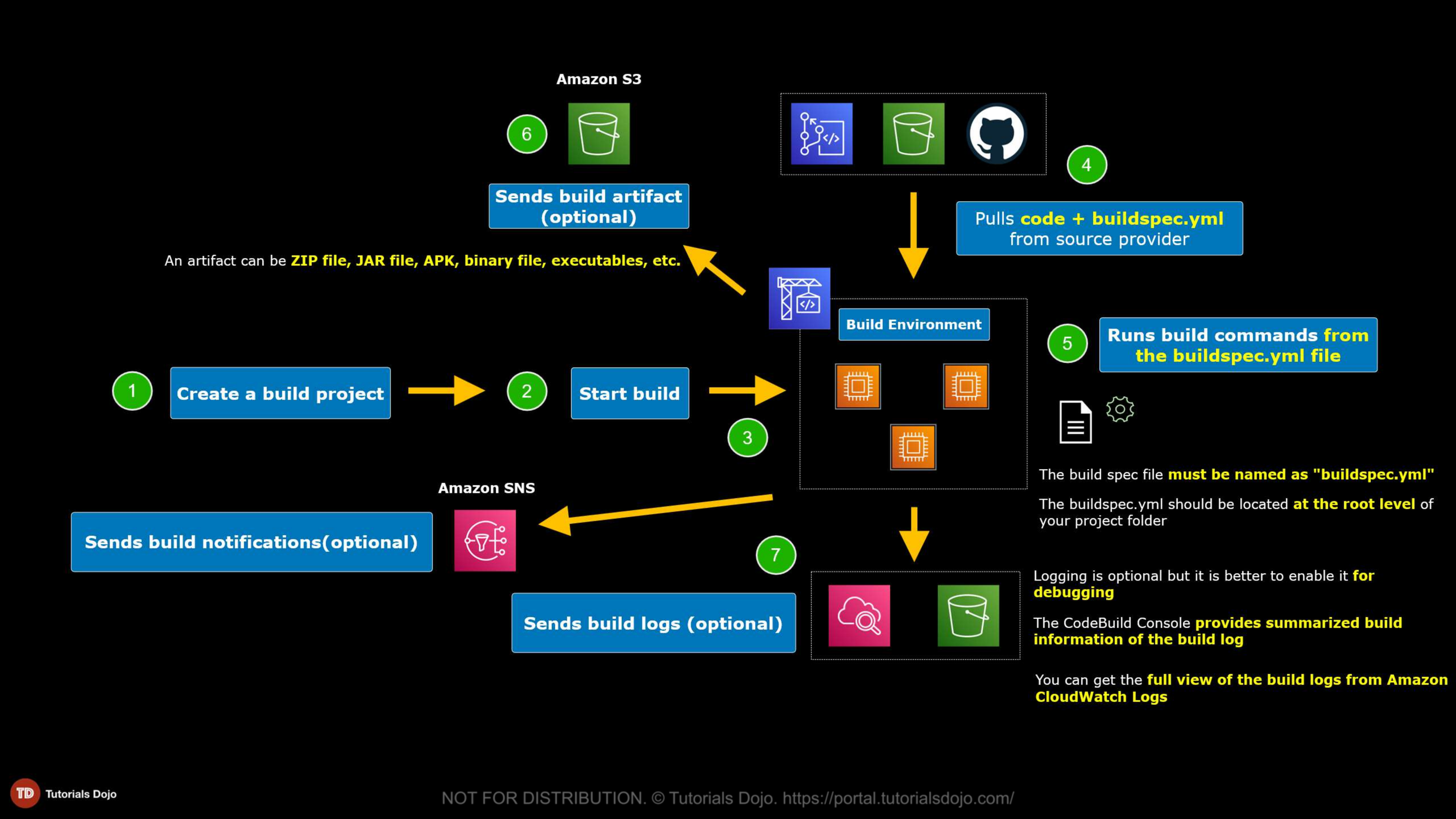
Compute type	computeType value	Memory	vCPUs	Disk space	Environment type
build.general1.small	BUILD_GENERAL1_SMALL	3 GB	2	64 GB	LINUX_CONTAINER
build.general1.medium	BUILD_GENERAL1_MEDIUM	7 GB	4	128 GB	LINUX_CONTAINER
build.general1.large	BUILD_GENERAL1_LARGE	15 GB	8	128 GB	LINUX_CONTAINER
build.general1.large	BUILD_GENERAL1_LARGE	255 GB	32	50 GB	LINUX_GPU_CONTAINER
build.general1.large	BUILD_GENERAL1_LARGE	16 GB	8	50 GB	ARM_CONTAINER
build.general1.2xlarge	BUILD_GENERAL1_2XLARGE	145 GB	72	824 GB (SSD)	LINUX_CONTAINER

AWS CodeBuild

- CodeBuild eliminates the need to set up, patch, update, and manage your own build servers and software
- It provides prepackaged build environments as Docker images
- A build environment is the combination of the operating system, programming language, and tools used by CodeBuild to run a build
- You can use the AWS CodeBuild agent to build and test applications locally
- CodeBuild automatically scales up and down according to build requests
- You only pay for the time it takes to complete your build



AWS CodeBuild Workflow



AWS CodeDeploy

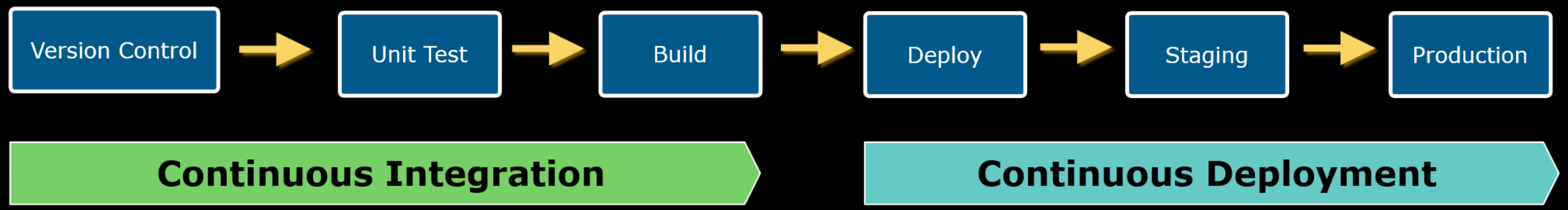
AWS CodeBuild

Continuous **integration** service

AWS CodeDeploy

Continuous **deployment** service

- Continuous deployment is a software development strategy wherein code changes that pass automated tests from a build service like CodeBuild are deployed automatically into an application environment



AWS CodeDeploy

- A **fully managed deployment service** that automates software deployments
  - You can release new application versions **without downtime**
- It supports deployments to **Amazon EC2**, **AWS Lambda**, **Amazon ECS**, **AWS Fargate, and on-premises servers**
- You need to install the **AWS CodeDeploy agent for EC2/On-premises deployments**

AWS CodeDeploy

- You can release new application versions **without downtime**
- It offers **two deployment options**:
  1. **In-place** deployment
  2. **Blue**/**Green** deployment
- It also supports deployment of different application assets such as **Lambda functions**, **configuration files**, **executables**, **packages**, **scripts**, or even **multimedia files**

AWS CodeDeploy

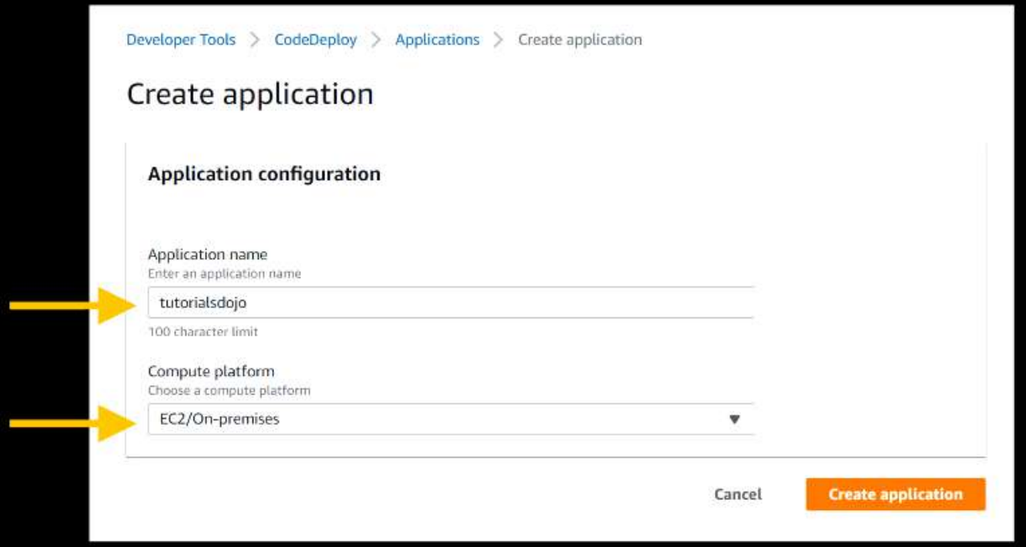
- CodeDeploy is **platform-agnostic**

- It can perform **automatic rollbacks**
- If you prefer, you can also perform rollbacks **manually**

AWS CodeDeploy Primary Components

- **Application**
  - identifies the **application version** that you want to deploy
  - must have a **unique name**

CodeDeploy: Primary Components



CodeDeploy: Primary Components

- **Application**
  - identifies the **application version** that you want to deploy
  - must have a **unique name**
- **Deployment group**
  - refers to the **set of instances or Lambda functions** where you deploy the code revision
  - You can **create multiple deployment groups** within an Application

CodeDeploy: Primary Components

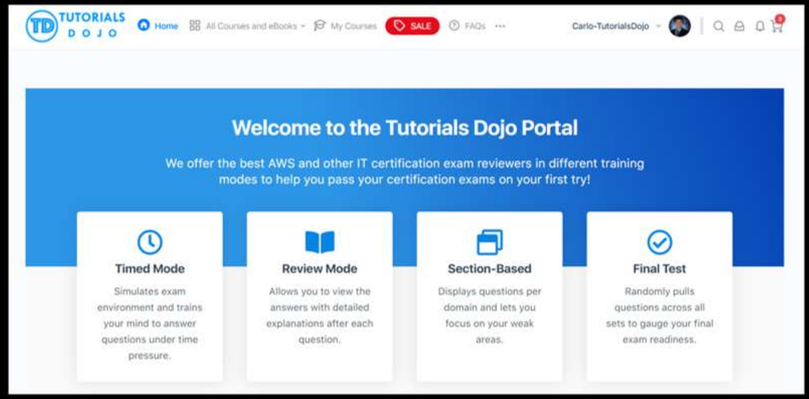
- **Application**
  - identifies the **application version** that you want to deploy
  - must have a **unique name**
- **Deployment group**
  - refers to the **set of instances or Lambda functions** where you deploy the code revision
  - You can **create multiple deployment groups** within an Application
- **Deployment configuration**
  - **set of conditions and deployment rules** that CodeDeploy applies during a deployment
  - deployment configurations **vary for the compute type** that you're using
- **Application Specification (AppSpec) file**
  - written in **YAML or JSON**
  - manages deployment stages as **lifecycle event hooks**

