Module 3

Public Cloud

Introduction to Public Cloud

According to an International Data Corporation (IDC) report, worldwide spending on public cloud provider services will reach $1.35 trillion in 2027.

A public cloud is a type of cloud computing in which a third-party *service provider makes computing resources available to users* over the public internet on a pay-per-usage basis.

It enables companies to automatically scale compute and storage resources up or down to meet their individual needs.

Famous public cloud providers are Amazon Web Services (AWS), Google Cloud Platform, IBM Cloud or Microsoft Azure.

How does public cloud computing work?

- ❑ A cloud service provider (CSP) owns and operates vast physical data centers that run client workloads.
- ❑ Public cloud services are built on extensive *data center infrastructure*. Those data centers house the physical hardware and servers that deliver cloud services to users and businesses.
- ❑ Data centers are distributed across various geographical regions (*Google has 23 active data center locations*).
- ❑ A distributed infrastructure *improves performance* by reducing latency, provides redundancy, and ensures data resilience in case of hardware failures.

How does public cloud computing work?

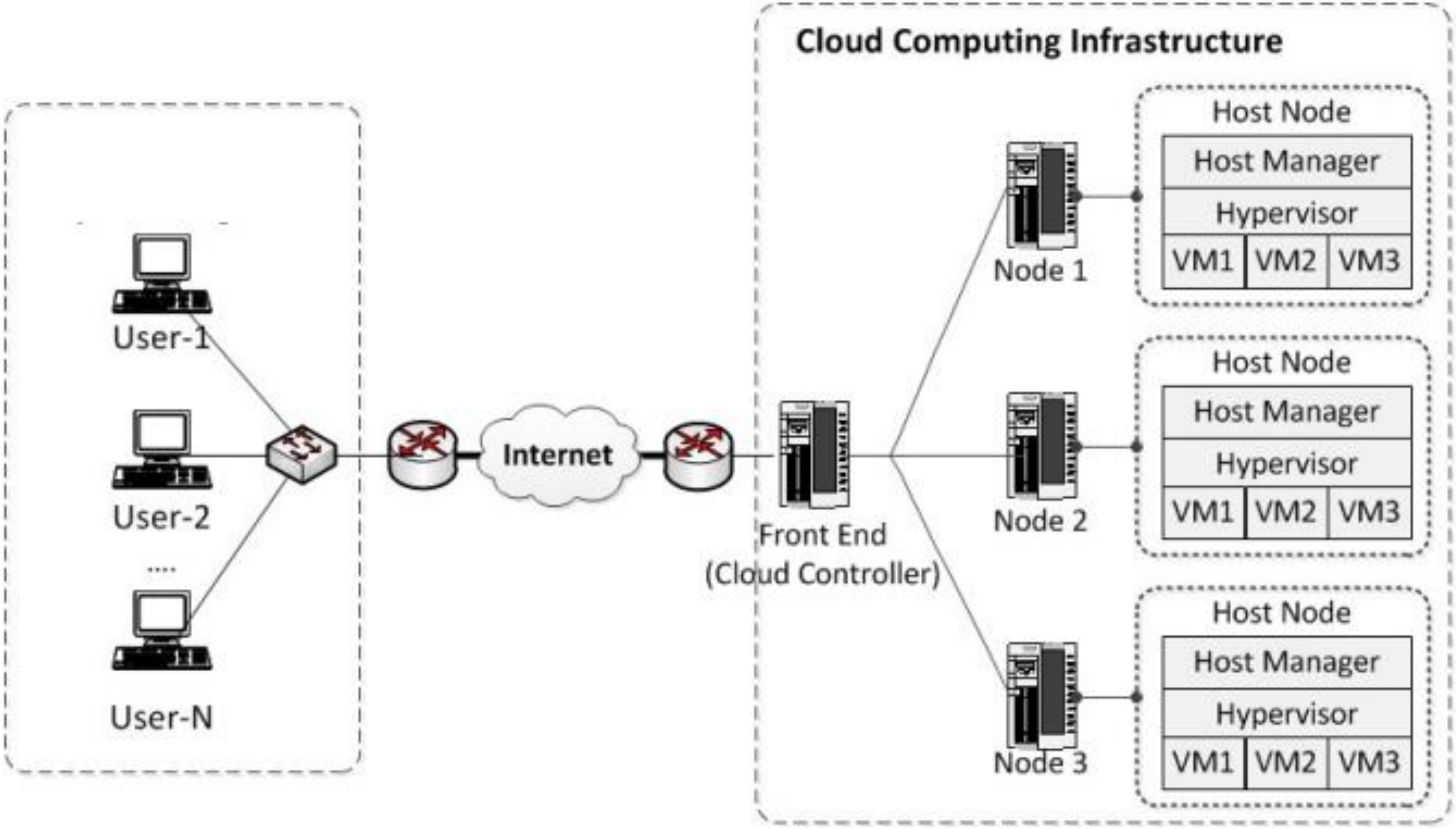
- ❑ Public cloud environments are multi-tenant i.e. multiple customers share the same physical infrastructure and computing resources, while their data and operations remain isolated from each other. The service is provided through a self-service API interface.

- ❑ Multi-tenant hosting allows cloud service providers to *maximize utilization of their data centers and infrastructure resources* to offer services at much lower costs than a company-owned, on-premises data center.

Public cloud computing - Steps

1. **Infrastructure Setup (by Cloud Provider)**
2. **Resource Pooling & Virtualization**
3. **Self-Service Portal or API Access**
4. **Service Provisioning**
5. **Scalability & Elasticity**
6. **Security & Isolation** (User authentication and access control/ Data encryption/ Network isolation)
7. Monitoring & Billing

Public cloud Architecture



Public cloud Architecture

Users can be individuals, companies, or applications needing compute, storage, or platform services.

A Cloud Controller is defined as a component responsible for creating, maintaining, and deleting resources in a Cloud.

It schedules virtual resources in the cloud and manages resource allocation within the cloud environment.

It is responsible for: *Resource management, Service provisioning, User authentication and authorization,, Monitoring and billing*

Public cloud Architecture

In the backend,

- Each node is a physical server in a data center. Nodes are grouped together to form a resource pool for computing tasks.

- Host Manager in each host node is a software that oversees the node’s resources. *It handles orchestration, resource scheduling, and communication with the Cloud Controller.*

- The virtualization layer that enables multiple virtual machines (VMs) to run on one physical host.

- *Each VM runs its own OS and applications, isolated from others. Users may be assigned one or more VMs based on their requested resources.*

Public cloud Architecture

Node Manager enables you to perform these tasks:

- *Start and stop remote Managed Servers.*
- *Monitor the self-reported health of Managed Servers and automatically kill server instances whose health state is "failed".*

For example, In Google Cloud, Managed Instance Groups (MIGs) restart unhealthy instances.

Public cloud service models

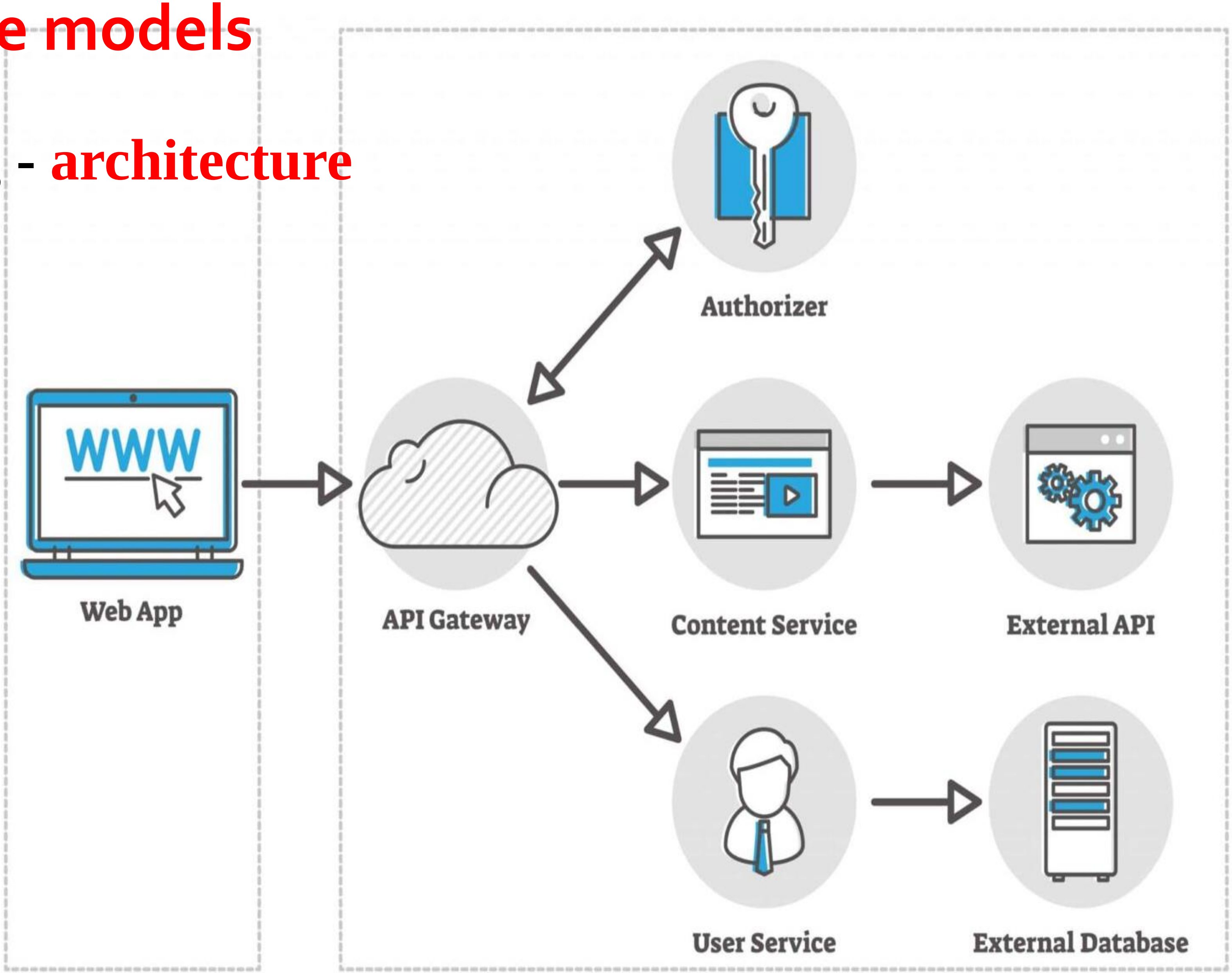
1. Software-as-a-Service (SaaS)
2. Platform-as-a-Service (PaaS)
3. Infrastructure-as-a-Service (IaaS)
4. Serverless computing (Serverless)

Serverless computing

Serverless computing is an application development and execution model that enables developers to build and run application code without provisioning or managing servers or back-end infrastructure.

Public cloud service models

Serverless computing - architecture



Public cloud service models

Serverless computing - Characteristics

- *No server management*
- *Event driven - Functions are triggered by events (HTTP request, file upload, DB change).*
- *Granular billing*
- *Auto-scaling*

Public cloud service models

Serverless computing

In serverless, you write and deploy code, and the cloud provider handles everything else:

- Server provisioning
- Scaling
- High availability
- Patching
- Billing (based on actual execution time)

Public cloud service models

Serverless computing vs. PaaS

Feature	PaaS (Traditional)	Serverless (FaaS)
Deployment unit	Deploy a full application	Deploy individual functions
Always running?	Yes, the app runs even when idle	No, functions run only when triggered
Scaling	Manual or limited auto-scaling	Auto-scales instantly per request/event
Billing	Pay for uptime or resources allocated	Pay only for actual execution time
Management level	No server management, but app runtime is managed	No server or app runtime management at all
Startup time (cold start)	N/A or very low	Functions may have a slight cold start delay
Use case	Web apps, APIs, backend services	Event-driven tasks, lightweight microservices

Benefits of Cloud Computing

- **Cost-effectiveness**
- **Efficiency**
- **Elasticity**
- **Scalability**
- **Innovation –** *include advanced technologies*
- **Spending predictability**
- **Team collaboration**
- **High availability and reliability –** *automatic backup and disaster recovery*
- **Sustainability –** *reduce carbon footprint*

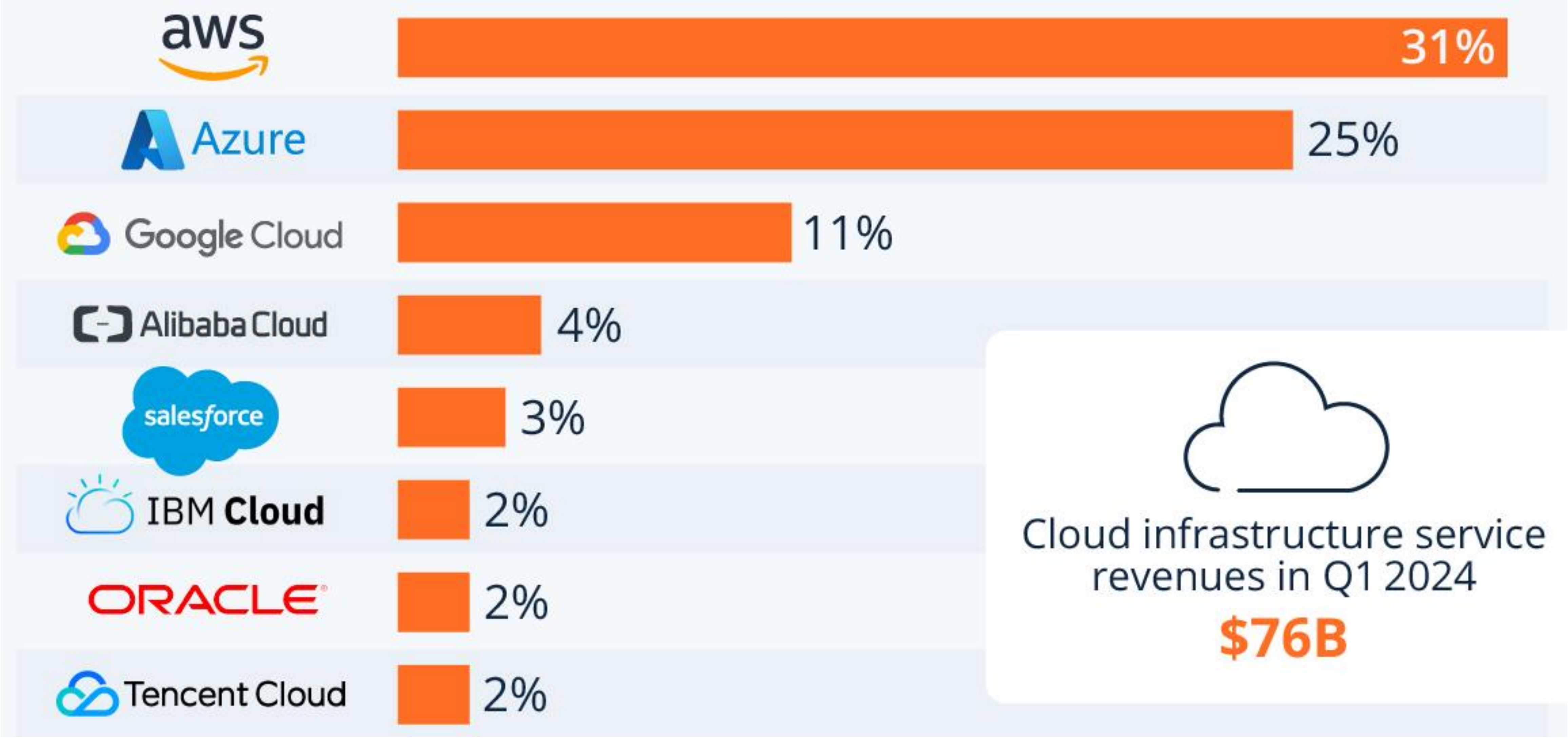
Challenges in Public Cloud

- ❑ **Security Concerns**: While public cloud providers offer strong security, the shared nature of the infrastructure can raise concerns about data privacy and compliance.
- ❑ **Control and Customization**: Public cloud environments may offer less control and customization compared to private cloud or on-premises solutions.
- ❑ **Cost Management**: Without proper management, cloud costs can escalate, especially with unpredictable usage patterns.

