Study Guide

Version 2026-01-28

Table of Contents

Study Guide

Topics

Domain 1: Cloud Concepts

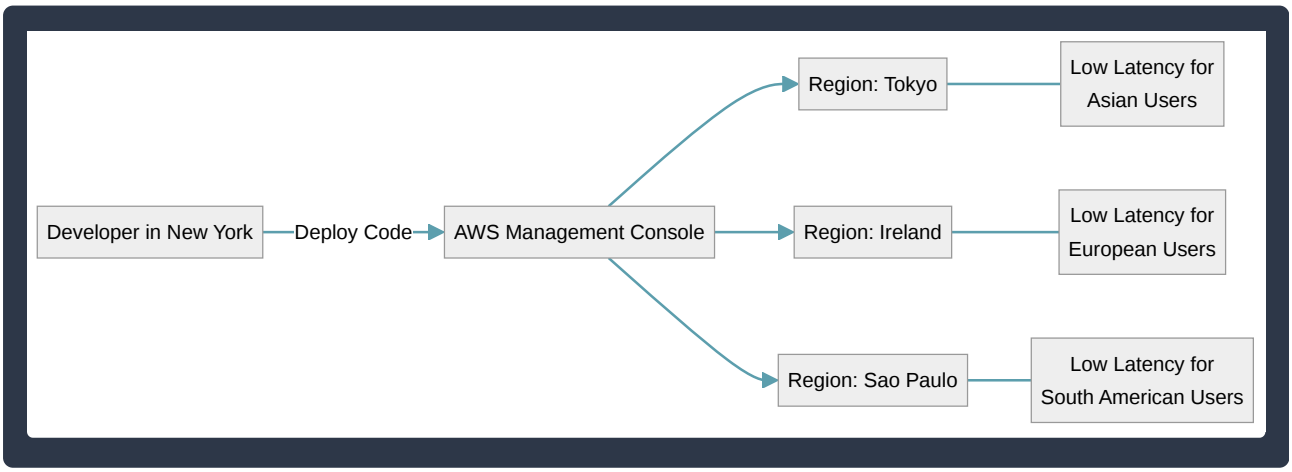
Benefits of the AWS Cloud: Global Infrastructure and Operational Advantages

The AWS Cloud offers a significant departure from traditional on-premises computing by providing a massive, distributed infrastructure that allows organizations to trade fixed capital expenses for variable operational expenses. This shift enables businesses to innovate faster and reach a global audience with minimal effort.

Global Infrastructure Benefits

The AWS Global Infrastructure is built around **AWS Regions** and **Availability Zones (AZs)**. Leveraging this footprint provides two primary advantages:

- **Speed of Deployment:** In a traditional environment, procuring and installing hardware can take weeks or months. In AWS, you can deploy hundreds or thousands of servers in minutes. This “Go global in minutes” capability allows businesses to respond to market changes instantly.
- **Global Reach:** AWS allows you to deploy applications in multiple Regions around the world. By placing applications closer to end-users, you significantly reduce **latency** (the delay before a transfer of data begins following an instruction) and provide a better user experience.

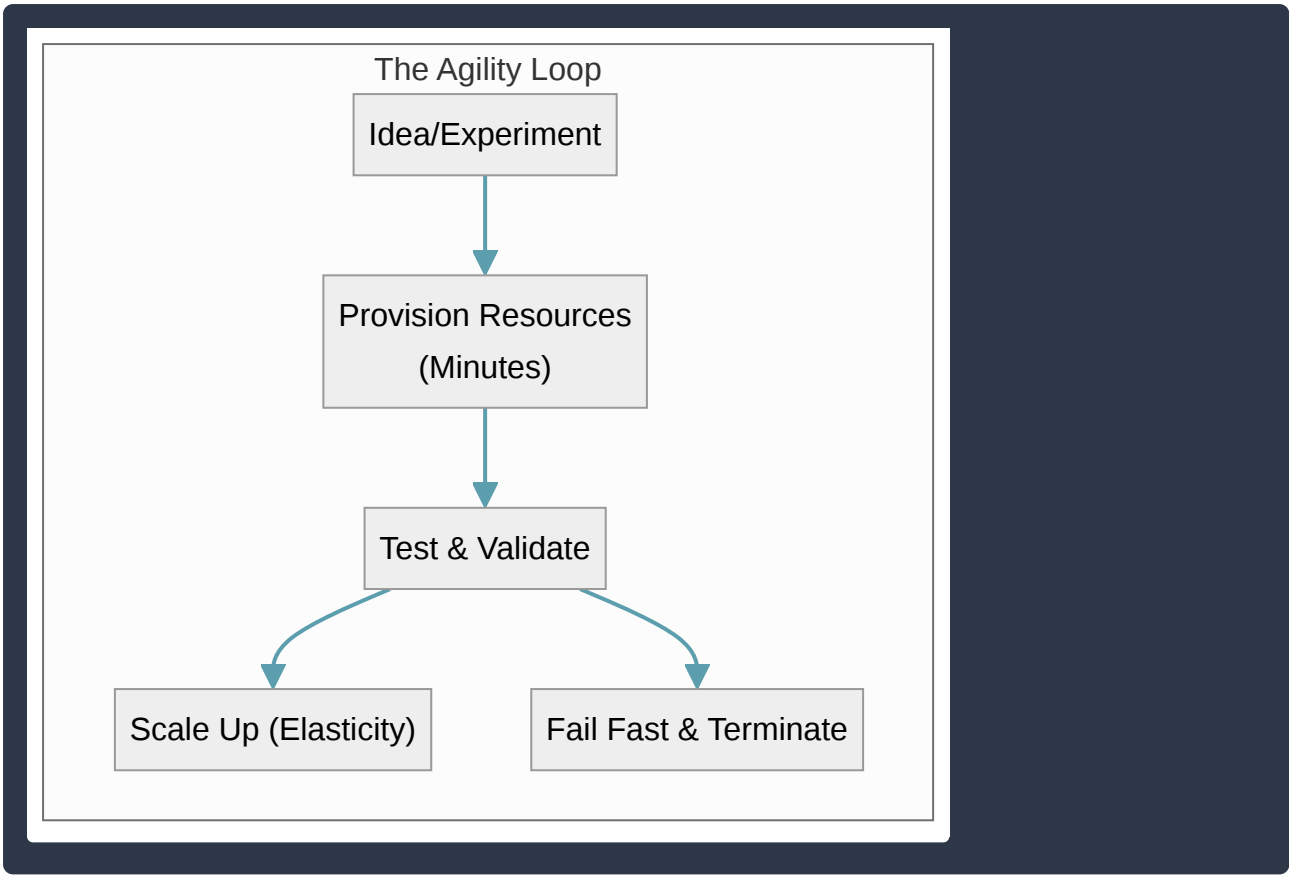


High Availability, Elasticity, and Agility

Beyond its physical reach, the AWS Cloud provides operational benefits that ensure applications remain functional, cost-effective, and innovative.

Concept	Definition	Business Value
High Availability	Ensuring a system remains operational and accessible even if a component fails.	Minimizes downtime and prevents loss of revenue or reputation.
Elasticity	The ability to scale resources up or down automatically to match current demand.	Ensures you have enough capacity during peaks and don't pay for idle resources during lulls.
Agility	The ability to innovate and develop applications faster by reducing the time required to provision resources.	Allows for rapid experimentation and faster time-to-market for new features.

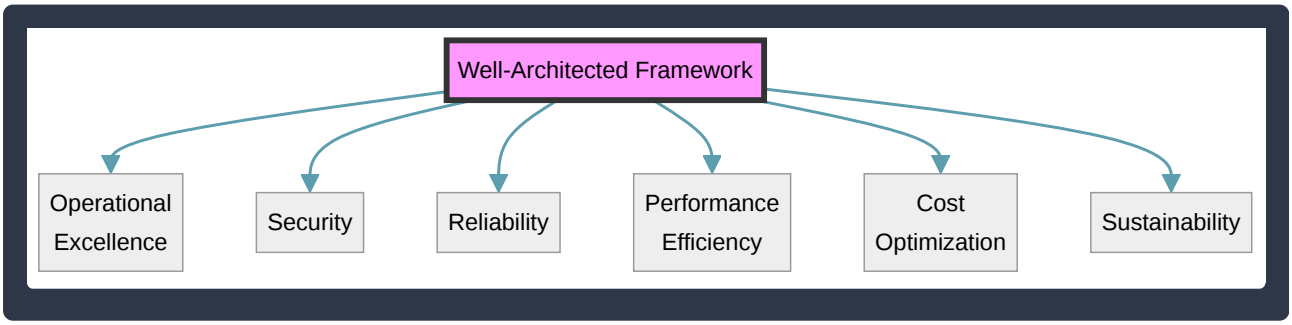
- **High Availability (HA):** This is achieved by deploying resources across multiple **Availability Zones**. If one data center experiences an outage, the application continues to run from another AZ without manual intervention.
- **Elasticity:** Unlike "scalability" (which is the ability to handle growth), elasticity is about the *fluidity* of resources. For example, an e-commerce site uses `Amazon EC2 Auto Scaling` to add servers during a "Black Friday" sale and remove them immediately afterward to save costs.
- **Agility:** AWS provides a broad set of services (compute, storage, databases, AI) that are available on-demand. This allows developers to experiment with new technologies without a long-term contract or heavy upfront investment. If an experiment fails, you can simply terminate the resources and stop paying for them.