AWS CodeDeploy

In-place vs Blue/Green Deployment

In-Place Deployment

ver. 2 + AppSpec = Archive file



Must be located at the **root level** of your application folder

ver. 1 ver. 1 ver. 1

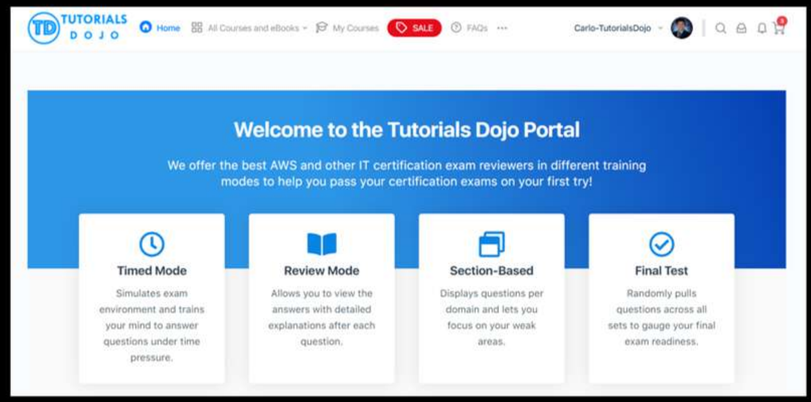
In-Place Deployment

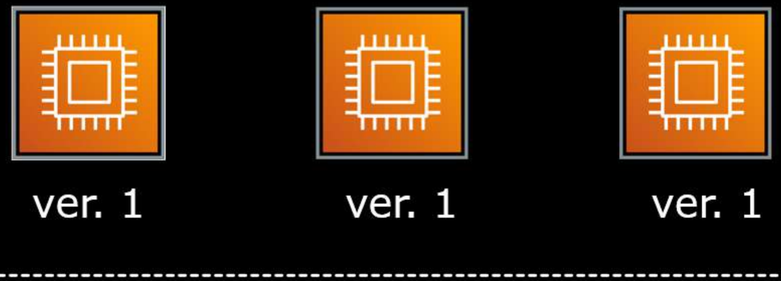
GitHub

BitBucket

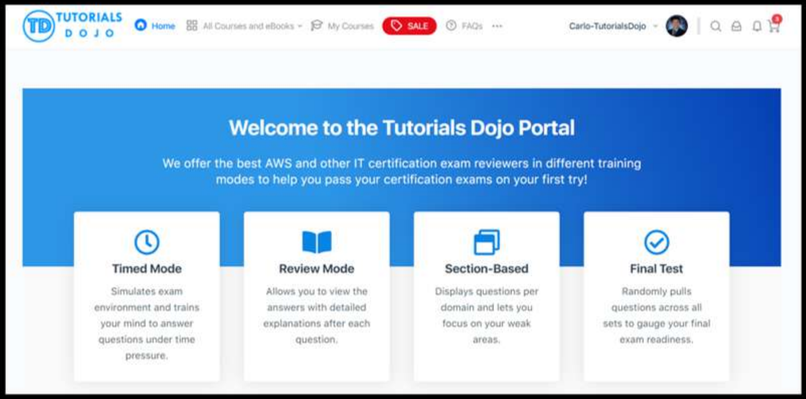
Amazon S3

For **EC2/On-premises** deployments only





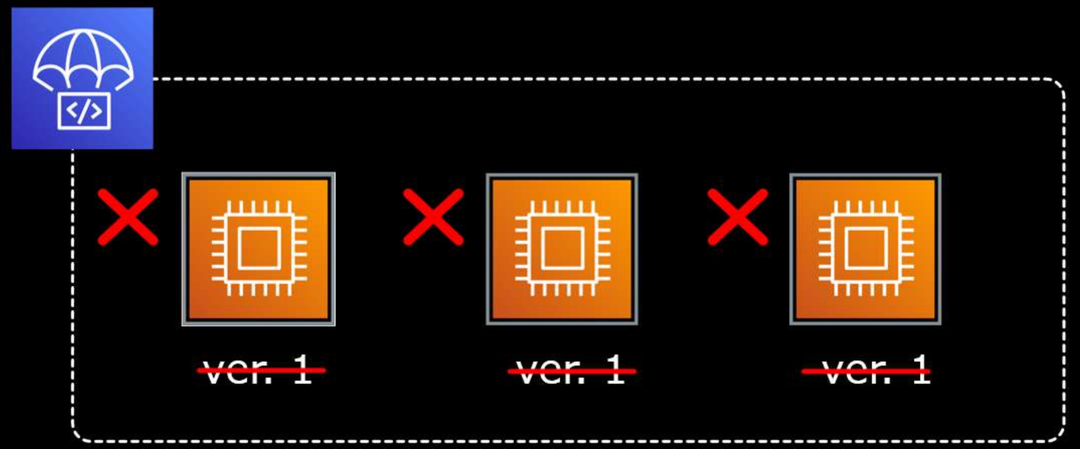
In-Place Deployment



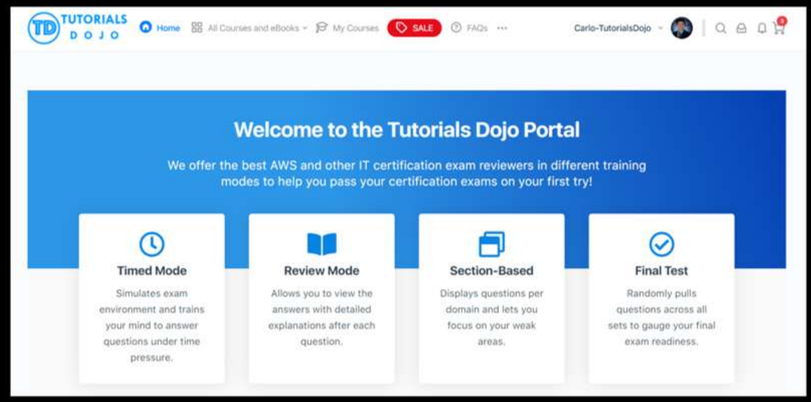
GitHub

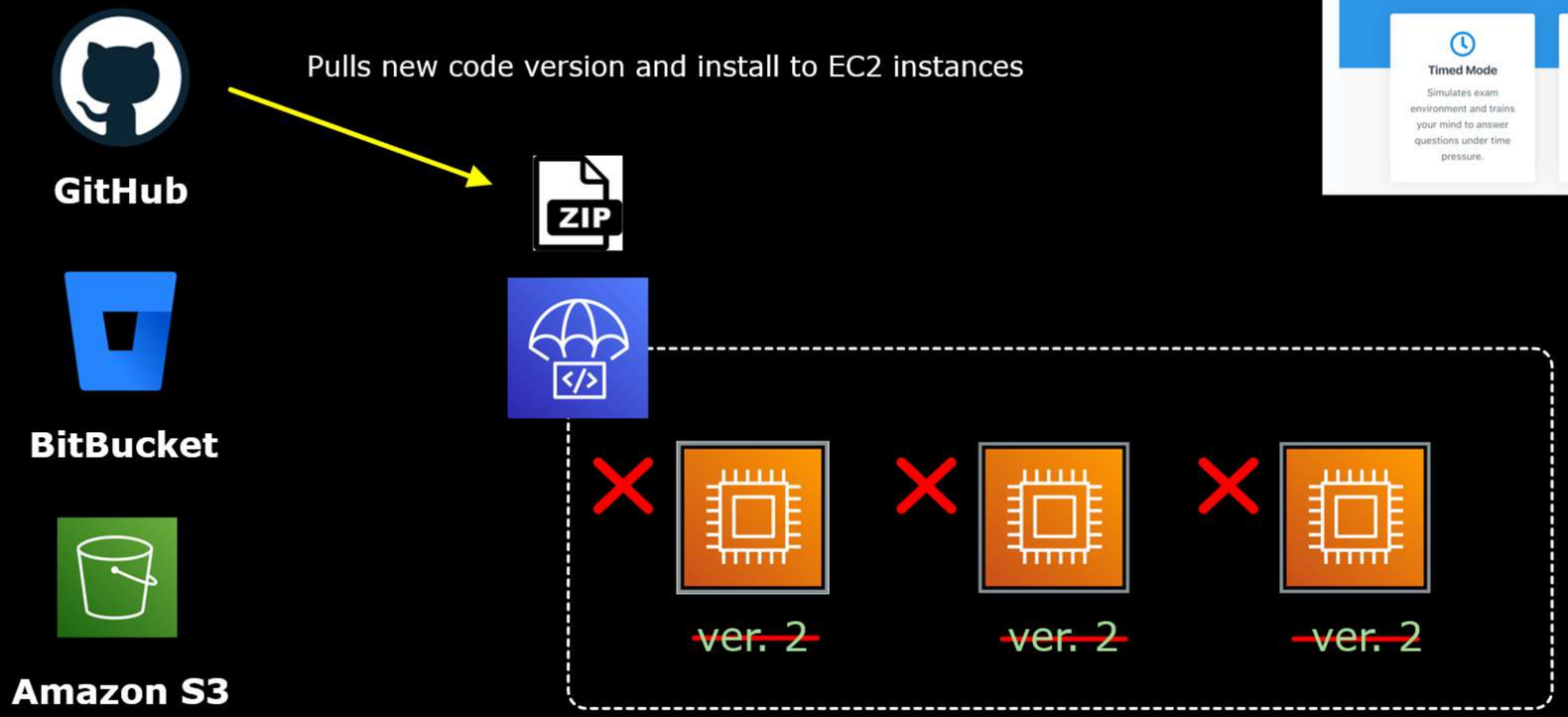
BitBucket

Amazon S3



In-Place Deployment





In-Place Deployment

Only EC2/On-premises deployments can use In-place deployment

Cons

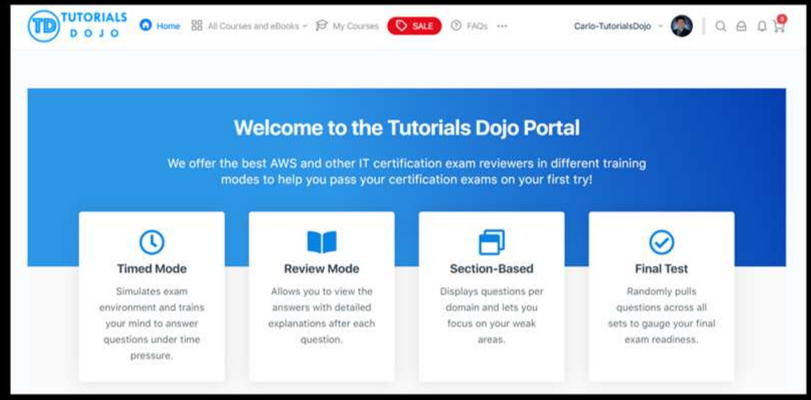
There might be some **downtime**

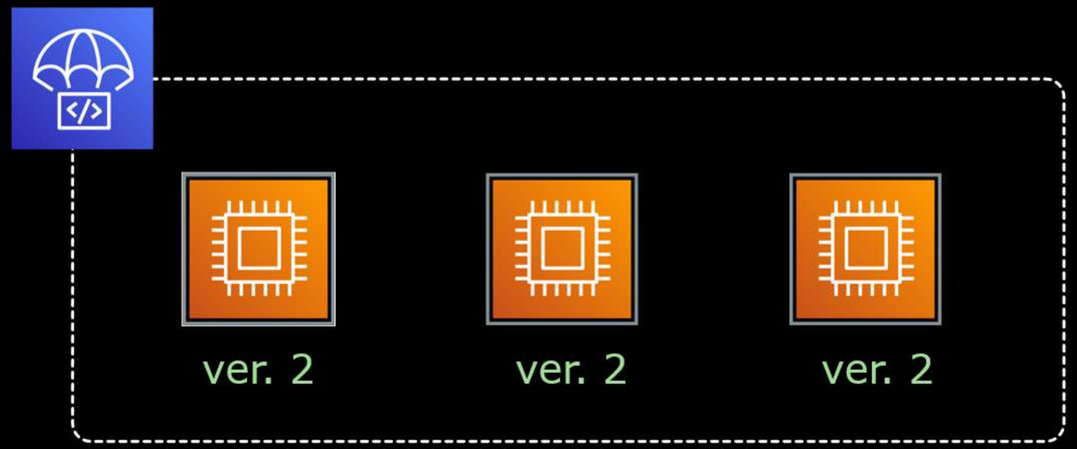
Slow rollback compared to blue/green deployment

Pros

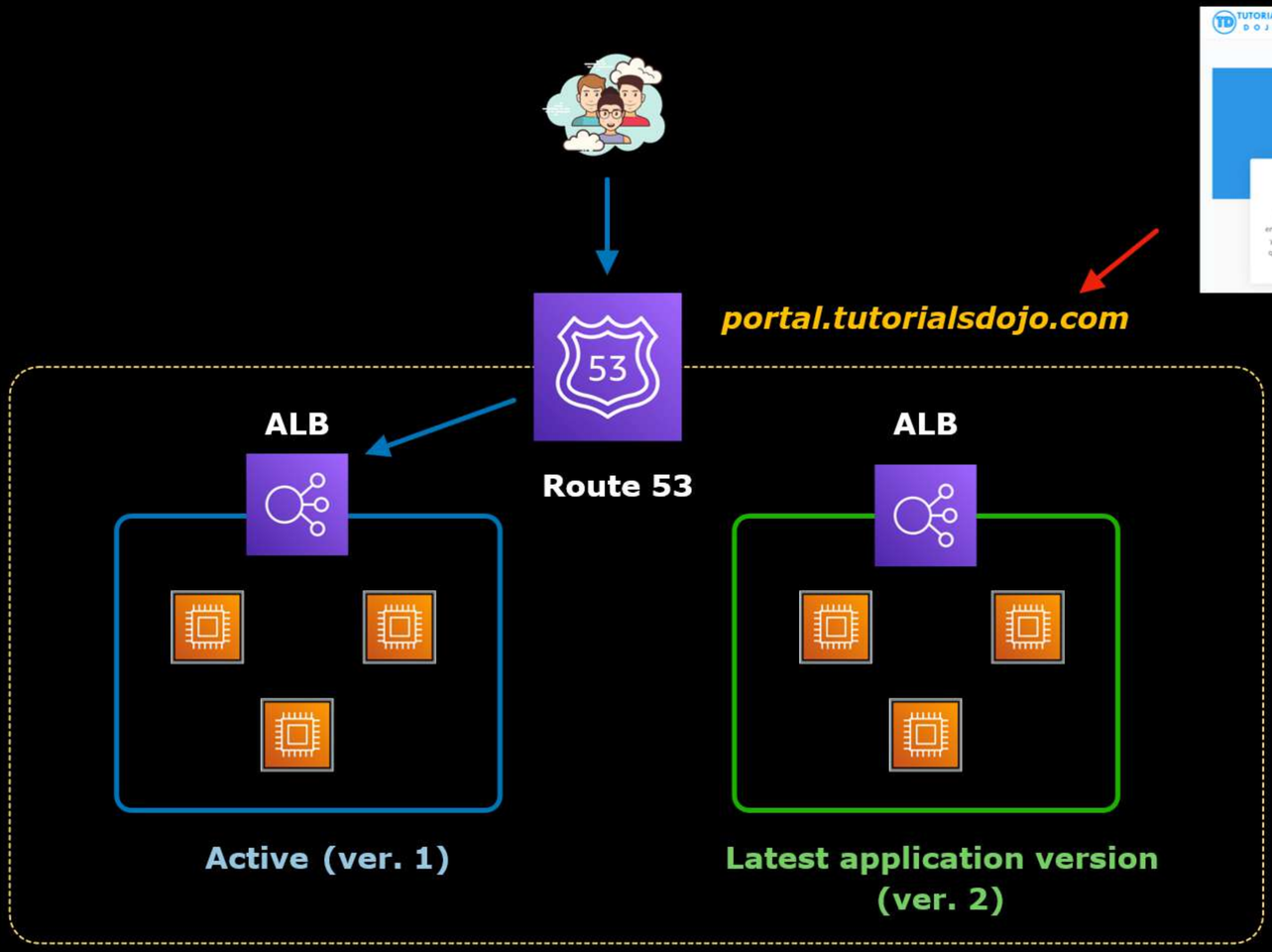
Cheap implementation cost

You're just maintaining a **single environment**

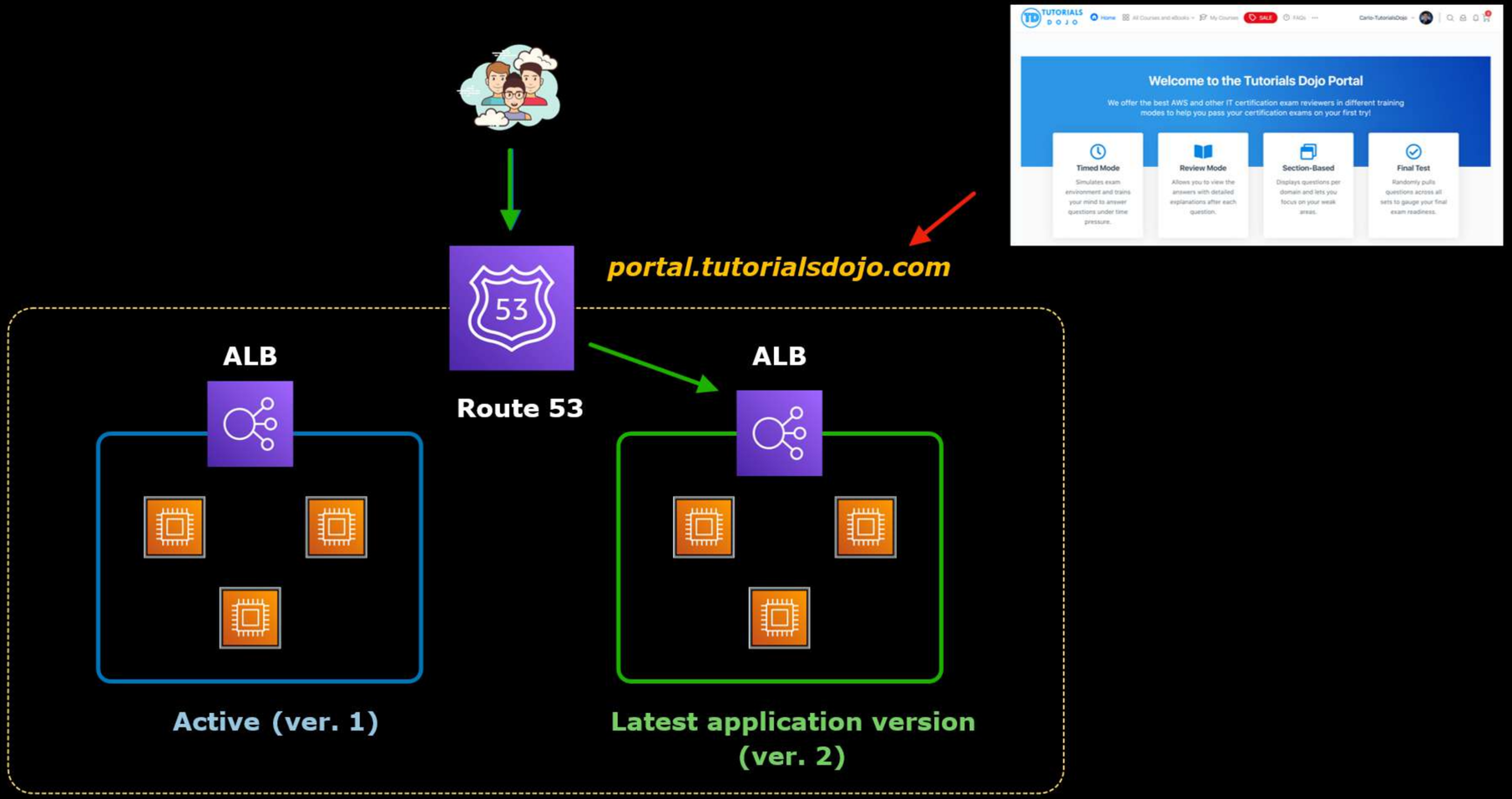




Blue/Green Deployment



Blue/Green Deployment



Blue/Green Deployment

Pros

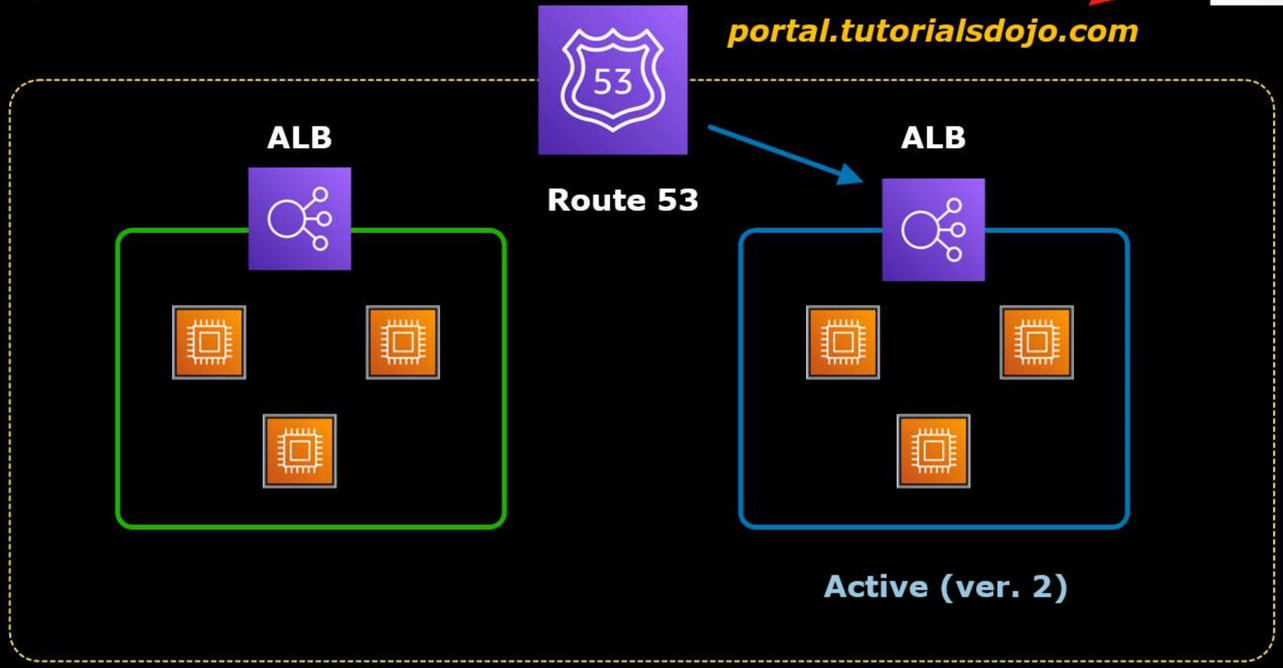
Switching to new application version has **no downtime**

Faster rollback since switching is done at the network level

You can **perform tests** on the green environment without affecting the blue environment