Challenges of Public Cloud

- ❑ **Performance and Reliability**: Even a little latency while loading an app or a web page can result in a huge drop in the percentage of user’s. Ex: Service outages, latency and bandwidth constrains
- ❑ **Vendor lock-in:** Use of proprietary technologies make it difficult to move workloads to another provider, limited interoperability
- ❑ **Data Governance**
- ❑ **Compliance and Legal Issues**
- ❑ **Skilled Workforce Shortage**

Public Cloud Service Providers (as of 2024)



Cloud infrastructure service revenues in Q1 2024

$76B

Problems for Practice

A startup wants to build and launch a highly customized e-commerce platform with full control over:

1. Operating system
2. Runtime environment
3. Security settings
4. Networking
5. Storage and scaling architecture

Recommend a suitable public cloud service for the above requirement and justify your answer.

References

https://www.ibm.com/blog/public-cloud-use-cases/

Module 3

Lecture 2

AWS Compute Services

AWS

Amazon Web Services offers a broad set of global cloud-based products including compute, storage, databases, analytics, networking, mobile, developer tools, management tools, IoT, security, and enterprise applications, on-demand, available in seconds, with pay-as-you-go pricing.

Started in 2006 to offer IT infrastructure services to businesses as web services

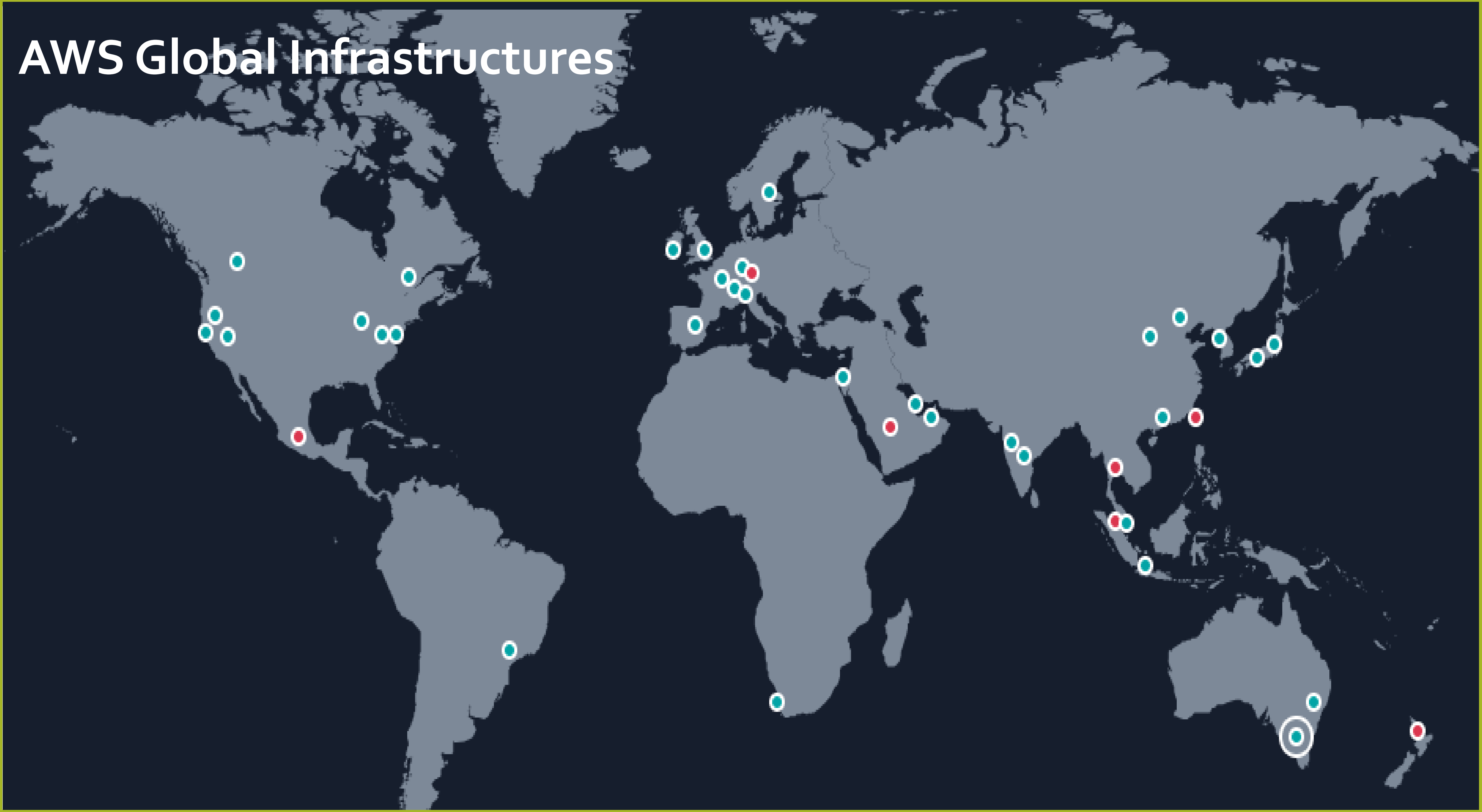
Amazon AWS

- ❑ With AWS, businesses no longer need to plan for and procure servers and other IT infrastructure weeks or months in advance.
- ❑ Instead, they can instantly spin up hundreds or thousands of servers in minutes and deliver results faster.
- ❑ AWS provides a highly reliable, scalable, low-cost infrastructure platform in the cloud
- ❑ It powers hundreds of thousands of businesses in 190 countries around the world.

AWS Global Infrastructure

- ❑ The AWS Cloud infrastructure is built around AWS Regions and Availability Zones.
- ❑ An AWS Region is a physical location in the world with multiple Availability Zones.
- ❑ Availability Zones consist of one or more discrete data centers, each with redundant power, networking, and connectivity, housed in separate facilities.
- ❑ These Availability Zones offer you the ability to operate production applications and databases
- ❑ There are 33 regions and 105 availability zones

AWS Global Infrastructures



Benefits of AWS

1. **Keep Your data safe** — The AWS infrastructure puts strong safeguards in place to help protect your privacy. All data is stored in highly secure AWS data centers.
2. **Meet compliance requirements** — AWS manages dozens of compliance programs in its infrastructure. This means that segments of your compliance have already been completed.
3. **Save money** — Cut costs by using AWS data centers. Maintain the highest standard of security without having to manage your own facility
4. **Scale quickly** — Security scales with your AWS Cloud usage. No matter the size of your business, the AWS infrastructure is designed to keep your data safe.

AWS Security and Compliance

AWS builds security into the core of the cloud infrastructure, and offers foundational services to help organizations meet their unique security requirements in the cloud.

Security in the cloud is much like security in your on-premises data centers—only without the costs of maintaining facilities and hardware.

Software-based security tools are used to monitor and protect the flow of information into and out of your cloud resources.

The AWS Cloud enables a shared responsibility model. While AWS manages security of the cloud, you are responsible for security in the cloud.

AWS Security and Compliance

AWS provides security-specific tools and features across network security, configuration management, access control, and data encryption.

The IT infrastructure that AWS provides to its customers is designed and managed in alignment with best security practices and a variety of IT security standards.

AWS environments are continuously audited, with certifications from accreditation bodies across geographies.

Compliance – management, operational and functional.

AWS Compute Services

1. Amazon EC2

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

It enables businesses to run application programs in the AWS public cloud.

The simple web interface of Amazon EC2 allows you to obtain and configure the required computing resources.

It provides you with complete control of your computing resources and lets you run on Amazon’s proven computing environment.

It reduces the time required to obtain and boot new server instances to minutes, allowing you to quickly scale capacity, both up and down, as your computing requirements change.

Amazon EC2

EC2 offers the broadest and deepest compute platform, with over 750 instances and choice of the latest processor, storage, networking, and operating system.

It supports Intel, AMD, and Arm processors, the only cloud with on-demand EC2 Mac instances, and the only cloud with 400 Gbps Ethernet networking.

It provides developers and system administrators the tools to build failure resilient applications and isolate themselves from common failure scenarios.

Amazon EC2 Benefits

- ❑ SLA commitment of 99.99% availability
- ❑Provide secure compute for your applications using AWS Nitro System
- ❑Optimize performance and cost
- ❑AWS Migration Tools

Amazon EC2 instance types

An Amazon EC2 instance is a virtual server offering an appropriate mix of resources used to run applications on the AWS infrastructure without having to purchase any hardware.

These instances, which differ in terms of the central processing unit (CPU), memory, storage and networking capacity.

1. General purpose – T2, T3
2. Compute optimized – C4, C5
3. Memory optimized – R4, R5
4. Accelerated computing – P4, P5
5. Storage optimized – D2, D3
6. HPC optimized

Amazon EC2 instance types

1. General Purpose

General purpose instances provide a *balance of compute, memory and networking resources*, and can be used for a variety of diverse workloads.

These instances are ideal for applications that use these resources in equal proportions such as *web servers and code repositories*.

Each general purpose instance comes with different processors, memory (DDR5), networking and storage.

Example instances: M7g, M7i, T2, T3

Amazon EC2 instance types

2. Compute Optimized

Compute Optimized instances are ideal for compute bound applications that benefit from high performance processors.

Example applications: *batch processing workloads, media transcoding, high performance web servers, high performance computing (HPC), scientific modeling, dedicated gaming servers and ad server engines, machine learning inference and other compute intensive applications.*

Example instances: C4, C5, C6i

Amazon EC2 instance types

3. Memory Optimized

Memory optimized instances are designed to deliver fast performance for workloads that process large data sets in memory.

Example instances: R4, R5, R7g, R8g

4. Accelerated Computing

Accelerated computing instances use hardware accelerators, or co-processors, to perform functions, such as floating point number calculations, graphics processing, or data pattern matching, more efficiently than is possible in software running on CPUs.

Example instances: P2, P3, P4, P5

Amazon EC2 instance types

5. Storage Optimized

Storage optimized instances are designed for workloads that require high, sequential read and write access to very *large data sets on local storage*.

They are optimized to deliver *tens of thousands of low-latency, random I/O operations per second (IOPS)* to applications.

Example instances: D2, D3, 14g

Amazon EC2 instance types

6. HPC Optimized

High performance computing (HPC) instances are purpose built to offer the best price performance for running HPC workloads at scale on AWS.

HPC instances are ideal for applications that benefit from *high-performance processors* such as large, complex simulations and deep learning workloads.

Example instances: HPC6a, HPC7a, HPC7g

Steps to create an instance using EC2

Step 1. Select a region
Step 2. Navigate to the EC2 Console
Step 3. Create the EC2 instance
Step 4. Choose an instance type
Step 5. Configure storage
Step 6. Tag the instance
Step 7. Build in security
Step 8. Enable SSH access with a key

Amazon EC2 Spot Instances

- ❑ Amazon EC2 Spot Instances let you take advantage of unused EC2 capacity in the AWS cloud and are available at up to a 90% discount compared to On-Demand prices.

- ❑ Cloud provider must have spare capacity available for any surge in customer demand. To offset the loss of this idle infrastructure, AWS offers this excess capacity at a massive discount to drive usage.

- ❑ Use Spot Instances for various stateless, fault-tolerant, or flexible applications such as big data, containerized workloads, CI/CD, web servers, high-performance computing (HPC), and test & development workloads.

Amazon EC2 Spot Instances Pricing

❑ AWS's spot instance pricing can be viewed at *pricing page* as well as on the spot instance *advisor page*.

Two pricing options:

❑ ***Regular spot pricing***—instances can be terminated with 2 minutes notice.

❑ ***Defined duration***—you can get a spot instance guaranteed to run for a period of 1-6 hours. The longer the defined duration, the lower the discount provided for the spot instance. Defined duration instances grant discounts of 30-50% vs On-Demand pricing.

Amazon EC2 Auto Scaling

- ❑ It helps you *maintain application availability* and lets you *automatically add or remove EC2 instances* using scaling policies that you define.

- ❑ It helps you to ensure that you have the correct number of Amazon EC2 instances available to handle the load for your application

- ❑ Dynamic or predictive scaling policies let you add or remove EC2 instance capacity to service established or real-time demand patterns.

- ❑ The fleet management features of Amazon EC2 Auto Scaling help maintain the health and availability of your fleet.

Amazon EC2 Auto Scaling

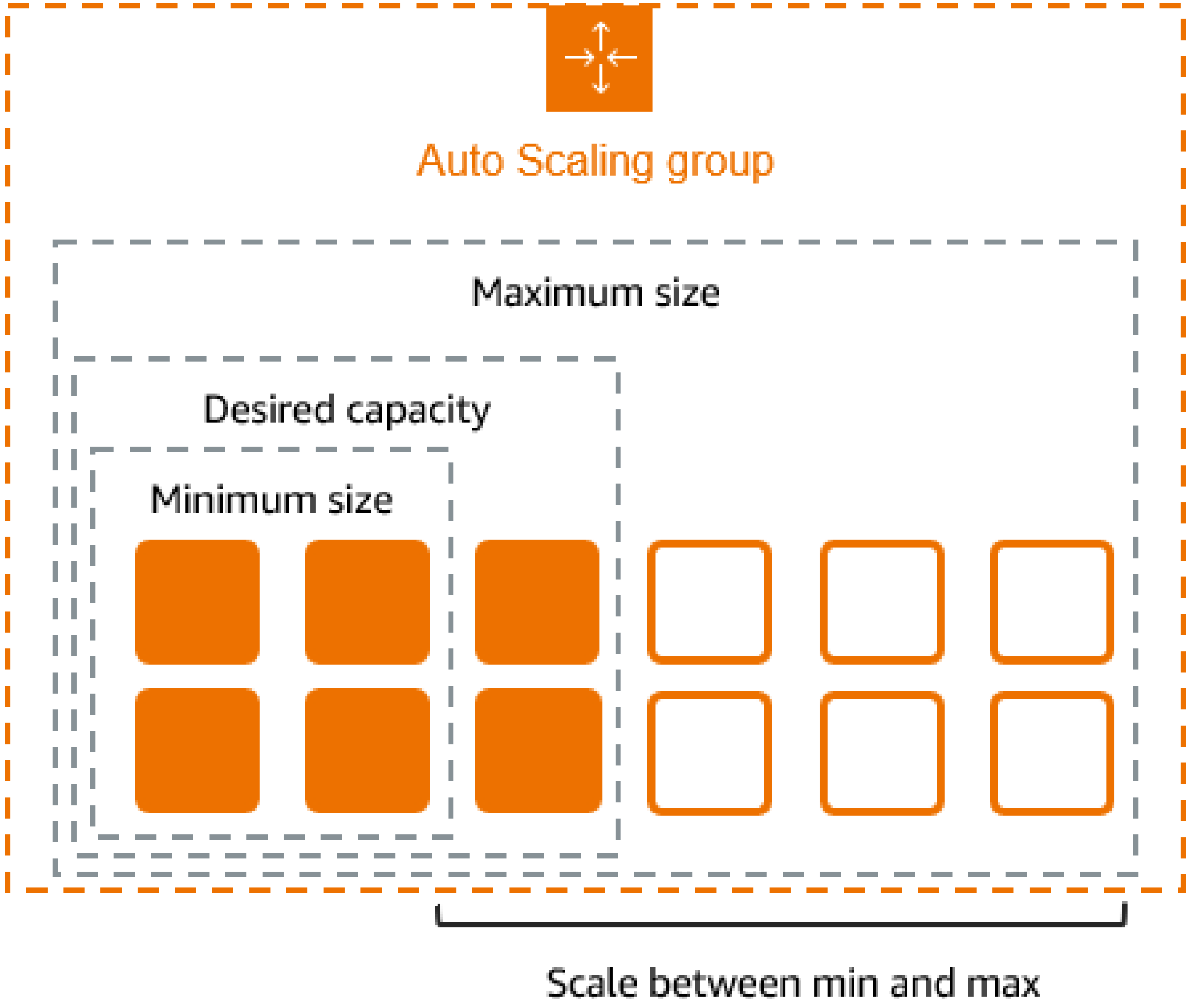
- ❑ For example, create an Auto Scaling groups i.e. collections of EC2 instances.