By combining **Global Reach** with **High Availability**, organizations can build “fault-tolerant” systems that are resilient to localized disasters while maintaining high performance for a worldwide user base.

AWS Well-Architected Framework Pillars

The AWS Well-Architected Framework provides a consistent set of best practices for customers and partners to evaluate architectures and implement designs that can scale over time. It is organized into six pillars, which serve as the foundation for building stable and efficient systems in the cloud.



The Six Pillars Defined

- **Operational Excellence**: Focuses on running and monitoring systems to deliver business value, and continually improving processes and procedures. Key concepts include performing operations as **code**, making frequent, small, reversible changes, and anticipating failure.

- **Security**: Focuses on protecting information and systems. Key topics include confidentiality and integrity of data, managing user permissions (**IAM**), and establishing controls to detect security events.
- **Reliability**: Ensures a workload performs its intended function correctly and consistently when it's expected to. This includes the ability to operate and test the workload through its total lifecycle. Key concepts include distributed system design, recovery planning, and adapting to changing requirements.
- **Performance Efficiency**: Focuses on using IT and computing resources efficiently. Key concepts include selecting the right resource types and sizes based on workload requirements, monitoring performance, and making informed decisions to maintain efficiency as business needs evolve.
- **Cost Optimization**: Focuses on avoiding unnecessary costs. Key concepts include understanding spending over time, providing funds for only what is needed, and selecting the most manual or managed services to reduce **Total Cost of Ownership (TCO)**.
- **Sustainability**: Focuses on minimizing the environmental impacts of running cloud workloads. Key topics include a shared responsibility model for sustainability, understanding impact, and maximizing utilization to minimize required resources and reduce downstream impact.

Identifying Differences Between Pillars

While the pillars are interrelated, they focus on different primary outcomes. Understanding these differences is essential for making architectural trade-offs.

Pillar	Primary Focus	Key Question
Operational Excellence	Process and Change	How do we manage and evolve our environment?
Security	Protection and Compliance	How do we protect our data and infrastructure?
Reliability	Resilience and Recovery	How do we prevent and recover from failure?
Performance Efficiency	Speed and Scalability	How do we use resources to meet requirements?
Cost Optimization	Value and Savings	How do we eliminate unneeded expenses?
Sustainability	Environmental Impact	How do we reduce our carbon footprint?

Practical Examples and Use Cases

- **Reliability vs. Cost**: To increase **Reliability**, an architect might deploy resources across multiple **Availability Zones**. While this increases cost, it ensures the system remains available if one zone fails.

- **Performance Efficiency**: Using **Amazon CloudFront** to cache content closer to users improves performance by reducing latency, demonstrating the efficient use of network resources.
- **Sustainability**: Choosing a “Serverless” architecture using **AWS Lambda** can improve sustainability because AWS manages the underlying infrastructure to maximize server utilization, reducing the energy wasted on idle resources.

AWS Cloud Adoption Framework and Migration Strategies

Migrating to the AWS Cloud is a transformative process that requires a structured approach to align business goals with technical implementation. AWS provides the **Cloud Adoption Framework (AWS CAF)** to guide organizations through this transition, ensuring they realize the full value of the cloud while minimizing disruption.

The AWS Cloud Adoption Framework (AWS CAF)

The AWS CAF identifies specific transformation domains that help organizations close the gap between where they are and where they want to be. By following this framework, businesses can achieve several key outcomes:

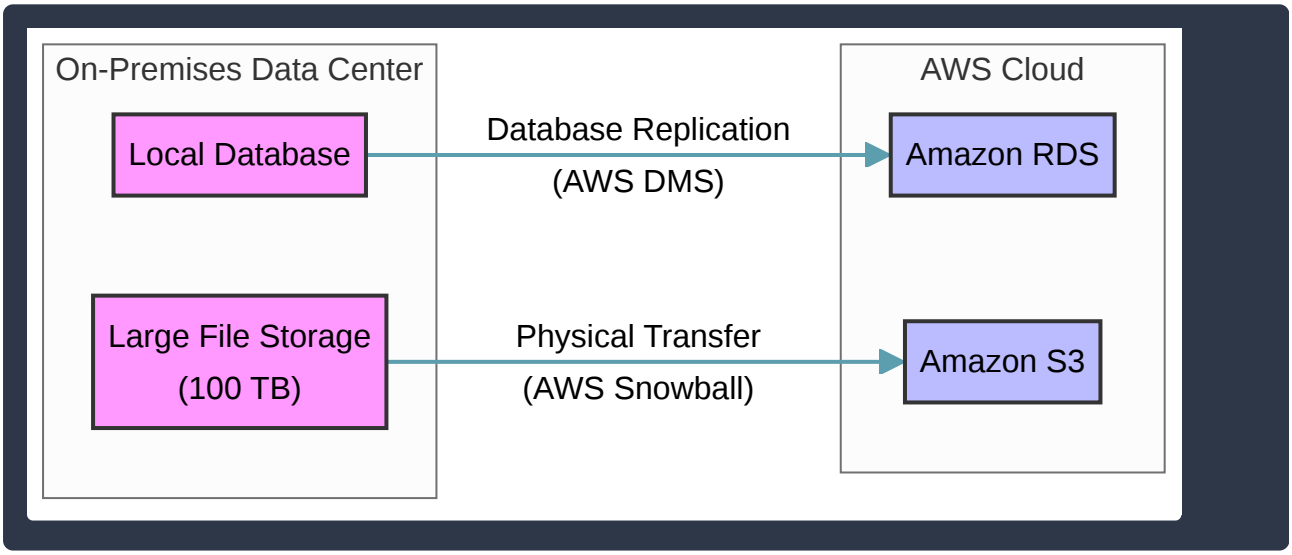
- **Reduced Business Risk:** AWS provides high availability, automated backups, and robust security features. This reduces the risk of data loss, system outages, and security breaches compared to traditional on-premises environments.
- **Improved Environmental, Social, and Governance (ESG) Performance:** AWS allows organizations to reduce their carbon footprint by using shared, energy-efficient infrastructure. AWS also provides tools to track sustainability goals and improve transparency in governance.
- **Increased Revenue:** Cloud adoption enables faster time-to-market for new products and services. Organizations can experiment quickly and scale globally in minutes, leading to new business opportunities and revenue streams.
- **Increased Operational Efficiency:** By moving away from managing physical hardware, IT teams can focus on innovation. Automation of routine tasks (like patching and provisioning) reduces manual labor and operational costs.

CAF Outcome	Description	Example
Business Risk	Enhancing reliability and security.	Using **AWS Shield** to prevent DDoS attacks.
ESG Performance	Meeting sustainability and social goals.	Moving to **Graviton** processors for better energy efficiency.
Revenue	Driving growth through agility.	Launching a mobile app in 10 new countries in one day.
Efficiency	Optimizing resource utilization.	Using **Auto Scaling** to only pay for what you use.

Migration Strategies and Tools

When moving workloads to AWS, organizations must choose the right strategy based on their data volume, time constraints, and technical requirements.

- **Database Replication:** This strategy involves continuously copying data from an on-premises database to an AWS database (like **Amazon RDS**). It is ideal for migrations requiring minimal downtime, as the cloud database stays synchronized with the source until the final cutover.
- **AWS Snowball:** This is a physical migration service. AWS sends a ruggedized hardware device to your data center; you load your data onto it and ship it back to AWS.
  - **Use Case:** Use Snowball when you have massive amounts of data (terabytes or petabytes) and limited network bandwidth, making over-the-wire transfer impractical.
- **The "7 Rs" of Migration:** While replication and Snowball are tools, they support broader strategies such as **Rehosting** (Lift-and-Shift), **Replatforming** (Lift-and-Reshape), and **Refactoring** (re-architecting for the cloud).



Choosing the Right Approach

- **Small Data Volumes:** Use internet-based transfers or **AWS DataSync** for efficient online movement.
- **Large Data Volumes (Limited Bandwidth):** Use the **AWS Snow Family** (Snowcone, Snowball, or Snowmobile) to bypass network bottlenecks.
- **Critical Applications:** Use **AWS Application Migration Service (MGN)** or database replication to ensure the application remains online during the migration process.

Cloud Economics and Cost Optimization

Cloud economics focuses on the shift from traditional hardware-centric spending to a flexible, consumption-based model. Understanding these concepts helps organizations maximize the value of their AWS investment while minimizing waste.