Cons

Expensive implementation cost

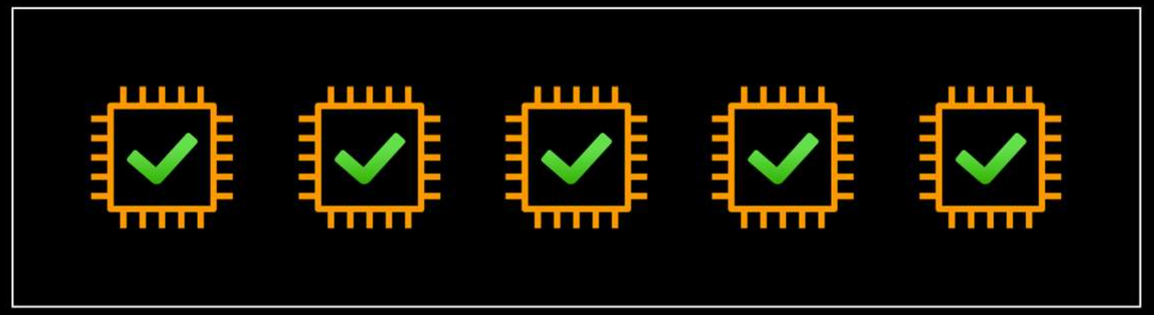


AWS CodeDeploy Deployment Configuration

All at once

- Deploys application to all instances at the same time

Deployment configurations



All at once

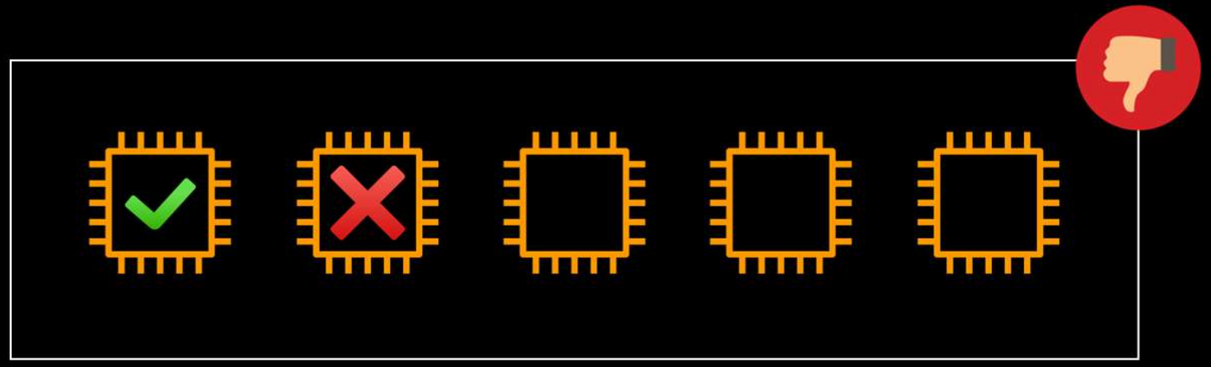
Deployment configurations

- Deploys application to all instances at the same time
- The application must be deployed to at least one instance for the deployment to be successful
- Not recommended for production
- Typically used for quickly pushing changes to test environments

One at a time

Deployment configurations

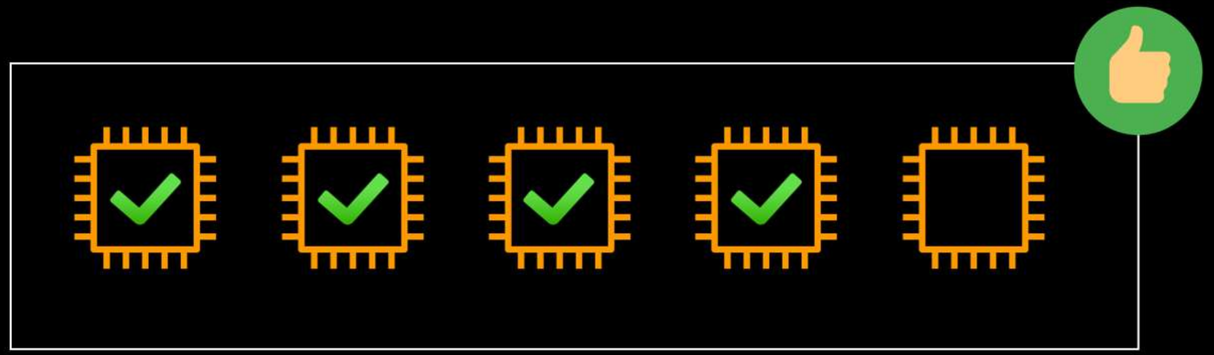
- The application is deployed to each instance one at a time
- Deployment is successful:
  - if the new application version is deployed to all of the instances
  - even if deployment to the final instance fails



Half at a time

Deployment configurations

- The application is deployed to at least 50% of your instances at once (with fractions rounded down)
- Deployment succeeds if the application revision is deployed to at least half of the instances (with fractions rounded up)



Deployment configurations

Canary

- Only applicable to a Blue/Green deployment type
- Traffic is shifted in two increments
- You specify how much traffic will be shifted to the green environment at first increment
- You specify how many minutes CodeDeploy waits before shifting the remaining traffic

Example:

10% of traffic is shifted at the first increment

`LambdaCanary10Percent5Minutes`

90% of traffic is shifted after 5 minutes of the first increment

Deployment configurations

Linear

- Only applicable to a Blue/Green deployment type
- Traffic is shifted in equal increments

Example:

```
CodeDeployDefault.LambdaLinear10PercentEvery1Minute
```

Deployment configurations

- Only All-at-once, Canary, and Linear deployment configurations an be applied to AWS Lambda and ECS deployments

- EC2 and on-premises deployments supports all deployment configurations

AWS CodeDeploy Lifecycle event hooks

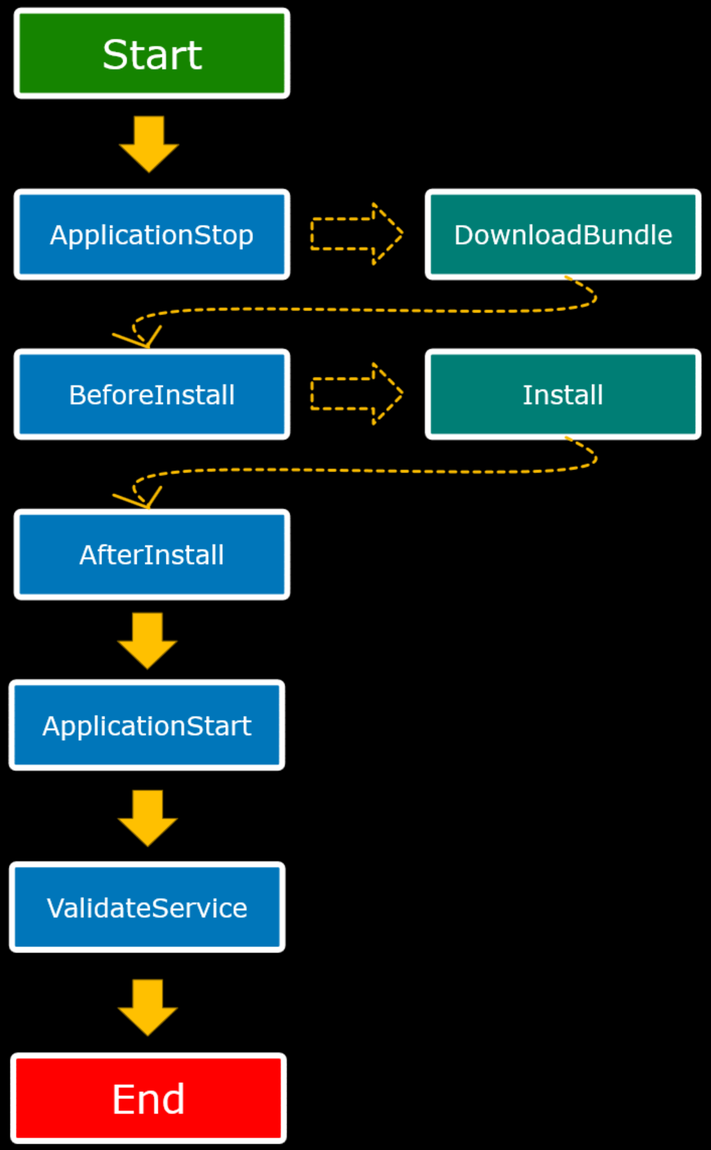
Lifecycle event hooks

- The **hooks section** contains a set of logical groupings called **lifecycle event hooks**
- You can **define one or more scripts** on a lifecycle event hook
- Lifecycle event hooks are **executed in sequence**

Example of an appspec file

```yaml
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/tutorialsdojo

hooks:
  ApplicationStop:
    - location: scripts/deactivate_server.sh
  BeforeInstall:
    - location: scripts/backup.sh
  AfterInstall:
    - location: scripts/decrypt_mysql_credentials.sh
  ApplicationStart:
    - location: scripts/start_service.sh
  ValidateService:
    - location: scripts/test_portal.sh
```

Lifecycle event hooks



1. **Start** - initiates the deployment process
2. **ApplicationStop** - use this event to deactivate the old application
3. **DownloadBundle** - pulls the new application bundle onto the instance
4. **BeforeInstall** - you can use this event hook for installing dependencies or backing up log files before installing the new application
5. **Install** - copies new application files into the folder that you specified
6. **AfterInstall** - you can use this event hook for decrypting data such as database credentials or application secrets
7. **ApplicationStart** - use this event to restart your new application back into service
8. **ValidateService** - use this event to test if your application is behaving as expected
9. **End** - terminates the deployment process

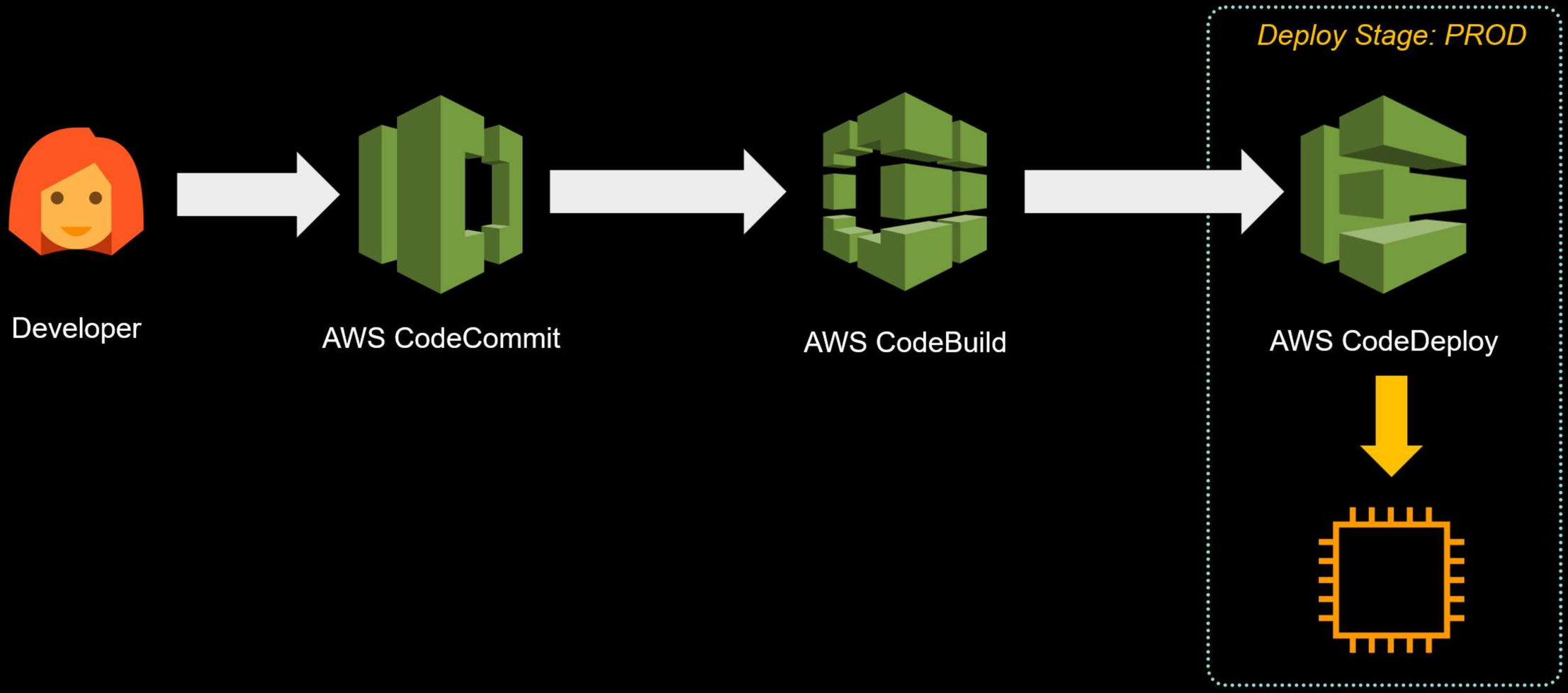
AWS CodePipeline

C ontinuous I ntegration

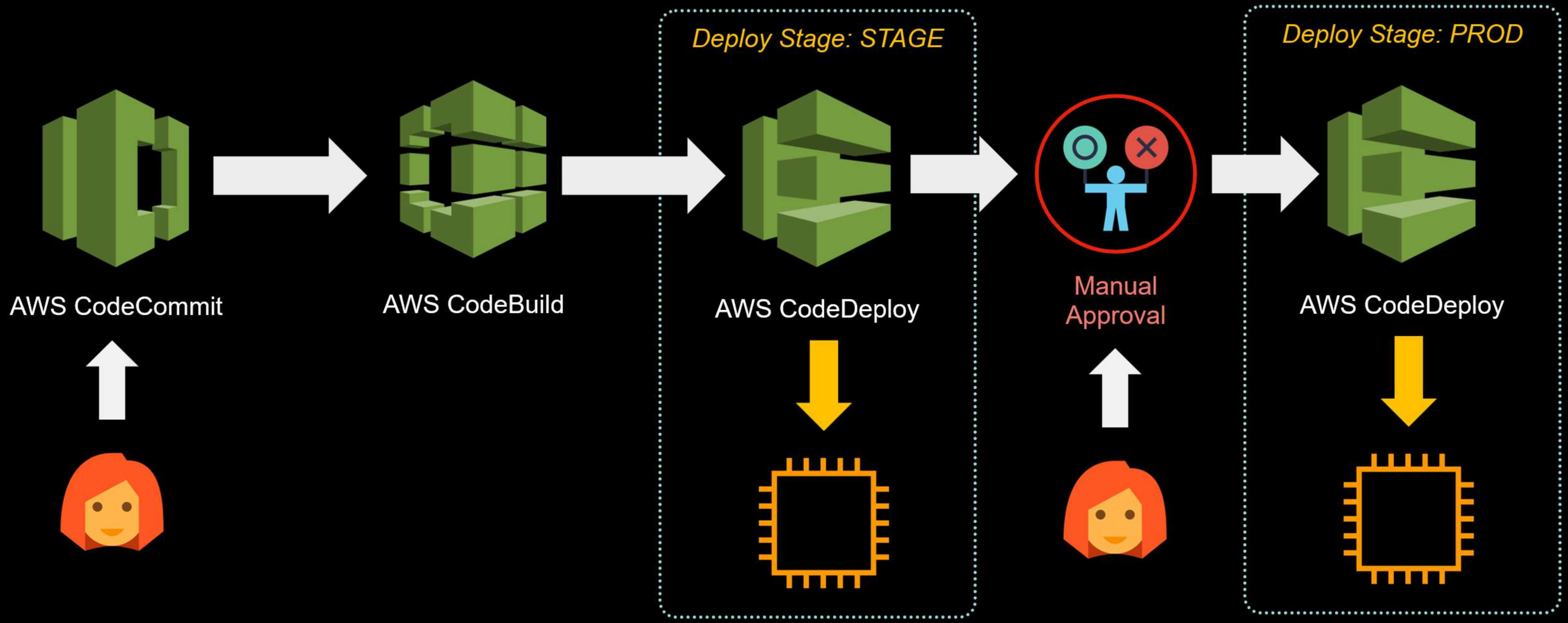
CD → Continuous Deployment

CD → Continuous Delivery

Continuous Deployment

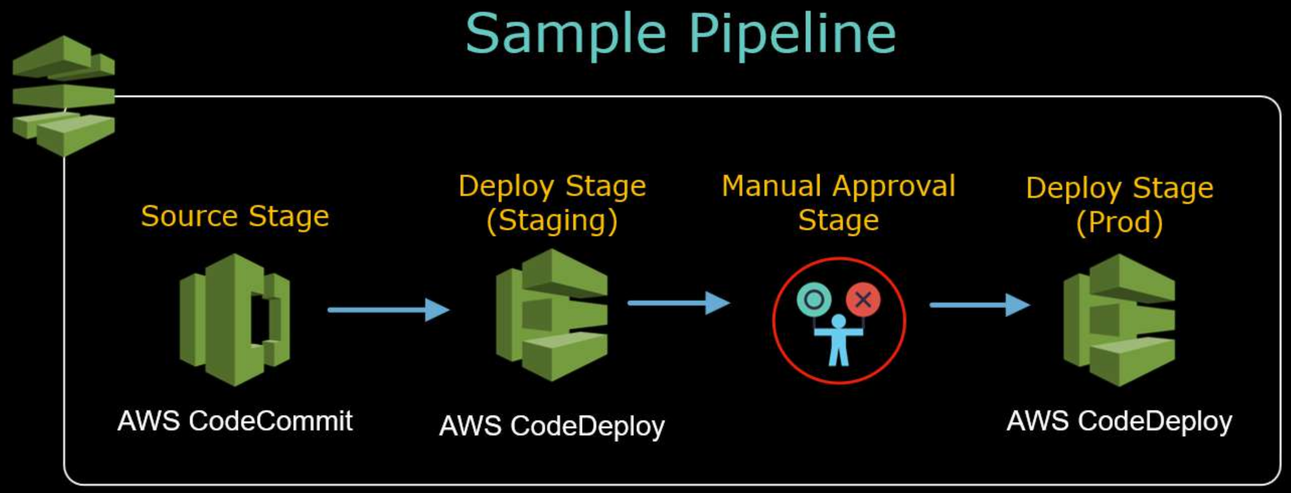


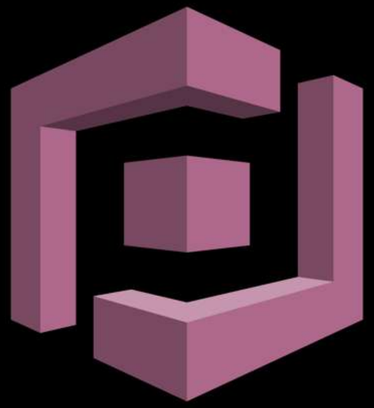
Continuous Delivery



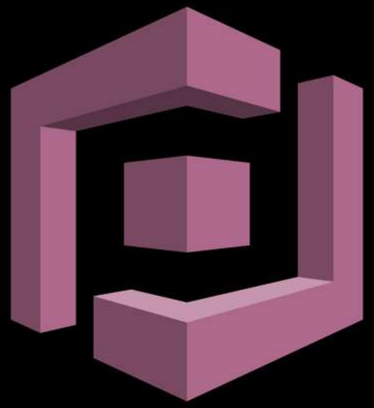
AWS
CodePipeline

- A fully managed continuous delivery service that can help you easily build your own CI/CD pipeline