- ❑ Specify the minimum/maximum number of instances in each Auto Scaling group, and Auto Scaling ensures that your group never goes below/above this size.

- ❑ If you specify the desired capacity, Auto Scaling ensures that your group has this many instances.

- ❑ Based on the scaling policies, Amazon EC2 Auto Scaling can launch or terminate instances as demand on your application increases or decreases.

Amazon EC2 Auto Scaling



Amazon EC2 Auto Scaling Features

- ❑ Monitoring the health of running instances
- ❑ Custom health checks
- ❑ Balancing capacity across Availability Zones
- ❑ Multiple instance types and purchase options
- ❑ Automated replacement of Spot Instances
- ❑ Load balancing
- ❑ Scalability

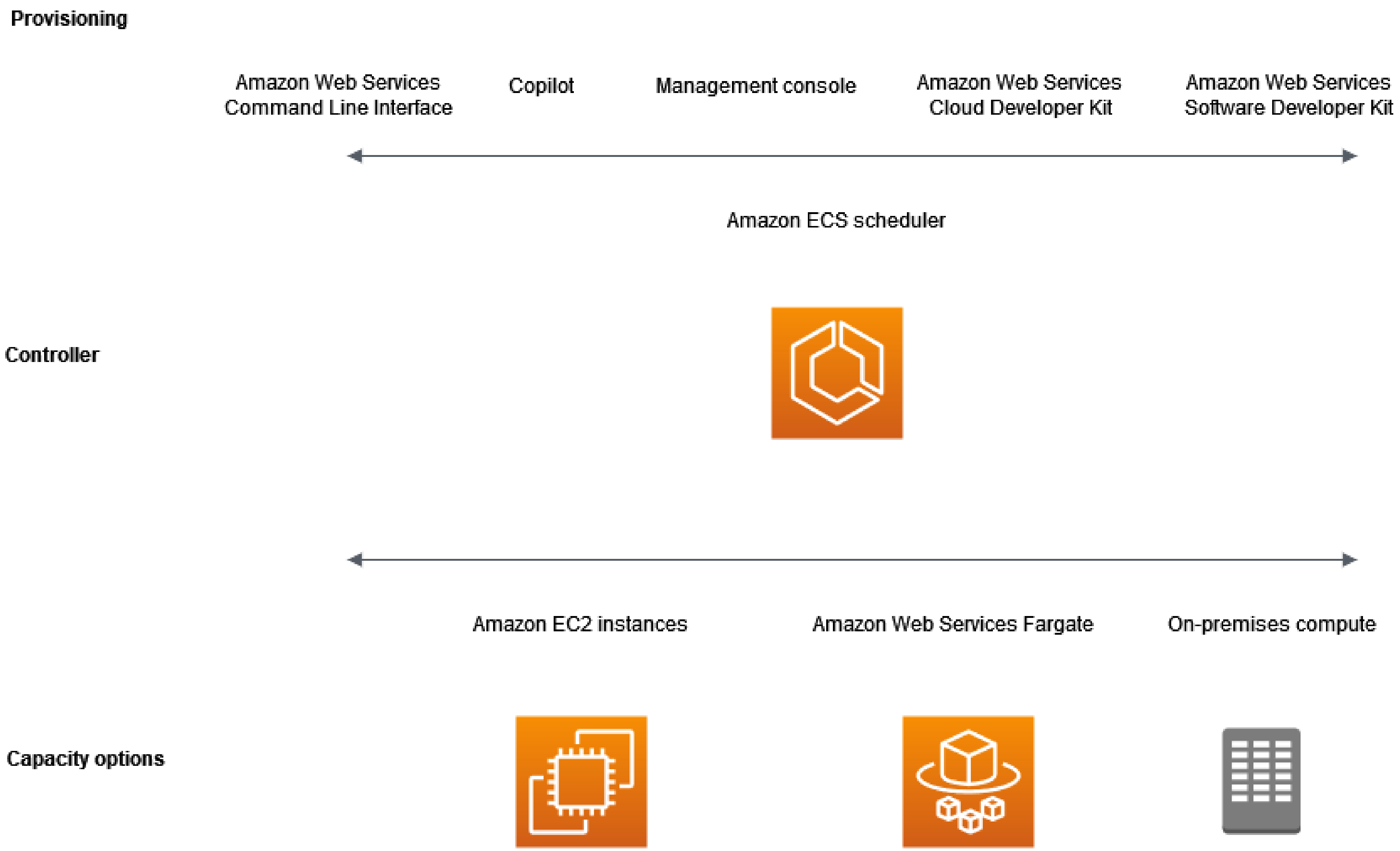
Amazon Elastic Container Service (ECS)

Amazon Elastic Container Service (Amazon ECS) is a fully managed container orchestration service that helps you *easily deploy, manage, and scale containerized applications*.

It's integrated with both *AWS and third-party tools*, such as Amazon Elastic Container Registry and Docker. This integration makes it easier for teams to focus on building the applications, not the environment.

A container is a standardized unit of software development that holds everything that your software application requires to run. This includes relevant code, runtime, system tools, and system libraries.

Amazon Elastic Container Service (ECS) Layers



Provisioning

Amazon Web Services Command Line Interface

Copilot

Management console

Amazon Web Services Cloud Developer Kit

Amazon Web Services Software Developer Kit

Amazon ECS scheduler

Controller

Amazon EC2 instances

Amazon Web Services Fargate

On-premises compute

Capacity options

Amazon Elastic Container Service (ECS) layers

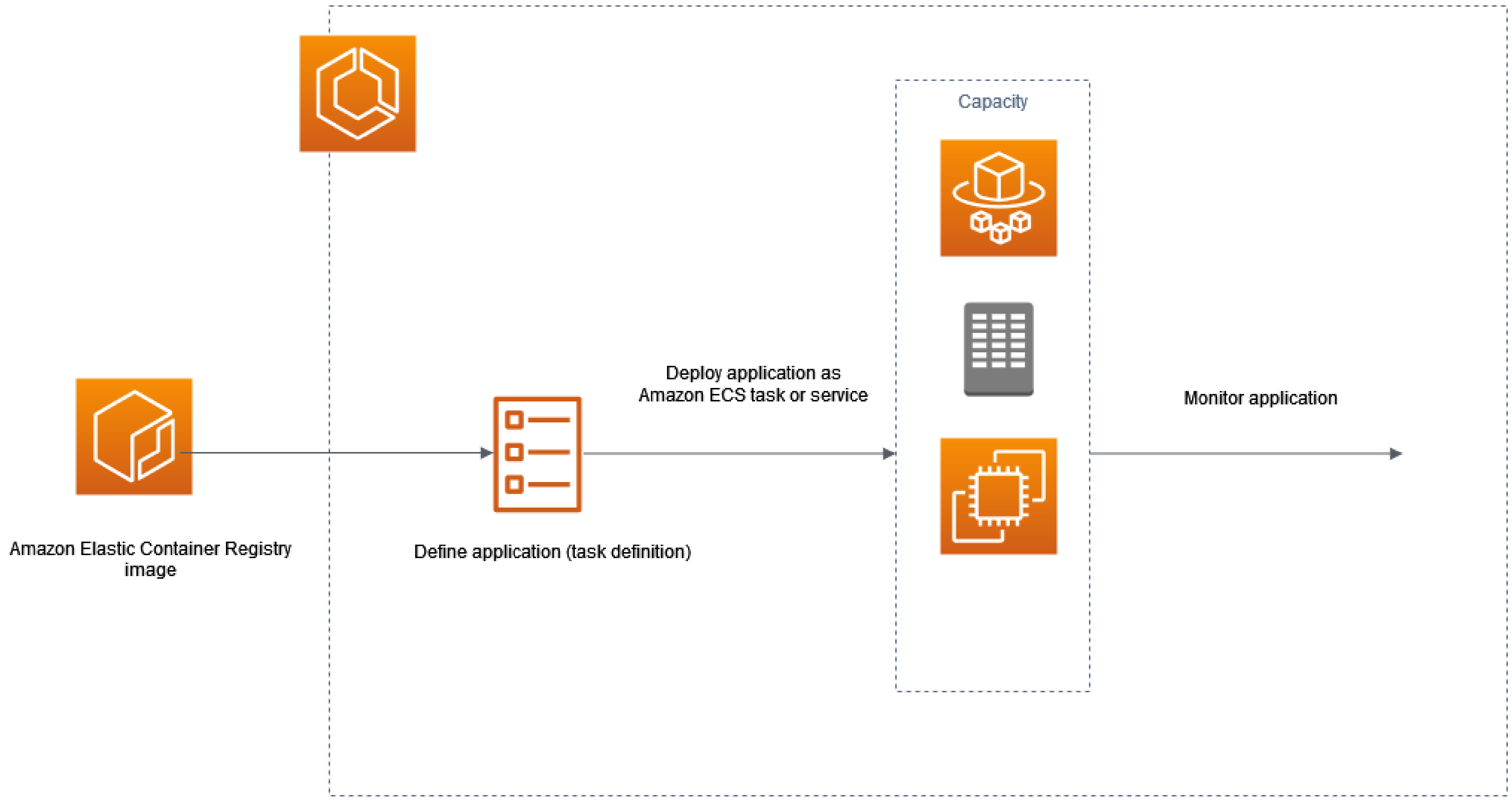
There are three layers in Amazon ECS:

- ❑ Capacity - The infrastructure where your containers run

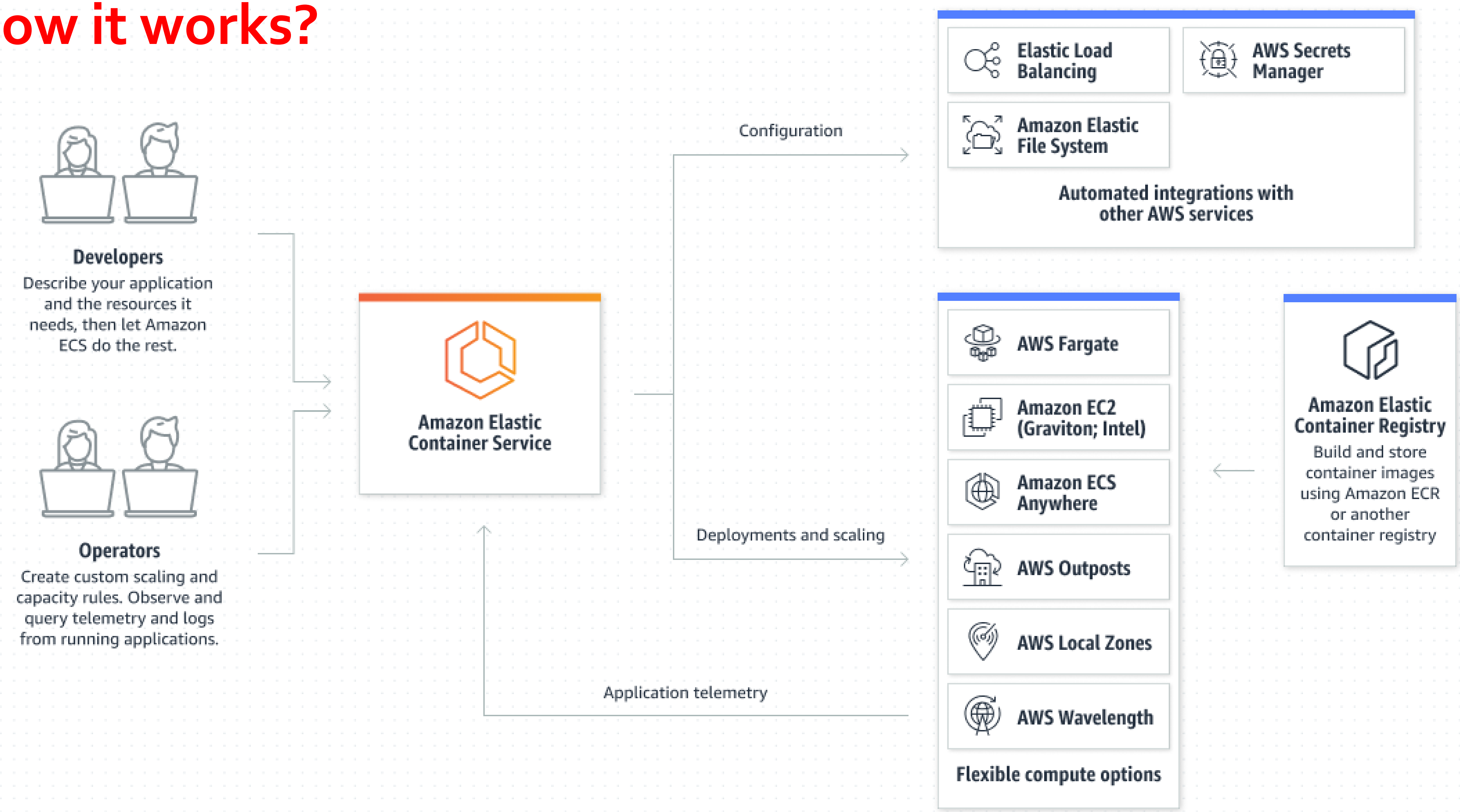
- ❑ Controller - Deploy and manage your applications that run on the containers. It is the software that manages your applications.

- ❑ Provisioning - The tools that you can use to interface with the scheduler to deploy and manage your applications and containers

Amazon ECS – Application Life Cycle


Capacity
Amazon Elastic Container Registry image
Define application (task definition)
Deploy application as Amazon ECS task or service
Monitor application

How it works?



Developers
Describe your application and the resources it needs, then let Amazon ECS do the rest.

Operators
Create custom scaling and capacity rules. Observe and query telemetry and logs from running applications.

Amazon Elastic Container Service

Configuration

Elastic Load Balancing

AWS Secrets Manager

Amazon Elastic File System

Automated integrations with other AWS services

Deployments and scaling

AWS Fargate

Amazon EC2 (Graviton; Intel)

Amazon ECS Anywhere

AWS Outposts

AWS Local Zones

AWS Wavelength

Flexible compute options

Amazon Elastic Container Registry
Build and store container images using Amazon ECR or another container registry

Application telemetry

Amazon Elastic Container Service (ECS) Benefits

- ❑ Launch containers on AWS
- ❑ Reduce costs with automatic scaling and pay-as-you-go pricing
- ❑ Deploy faster and focus on your applications using AWS Fargate serverless compute for containers
- ❑ Optimize for security and compliance

AWS Elastic Beanstalk

It is an easy-to-use service for *deploying and scaling web applications* and services developed with Java, .NET, .NET Core, PHP, Node.js, Python, Ruby, Go, or Docker.