Fixed Costs vs. Variable Costs In a traditional environment, organizations deal with **Fixed Costs**, also known as **Capital Expenditures (CapEx)**. These are upfront investments in physical assets like servers and data centers. In contrast, AWS operates on a **Variable Cost** model, or **Operating Expenditures (OpEx)**.

- **Fixed Costs (CapEx):** You pay for capacity regardless of whether you use it. This often leads to “shelfware” or wasted resources.
- **Variable Costs (OpEx):** You pay only for what you consume. If you turn off a resource, you stop paying for it. This allows for better alignment between expenses and actual business demand.

On-Premises Costs Running an on-premises data center involves several “hidden” costs beyond just the servers:

- **Facilities:** Rent, floor space, power, and cooling systems.
- **Hardware:** Servers, storage arrays, and networking equipment (switches, routers).
- **Labor:** Salaries for staff to rack, stack, cable, and maintain hardware.
- **Maintenance:** Ongoing costs for hardware replacements and software updates.

Cost Category	On-Premises (CapEx)	AWS Cloud (OpEx)
Hardware	High upfront purchase costs	No upfront cost; pay-as-you-go
Facilities	Costs for power, cooling, and space	Included in the service price
Maintenance	Manual patching and hardware repair	Managed by AWS
Scaling	Slow (weeks/months for procurement)	Instant (Elasticity)

Licensing Strategies AWS offers flexibility in how you handle software licenses (e.g., Windows Server, SQL Server):

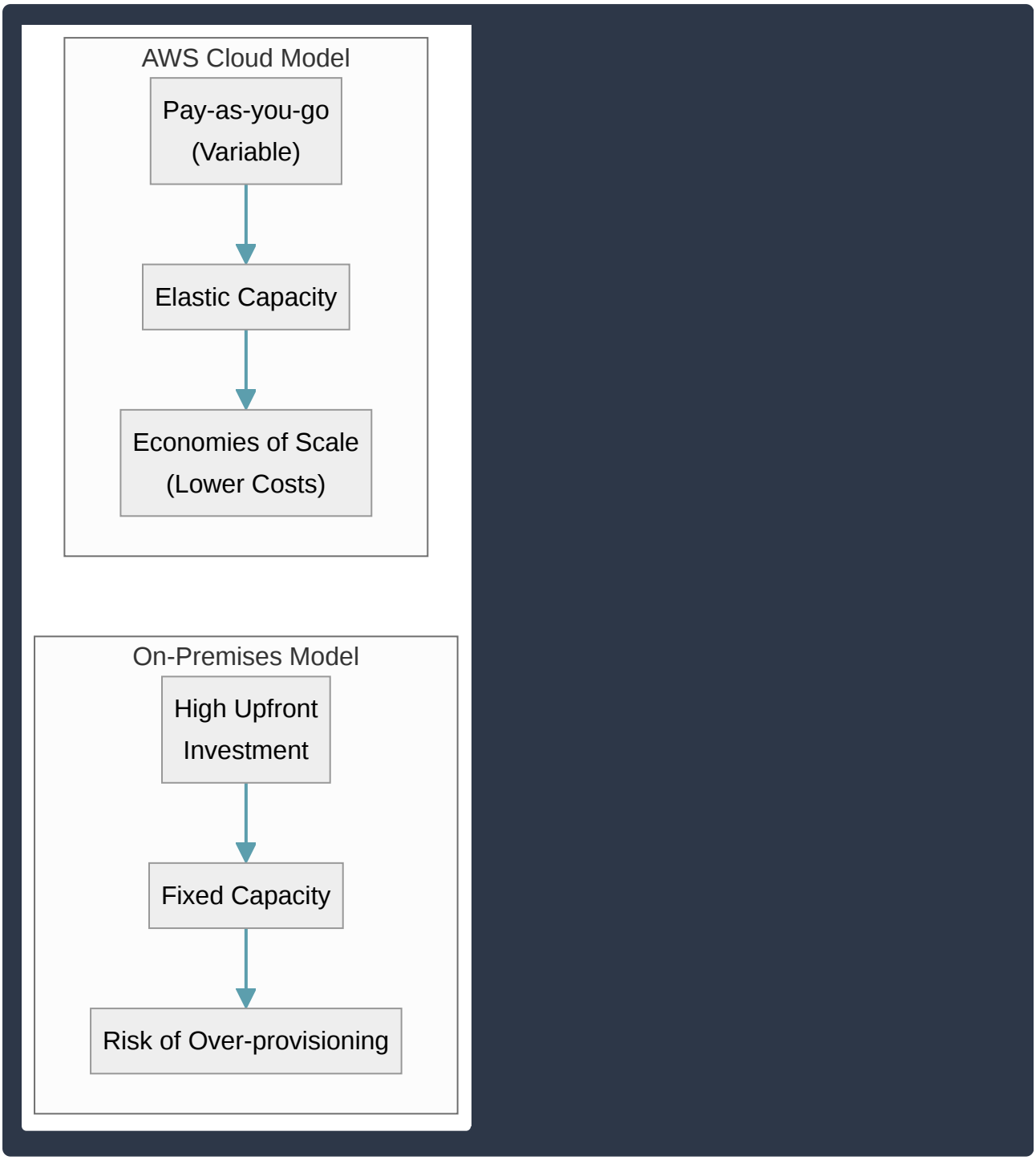
- **License Included:** The cost of the software license is bundled into the hourly price of the AWS service. This is ideal for new projects or when you want to avoid long-term contracts.
- **Bring Your Own License (BYOL):** You use your existing enterprise licenses on AWS. This is beneficial for organizations that have already invested heavily in perpetual licenses and want to reduce migration costs.

Rightsizing Rightsizing is the process of matching instance types and sizes to your workload performance and capacity requirements at the lowest possible cost. It involves looking at metrics like CPU, memory, and disk usage to ensure you aren’t paying for “over-provisioned” resources.

- **Example:** If an `m5.large` instance is only using 10% of its CPU, rightsizing might involve moving that workload to a smaller `t3.medium` instance to save money.

Economies of Scale AWS achieves **economies of scale** by aggregating usage from hundreds of thousands of customers. Because AWS purchases hardware in massive quantities, they achieve

lower costs per unit than any single organization could. AWS then passes these savings back to customers in the form of lower pay-as-you-go prices.



Benefits of Automation Automation is a key driver of cloud economics by reducing manual effort and increasing efficiency:

- **Cost Savings:** Automatically shutting down non-production instances (like dev/test environments) after business hours.
- **Agility:** Using tools like AWS CloudFormation to deploy entire environments in minutes rather than days.

- **Consistency:** Reducing human error by using code to manage infrastructure, ensuring environments are identical and secure.

Domain 2: Security and Compliance

The AWS Shared Responsibility Model

The **AWS Shared Responsibility Model** is a fundamental security framework that clarifies the security obligations of both AWS and the customer. By defining who is responsible for specific aspects of the environment, this model helps reduce operational overhead and ensures there are no gaps in security coverage.

AWS Responsibilities: Security “of” the Cloud

AWS is responsible for protecting the infrastructure that runs all the services offered in the AWS Cloud. This is often referred to as **Security of the Cloud**.

- **Physical Infrastructure**: AWS manages the physical security of data centers, including access control, environmental monitoring, and power redundancy.
- **Hardware and Software**: AWS is responsible for the physical servers, storage devices, and the virtualization layer (hypervisor) that runs the services.
- **Global Infrastructure**: This includes the maintenance and protection of **Regions**, **Availability Zones**, and **Edge Locations**.
- **Networking**: AWS manages the physical networking components and the software-defined networks that connect their infrastructure.

Customer Responsibilities: Security “in” the Cloud

The customer is responsible for how they use the services and how they protect the data they place on AWS. This is known as **Security in the Cloud**.

- **Customer Data**: Managing data encryption (both at rest and in transit) and ensuring proper data classification.
- **Identity and Access Management (IAM)**: Managing user accounts, permissions, and multi-factor authentication (MFA).
- **Operating System (OS) Configuration**: For services like `Amazon EC2`, the customer is responsible for patching and maintaining the guest OS.
- **Network Traffic Protection**: Configuring security groups, network ACLs, and firewall settings.

Shared Responsibilities

Some responsibilities are shared between AWS and the customer, depending on the context:

- **Patch Management**: AWS patches the physical infrastructure and managed services, while the customer patches guest operating systems and applications.
- **Configuration Management**: AWS manages the configuration of infrastructure devices; the customer manages the configuration of their own resources and guest OS.
- **Awareness and Training**: Both parties must train their employees on security best practices and compliance.

Shifting Responsibilities Based on Service Type

The boundary of responsibility shifts depending on whether you use Infrastructure as a Service (IaaS), Platform as a Service (PaaS), or Serverless services.

Service	Service Type	Customer Responsibility	AWS Responsibility
Amazon EC2	IaaS	OS patching, application security, firewall rules, data encryption.	Physical hardware, virtualization layer, global infrastructure.
Amazon RDS	PaaS	Application optimization, database settings, IAM, data encryption.	OS patching, database engine patching, hardware maintenance.
AWS Lambda	Serverless	Code security, IAM permissions, data protection.	Managing the runtime environment, scaling, OS, and hardware.