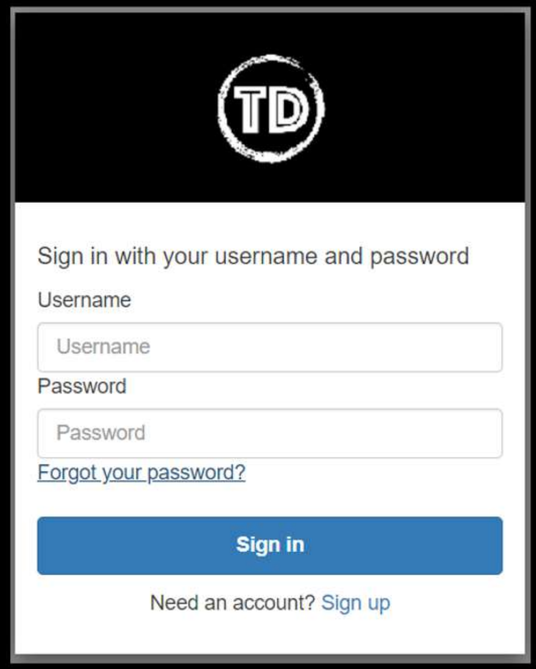


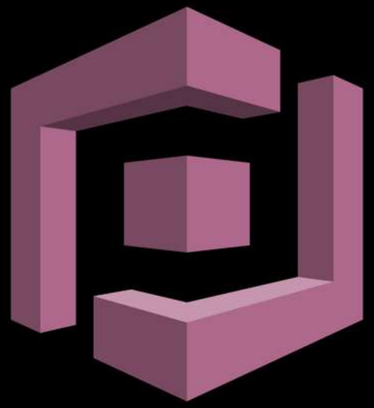
Amazon Cognito User Pool



Cognito User Pool

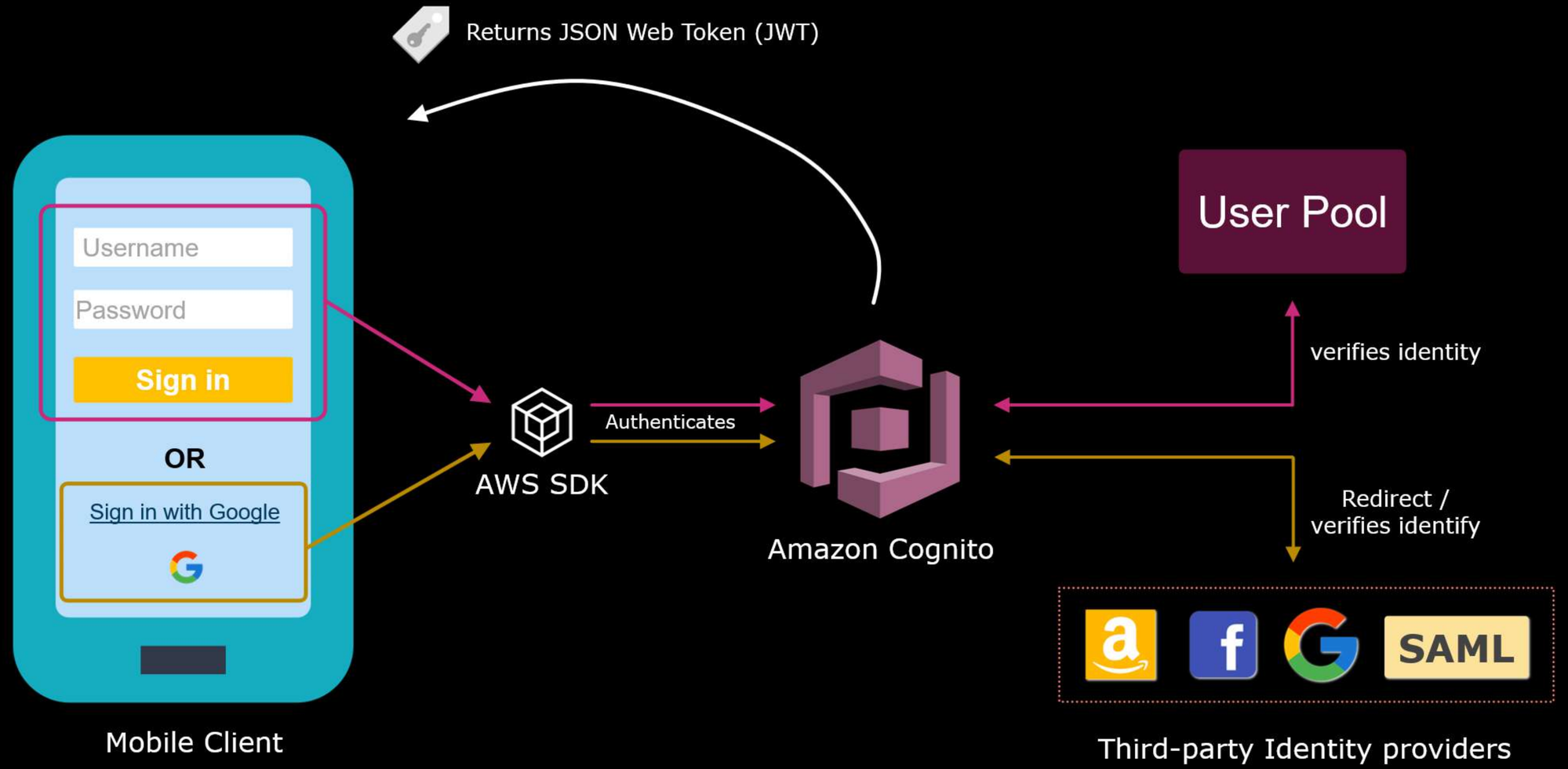
- Managed user directory service
- Provides sign-up and sign-in functionalities to your web and mobile applications via APIs
- Has a built-in UI for sign-up and sign-in:
- Security features:
  - Multi-factor authentication
  - checks for compromised credentials
  - account takeover protection
  - phone and email verification
  - enforce strong password requirements

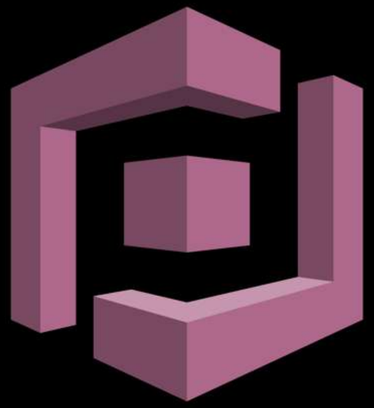




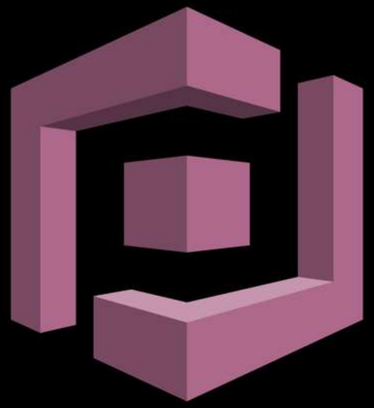
Cognito User Pool

- Has integrations with external IdPs (Amazon, Facebook, Google, SAML) for user authentication
- Users can directly sign-in using their existing accounts (external IdPs) rather than creating one in your User Pool
- Cognito manages the interactions with the external IdPs



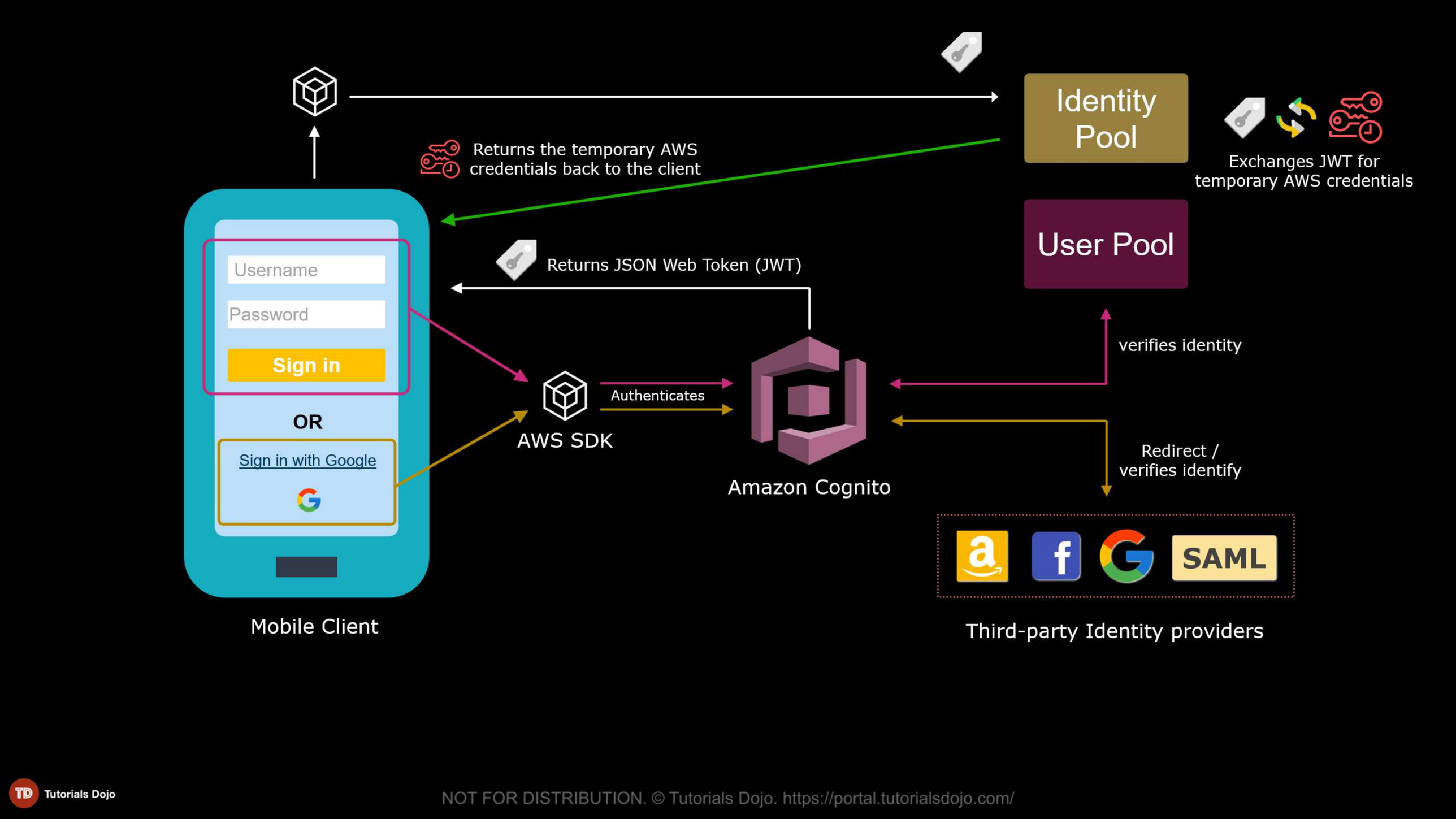


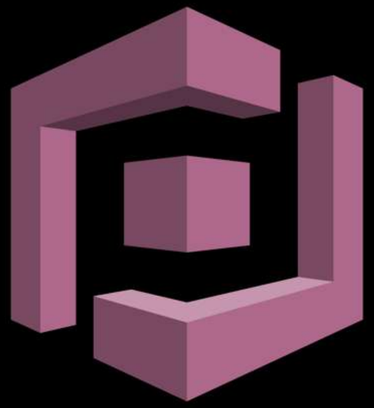
Amazon Cognito Identity Pool



Cognito Identity Pool

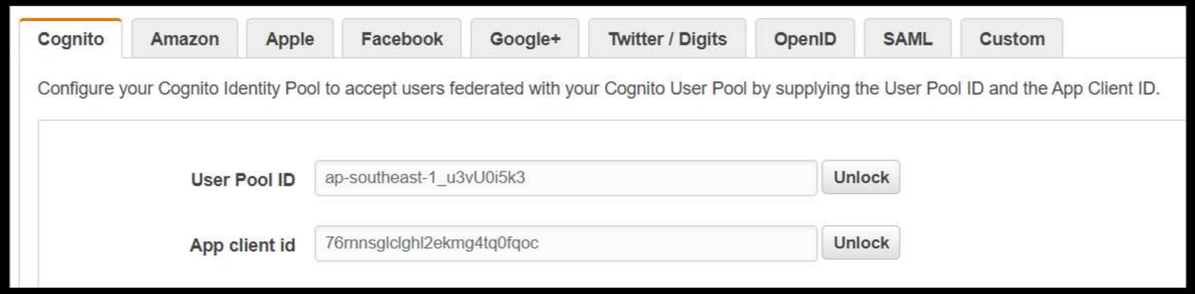
- User Pool - authentication
- Identity Pool - authorization
- Identity Pool provides temporary AWS credentials than can be used to access other AWS services (e.g., Amazon S3, Amazon DynamoDB)





Cognito Identity Pool

- You get to choose an authentication provider



- You can implement a custom authentication process using the developer authenticated identities

- You may use Identity Pool without configuring a User Pool

Cognito Identity Pool

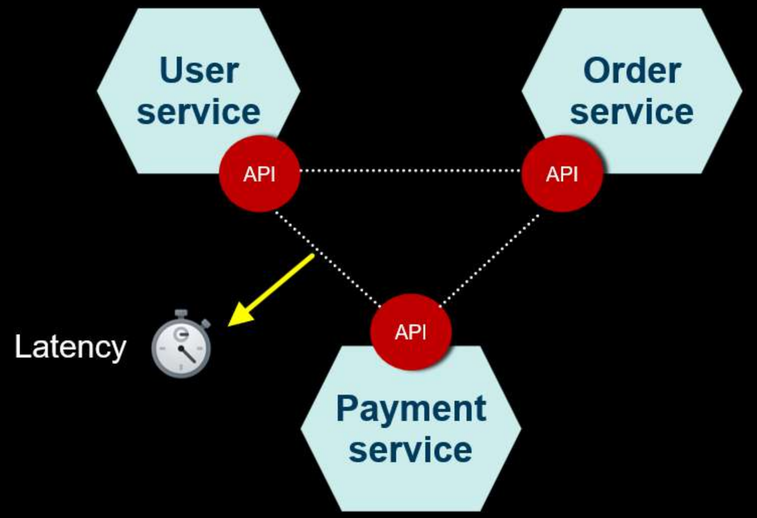
Two types of identities:

1. Authenticated
   - users who are authenticated by a trusted identity provider
   - permissions are defined by the policy associated with authenticated identities' role
2. Unauthenticated (guest users)
   - users who don't have to be logged in to access your application
   - as best practice, guest users must be given limited access
   - permissions are defined by the policy associated with unauthenticated identities' role

AWS X-Ray

- A distributed tracing service that can help you analyze and debug potential issues in a distributed application