Elastic Beanstalk is a complete application management solution that manages all infrastructure and platform tasks. It reduces management complexity for hosting the web applications.

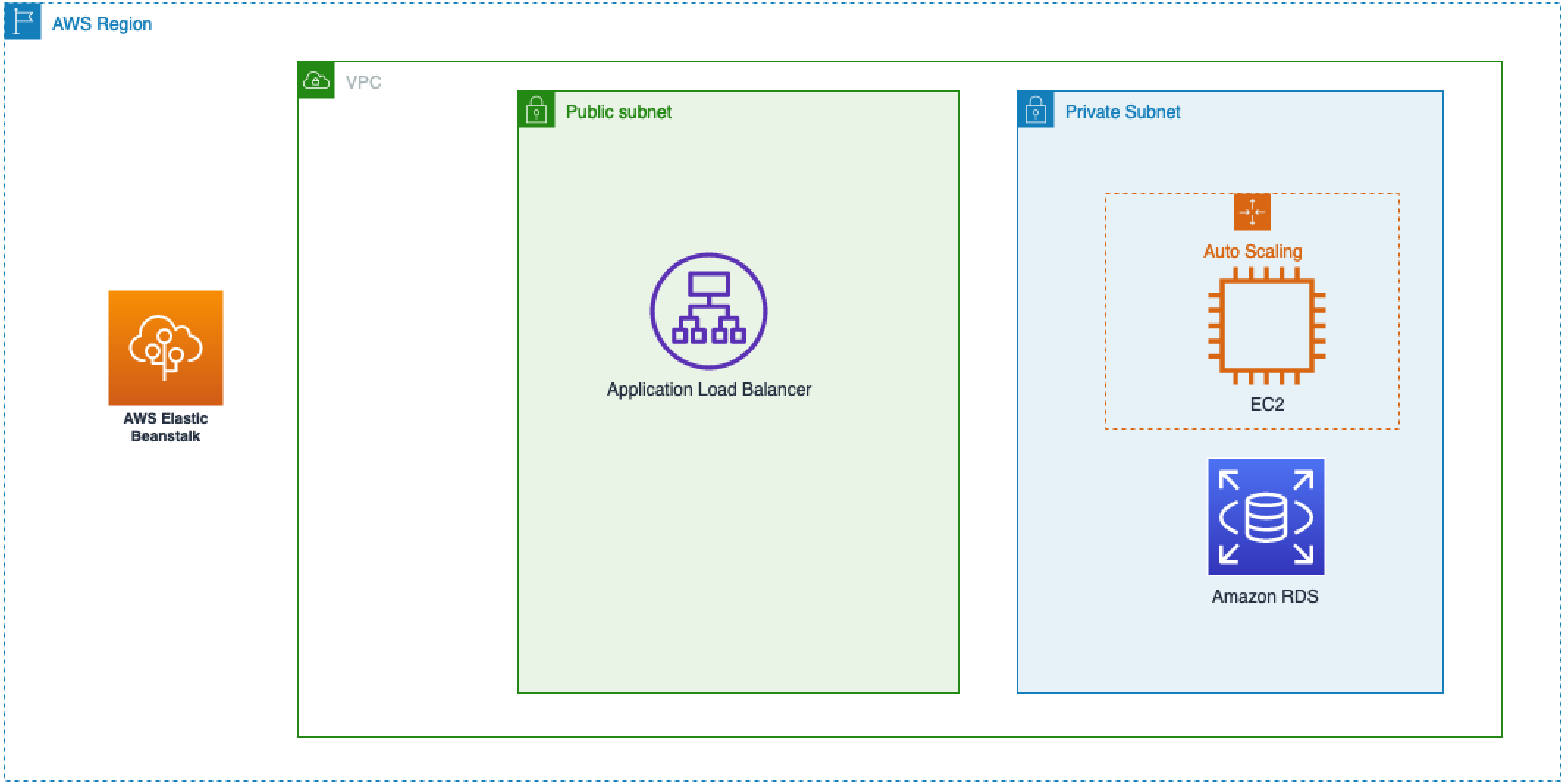
AWS Elastic Beanstalk

For deployment, simply upload your source code and Elastic Beanstalk will provision and operate all necessary infrastructure, including servers, databases, load balancers, networks, and auto scaling groups.

- Full control and customize the resources as needed

- It can process regulated financial data or protected health information (PHI).

AWS Elastic Beanstalk Usecase



AWS Elastic Beanstalk Benefits

- ❑ Upload and deploy web applications in a simplified, fast way.
- ❑ Focus on writing code instead of provisioning and managing infrastructure.
- ❑ Full control of the optimal AWS resources
- ❑ Scale your application for handling peaks in traffic

AWS Lambda

- ❑ It is used to run code without provisioning or managing servers.
- ❑ It runs your code on a high-availability compute infrastructure and performs all of the administration of the compute resources, including server and operating system maintenance, capacity provisioning and automatic scaling, and logging.
- ❑ The Lambda service runs your function only when needed and scales automatically.

When to use AWS Lambda

❑ Lambda is an ideal compute service for application scenarios that need to scale up rapidly, and scale down to zero when not in demand.

For example, you can use Lambda for:

❑ File processing: Use Amazon Simple Storage Service (Amazon S3) to trigger Lambda data processing in real time after an upload.

❑ Web applications: Combine Lambda with other AWS services to build powerful web applications that automatically scale up and down and run in a highly available configuration across multiple data centers.

References

1. https://docs.aws.amazon.com/autoscaling/ec2/userguide/what-is-amazon-ec2-auto-scaling.html
2. https://docs.aws.amazon.com/AmazonECS/latest/developerguide/Welcome.html
3. https://docs.aws.amazon.com/whitepapers/latest/overview-deployment-options/aws-elastic-beanstalk.html

Module 3

Lecture 2

AWS Storage Services

AWS Storage Services

A cloud computing provider manages and runs data storage as a service model for storing data online.

It eliminates the need to purchase and manage your own data storage infrastructure because it is offered on-demand with just-in-time capacity and pricing.

Cloud Storage Types

1. Object Storage

- ❑ Object storage allows any data for the desired duration, facilitates easy retrieval of data, and is ideal for unstructured and binary data

- ❑ It adapts to frequent component failures of systems via continual monitoring, fault tolerance error detection, automatic recovery.

- ❑ Object storage can accommodate massive data sets and files.

Cloud Storage Types

1. Object Storage

- ❑ It is highly scalable, distributed, and more efficient

- ❑ Few use cases includes data lakes, metadata, backup and archival systems, data analytics, machine, and deep learning, source code management, disaster recovery, continuous integration and delivery pipelines, websites, and documentation.

Cloud Storage Types

2. File Storage

- ❑ The data is stored in files. These files are then sorted and set up in folders arranged into directories, and subdirectories
- ❑ Files in file storage are generally easy to name, delete, or customize as needed
- ❑ As a fully managed network-attached storage solution, it is ideal for unstructured data/shared file storage.

Cloud Storage Types

2. File Storage

- ❑ File storage facilitates sharing and collaboration.
- ❑ Common uses for file storage are storage for office directories in content repositories, application migration, media processing, machine learning, and data storage that need data protection and easy deployment capabilities

Cloud Storage Types

3. Block Storage

- ❑ In this type of cloud storage, data is divided into sections called blocks and stored in a system that can be physically distributed.
- ❑ Each block has a unique identifier, allowing the system to track and assemble them as needed.
- ❑ Many cloud enterprise workloads are currently run using block storage.

Cloud Storage Types

3. Block Storage

- ❑ It offers low latency and high performance and is Ideal for VMs and stateless workloads.
- ❑ A block storage system is used in cases where quick retrieval and manipulation of data are needed.
- ❑ Some use cases include VM disks, scale-out analytics, media rendering, database storage, and flash-optimized databases.

Why Cloud Storage on AWS?

❑ Make resources available in minutes. Speed time to market, avoid complex capacity planning, and reduce over-provisioning with just a few clicks.

❑ Secure and protect your data

❑ Minimize your total cost of ownership (TCO) with managed services that eliminate infrastructure maintenance.

❑ Choose from a variety of tools to get more from your data and accelerate new product and service delivery.

Amazon Simple Storage Service (Amazon S3)

Amazon Simple Storage Service (S3) is an object storage service that offers industry-leading *scalability, data availability, security, and performance*.

Customers of all sizes and industries can use it to store and protect any amount of data for a range of use cases

The service is designed for *online backup and archiving of data and applications* on Amazon Web Services.

It provides easy-to-use management features so you can organize your data and configure finely-tuned access controls to meet your specific business, organizational, and compliance requirements

Amazon Simple Storage Service (Amazon S3)

Amazon S3 is designed for 99.999999999% of durability, and stores data for millions of applications for companies all around the world.

An administrator can also link S3 to other AWS security and monitoring services

How S3 works?

Amazon S3 is an object storage service that stores data as objects within buckets.

An *object* is a file and any metadata that describes the file. Objects consist of object data and metadata. The metadata is a set of name-value pairs that describe the object.

A *bucket* is a container for objects. Store any number of objects in a bucket and can have up to 100 buckets in your account.

To store your data in Amazon S3, create a bucket and specify a bucket name and AWS Region.

How S3 works?

Then, upload the data to that bucket as objects in Amazon S3. Each object has a *key* (or *key name*), which is the unique identifier for the object within the bucket.

To list and access your Amazon S3 buckets, you can use various tools such as Amazon S3 console, AWS CLI and Amazon S3 REST API

To list all of your buckets, you must have the s3:ListAllMyBuckets permission.

To access a bucket, make sure to also obtain the required AWS Identity and Access Management (IAM) permissions to list the contents of the specified bucket.

How S3 works?

Buckets also organize the Amazon S3 namespace at the highest level.

It provide access control options, such as bucket policies, access control lists (ACLs), and S3 Access Points, that you can use to manage access to your Amazon S3 resources.

S3 Versioning allows to keep multiple versions of an object in the same bucket, which allows to restore objects that are accidentally deleted or overwritten.

The combination of a bucket, object key, and optionally, version ID uniquely identify each object.

AWS S3 Data Migration

Data can be transferred to S3 over the public internet via access to S3 application programming interfaces (APIs).

Amazon S3 Transfer Acceleration for faster movement over long distances, as well as AWS Direct Connect for a private, consistent connection between S3 and an enterprise's own data center.

An administrator can also use *AWS Snowball*, a physical transfer device, to ship large amounts of data from an enterprise data center directly to AWS, which will then upload it to S3.

Users can integrate other AWS services with S3 easily. For example, an analyst can query data directly on S3 using Amazon Athena.

AWS S3 Storage classes

It offers a range of storage classes that you can choose from based on the performance, data access, resiliency, and cost requirements of your workloads.

S3 storage classes are purpose-built to provide the lowest cost storage for different access patterns.

S3 storage classes are ideal for virtually any use case, including those with demanding performance needs, data lakes, or archival storage.

Example: S3 Standard for frequently accessed data, S3 Express One Zone for your most frequently accessed data, Amazon S3 Glacier Deep Archive for long-term archive

Amazon Elastic File System

Amazon Elastic File System

It provides a simple, scalable, elastic file system for Linux-based workloads for use with AWS Cloud services and on-premises resources.

It is built to scale on demand to petabytes without disrupting applications, growing and shrinking automatically as you add and remove files.

It is designed to provide massively parallel shared access to thousands of Amazon EC2 instances, to achieve high levels of aggregate throughput and IOPS with consistent low latencies.

Amazon EFS is a fully managed service that requires no changes to your existing applications and tools, providing access through a standard file system interface

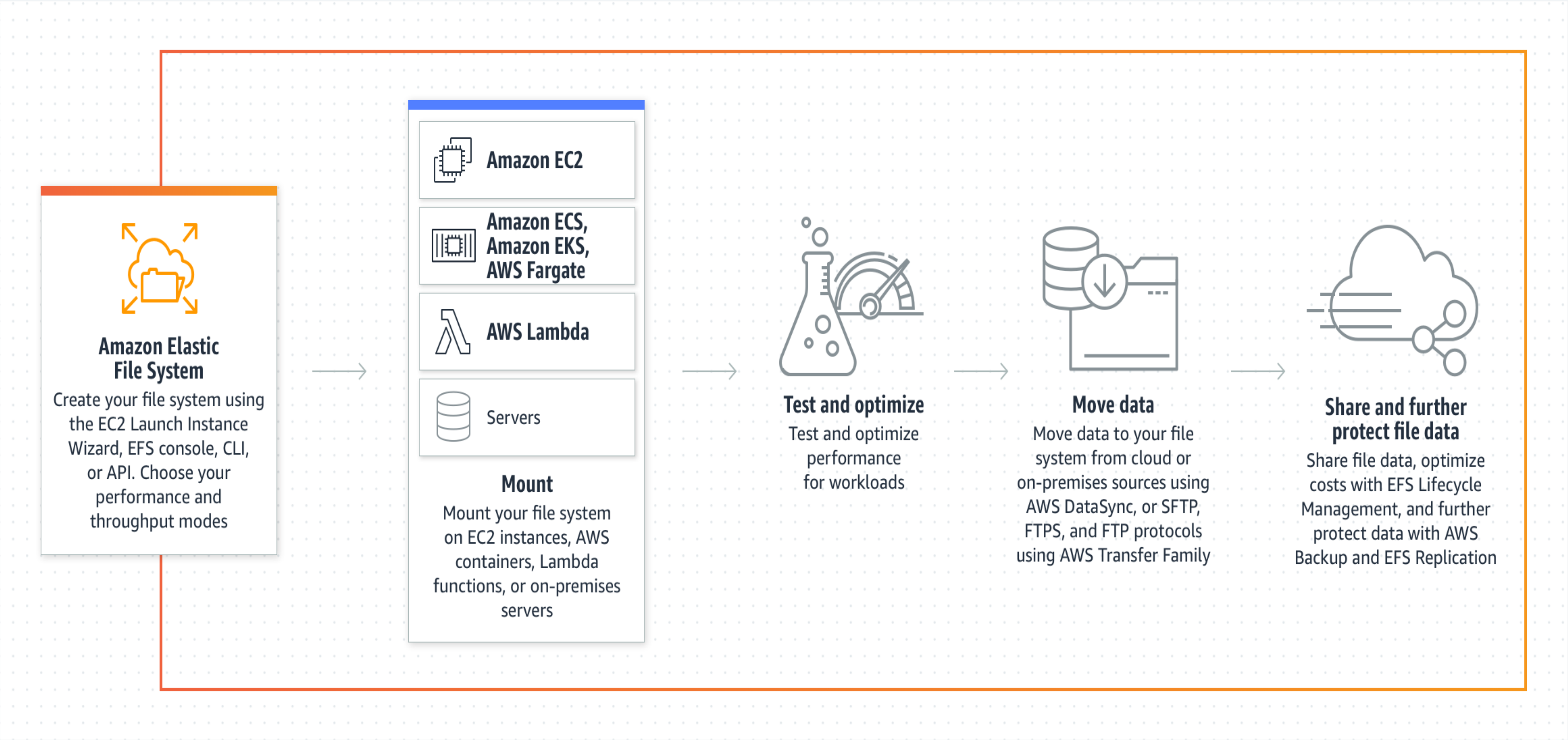
Amazon Elastic File System

File systems can be accessed across Availability Zones and AWS Regions

Share files between thousands of Amazon EC2 instances and on-premises servers via AWS Direct Connect or AWS VPN

Example Use cases: Enterprise applications, big data analytics, web serving and content management, application development and testing, media and entertainment workflows, database backups, and container storage.

Amazon Elastic File System – How it works?