- **Amazon EC2 (Elastic Compute Cloud)**: Provides virtual servers. Use this when you need full control over the operating system and software stack.
- **Amazon RDS (Relational Database Service)**: A managed database service. Use this to reduce the administrative burden of managing database engines like MySQL or PostgreSQL.
- **AWS Lambda**: A serverless compute service that runs code in response to events. Use this to run applications without managing any underlying servers or runtimes.

AWS Cloud Security, Governance, and Compliance

Maintaining a secure and compliant environment in the cloud requires a combination of AWS-managed infrastructure and customer-configured tools. AWS provides a suite of services to automate monitoring, auditing, and threat detection, ensuring that organizations can meet specific industry or geographic regulatory requirements.

Finding Compliance Information

AWS simplifies the process of proving compliance to auditors through centralized portals and documentation.

- **AWS Artifact**: This is the primary resource for AWS compliance information. It provides on-demand access to AWS security and compliance reports (such as SOC and PCI reports) and select online agreements.
- **Geographic and Industry Compliance**: Compliance needs vary by location (e.g., **GDPR** in Europe) and industry (e.g., **HIPAA** for healthcare or **PCI DSS** for payment processing). AWS maintains a vast array of certifications, but customers must verify which services are "in-scope" for specific compliance programs.

Securing Resources on AWS

AWS offers several specialized services to protect workloads from vulnerabilities and active threats.

Service	Primary Function	Use Case
Amazon Inspector	Automated vulnerability management	Scanning EC2 instances and container images for software vulnerabilities and unintended network exposure.
Amazon GuardDuty	Intelligent threat detection	Monitoring for malicious activity like crypto-mining or unauthorized access using machine learning.
AWS Security Hub	Security posture management	Providing a central dashboard to aggregate security alerts (findings) from multiple AWS services.
AWS Shield	DDoS protection	Protecting applications from Distributed Denial of Service attacks. **Shield Standard** is free for all; **Shield Advanced** provides higher-level protection.

Encryption Options

Protecting data involves two primary states of encryption:

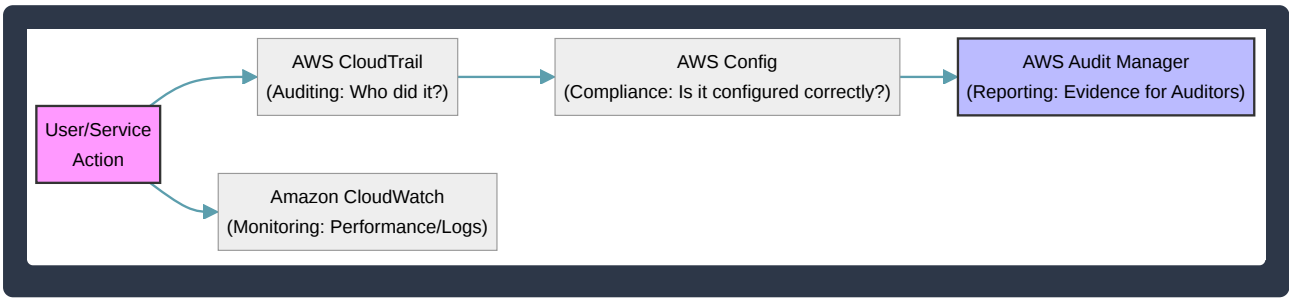
- **Encryption at Rest**: Protects data stored on physical media. This applies to data in services like **Amazon S3**, **Amazon EBS**, and **Amazon RDS**.
- **Encryption in Transit**: Protects data as it moves between a client and a server or between services. This is typically achieved using **TLS/SSL** certificates.

Governance, Auditing, and Monitoring

Governance services help organizations track changes, monitor performance, and prepare for audits.

- **Amazon CloudWatch**: A monitoring and observability service that collects **metrics**, logs, and events. It is used to trigger alarms when thresholds are met (e.g., high CPU usage).
- **AWS CloudTrail**: Records **API calls** made within an account. It provides a history of "who did what, when, and from where," which is essential for security auditing.

- **AWS Config**: Tracks the **configuration history** of AWS resources. It allows you to see how a resource was configured in the past and whether it currently complies with internal guidelines.
- **AWS Audit Manager**: Automates the collection of evidence to assess if your use of AWS services aligns with industry standards and regulations.
- **Access Reports**: Tools like the **IAM Credential Report** or **IAM Access Analyzer** help identify who has access to what resources and when passwords or access keys were last rotated.



Varying Compliance Requirements

It is critical to recognize that compliance is not “one size fits all” across the AWS platform.

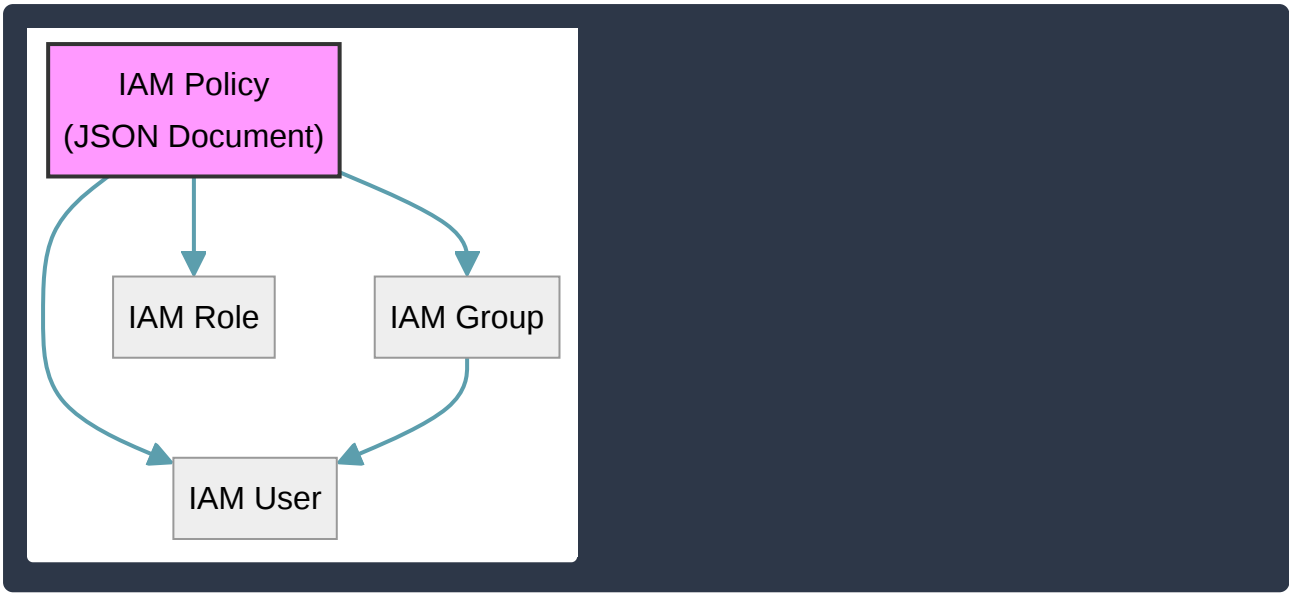
- **Service Eligibility**: Not every AWS service is eligible for every compliance standard. For example, a database service might be **HIPAA-eligible**, while a newer analytics service might still be undergoing the certification process.
- **Shared Responsibility**: While AWS is responsible for the security **of** the cloud (the physical data centers and hardware), the customer is responsible for security **in** the cloud (how they configure the services and protect their data).

AWS Access Management and Identity Security

AWS Identity and Access Management (IAM) is the foundational service used to manage access to AWS resources securely. It allows you to control who is authenticated (signed in) and authorized (has permissions) to use resources.

IAM Entities and the Principle of Least Privilege To maintain a secure environment, AWS recommends the **Principle of Least Privilege**: granting only the minimum permissions required to perform a task.

- **IAM Users**: Persistent identities representing a person or service.
- **IAM Groups**: Collections of users. Assigning a policy to a group grants those permissions to all users within it.
- **IAM Roles**: Temporary identities assumed by users, applications, or services. They do not have long-term credentials like passwords.
- **Managed Policies**: Pre-defined permissions created and maintained by AWS (e.g., `ReadOnlyAccess`).
- **Custom Policies**: Customer-managed policies tailored to specific organizational needs.



Authentication and Identity Management AWS supports various methods to verify identities and manage access across multiple accounts:

- **Multi-Factor Authentication (MFA)**: A security best practice that requires a second form of authentication (like a virtual token or hardware key) in addition to a password.
- **IAM Identity Center**: The successor to AWS Single Sign-On (SSO). It provides a central place to manage access to multiple AWS accounts and business applications.
- **Federated Identity**: Allows users to sign in using external identity providers (IdP) like Login with Amazon, Google, or corporate Active Directory via SAML 2.0.
- **Cross-Account IAM Roles**: Allows a user in one AWS account to perform tasks in another account without needing a separate set of credentials.