AWS X-Ray



Microservices

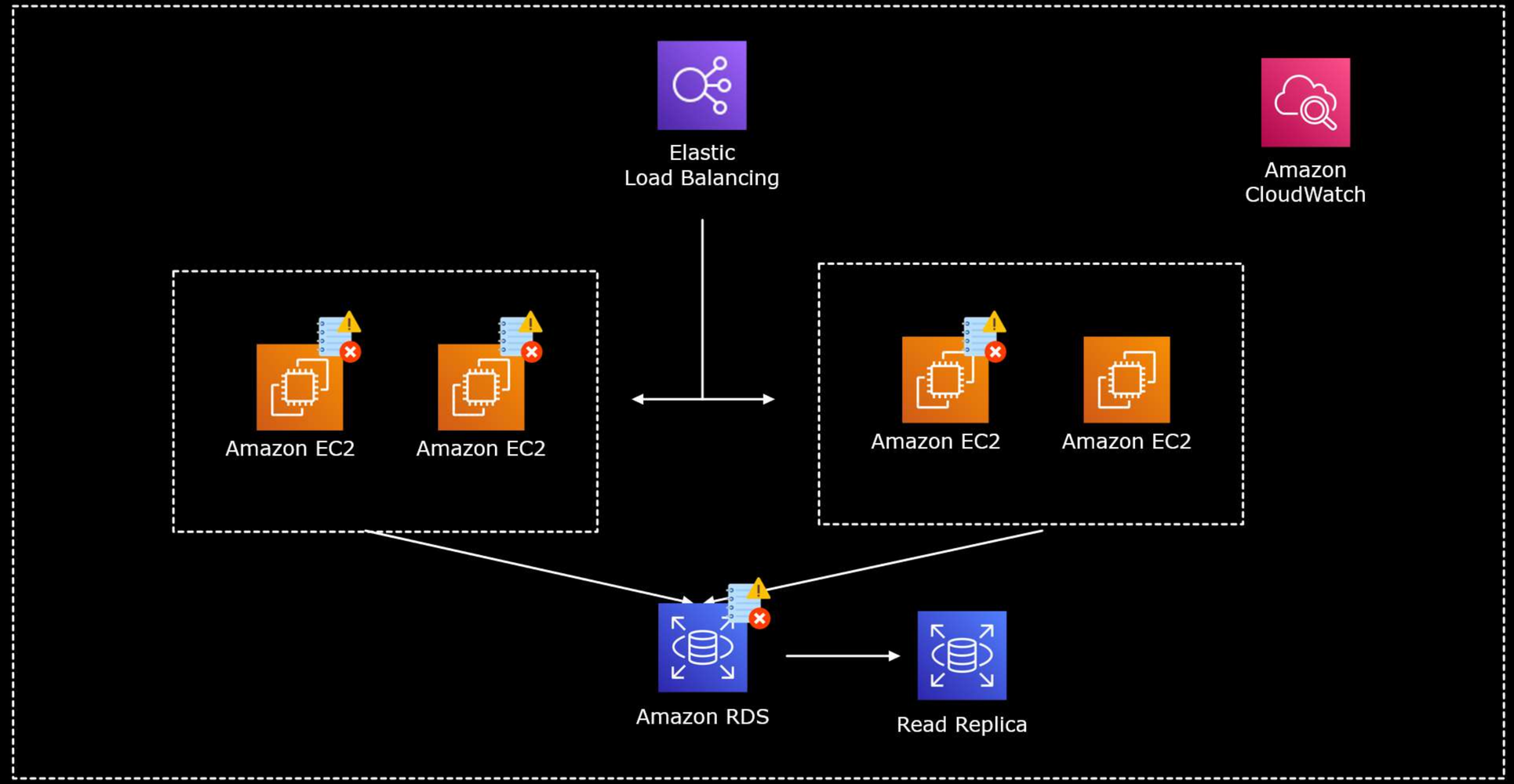
AWS X-Ray

- A distributed tracing service that can help you analyze and debug potential issues in a distributed application
- Helps you pinpoint which part causes the performance bottleneck in your application
- The service map feature gives you visibility on the user requests as they travel through the components of your application

AWS X-Ray

- A distributed tracing service that can help you analyze and debug potential issues in a distributed application
- Helps you pinpoint which part causes the performance bottleneck in your application
- The service map feature gives you visibility on the user requests as they travel through the components of your application
- The end-to-end tracing feature tracks the path of an individual request as it passes through each service node
- Supports applications running on Amazon EC2, Amazon ECS, AWS Elastic Beanstalk and AWS Lambda

Amazon CloudWatch Overview



Amazon CloudWatch

Collect

Collect metrics and logs from your AWS resources, application and on-premises servers

Monitor

Visualize your operational data using dashboards and set up alerts to notify you.

Act

Automate the process of responding to operational changes and alarms.

Analyze

Extended data retention up to 15 months with real-time analytics.

Amazon CloudWatch

Collect

Collect metrics and logs from your AWS resources, application and on-premises servers

Monitor

Visualize your operational data using dashboards and set up alerts to notify you.

Act

Automate the process of responding to operational changes and alarms.

Analyze

Extended data retention up to 15 months with real-time analytics.

Amazon CloudWatch

Collect

Collect metrics and logs from your AWS resources, application and on-premises servers

- collect operational data like logs, metrics, and events
- use the data to improve performance of your applications and resources

Amazon CloudWatch

Collect

Collect metrics and logs from your AWS resources, application and on-premises servers

Monitor

Visualize your operational data using dashboards and set up alerts to notify you.

Act

Automate the process of responding to operational changes and alarms.

Analyze

Extended data retention up to 15 months with real-time analytics.

Amazon CloudWatch

Monitor

Visualize your operational data using dashboards and set up alerts to notify you.

- setup monitoring solutions
- unified view of your application and resource's operational health
- create dashboards to get graphical view of your systems
- setup alerts to get notified

Amazon CloudWatch

Collect

Collect metrics and logs from your AWS resources, application and on-premises servers

Monitor

Visualize your operational data using dashboards and set up alerts to notify you.

Act

Automate the process of responding to operational changes and alarms.

Analyze

Extended data retention up to 15 months with real-time analytics.

Amazon CloudWatch

Act

Automate the process of responding to operational changes and alarms.

- automatically respond when things go wrong for your application and resources
- trigger actions like Auto Scaling, stopping instances to reduce overages, and trigger other workflows with services like Lambda, SNS, and CloudFormation

Amazon CloudWatch

Collect

Collect metrics and logs from your AWS resources, application and on-premises servers

Monitor

Visualize your operational data using dashboards and set up alerts to notify you.

Act

Automate the process of responding to operational changes and alarms.

Analyze

Extended data retention up to 15 months with real-time analysis.

Amazon CloudWatch

Analyze

Extended data retention up to 15 months with real-time analysis.

- deep dive and analyze existing metrics and logs
- come up with a strategy and implement them
- create metrics and logs again to verify and monitor the performance of the improvement

Amazon CloudWatch

Collect

Collect metrics and logs from your AWS resources, application and on-premises servers

Monitor

Visualize your operational data using dashboards and set up alerts to notify you.

Act

Automate the process of responding to operational changes and alarms.

Analyze

Extended data retention up to 15 months with real-time analytics.

AWS CloudWatch

Metrics

Logs

Alarms

Events

Dashboard

Metrics

Logs

Alarms

Events

Dashboard

Metrics | Logs | Alarms | Events | Dashboard

Amazon CloudWatch

Metrics - data that describes your the performance of your application or resources

- By default, many AWS services already provide free metrics for resources like EC2, EBS, and RDS.
- Most AWS services send metric data to CloudWatch every one minute.
- EC2 instances send metrics data to CloudWatch every 5 minutes by default but you can also enable detailed monitoring to send metric data at a 1-minute interval.

Metrics

Logs

Alarms

Events

Dashboard

Amazon CloudWatch

- You can publish your own custom metrics
- Metric data in CloudWatch is kept for 15 months

Metrics

Logs

Alarms

Events

Dashboard

Amazon CloudWatch

- Helps you monitor, store, and access your log files from:
  - Amazon EC2 instances
  - AWS CloudTrail
  - Route 53
  - on-premises servers, and other sources.
- Think of CloudWatch Logs as a highly scalable log repository that allows you to centralize your logs.

Metrics

Logs

Alarms

Events

Dashboard

Amazon CloudWatch

- view logs
- query for specific error codes
- filter logs based on specific fields
- archive logs for future analysis or investigation

Vended Logs

Published by AWS

and more..

Logs from on-premises server

CloudWatch Agent or **PutLogData API**

Metrics

Logs

Alarms

Events

Dashboard

Amazon CloudWatch

Alarms - triggers one or more actions based on the value of the metric relative to the threshold you set

Actions can be in the form of:

- sending notification to an Amazon SNS topic
- performing an Amazon EC2 action
- scaling an Auto Scaling Group
- creating an incident in Systems Manager
- sending a billing alert
- invoking a Lambda function

Metrics

Logs

Alarms

Events

Dashboard

Amazon CloudWatch

Events - near real-time stream of system events that describe changes in your AWS resources

Rules - matches events that you want to monitor and route them to one or more targets

Targets - can be an EC2 instance action, triggering a Lambda function, invoking ECS task, setting off a Systems Manager Automation document, publishing a message to an SNS topic, and many more

Once a rule is set up, CloudWatch Events becomes aware of operational changes once they occur

Metrics

Logs

Alarms

Events

Dashboard

CloudWatch Events

Amazon EventBridge

Metrics

Logs

Alarms

Events

Dashboard

Alarms

take actions when a metric reaches a certain threshold

Events

reacts to real-time system events performed on your AWS resources

Metrics

Logs

Alarms

Events

Dashboard

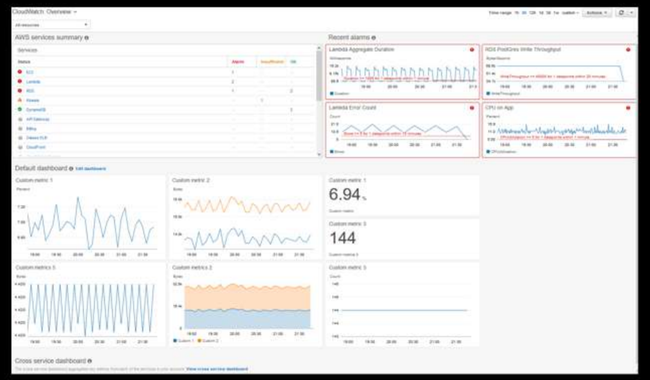
Amazon CloudWatch

- creates a reusable graphs and visualize your resources in a unified view
- graph metrics and logs data into a single dashboard which helps you to resolve issues and understand the root cause

Amazon EC2

CPU Utilization

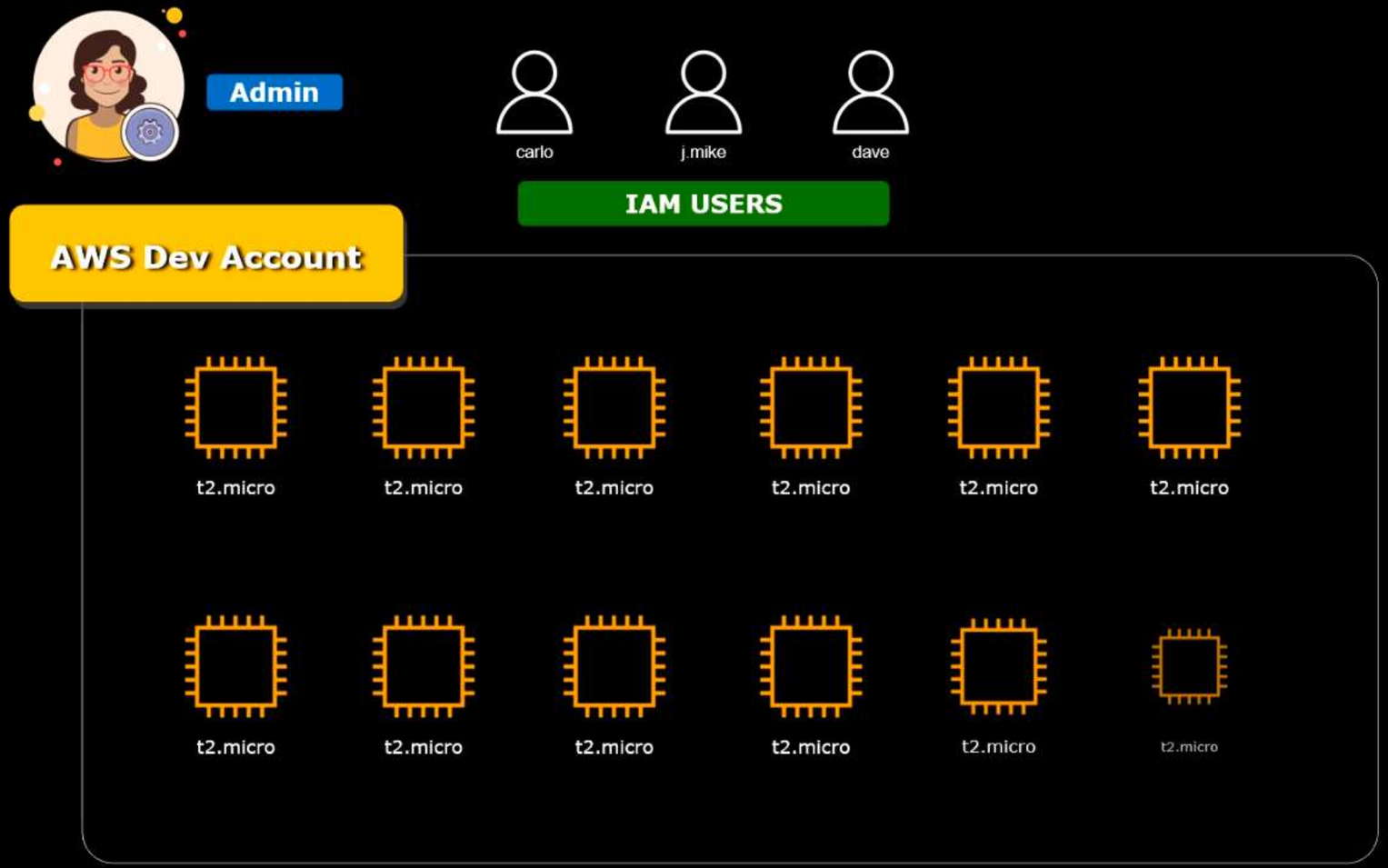
Memory Usage



Amazon CloudWatch Overview

AWS CloudTrail

- A managed service for monitoring and logging account activities
- AWS Management Console, API, or CLI actions are recorded as events
- It can help you facilitate governance, compliance, operational auditing, and risk auditing of your AWS account