Amazon Elastic File System

Create your file system using the EC2 Launch Instance Wizard, EFS console, CLI, or API. Choose your performance and throughput modes

- Amazon EC2
- Amazon ECS, Amazon EKS, AWS Fargate
- AWS Lambda
- Servers

Mount

Mount your file system on EC2 instances, AWS containers, Lambda functions, or on-premises servers

Test and optimize

Test and optimize performance for workloads

Move data

Move data to your file system from cloud or on-premises sources using AWS DataSync, or SFTP, FTPS, and FTP protocols using AWS Transfer Family

Share and further protect file data

Share file data, optimize costs with EFS Lifecycle Management, and further protect data with AWS Backup and EFS Replication

Amazon Elastic File System - Benefits

- ❑ Create and configure shared file systems simply and quickly
- ❑ Scale workloads on-demand to petabytes
- ❑ Reduce TCO with automatic lifecycle management
- ❑Securely and reliably access your files with a fully managed file system

Amazon Elastic Block Store

Amazon Elastic Block Store

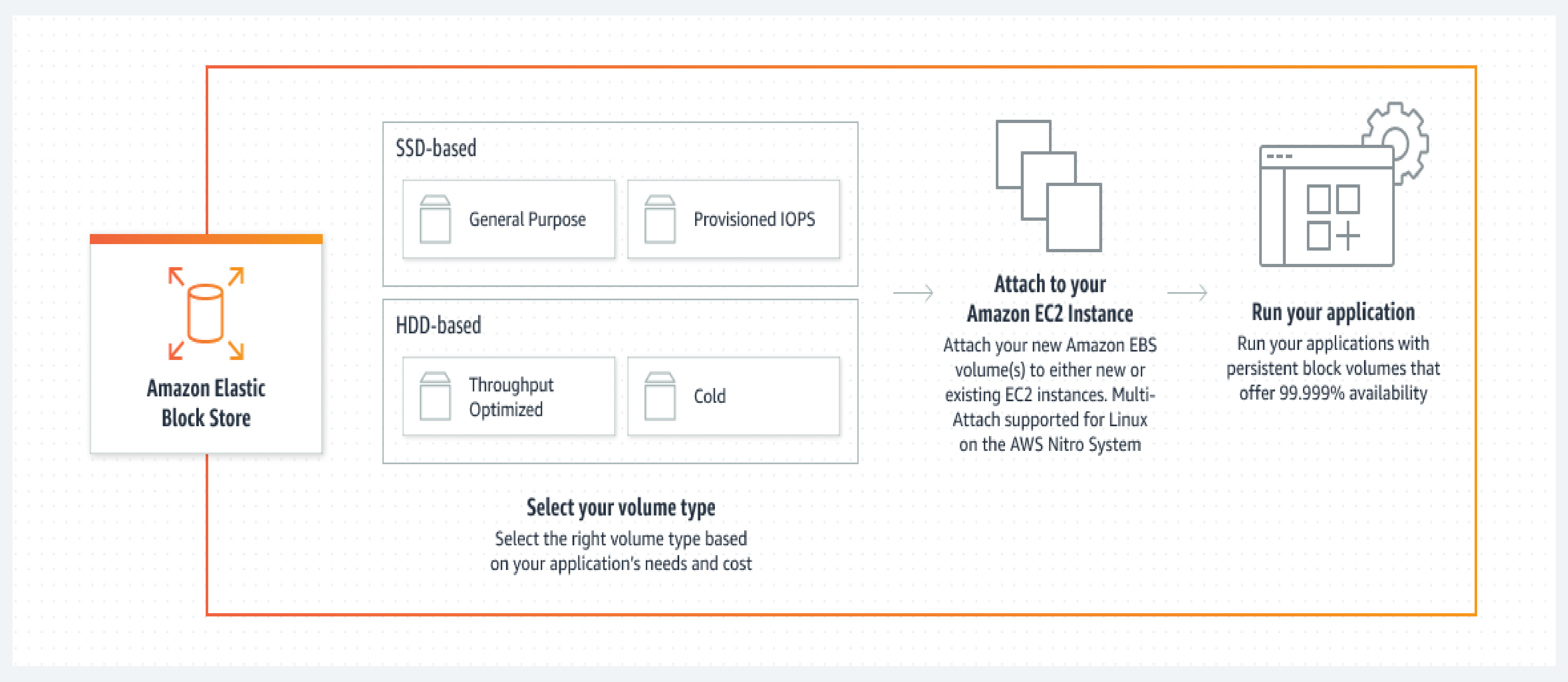
❑ Amazon Elastic Block Store (Amazon EBS) is an easy-to-use, scalable, high-performance block-storage service designed for Amazon Elastic Compute Cloud

❑ Each Amazon EBS volume is automatically replicated within its Availability Zone to protect from component failure, offering high availability and durability.

❑ Amazon EBS volumes offer the consistent and low-latency performance needed to run your workloads.

❑ Scale your usage up or down within minutes and pay a low price

Amazon Elastic Block Store – How it Works?



Select your volume type

Select the right volume type based on your application's needs and cost

Attach to your Amazon EC2 Instance

Attach your new Amazon EBS volume(s) to either new or existing EC2 instances. Multi-Attach supported for Linux on the AWS Nitro System

Run your application

Run your applications with persistent block volumes that offer 99.999% availability

Amazon Elastic Block Store – Benefits

- ❑ Scale fast for your most demanding, high-performance workloads
- ❑ Protect against failures with high availability
- ❑ Select the storage that best fits your workload
- ❑ Encrypt your block storage resources
- ❑ Prevent unauthorized access to your data by restricting public access and configuring locks on data backups
- ❑ Protect block storage data in the cloud, as well as on-premises block data using Amazon EBS Snapshots

References

1. https://docs.aws.amazon.com/whitepapers/latest/aws-overview/storage-services.html

Module 3

Lecture 3

AWS Network Services

Amazon VPC

Amazon VPC

Amazon Virtual Private Cloud (Amazon VPC) gives you full control over your virtual networking environment, including resource placement, connectivity, and security.

In addition, complete control on the selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.

Do provision a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network

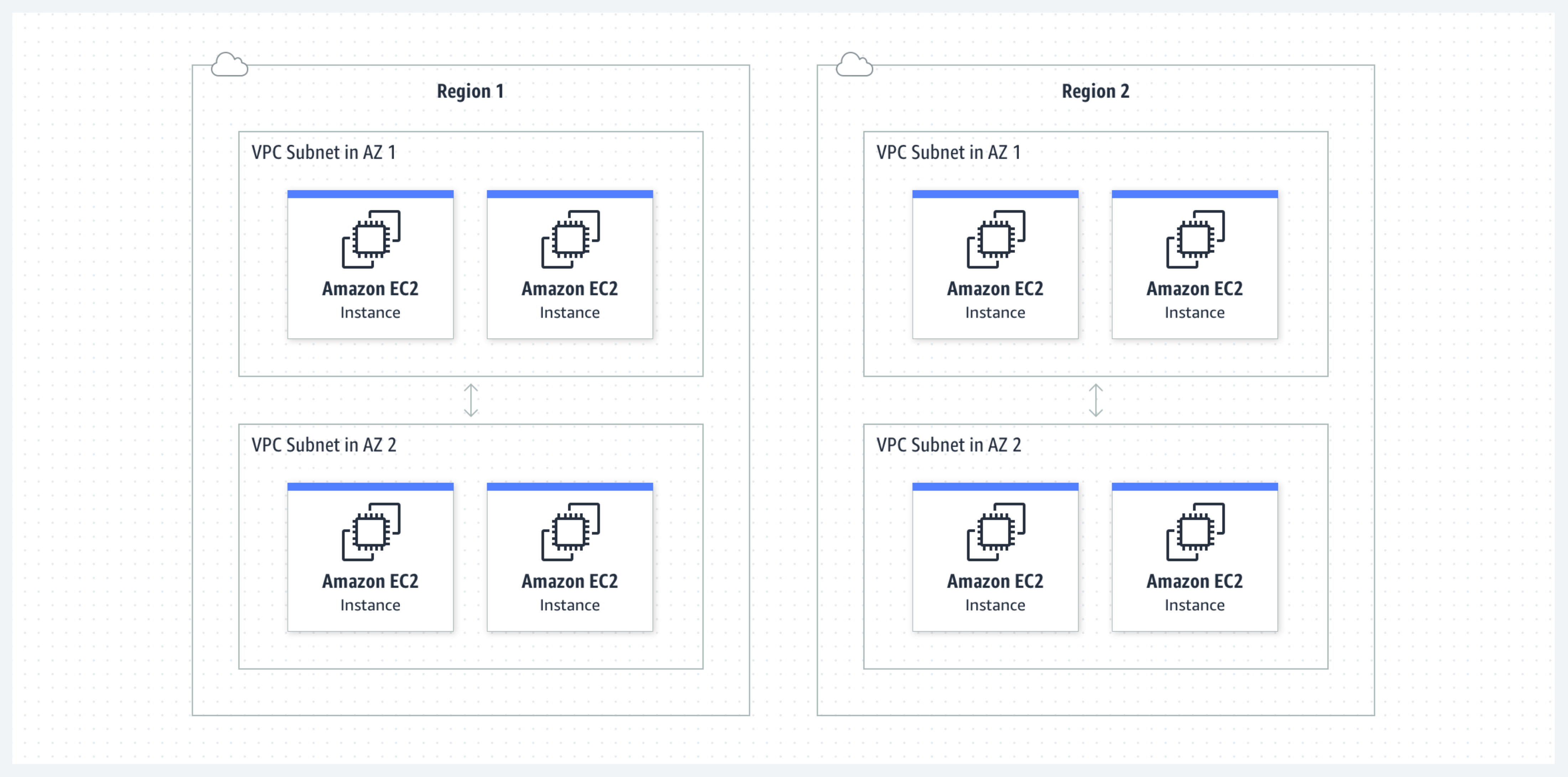
Amazon VPC

IPv4 and IPv6 can be used in your VPC for secure and easy access to resources and applications.

Easy customization of the network configuration in VPC

For example, create a public-facing subnet for your web servers that has access to the Internet, and place your backend systems, such as databases or application servers, in a private-facing subnet with no Internet access.

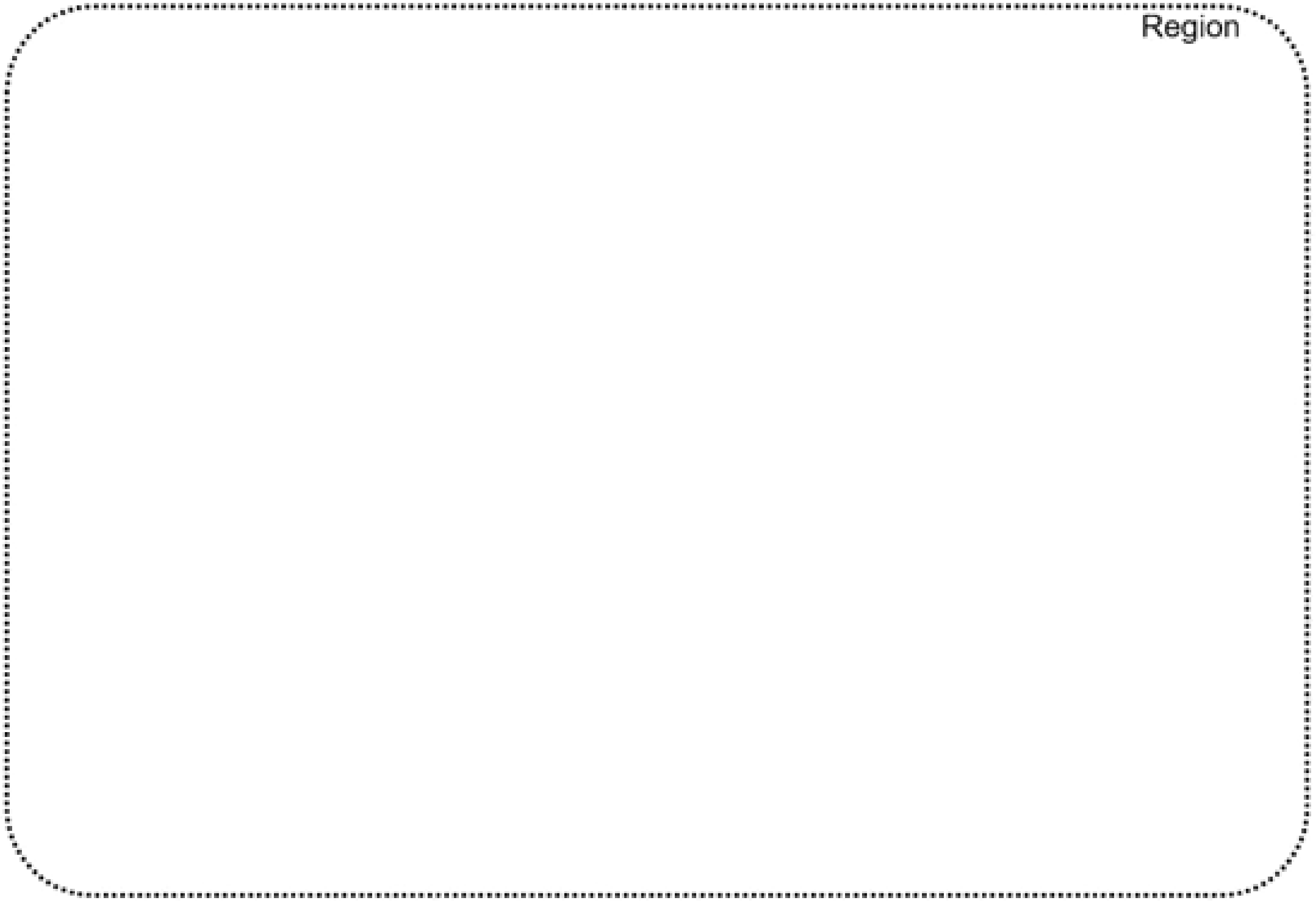
VPC - Example


Region 1
VPC Subnet in AZ 1
Amazon EC2
Instance
Amazon EC2
Instance
VPC Subnet in AZ 2
Amazon EC2
Instance
Amazon EC2
Instance
Region 2
VPC Subnet in AZ 1
Amazon EC2
Instance
Amazon EC2
Instance
VPC Subnet in AZ 2
Amazon EC2
Instance
Amazon EC2
Instance