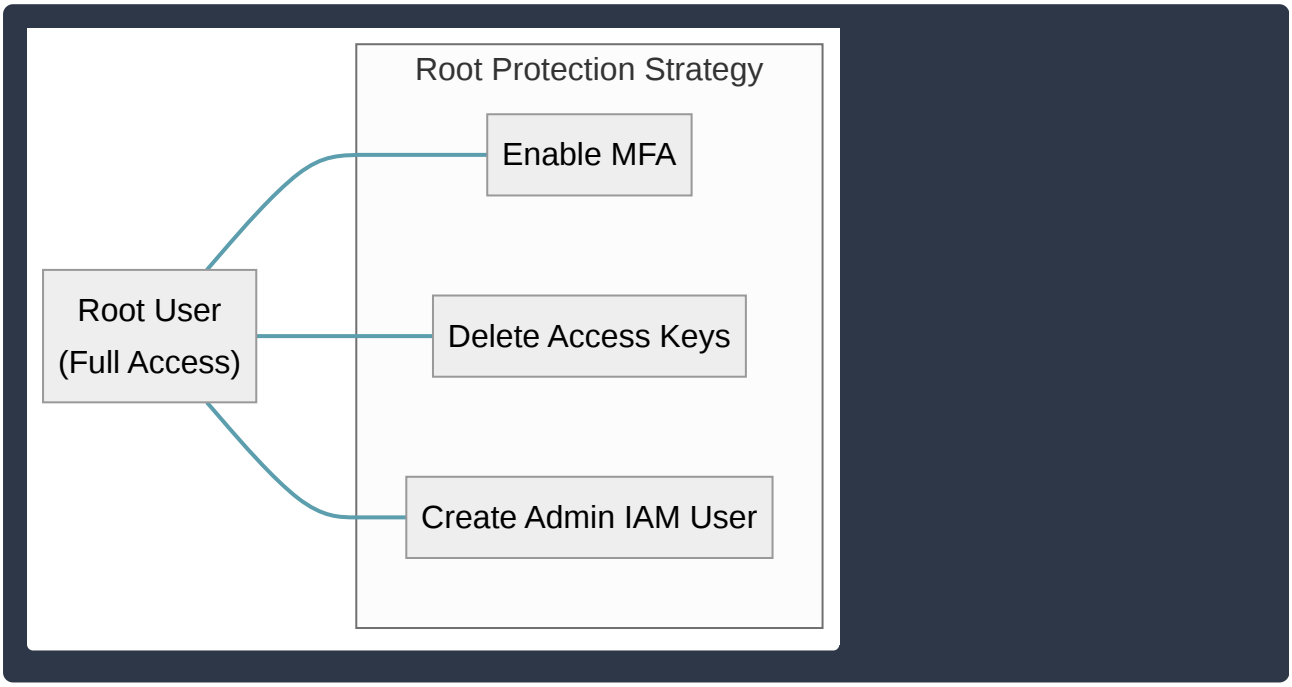
Credential Storage and Management Managing sensitive information requires specialized services to prevent hard-coding secrets in application code.

Service	Primary Use Case	Key Features
AWS Secrets Manager	Managing secrets like database credentials and API keys.	Supports automatic credential rotation and integration with RDS.
AWS Systems Manager (Parameter Store)	Storing configuration data and plain-text or encrypted strings.	Hierarchical storage for configuration and secrets; often used for environment variables.

- **Access Keys**: Consist of an **Access Key ID** and **Secret Access Key**. These are used for programmatic access via the AWS CLI or SDKs.
- **Password Policies**: Rules defined by administrators to enforce complexity, rotation, and expiration requirements for IAM user passwords.

The Account Root User The **Root User** is the identity created when the AWS account is first opened. It has complete, unrestricted access to all resources in the account.

- **Root-Only Tasks**: Only the root user can close the AWS account, change the account email address, change the AWS Support plan, or edit certain billing settings.
- **Root Protection**: Because of its power, you should never use the root user for daily tasks. Protection methods include:
  - Enabling a strong **MFA** device immediately.
  - Deleting root **access keys** to prevent programmatic access.
  - Creating an IAM user with administrative permissions for daily management.



Security Components and Resources

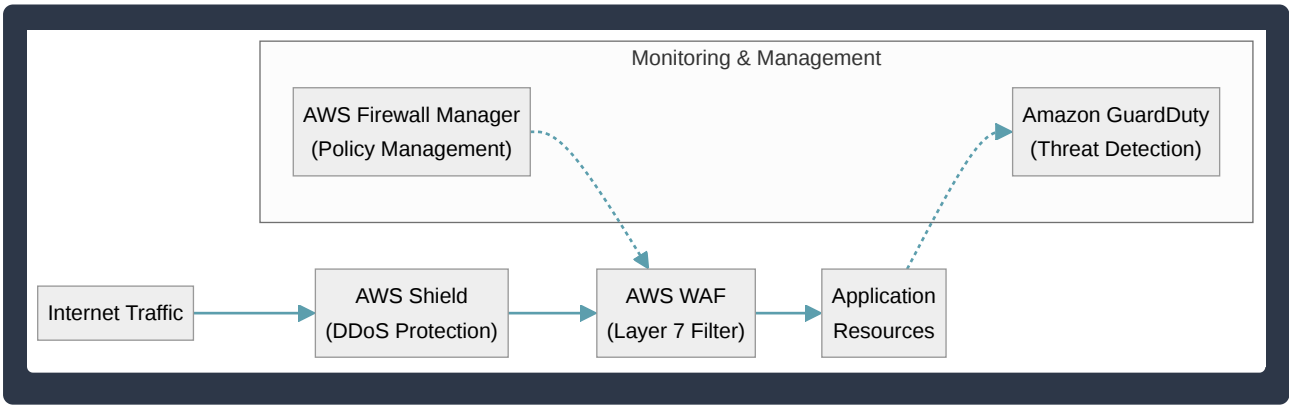
AWS provides a comprehensive suite of security services and resources designed to protect data, manage identities, and monitor infrastructure. These tools range from automated threat detection to centralized policy management, ensuring a robust security posture across the entire cloud environment.

Core AWS Security Services

AWS offers several native services to protect applications from external threats and manage security configurations at scale.

Service	Primary Function	Use Case
AWS WAF	Web Application Firewall	Protects web applications from common exploits like SQL injection and Cross-Site Scripting (XSS).
AWS Shield	DDoS Protection	Safeguards applications against Distributed Denial of Service (DDoS) attacks.
AWS Firewall Manager	Centralized Management	Simplifies administration of WAF, Shield, and Security Group rules across multiple accounts.
Amazon GuardDuty	Threat Detection	Uses machine learning to monitor for malicious activity and unauthorized behavior in your AWS accounts.

- **AWS WAF**: Allows you to create custom rules to block or allow traffic based on IP addresses, HTTP headers, or URI strings. It is typically deployed on Amazon CloudFront, Application Load Balancers, or Amazon API Gateway.
- **AWS Shield**: Comes in two tiers: **Shield Standard** (automatically enabled for all customers at no extra cost) and **Shield Advanced** (provides higher-level protection and 24/7 access to the Shield Response Team).
- **AWS Firewall Manager**: Essential for organizations using **AWS Organizations**. It ensures that new and existing resources comply with security policies automatically.
- **Amazon GuardDuty**: A continuous security monitoring service that analyzes data sources such as `AWS CloudTrail` management events, `VPC Flow Logs`, and `DNS Logs` to identify threats like crypto-mining or compromised credentials.



Identifying Security Issues and Best Practices

AWS provides tools to help customers proactively identify vulnerabilities and misconfigurations.

- **AWS Trusted Advisor**: This service inspects your AWS environment and makes recommendations in five categories, including **Security**. It identifies issues such as publicly accessible S3 buckets, unrestricted ports in security groups, and whether Multi-Factor Authentication (MFA) is enabled on the root account.

- **Third-Party Security Products**: While AWS offers many native tools, customers can also find, buy, and deploy third-party security software from the **AWS Marketplace**. This includes familiar vendors for firewalls, endpoint security, and vulnerability scanners, allowing customers to use existing licenses or pay-as-you-go.

Accessing Security Information

Staying informed about the latest security threats and AWS best practices is critical for maintaining a secure environment.

- **AWS Knowledge Center**: A repository of the most frequent questions and requests received by AWS Support. It contains “how-to” guides and troubleshooting steps for security configurations.
- **AWS Security Center**: A central hub that provides guidance on security fundamentals, compliance, and the shared responsibility model.
- **AWS Security Blog**: A resource for the latest security news, deep dives into new features, and expert advice on securing AWS workloads.

Domain 3: Cloud Technology and Services

Methods of Deploying and Operating in AWS

To effectively manage resources in the AWS Cloud, users must choose the right tools based on their technical requirements, scale, and frequency of tasks. AWS provides several interfaces to interact with its services, ranging from visual web consoles to automated code-based deployments.