Admin

CloudTrail log

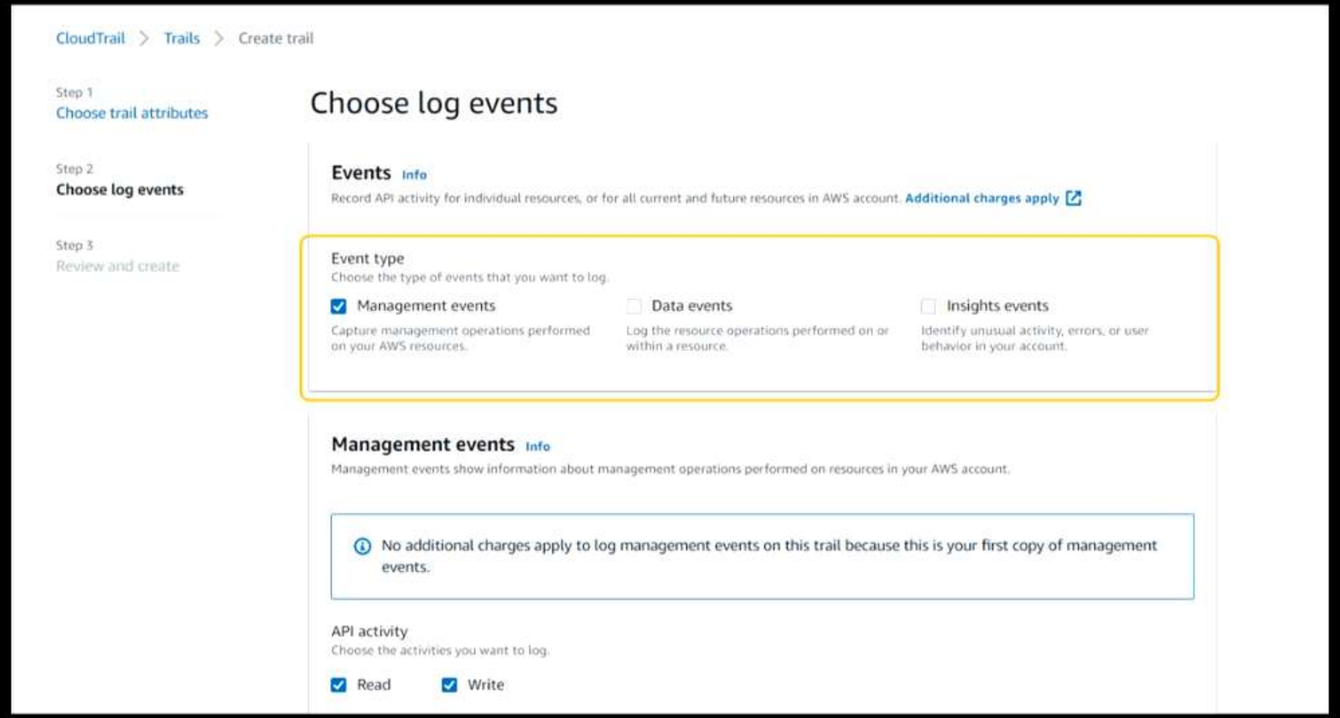

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDAS2OSY7CBGGXUXXKGJ",
        "arn": "arn:aws:iam::194051115896:user/carlo",
        "accountId": "194051115896",
        "accessKeyId": "ASIAS2LSY7CBL6QTYY4N",
        "userName": "carlo",
        "sessionContext": {
            "sessionIssuer": {},
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2021-06-28T16:07:31Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2021-06-28T16:14:56Z",
    "eventSource": "ec2.amazonaws.com",
    "eventName": "RunInstances",
    "awsRegion": "ap-southeast-1",
    "sourceIPAddress": "230.205.213.212",
    "userAgent": "console.ec2.amazonaws.com",
    "requestParameters": {
        "instancesSet": {
            "items": [
```


carlo

j.mike

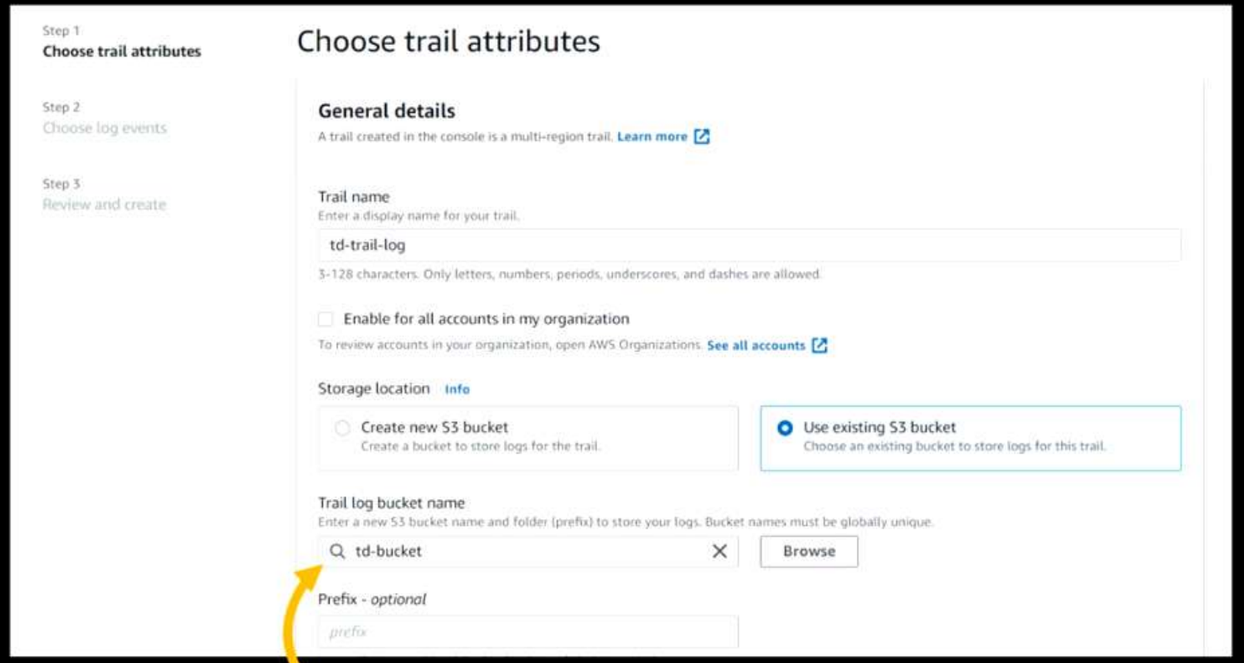
dave

IAM USERS

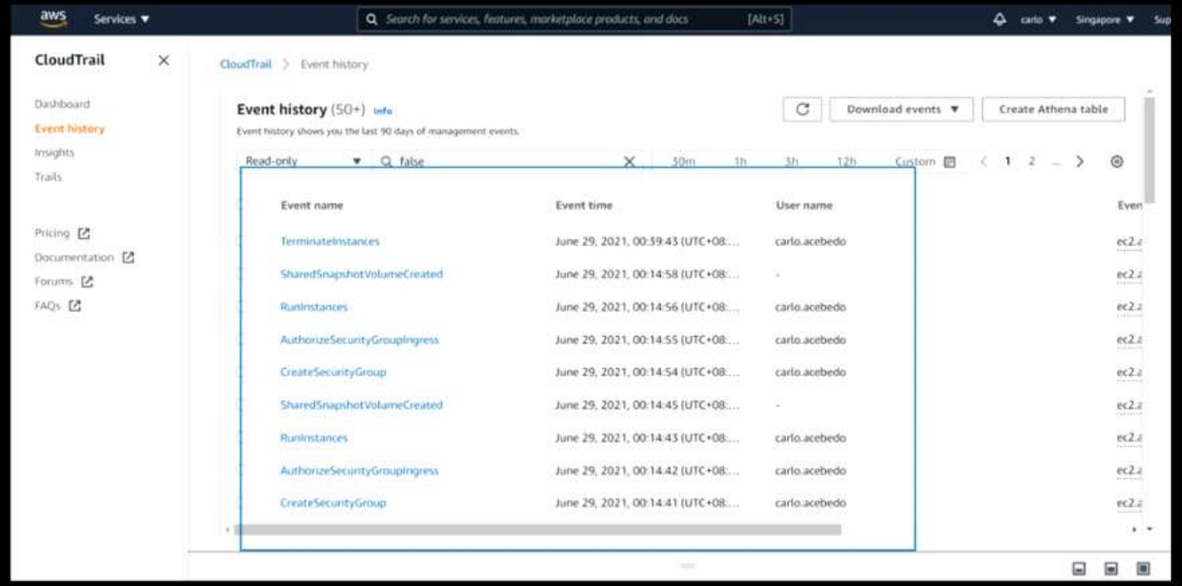


Event types

1. Management events
   - management operation events such as user logins
   - enabled by default
2. Data events
   - S3 object-level API events (ex., `GetObject`, `PutObject`)
   - Lambda function Invoke API
   - DynamoDB object-level API activity (ex., `PutItem`, `DeleteItem`)
   - Additional charges apply
3. Insight events
   - Unusual write API calls

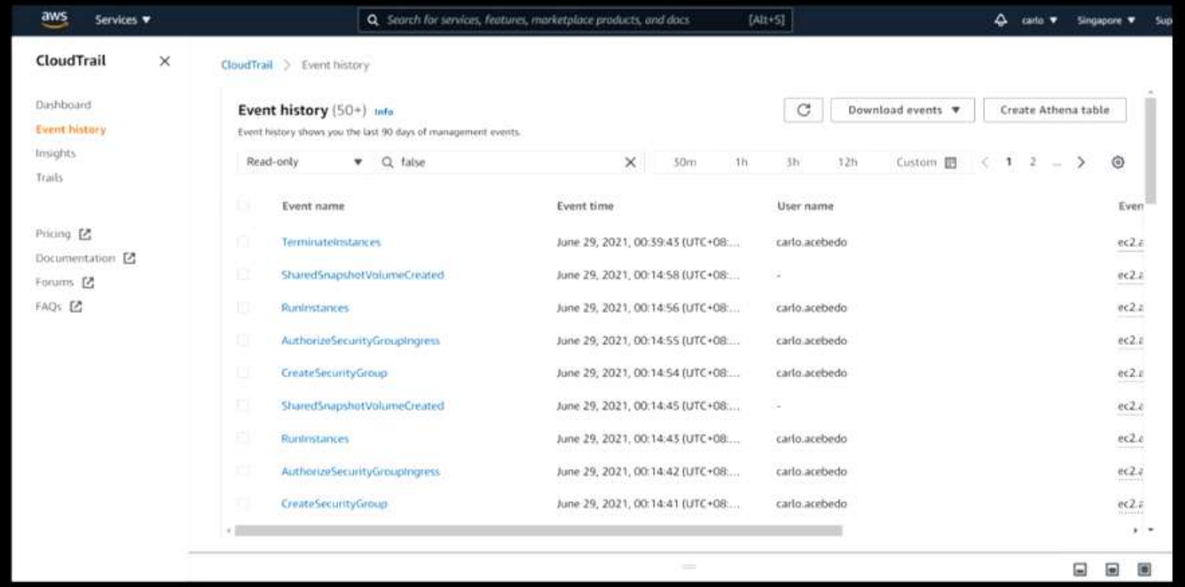


Trail logs are stored on Amazon S3



Event history

- Gives immediate visibility to management events over the past 90 days
- Data older than 90 days are deleted
- Create a trail to persist logs on an S3 bucket



AWS
Secrets Manager

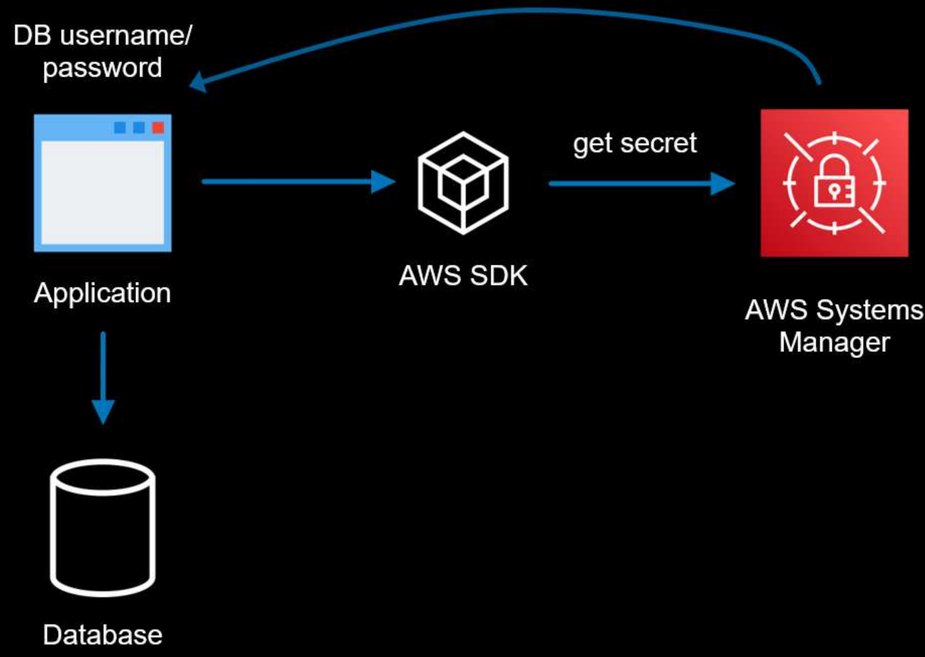
- A secret management service that enables you to easily rotate, manage, and retrieve database credentials, API keys, and other secrets throughout their lifecycle

Example of hardcoded credentials

```
connection = pymysql.connect(host = 'db-prod.czx0zp7fdpw7.us-east-2.rds.amazonaws.com',
                             user= 'tutorialsdojo',
                             port = 3306,
                             password= 'password123')
```

AWS
Secrets Manager

- A secret management service that enables you to easily rotate, manage, and retrieve database credentials, API keys, and other secrets throughout their lifecycle



AWS
Secrets Manager

- A secret management service that enables you to easily rotate, manage, and retrieve database credentials, API keys, and other secrets throughout their lifecycle
- Secrets are resources that save and retrieve in Secrets Manager
- A secret can be a DB username and password, key-value pair, JSON document, and TLS certificate

AWS
Secrets Manager

- Secrets Manager supports automatic rotation of database credentials for Amazon RDS, Amazon Redshift, and Amazon DocumentDB without writing your own rotation function
- You have to author a custom Lambda function to automatically rotate credentials from external databases and other secrets
- Secrets Manager uses AWS KMS to enforce encryption at rest
- Storing of data in plaintext is not possible

How to retrieve a secret?

AWS
Secrets Manager

```
secrets_manager = boto3.client('secretsmanager')

secret = secrets_manager.get_secret_string('mysecret')
response = secrets_manager.get_secret_value(SecretId='dev/carlo/tdojo')

DB_Credentials = json.loads(response['SecretString'])

DB_HOST = DB_Credentials['host']
DB_USER = DB_Credentials['username']
DB_PWD = DB_Credentials['password']

connection = pymysql.connect(host = DB_HOST,
                             user= DB_USER,
                             port = 3306,
                             password= DB_PWD)
```

Secret name

DB credentials are no longer hardcoded

Without AWS Secrets Manager

AWS
Secrets Manager

```
MyRDSInstance:
  Type: 'AWS::RDS::DBInstance'
  Properties:
    DBName: MyRDSInstance
    AllocatedStorage: '20'
    DBInstanceClass: db.t2.micro
    Engine: mysql
    MasterUsername: 'tutorialsdojo'
    MasterUserPassword: 'password123'
```

AWS
Secrets Manager

With AWS Secrets Manager

```yaml
MyRDSInstance:
  Type: 'AWS::RDS::DBInstance'
  Properties:
    DBName: MyRDSInstance
    AllocatedStorage: '20'
    DBInstanceClass: db.t2.micro
    Engine: mysql
    MasterUsername: '{{resolve:secretsmanager:mySecret:SecretString:username}}'
    MasterUserPassword: '{{resolve:secretsmanager:mySecret:SecretString:password}}'
```

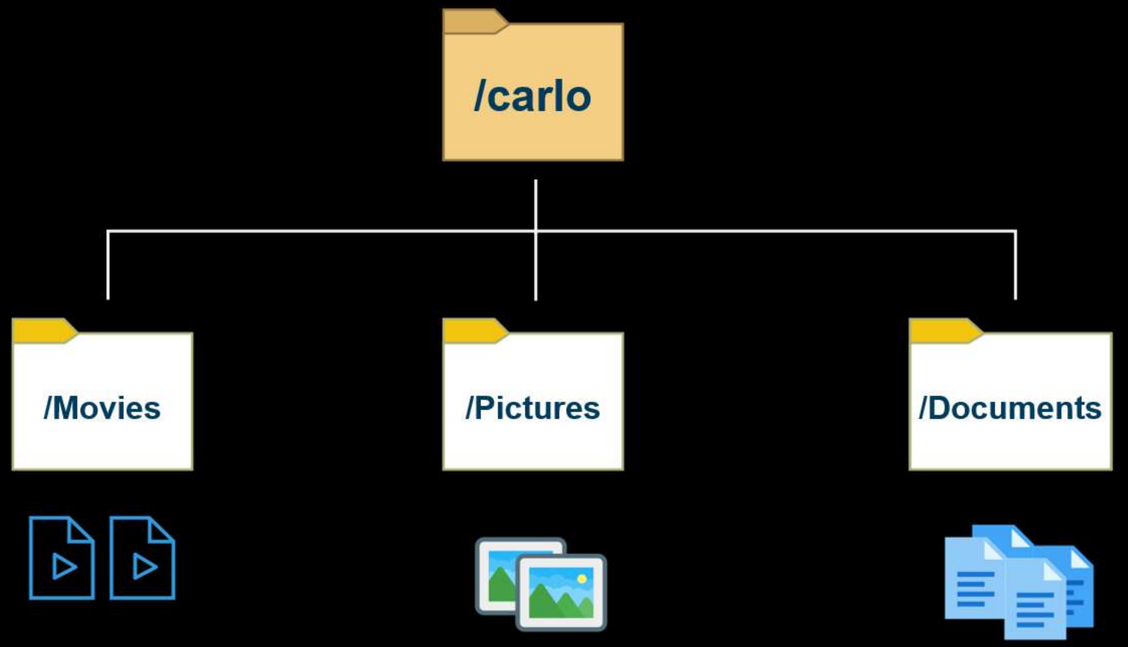
AWS
SSM Parameter Store

- Parameter Store allows you to create key-value parameters where you can store your application configurations, custom environment variables, product keys, and credentials

- Parameter Store is suitable for storing parameters that don't require encryption (AMI ID, security group ID, URL, etc)

AWS
SSM Parameter Store

- Parameter hierarchy is a method of grouping parameters to make management easier



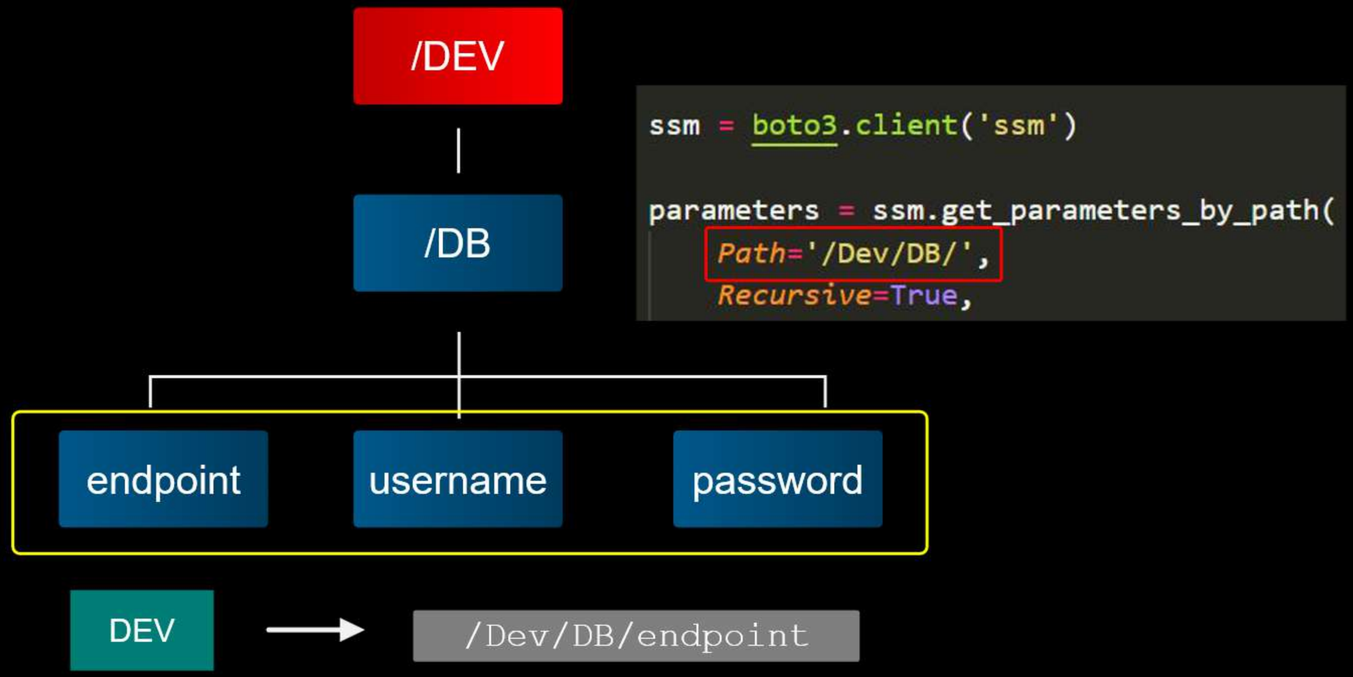
AWS
SSM Parameter Store

- Parameter hierarchy is a method of grouping parameters to make management easier
- Parameter hierarchy is helpful in determining the right parameter for a job, preventing you from accidentally using the wrong parameter

PROD → `/Prod/DB/endpoint`

DEV → `/Dev/DB/endpoint`

AWS
SSM Parameter Store