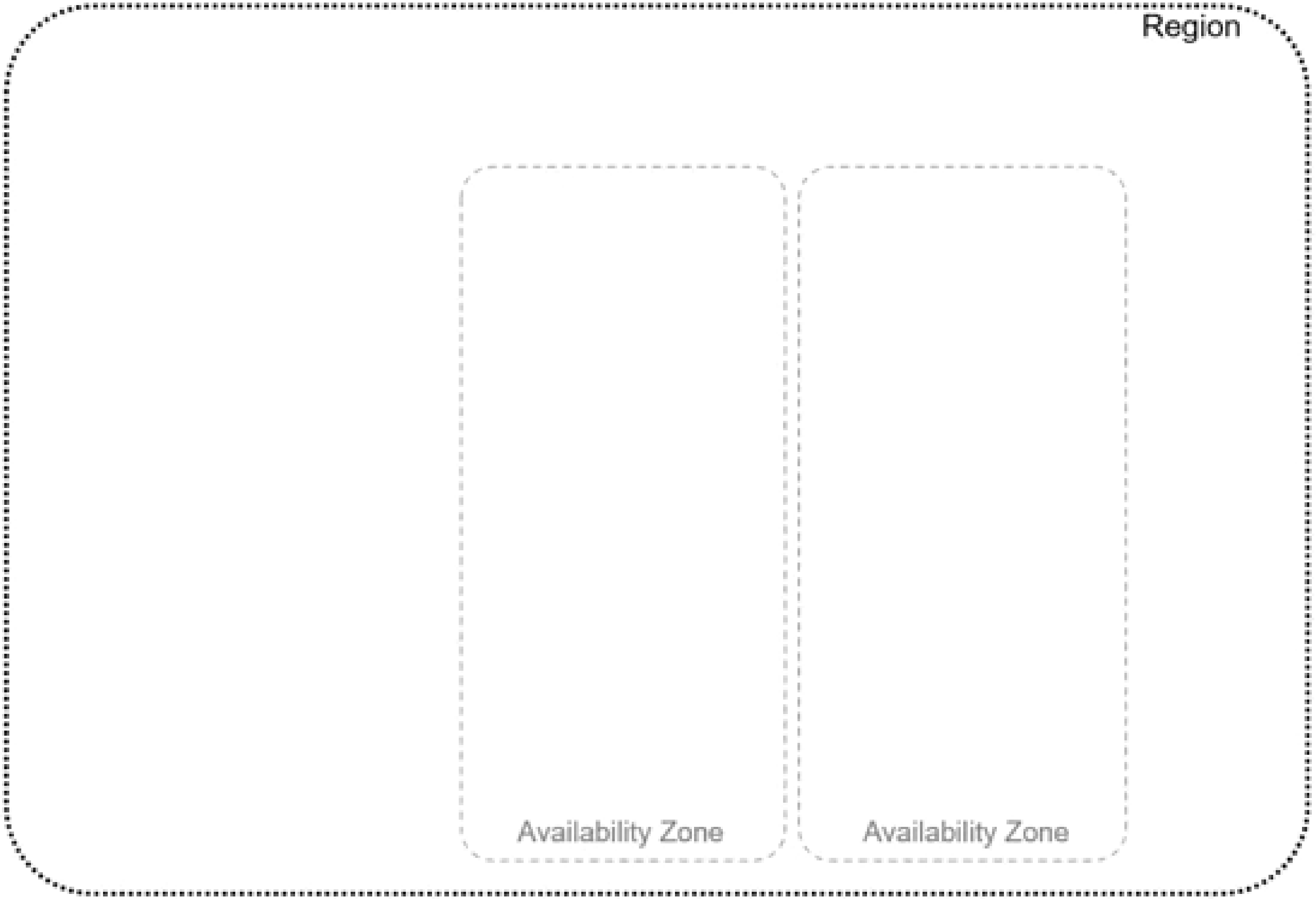
Steps to create a VPC

1. Set up a VPC in the AWS service console.
2. Add resources to it such as Amazon Elastic Compute Cloud (EC2) and Amazon Relational Database Service (RDS) instances.
3. Define how your VPCs communicate with each other across accounts, Availability Zones, or AWS Regions