AWS Interaction Methods

There are three primary ways to interact with AWS services: the Management Console, programmatic tools (CLI and SDKs), and Infrastructure as Code (IaC).

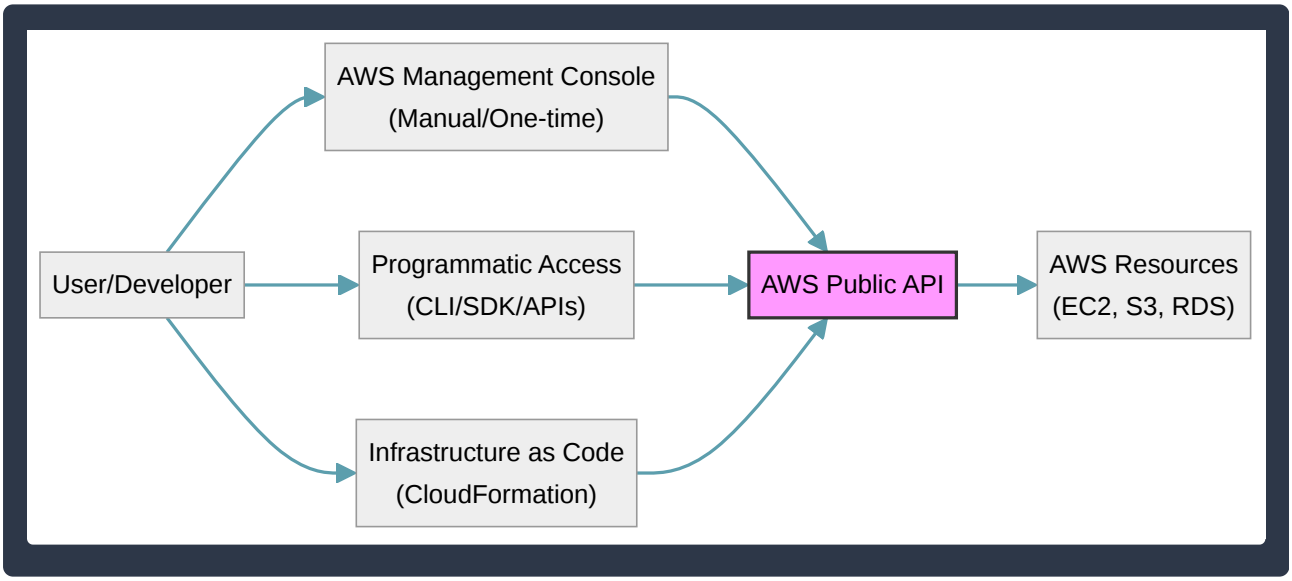
- **AWS Management Console**: A web-based graphical user interface (GUI) used to manage AWS resources. It is ideal for beginners, performing one-time configuration tasks, or visually inspecting resources.
- **AWS Command Line Interface (CLI)**: A unified tool that allows users to control AWS services from the command line and automate them through scripts. It is faster for experienced users than navigating the GUI.
- **Software Development Kits (SDKs)**: These provide language-specific APIs (e.g., for Python, Java, or JavaScript) that allow developers to integrate AWS services directly into their application code.
- **Infrastructure as Code (IaC)**: A method of provisioning and managing resources using machine-readable definition files. **AWS CloudFormation** is the primary service for IaC, allowing users to treat their infrastructure just like application code.

Method	Best Use Case	Skill Level
Management Console	Visual exploration, manual tasks, learning	Beginner
AWS CLI	Quick commands, shell scripting, automation	Intermediate
AWS SDKs	Building applications that interact with AWS	Developer
CloudFormation (IaC)	Repeatable, consistent environment deployments	Advanced

One-Time Operations vs. Repeatable Processes

When deciding how to deploy, organizations must evaluate whether a task is a **one-time operation** or a **repeatable process**.

- **One-time Operations**: These are manual tasks performed once, such as checking a billing dashboard or testing a new service. These are typically handled via the **AWS Management Console**.
- **Repeatable Processes**: For production environments, consistency is critical. Using **Infrastructure as Code (IaC)** or scripts ensures that environments (Development, Testing, Production) are identical, reducing human error and configuration drift.



Cloud Deployment Models

AWS supports different deployment models depending on where the infrastructure resides and how it is managed.

- **Cloud (All-in)**: A cloud-based application is fully deployed in the cloud, and all parts of the application run in the cloud. Applications in this model are either created in the cloud or migrated from existing infrastructure.
- **Hybrid**: A hybrid deployment connects infrastructure and applications between cloud-based resources and existing resources located on-premises. This is common for organizations

migrating to the cloud or those with specific regulatory requirements that require some data to stay local.

- **On-premises (Private Cloud)**: Also known as “private cloud” deployment. While this uses virtualization and resource management tools, it resides in a local data center. AWS services like **AWS Outposts** allow users to run native AWS services on-premises for a consistent experience.

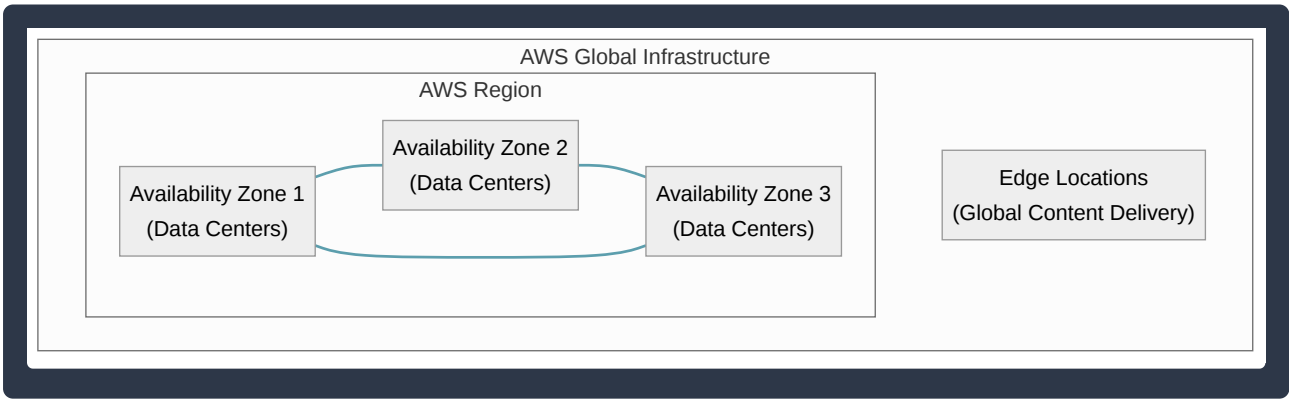
AWS Global Infrastructure: Regions, AZs, and Edge Locations

The AWS Global Infrastructure is the physical backbone that supports all AWS services. It is designed to be the most flexible and secure cloud computing environment available, organized into a hierarchy of Regions, Availability Zones, and Edge Locations.

Core Components and Relationships

AWS infrastructure is categorized into three primary layers that work together to provide global reach and local performance:

- **Regions**: A **Region** is a physical location in the world where AWS clusters data centers. Each Region is geographically isolated from other Regions to ensure maximum fault tolerance and stability.
- **Availability Zones (AZs)**: Each Region consists of multiple, isolated **Availability Zones**. An AZ is one or more discrete data centers with redundant power, networking, and connectivity.
- **Edge Locations**: These are specialized sites used by services like **Amazon CloudFront** to cache content closer to end users, reducing latency. Edge locations are typically located in major cities and are more numerous than Regions.



High Availability and Fault Tolerance

AWS enables users to build highly available applications by leveraging the physical separation of Availability Zones.

- **Achieving High Availability (HA)**: By deploying resources (like `Amazon EC2` instances or `Amazon RDS` databases) across multiple AZs, an application can remain operational even if one AZ experiences a failure. If one zone goes offline, traffic is automatically routed to the remaining healthy zones.

- **No Single Points of Failure**: Availability Zones are physically separated by a meaningful distance (miles) within a Region. They do not share **single points of failure**; each AZ has its own independent power infrastructure, cooling systems, and network connectivity. They are connected to each other via fast, private fiber-optic networking.

Multi-Region Strategies

While most applications function well within a single Region using multiple AZs, certain business requirements necessitate a **Multi-Region** architecture.

Use Case	Description
Disaster Recovery (DR)	If an entire geographic area is impacted by a natural disaster, having a backup in a different Region ensures data can be recovered.
Business Continuity	Ensures that critical business functions can continue to operate with minimal downtime during a large-scale regional outage.
Low Latency	Deploying applications in Regions physically closer to your global customers reduces the time it takes for data to travel (lag).
Data Sovereignty	Some countries require that specific types of data (like financial or health records) remain within their national borders to comply with local laws.

By understanding these components, organizations can design architectures that balance cost, performance, and resilience according to their specific needs.

AWS Compute Services

AWS compute services provide the processing power required to run applications. These services range from virtual servers where you manage the operating system to serverless options where AWS handles all underlying infrastructure.