```
ssm = boto3.client('ssm')

parameters = ssm.get_parameters_by_path(
    Path='/Dev/DB/',
    Recursive=True,
```

AWS
SSM Parameter Store

3 types of parameters:

1. String - any block of text that you wish to store unencrypted

Example:

https://rpoz5vq5wf.execute-api.ap-southeast-1.amazonaws.com

Example abc 123

2. StringList - comma-separated list of strings

Example:

aaa,bbb,ccc

#FF5733,#3B98B4,#AA3BB4

3. SecureString - any sensitive data that needs to be encrypted

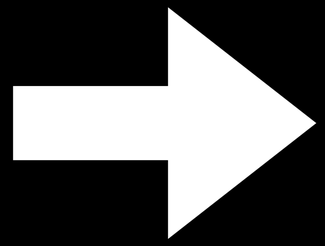
AWS
SSM Parameter Store

2 parameter tiers:

1. Standard
   - has a limit of 10,000 parameters
   - has a parameter value size limit of up to 4KB
   - storing of parameters is free
2. Advanced
   - can have more than 10,000 parameters
   - has a parameter value size limit of up to 8KB
   - $0.05 per advanced parameter per month

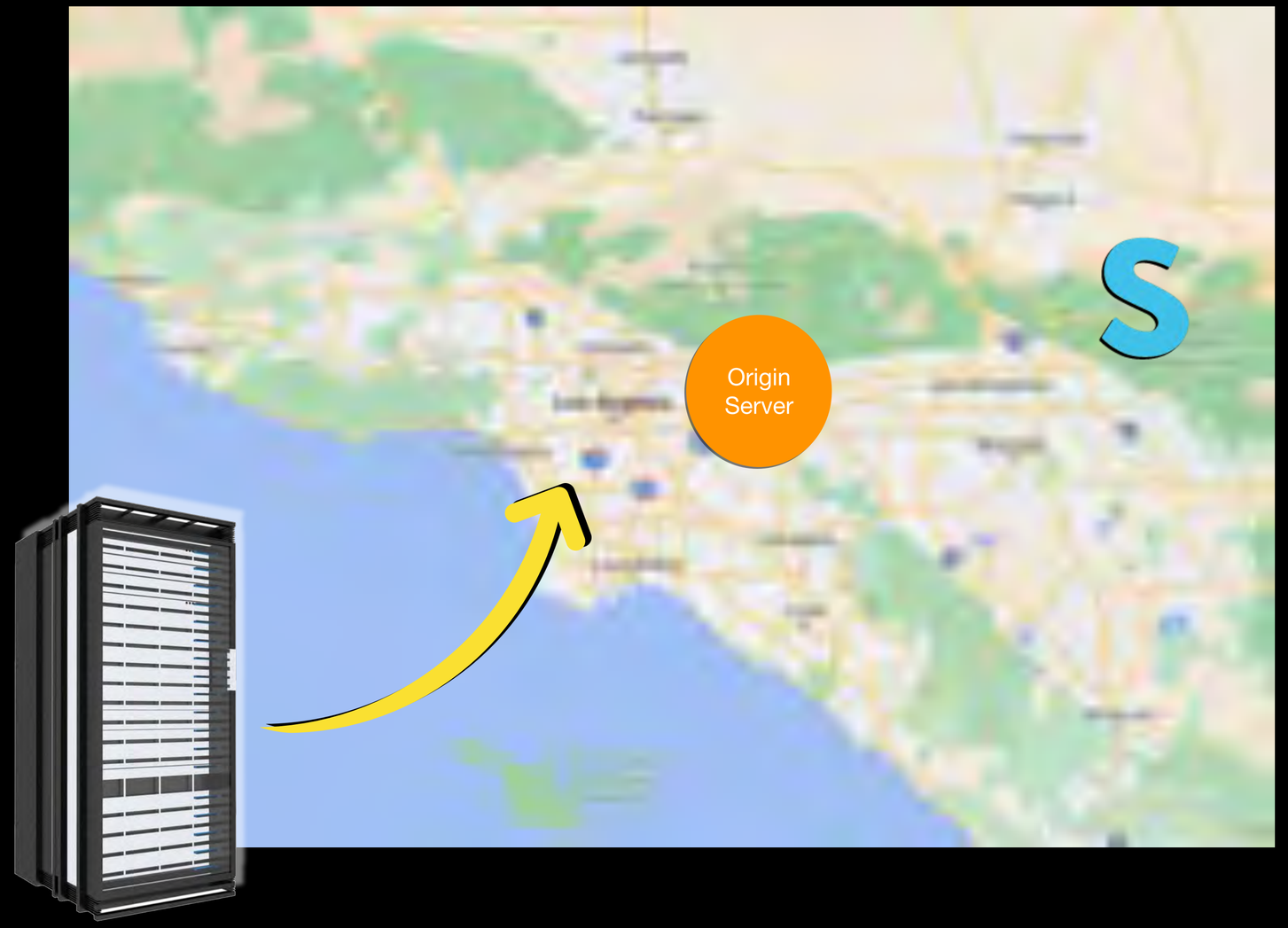
Amazon CloudFront
Overview

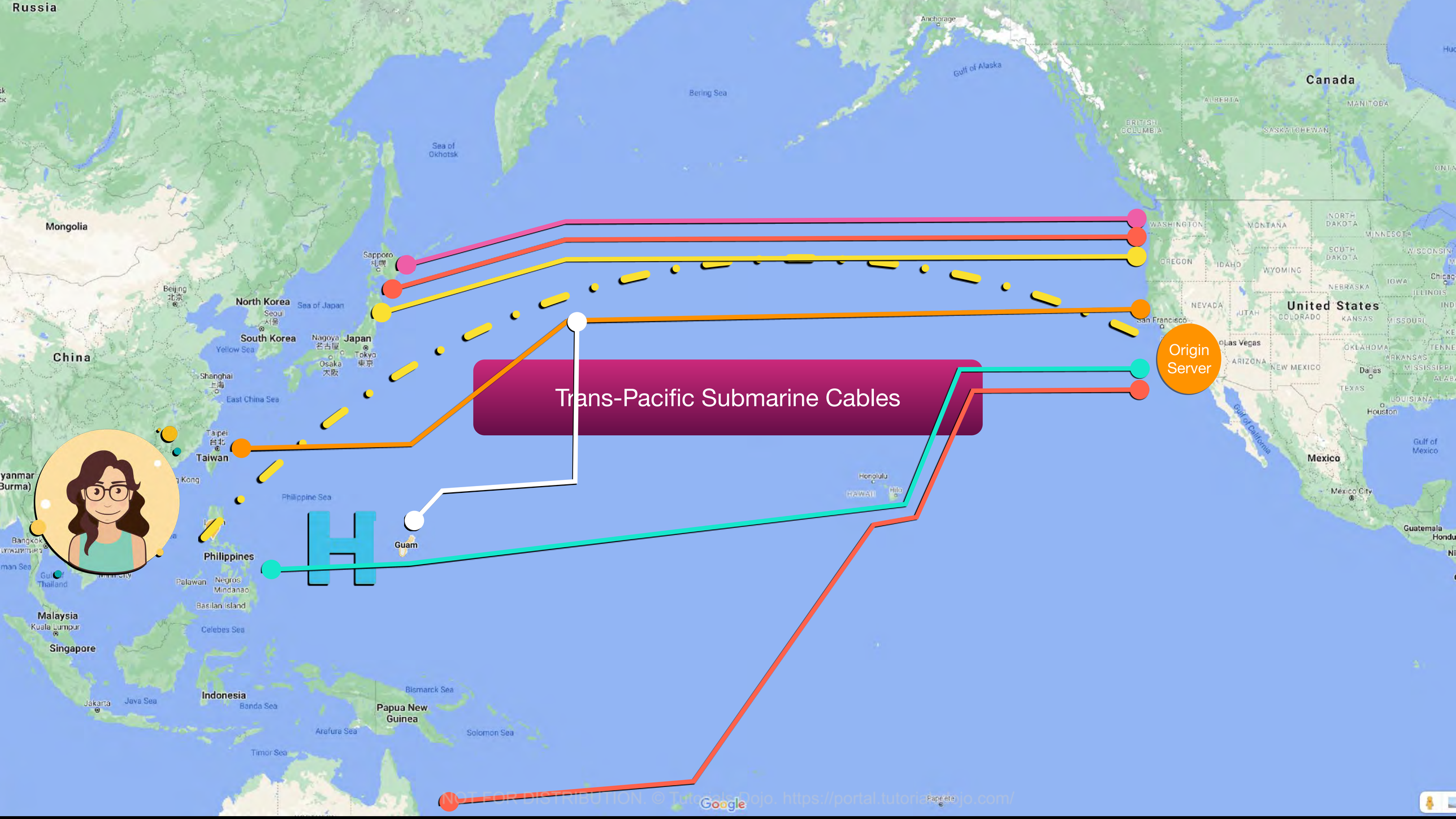
CloudFront

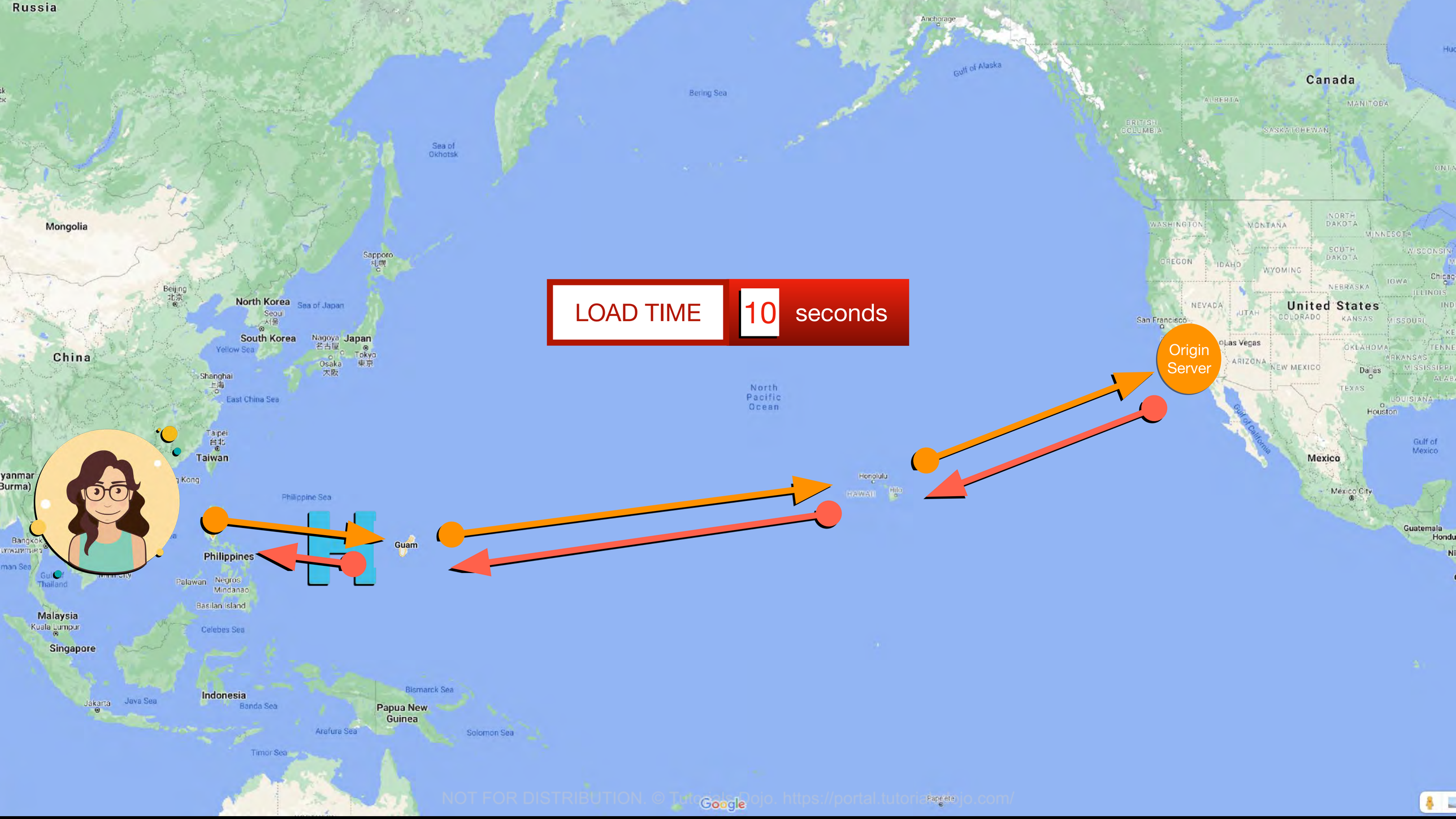


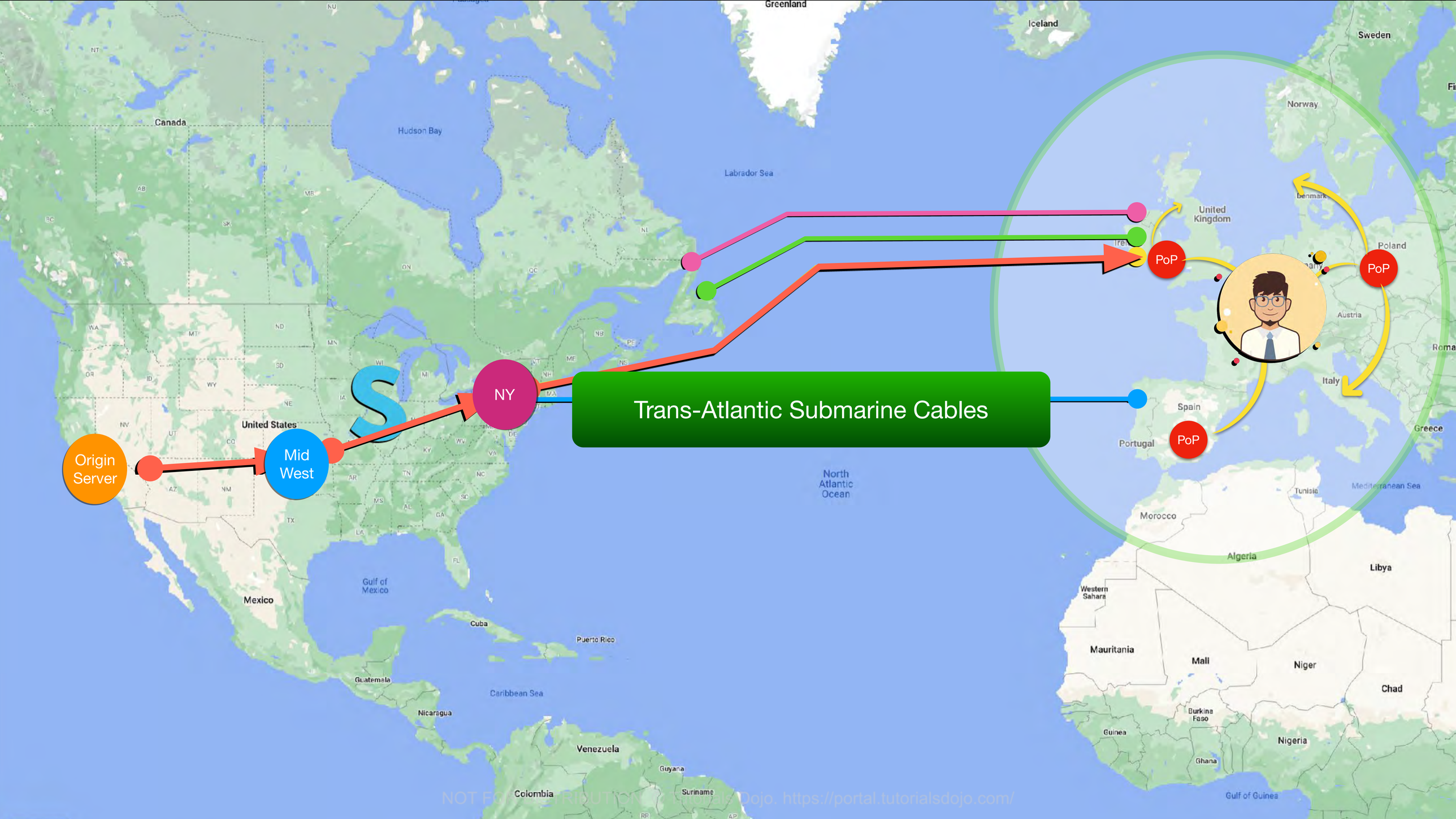
Content Delivery Network

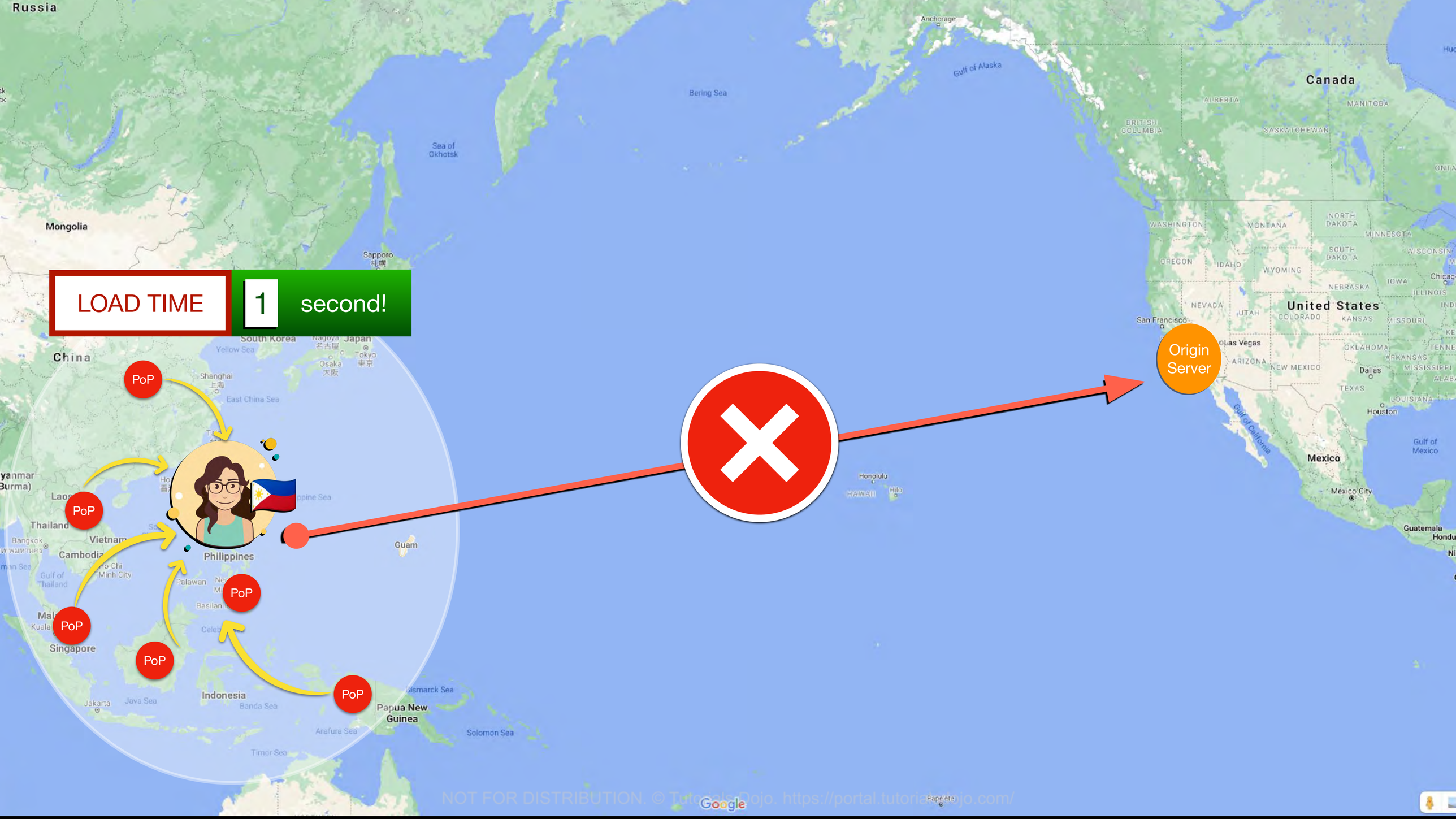
Content Delivery Network

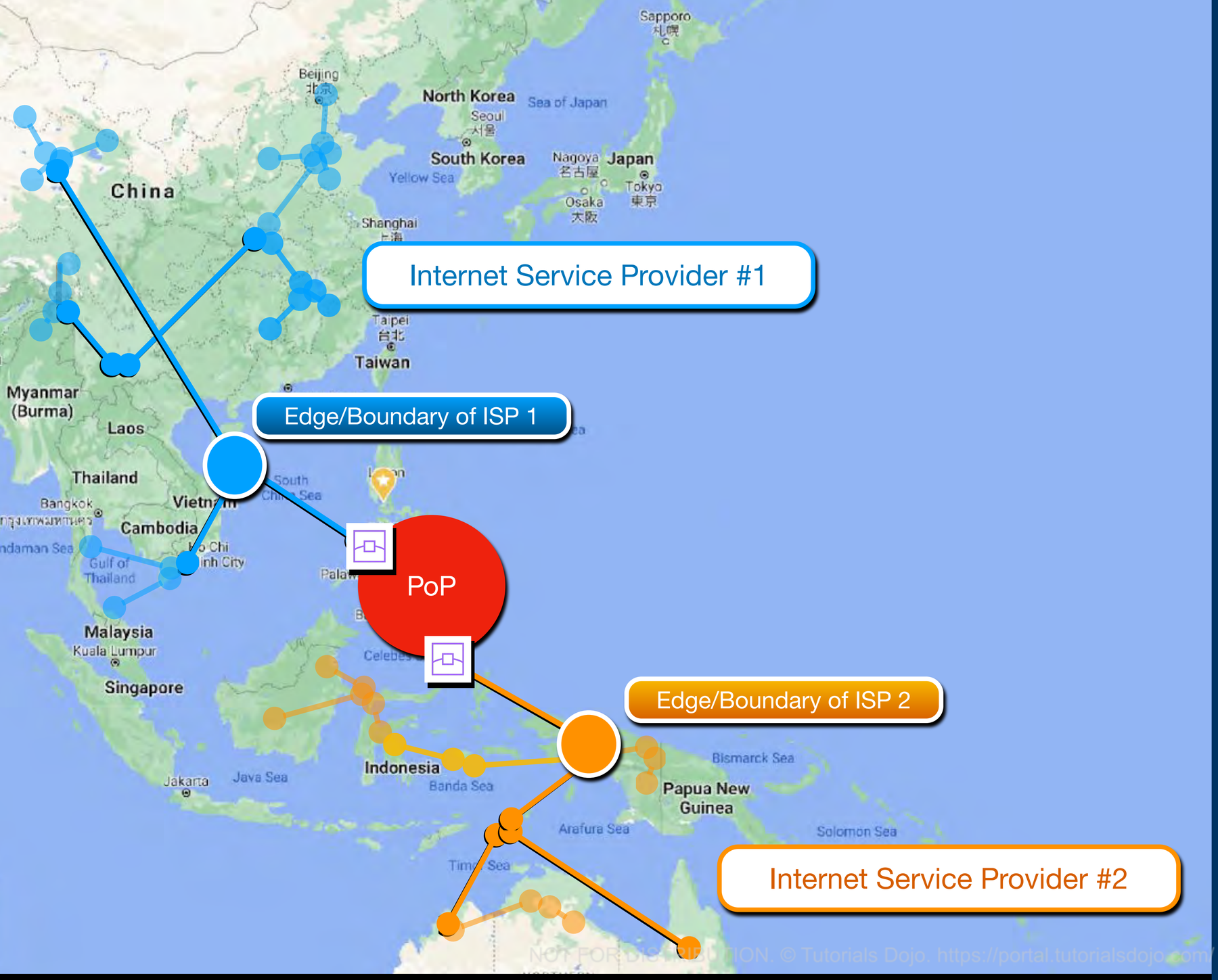


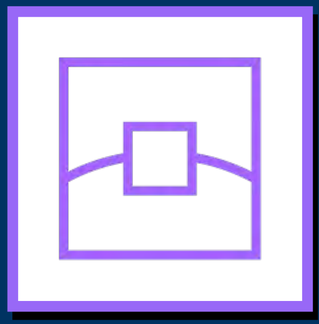








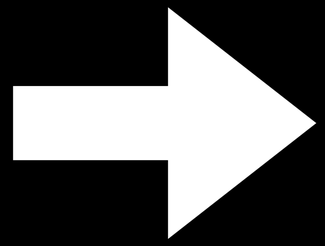




Edge Location

- Refers to the ‘edge’ or the boundary of the network
- Connects the different networks of various Internet Service Providers (ISPs) or Telecommunications companies

CloudFront



Content Delivery Network

CloudFront

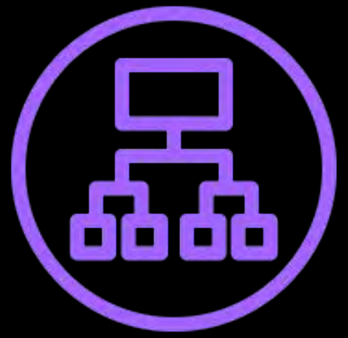
ORIGIN

DISTRIBUTION

VIEWER

ORIGIN

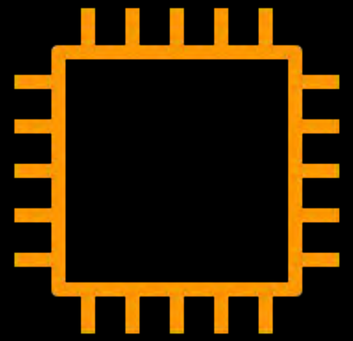
Amazon S3 Bucket



Elastic Load Balancer

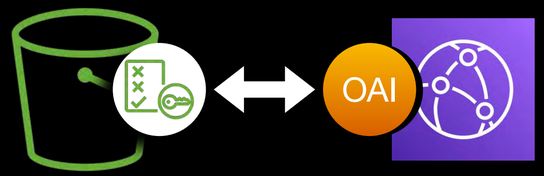
AWS Elemental
MediaPackage Endpoint

AWS Elemental
MediaStore Container



Amazon EC2 Instance or
Your On-Premises Server

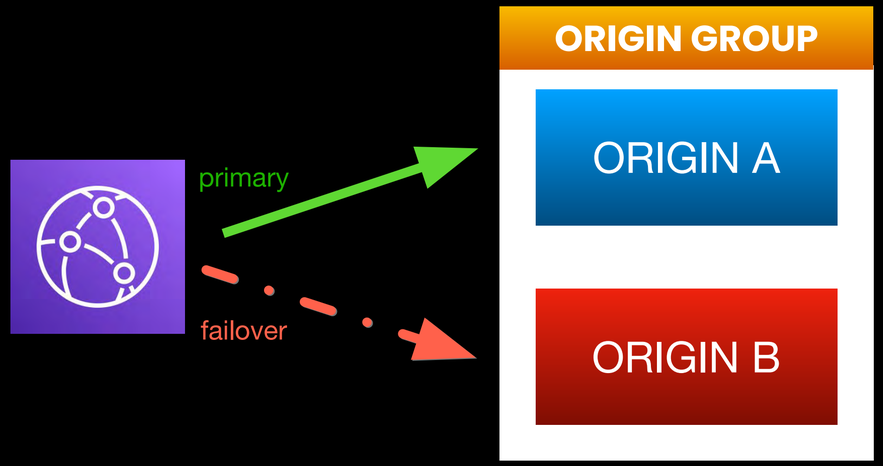
Amazon CloudFront Features



ORIGIN ACCESS IDENTITY
(OAI)

GEO-RESTRICTION

Lambda@Edge
and
CloudFront Functions



ORIGIN GROUP and ORIGIN FAILOVER

Amazon CloudFront Features

Signed URLs

Signed Cookies

Custom Domain Name and Custom SSL
(SNI / Dedicated IP)

AWS WAF – CloudFront Integration

	Global Secondary Index	Local Secondary Index
Key Attributes	PARTITION KEY + SORT KEY + NON-KEY Attributes	SORT KEY + NON-KEY Attributes
Span query	Queries span all data in the base table, across all partitions	Queries are scoped on the partition key of its base table
Index operations	Can be created anytime	Can be created only during the creation of a table
Size restrictions per partition key value	No restriction	Total size of indexed items under a partition key value ≤ 10GB
Read consistency	Eventual Consistency	Supports both Eventual and Strong Consistency
Provisioned Throughput consumption	Has its own provisioned throughput for read and write activities	Consumes capacity units from its base table
Projected Attributes	You can only request attributes that are projected in the GSI	Requested attributes that are not projected into the LSI are fetched from the base table by DynamoDB automatically

	Size value	Encryption	Versioning	Automatic Rotation	Cost
AWS Systems Manager Parameter Store	**STANDARD** • Up to 4KB per entry **ADVANCED** • Up to 8KB per entry				
AWS Secrets Manager	• **Up to 64KB per secret** • **Suitable for storing certificates**				

	Size value	Encryption	Versioning	Automatic Rotation	Cost