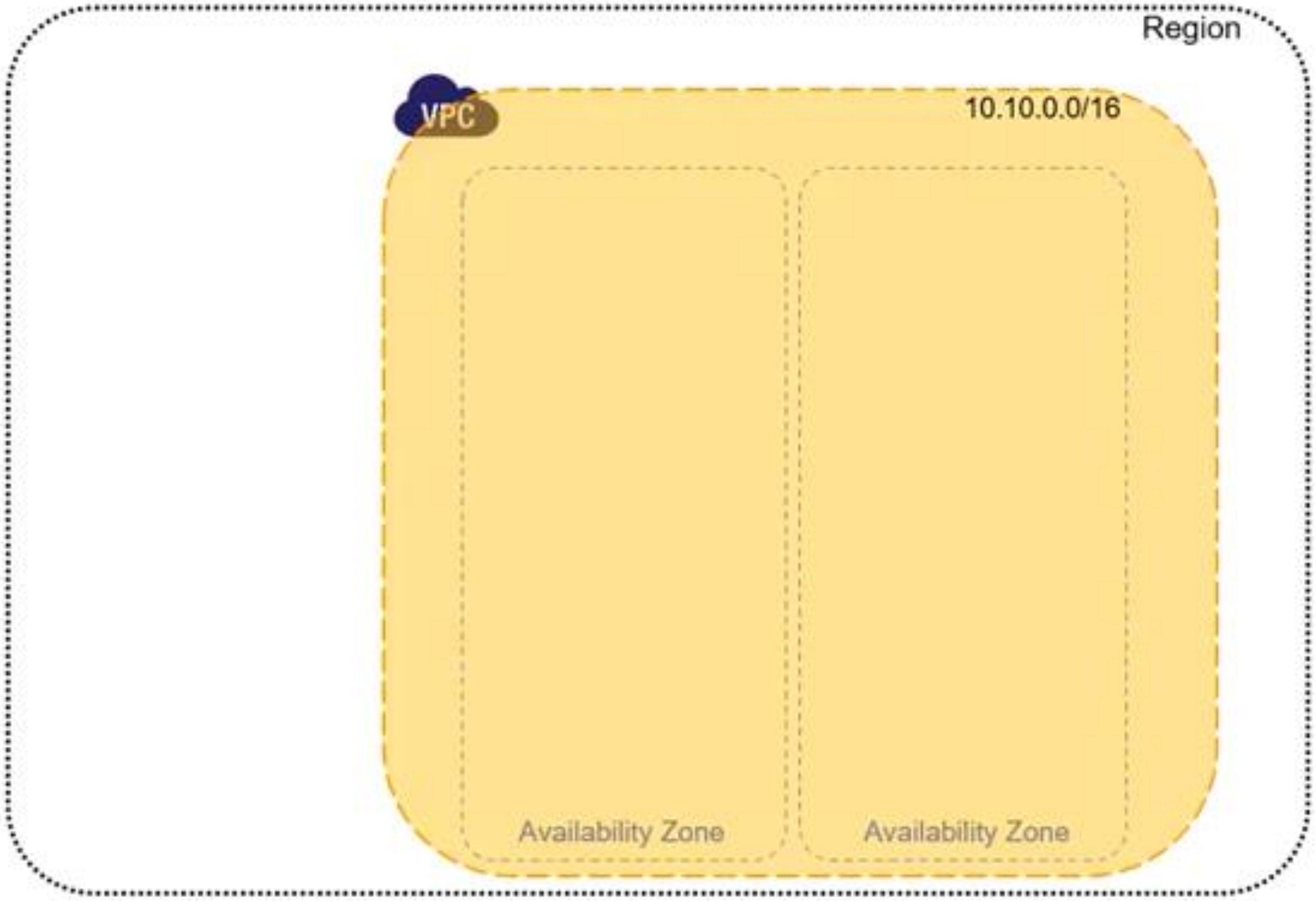
Networking with Virtual Private Cloud



Networking with Virtual Private Cloud

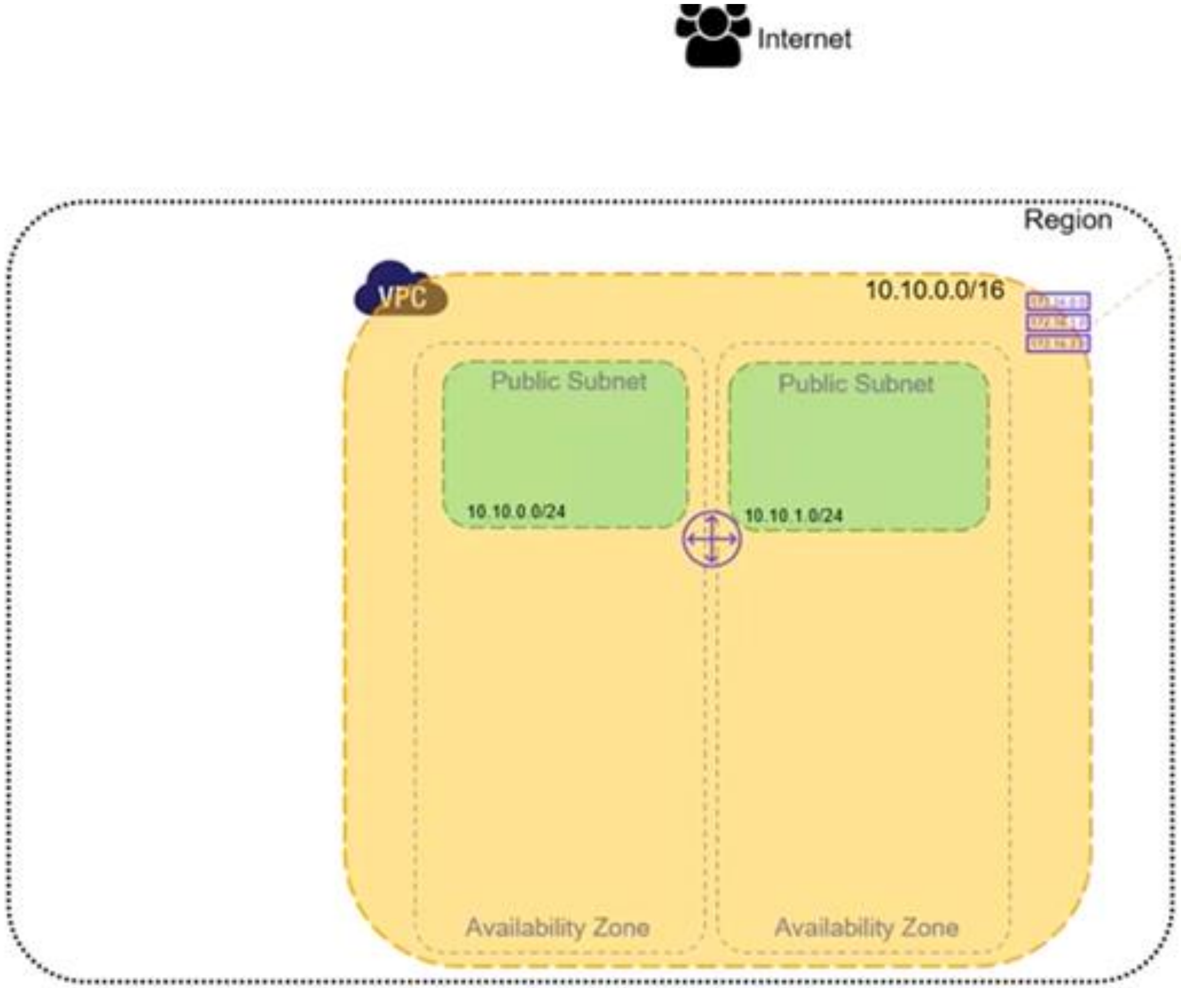


Networking with Virtual Private Cloud



CIDR - Classless Inter-Domain Routing

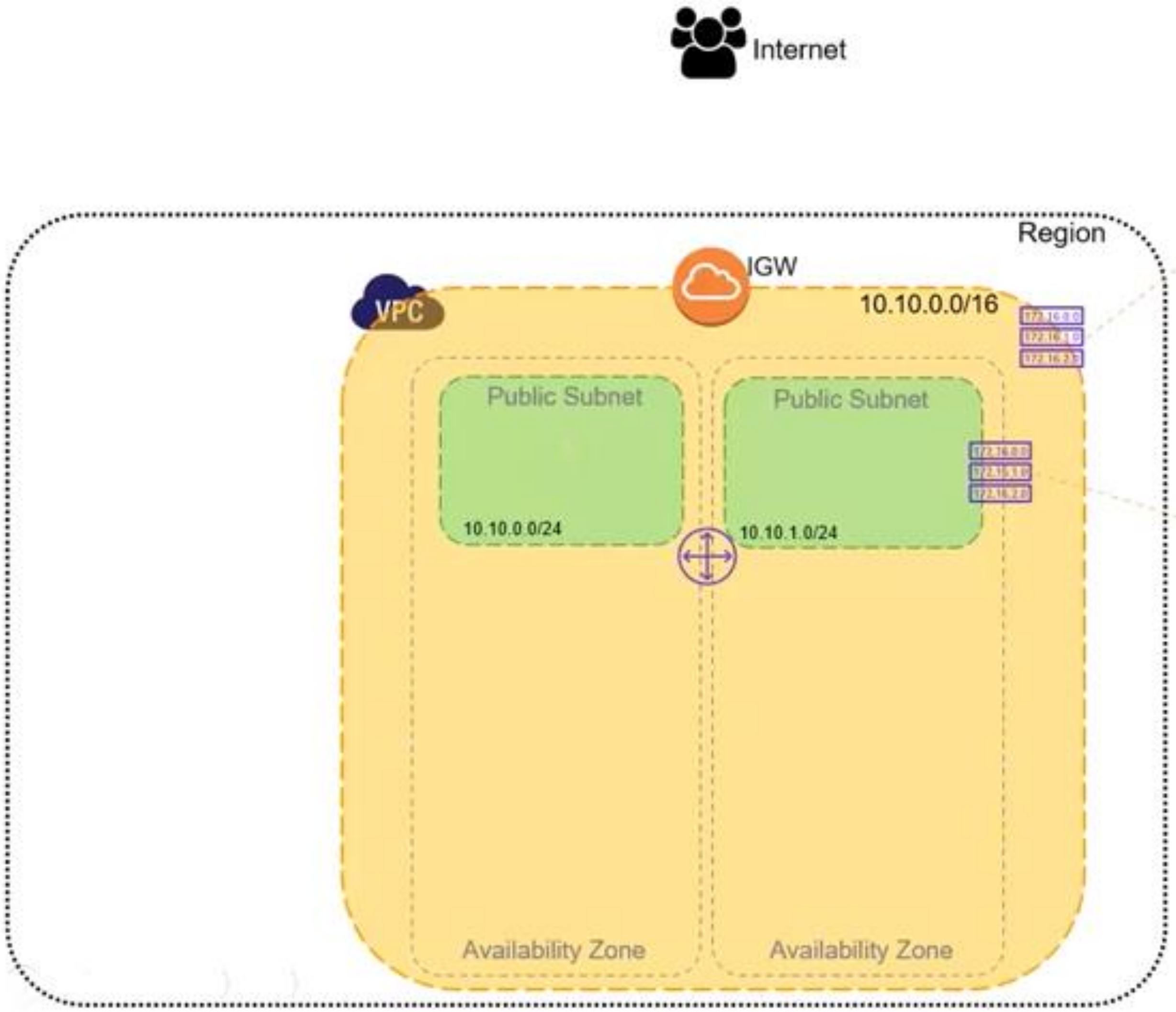
Networking with Virtual Private Cloud



VPC Main Route Table

Destination	Target
10.10.0.0/16	Local

Networking with Virtual Private Cloud



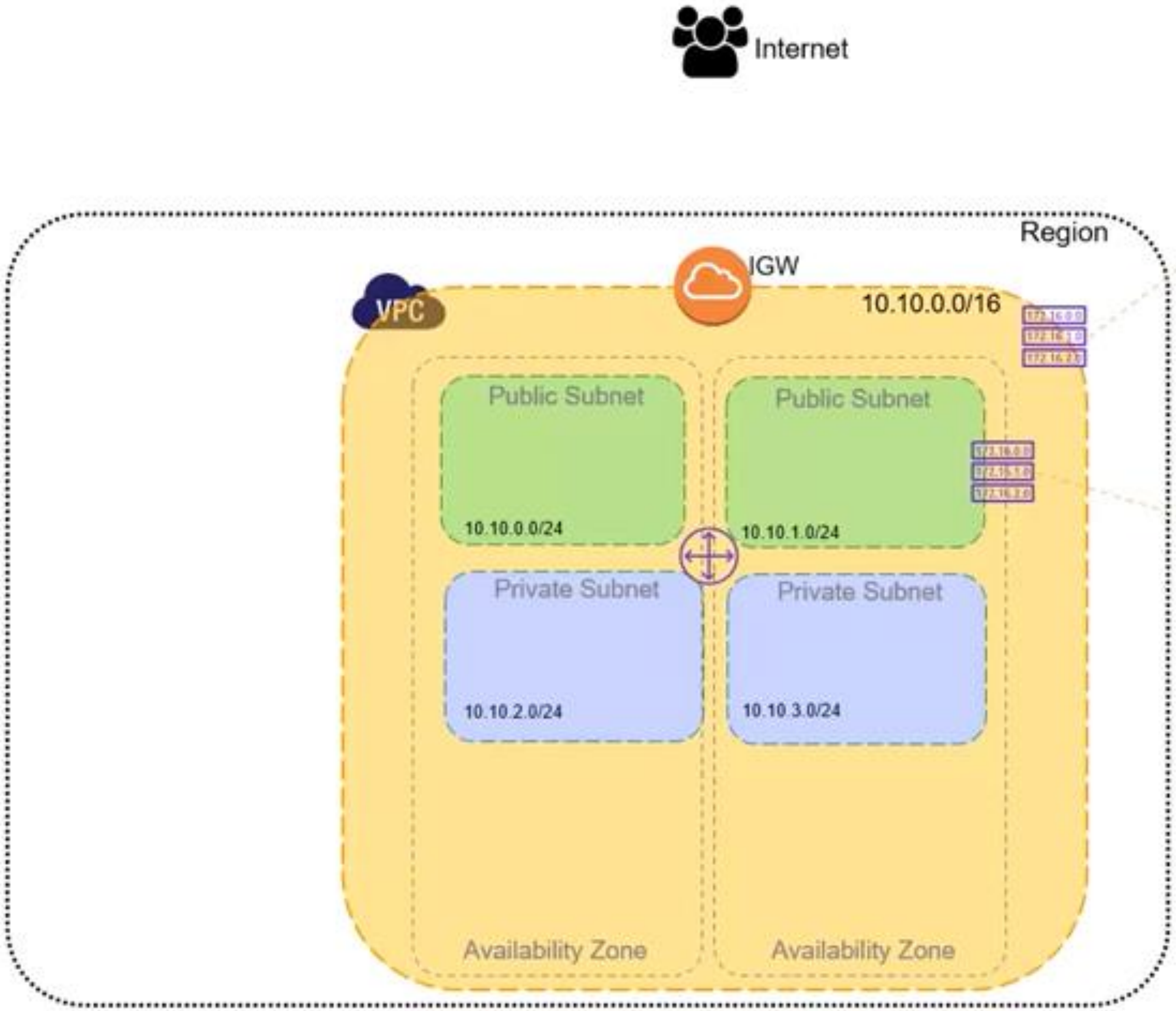
VPC Main Route Table

Destination	Target
10.10.0.0/16	Local

Public Subnet Route Table

Destination	Target
10.10.0.0/16	Local
0.0.0.0/0	igw-xxxxxxx

Networking with Virtual Private Cloud



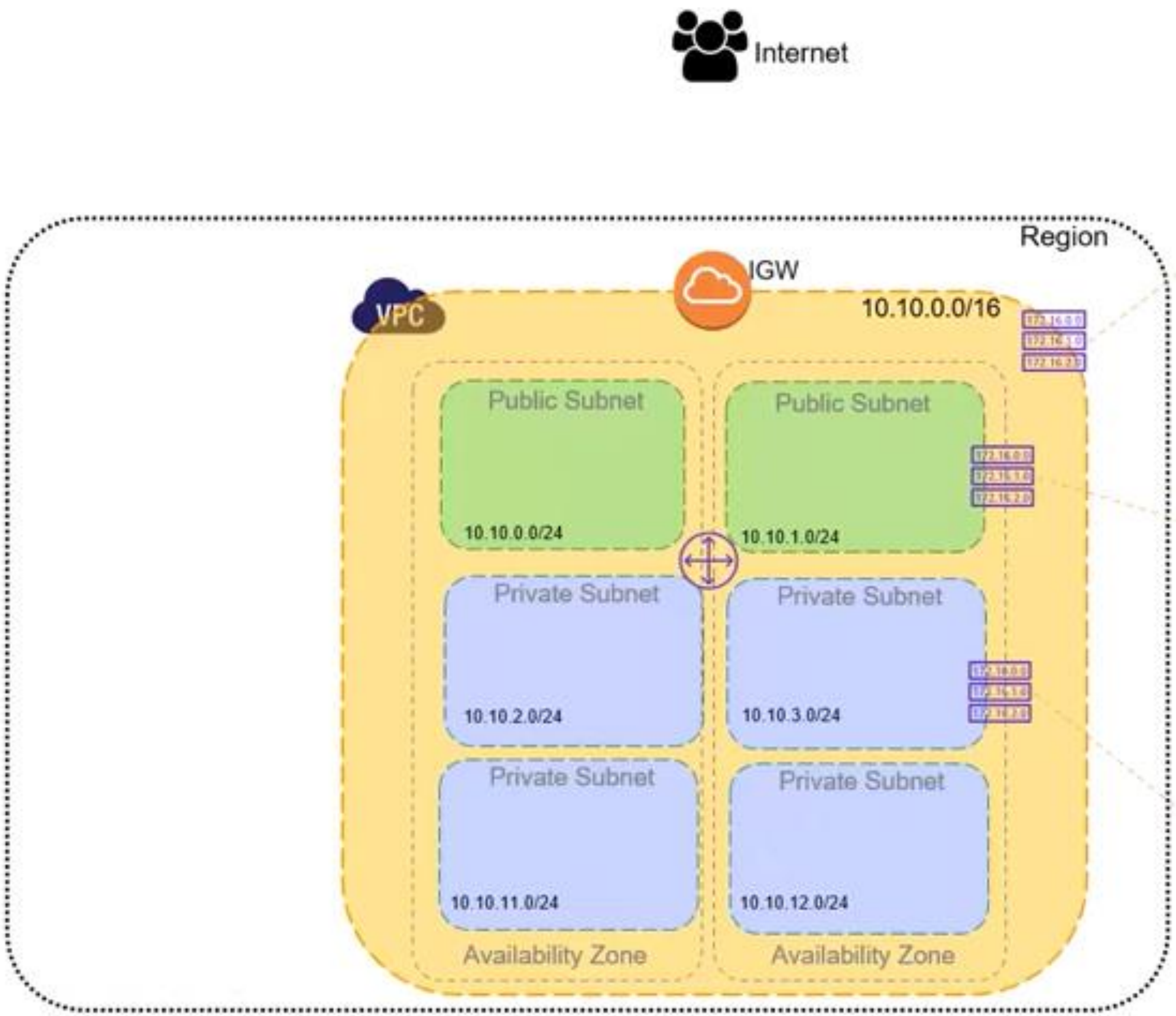
VPC Main Route Table

Destination	Target
10.10.0.0/16	Local

Public Subnet Route Table

Destination	Target
10.10.0.0/16	Local
0.0.0.0/0	igw-xxxxxxx

Networking with Virtual Private Cloud



VPC Main Route Table

Destination	Target
10.10.0.0/16	Local

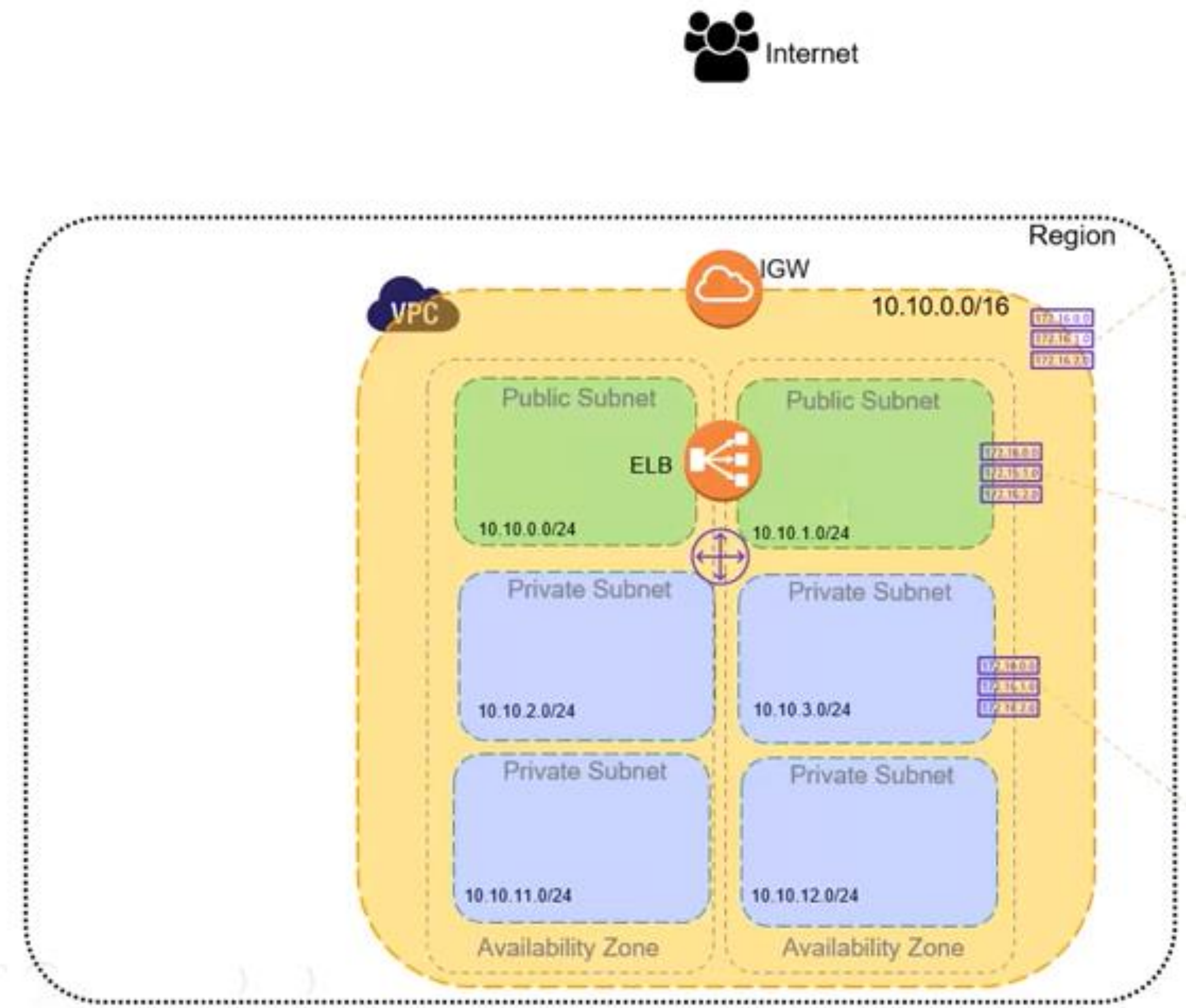
Public Subnet Route Table

Destination	Target
10.10.0.0/16	Local
0.0.0.0/0	igw-xxxxxxx

Private Subnet Route Table

Destination	Target
10.10.0.0/16	Local

Networking with Virtual Private Cloud



VPC Main Route Table

Destination	Target
10.10.0.0/16	Local

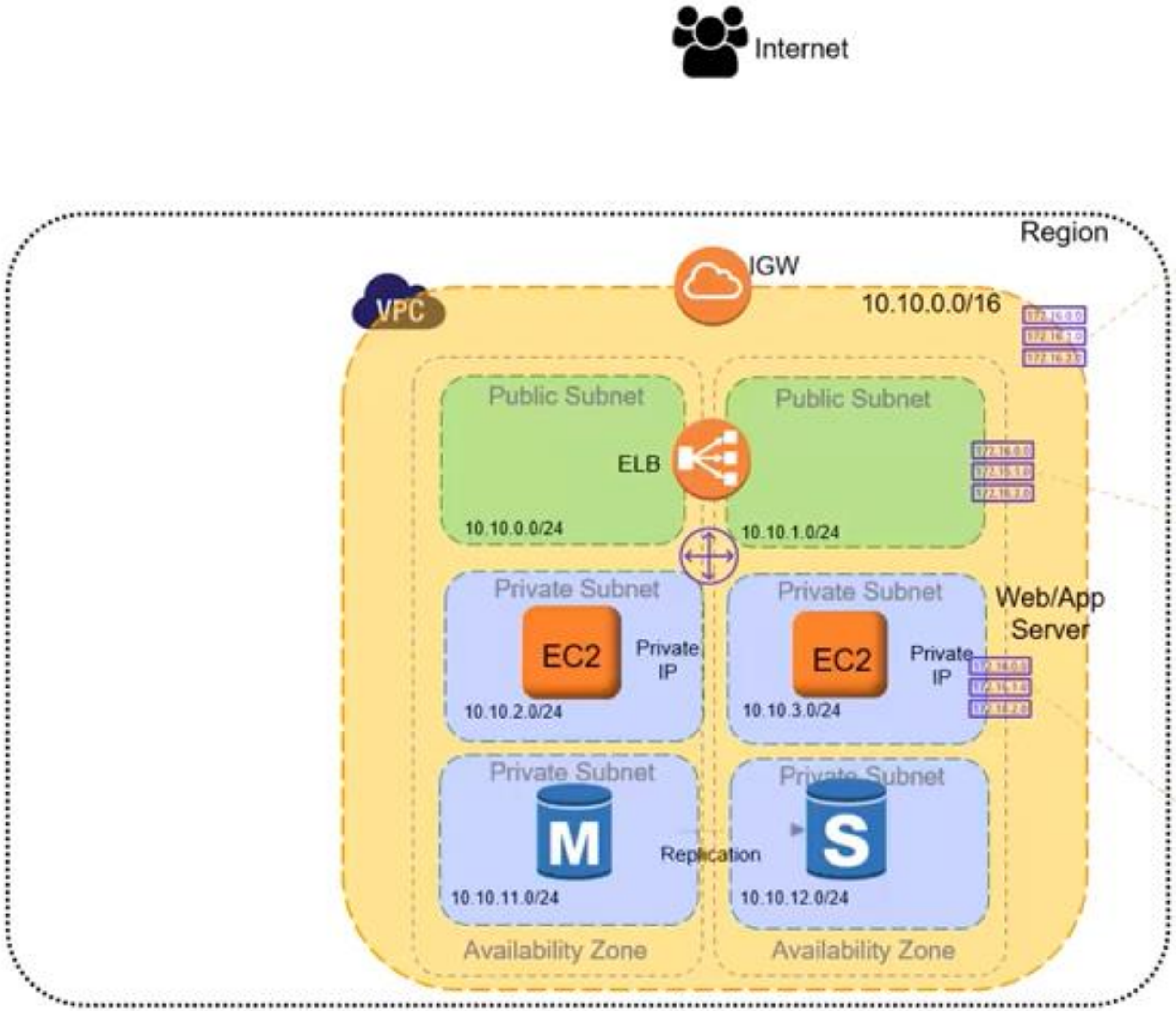
Public Subnet Route Table

Destination	Target
10.10.0.0/16	Local
0.0.0.0/0	igw-xxxxxxx

Private Subnet Route Table

Destination	Target
10.10.0.0/16	Local

Networking with Virtual Private Cloud



VPC Main Route Table

Destination	Target
10.10.0.0/16	Local

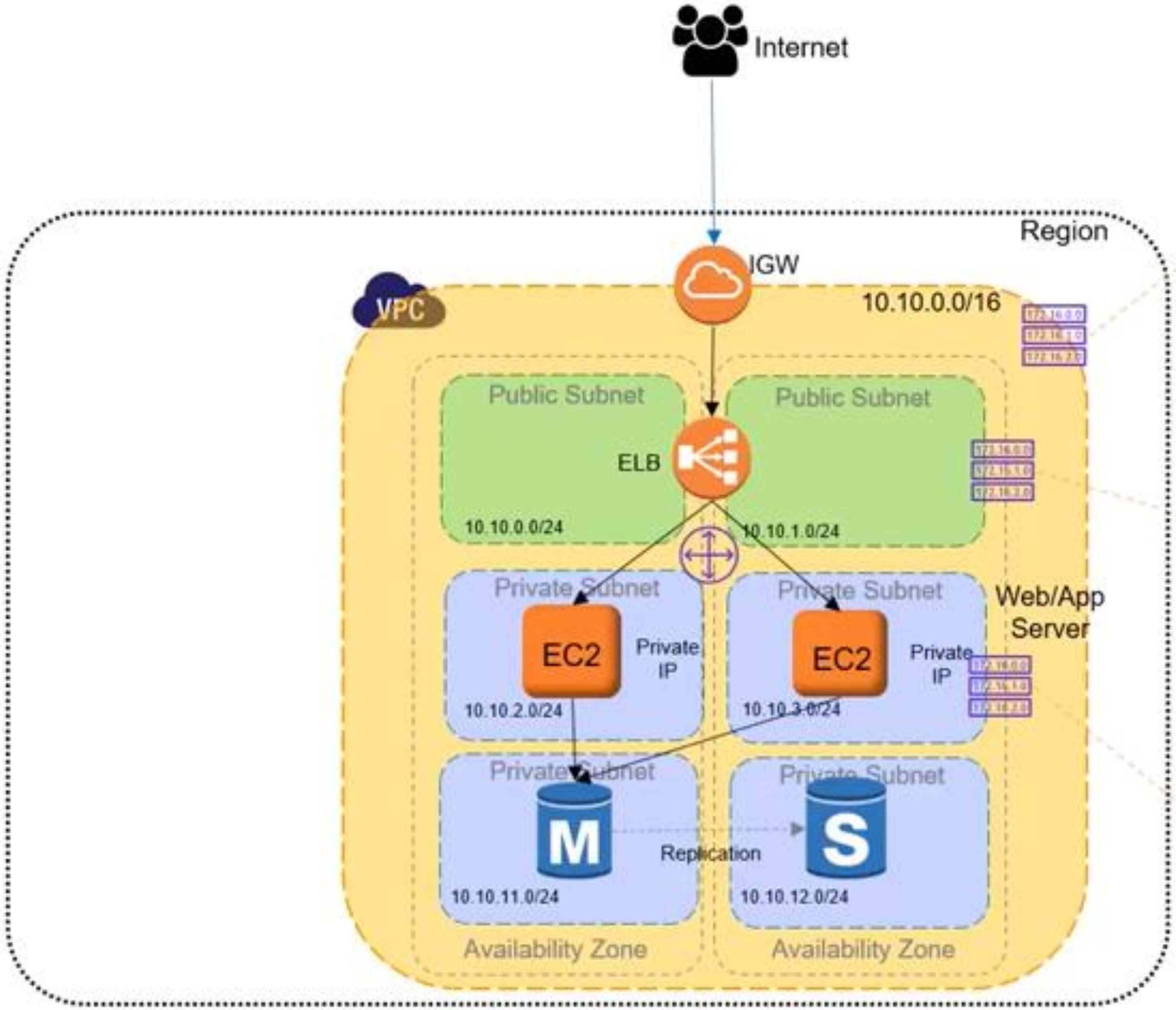
Public Subnet Route Table

Destination	Target
10.10.0.0/16	Local
0.0.0.0/0	igw-xxxxxxx

Private Subnet Route Table

Destination	Target
10.10.0.0/16	Local

Networking with Virtual Private Cloud



VPC Main Route Table

Destination	Target
10.10.0.0/16	Local

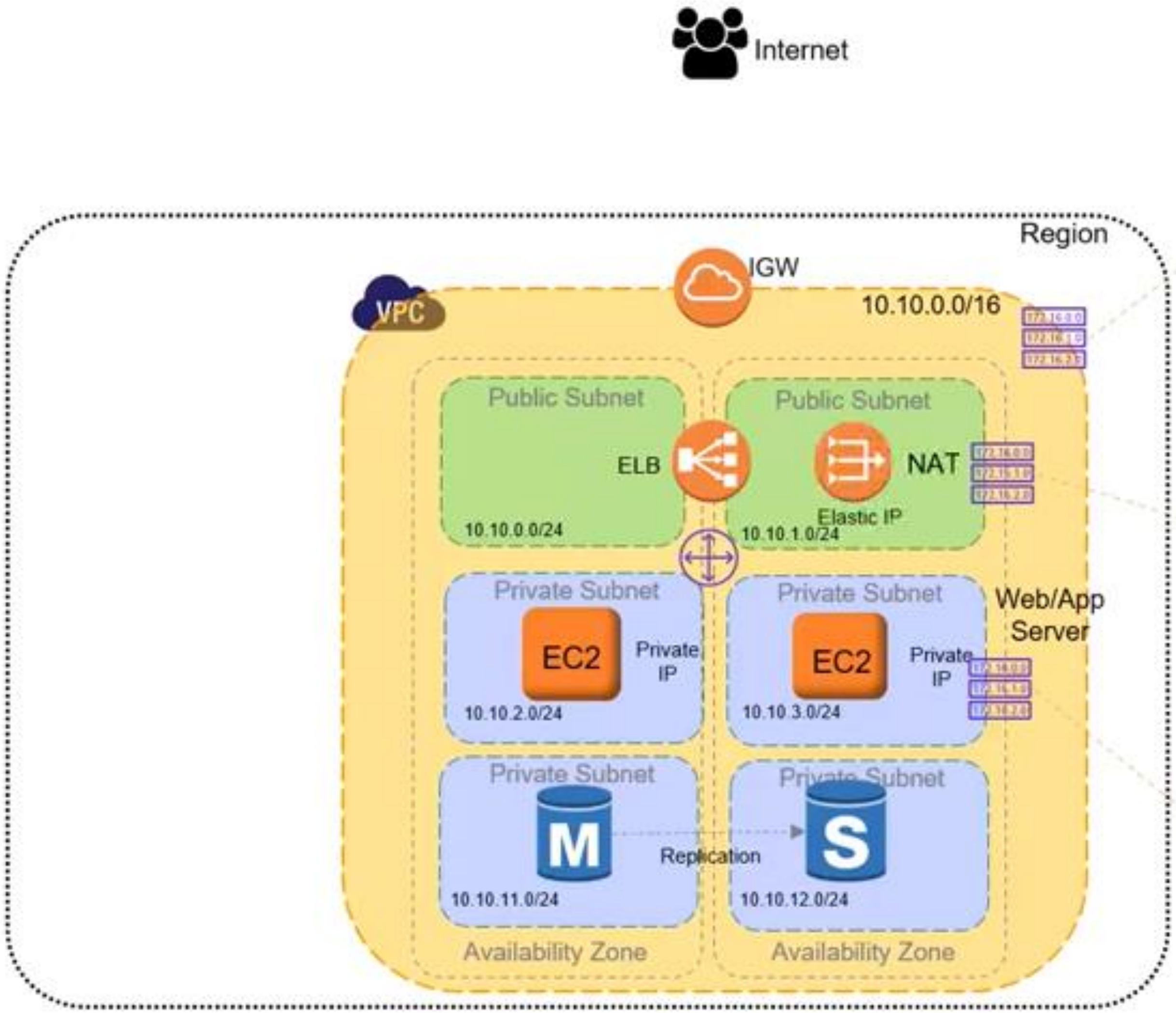
Public Subnet Route Table

Destination	Target
10.10.0.0/16	Local
0.0.0.0/0	igw-xxxxxxx

Private Subnet Route Table

Destination	Target
10.10.0.0/16	Local

Networking with Virtual Private Cloud – *Private Subnet communication with Internet*



VPC Main Route Table

Destination	Target
10.10.0.0/16	Local

Public Subnet Route Table

Destination	Target
10.10.0.0/16	Local
0.0.0.0/0	igw-xxxxxxx

Private Subnet Route Table

Destination	Target
10.10.0.0/16	Local
0.0.0.0/0	nat-xxxxx

Elastic Load Balancing

Elastic Load Balancing

It automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, and IP addresses.

It can handle the varying load of your application traffic in a single Availability Zone or across multiple Availability Zones.

It provides high availability, automatic scaling, and robust security necessary to make your applications fault tolerant

Types of Elastic Load Balancing

1. Application Load Balancer

It is best suited for load balancing of HTTP and HTTPS traffic

It routes traffic to targets within Amazon Virtual Private Cloud

2. Network Load Balancer

It is best suited for load balancing of TCP traffic where extreme performance is required

It routes traffic to targets within Amazon Virtual Private Cloud and is capable of handling millions of requests per second while maintaining ultra-low latencies.

Types of Elastic Load Balancing

3. Gateway Load Balancer

Easy to deploy, scale, and run third-party virtual networking appliances

Provides load balancing and auto scaling for fleets of third-party appliances

4. Classic Load Balancer

It provides basic load balancing across multiple Amazon EC2 instances and operates at both the request level and connection level.