Amazon EC2 and Instance Types

Amazon EC2 (Elastic Compute Cloud) provides secure, resizable virtual servers in the cloud. When launching an EC2 instance, you must choose an **Instance Type** optimized for specific workloads.

Instance Family	Best Use Case	Key Characteristics
General Purpose	Web servers, small databases	Balanced CPU, memory, and networking
Compute Optimized	High-performance web servers, batch processing	High-performance processors (CPU-intensive)
Memory Optimized	In-memory databases, real-time big data analytics	Large RAM capacity for processing large datasets
Storage Optimized	NoSQL databases, data warehousing	High, sequential read/write access to local storage
Accelerated Computing	Machine learning, graphics rendering	Hardware accelerators (GPUs or FPGAs)

Container Services

Containers allow you to package an application and its dependencies into a single image. AWS offers two primary orchestration services:

- **Amazon Elastic Container Service (Amazon ECS)**: A highly scalable, high-performance container management service that is native to AWS. It is the simplest way to run containers on AWS.
- **Amazon Elastic Kubernetes Service (Amazon EKS)**: A managed service that makes it easy to run Kubernetes on AWS without needing to install or operate your own Kubernetes control plane.

Serverless Compute

Serverless computing allows you to build and run applications without managing infrastructure.

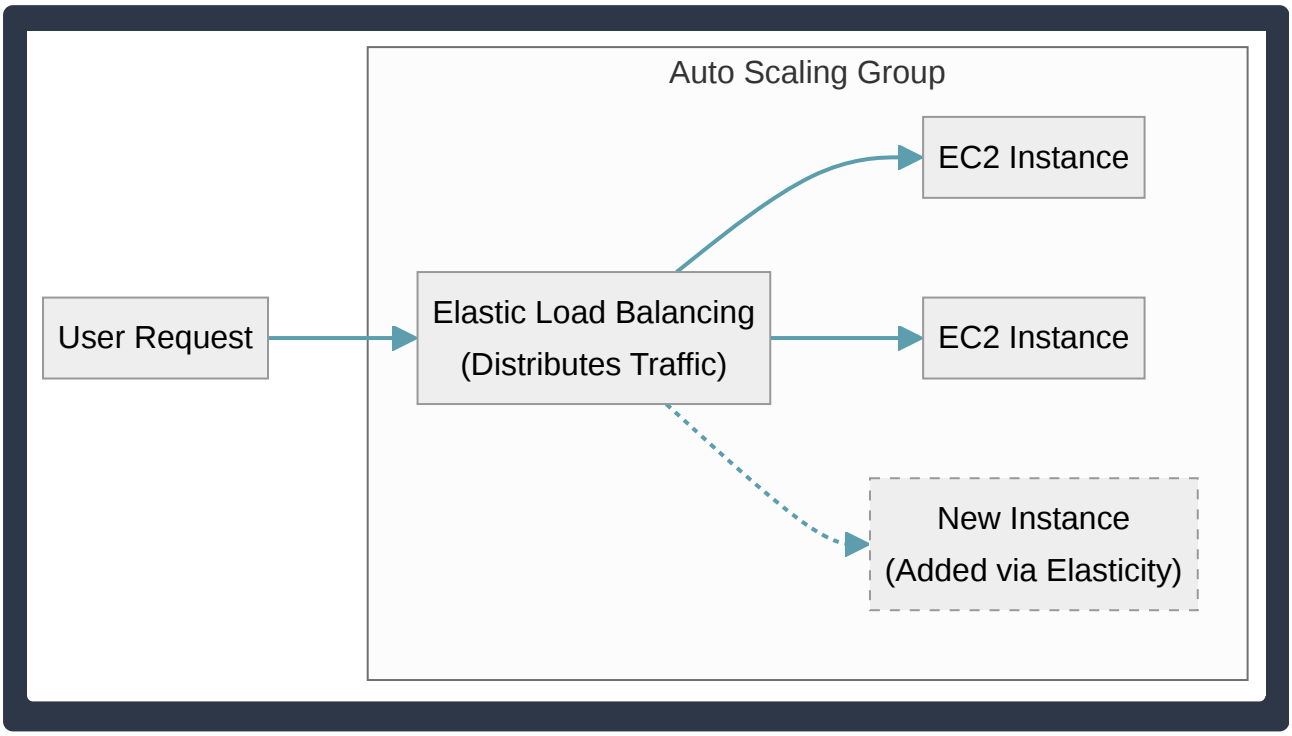
- **AWS Lambda**: A service that lets you run code in response to events (such as an image upload or a database update) without provisioning or managing servers. You pay only for the compute time you consume.
- **AWS Fargate**: A serverless compute engine for containers that works with both ECS and EKS. With Fargate, you no longer have to manage the underlying EC2 instances; you simply define the container’s requirements.

Elasticity and Load Balancing

To ensure applications remain available and cost-effective, AWS uses automation to handle changes in traffic.

- **Amazon EC2 Auto Scaling**: This service provides **elasticity** by automatically adding or removing EC2 instances based on defined conditions (such as CPU utilization). This ensures you have enough instances to handle the load but aren’t paying for idle resources.

- **Elastic Load Balancing (ELB)**: ELB automatically distributes incoming application traffic across multiple targets, such as EC2 instances, containers, or IP addresses. Its primary purposes are to increase application fault tolerance and ensure no single resource is overwhelmed.



Summary of Use Cases

- Use **Amazon EC2** when you need full control over the operating system or specific software requirements.
- Use **AWS Lambda** for short-running, event-driven tasks like processing data streams.
- Use **AWS Fargate** when you want to run containers but do not want to manage the underlying virtual servers.
- Use **Amazon EC2 Auto Scaling** and **ELB** together to create a highly available, elastic architecture that scales with user demand. AWS offers a wide range of purpose-built database services, allowing users to choose the right tool for specific workloads. Selecting the correct service depends on the data structure, performance requirements, and the level of management control desired.

EC2 Hosted vs. AWS Managed Databases

When deploying a database on AWS, you must choose between managing the database yourself on **Amazon EC2** or using a managed service like **Amazon RDS**.

Feature	EC2 Hosted Database	AWS Managed Database (RDS/Aurora)
Management	Customer manages OS, DB installation, and patching	AWS manages OS, DB patching, and hardware
Backups	Customer-configured and managed	Automated backups and snapshots included
High Availability	Customer must manually configure replication	Built-in Multi-AZ deployment options
Control	Full root/administrative access to the OS	Access limited to database-level configuration

- **Use EC2 Hosted** when you require a specific database engine or version not supported by AWS, or when you need deep access to the underlying operating system.
- **Use Managed Services** to reduce operational overhead ("undifferentiated heavy lifting") and focus on application development.

Relational Databases (SQL)

Relational databases store data in structured tables with predefined schemas. They are ideal for complex queries and transactional consistency.

- **Amazon RDS (Relational Database Service)**: A managed service that makes it easy to set up, operate, and scale relational databases. It supports six popular engines: **MySQL**, **PostgreSQL**, **MariaDB**, **Oracle**, **SQL Server**, and **Amazon Aurora**.
- **Amazon Aurora**: A cloud-native relational database engine compatible with MySQL and PostgreSQL. It is designed to be up to 5x faster than standard MySQL and features a self-healing, fault-tolerant storage system that automatically scales up to 128 TiB.

NoSQL and In-Memory Databases

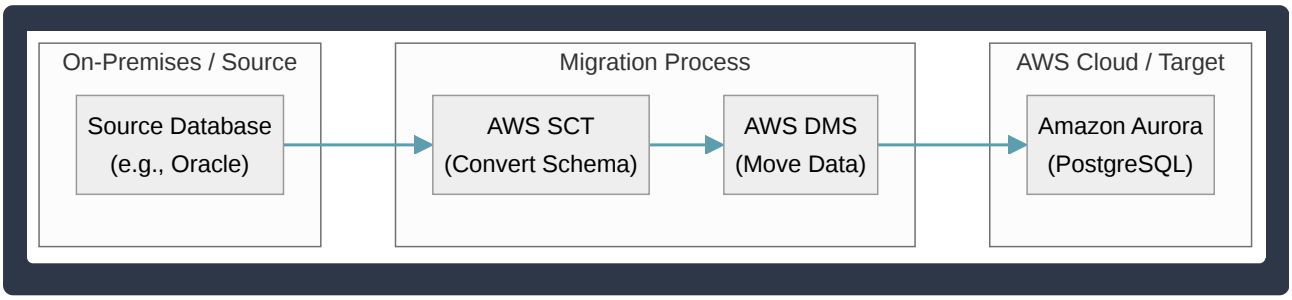
Non-relational databases are designed for specific data models and offer high scalability.

- **Amazon DynamoDB**: A fully managed, serverless **NoSQL** database service. It provides consistent, single-digit millisecond latency at any scale. It uses a key-value and document data model, making it ideal for mobile, web, gaming, and IoT applications.
- **Amazon ElastiCache**: A managed **in-memory** data store and cache service. It supports two open-source engines: **Redis** and **Memcached**. It is used to significantly improve application performance by retrieving data from high-speed memory instead of slower disk-based databases.