 AWS Systems Manager Parameter Store	**STANDARD** • Up to 4KB per entry **ADVANCED** • Up to 8KB per entry	• Uses **AWS KMS** for encryption • Encryption is **optional**			
 AWS Secrets Manager	• **Up to 64KB per secret** • **Suitable for storing certificates**	• Uses **AWS KMS** for encryption • Encryption is **enforced** • Storing plaintext data is not possible			

	Size value	Encryption	Versioning	Automatic Rotation	Cost
 AWS Systems Manager Parameter Store	**STANDARD** • Up to 4KB per entry **ADVANCED** • Up to 8KB per entry	• Uses **AWS KMS** for encryption • Encryption is **optional**	• One version can **only be** active at any given time		
 AWS Secrets Manager	• **Up to 64KB per secret** • **Suitable for storing certificates**	• Uses **AWS KMS** for encryption • Encryption is **enforced** • Storing plaintext data is not possible	• Multiple versions **can exist at the same time** when performing secret rotation		

	Size value	Encryption	Versioning	Automatic Rotation	Cost
AWS Systems Manager Parameter Store	**STANDARD** • Up to 4KB per entry **ADVANCED** • Up to 8KB per entry	• Uses **AWS KMS** for encryption • Encryption is **optional**	• One version can **only be** active at any given time		
AWS Secrets Manager	• **Up to 64KB per secret** • **Suitable for storing certificates**	• Uses **AWS KMS** for encryption • Encryption is **enforced** • Storing plaintext data is not possible	• Multiple versions **can exist at the same time** when performing secret rotation	• Can **rotate secrets automatically** • **Cross-account access** is supported • Can **generate random password**	

	Size value	Encryption	Versioning	Automatic Rotation	Cost
 AWS Systems Manager Parameter Store	**STANDARD** • Up to 4KB per entry **ADVANCED** • Up to 8KB per entry	• Uses **AWS KMS** for encryption • Encryption is **optional**	• One version can **only be** active at any given time		• It is **cheaper** Standard - FREE Advanced - $0.05/ parameter/month • $0.05 / 10,000 API calls
 AWS Secrets Manager	• **Up to 64KB per secret** • **Suitable for storing certificates**	• Uses **AWS KMS** for encryption • Encryption is **enforced** • Storing plaintext data is not possible	• Multiple versions **can exist at the same time** when performing secret rotation	• Can **rotate secrets automatically** • **Cross-account access** is supported • Can **generate random password**	• **More expensive** • $0.4/secret/month • $0.05 / 10,000 API calls

AWS Elastic Beanstalk

- All-at-once
- Rolling
- Rolling with additional batch
- Immutable
- Traffic Splitting

AWS CodeDeploy

- All-at-once
- Half-at-a-time
- One-at-a-time
- Canary
- Linear

AWS Elastic Beanstalk

- It is **NOT** a native deployment service
- A Platform-as-a-Service (PaaS) for deploying applications to AWS without dealing with infrastructure management
- Provides different deployment methods for deploying new application versions
- Uses the **EB CLI** for deploying codes
- Deployments are **limited to instances inside an EB environment**
- You can deploy incremental code changes from a **CodeCommit repository using EB CLI**

AWS CodeDeploy

- Purposely built to automate code deployments
- You can deploy code updates to EC2 instances, on-premises servers, and Lambda functions
- You can also deploy codes to instances managed by Elastic Beanstalk
- Uses the **AppSpec file** for managing deployments
- CodeDeploy **does not** deal with infrastructure configuration and orchestration
- It makes more sense to use AWS CodeDeploy for deployments to **existing on-premises servers or instances**