Amazon Route 53

Amazon Route 53

- ❑ Amazon's scalable DNS web service, Route 53, directs end users to web applications by translating domain names into numbered IP addresses.
- ❑ It is a highly available and scalable cloud Domain Name System (DNS) web service. It connects user requests to internet applications running on AWS or on-premises.
- ❑ Amazon Route 53 is fully compliant with IPv6
- ❑ It is used for three main functions: domain registration, DNS routing and health checking

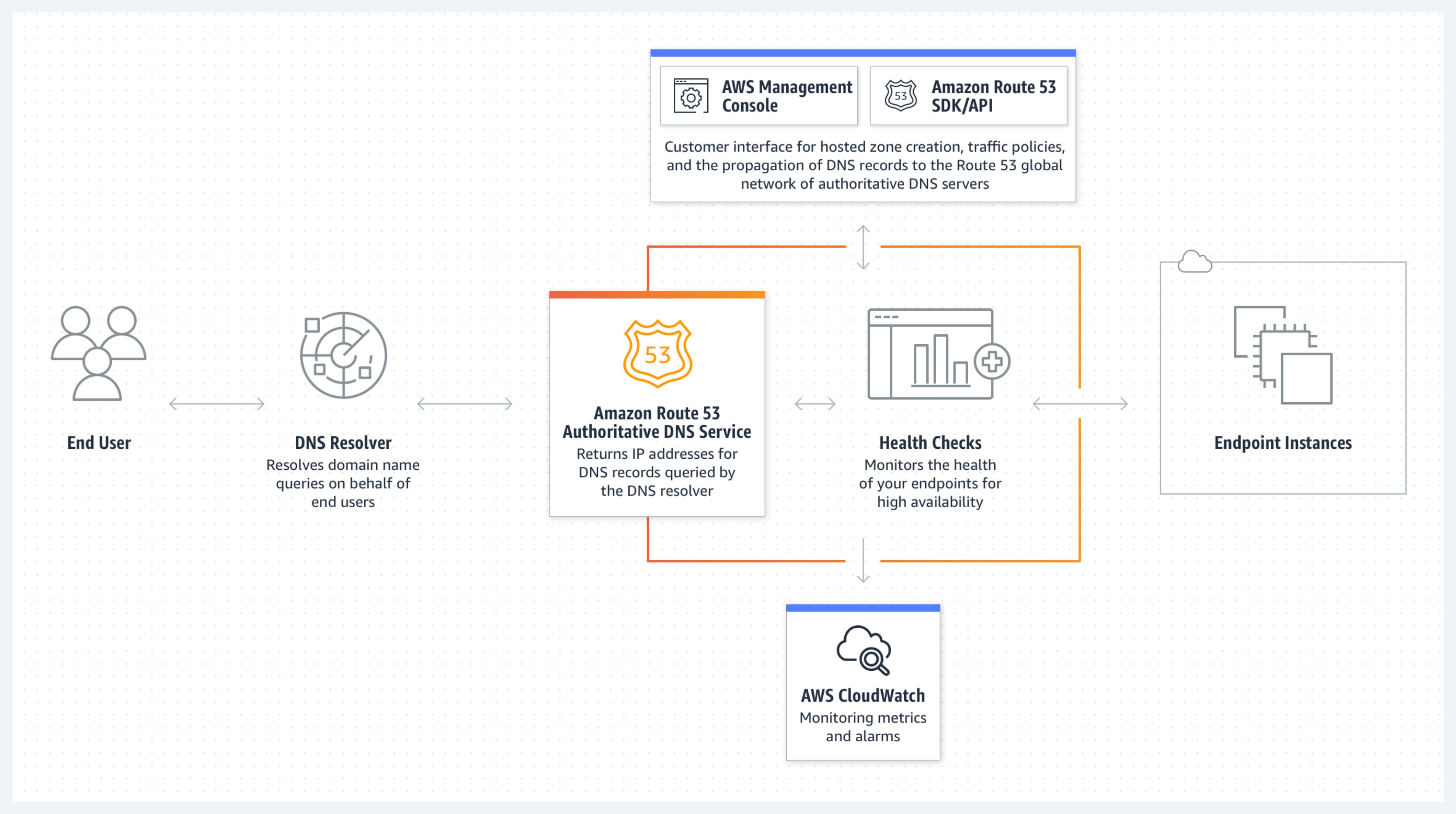
Amazon Route 53

It is used to configure DNS health checks to route traffic to healthy endpoints

Amazon Route 53 traffic flow makes it easy to manage traffic globally through a variety of routing types, including latency-based routing, Geo DNS, and weighted round robin, etc.

It offers Domain Name Registration—you can purchase and manage domain names. It will automatically configure DNS settings for your domains

Amazon Route 53



AWS Management Console

Amazon Route 53 SDK/API

Customer interface for hosted zone creation, traffic policies, and the propagation of DNS records to the Route 53 global network of authoritative DNS servers

End User

DNS Resolver
Resolves domain name queries on behalf of end users

Amazon Route 53 Authoritative DNS Service
Returns IP addresses for DNS records queried by the DNS resolver

Health Checks
Monitors the health of your endpoints for high availability

Endpoint Instances

AWS CloudWatch
Monitoring metrics and alarms

Amazon CloudFront

Amazon CloudFront

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

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

Improve security with traffic encryption and access controls, and use AWS Shield Standard to defend against DDoS attacks at no additional charge.

Reduces latency and cost optimization

Amazon CloudFront

Organizations use CloudFront to quickly distribute both static and dynamic content.

The CDN routes each request through the AWS network and to the nearest edge location to provide the fastest delivery path to end users.

It also reduces the number of networks a user's request passes through in the content delivery process.

It is commonly used to accelerate static website content delivery as well as enable video on demand and live streaming video.

AWS Direct Connect

AWS Direct Connect provides a private connection between a customer's on-premises data center and the cloud without using the public internet.

This Amazon networking service uses an Ethernet cable to connect an organization's internal workloads to one of AWS' Direct Connect locations.

Its offers connection speeds from 50 Mbps to 100 Gbps.

This connection creates multiple virtual interfaces to Amazon's publicly accessible cloud services or to private resources hosted on AWS.

Users can access private and public resources with the same connection

AWS Direct Connect

AWS cloud users can choose between two types of connections with this service:

Dedicated. The dedicated connection uses an Ethernet cable to create a connection with an individual customer. AWS cloud users request a dedicated connection through the AWS Direct Connect console, the command line interface or the API.

Hosted. The hosted connection requires an AWS Direct Connect Partner to provision the physical Ethernet connection on behalf of a customer. For the hosted connection, IT teams must choose a partner in the AWS Direct Connect Delivery Partners Program.

References

1. https://docs.aws.amazon.com/whitepapers/latest/aws-overview/networking-services.html