Database Migration Tools

AWS provides specialized tools to help move existing databases into the cloud with minimal downtime.

- **AWS Database Migration Service (AWS DMS)**: A service used to migrate data from a source database to a target database. The source remains fully operational during the migration. It can handle **homogeneous migrations** (e.g., Oracle to Oracle) and **heterogeneous migrations** (e.g., Oracle to Aurora).
- **AWS Schema Conversion Tool (AWS SCT)**: Used during heterogeneous migrations to automatically convert the source database schema and code (like views and stored procedures) into a format compatible with the target AWS database.

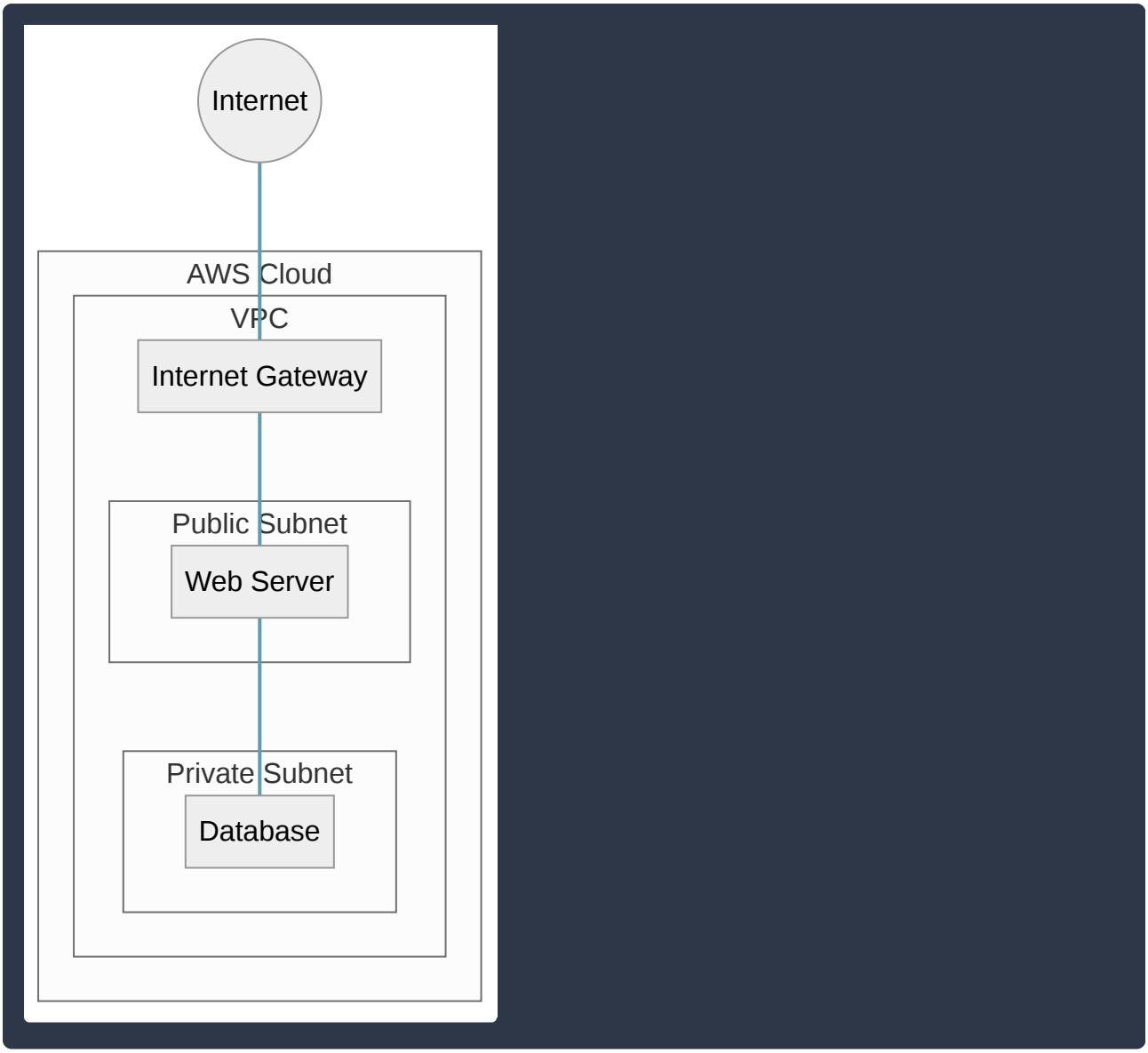


AWS Network Services

AWS provides a robust suite of networking services that allow you to create isolated virtual networks, manage traffic, and connect your on-premises environment to the cloud.

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

- **Subnets**: A range of IP addresses in your VPC.
  - **Public Subnet**: Directly connected to the internet via an Internet Gateway.
  - **Private Subnet**: Not directly accessible from the internet; typically used for databases or application servers.
- **Internet Gateway (IGW)**: A horizontally scaled, redundant, and highly available VPC component that allows communication between your VPC and the internet.
- **NAT Gateway**: Allows instances in a private subnet to connect to the internet (e.g., for software updates) while preventing the internet from initiating a connection with those instances.



VPC Security Security in a VPC is provided through multiple layers of defense. While **Amazon Inspector** is not a network service per se, it is a critical security tool that automatically discovers and scans EC2 instances and container images for software vulnerabilities and unintended network exposure.

Feature	Security Group	Network ACL (NACL)
Level	Operates at the **Instance** level.	Operates at the **Subnet** level.
Type	**Stateful**: Return traffic is automatically allowed.	**Stateless**: Return traffic must be explicitly allowed.
Rules	Supports “Allow” rules only.	Supports “Allow” and “Deny” rules.
Evaluation	All rules are evaluated before traffic is allowed.	Rules are processed in chronological order (lowest number first).

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

- **Domain Registration**: You can purchase and manage domain names.
- **Health Checking**: It monitors the health of your resources; if a resource is down, Route 53 can route traffic to a healthy resource.

Network Connectivity Options Organizations often need to connect their on-premises data centers to their AWS VPCs.

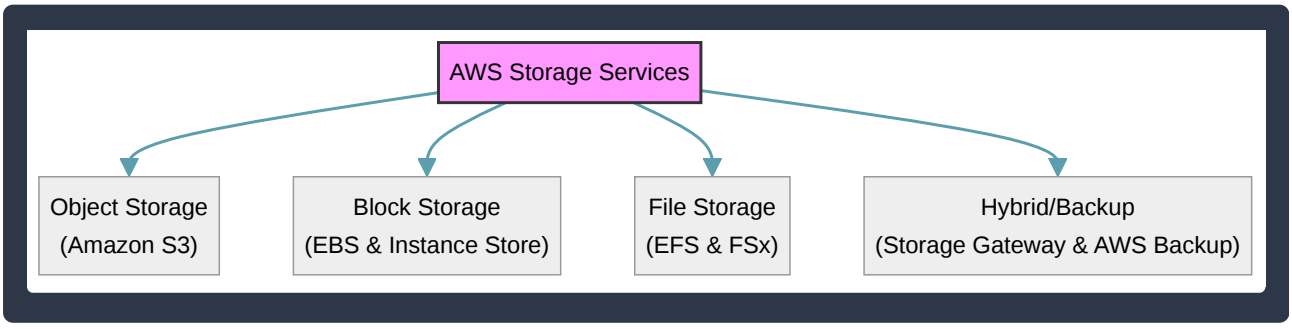
- **AWS Site-to-Site VPN**: Creates an encrypted tunnel between your network and your VPC. It is quick to set up and uses the public internet.
- **AWS Direct Connect**: A cloud service solution that makes it easy to establish a dedicated network connection from your premises to AWS. It bypasses the internet entirely, providing more consistent network performance and reduced bandwidth costs.

Connectivity Option	Use Case	Connection Type
AWS VPN	Quick setup, low-to-moderate bandwidth needs.	Encrypted via Public Internet
AWS Direct Connect	High-volume data transfer, consistent performance, increased security.	Private Dedicated Physical Fiber

AWS offers a diverse portfolio of storage services designed to meet specific needs for performance, durability, and cost. Understanding the differences between object, block, and file storage is essential for selecting the right service for a given workload.

AWS Storage Services and Use Cases

AWS storage is categorized by how data is accessed and stored. The following diagram illustrates the primary categories:



Object Storage: Amazon S3 Amazon S3 (Simple Storage Service) stores data as objects within “buckets.” It is highly scalable and designed for 99.999999999% (11 nines) of durability.

- **Uses**: Storing photos/videos, hosting static websites, data lakes for analytics, and software distribution.
- **S3 Lifecycle Policies**: These allow you to automate the movement of data between storage classes or delete data after a certain period to reduce costs. For example, moving data to archive storage after 30 days.

S3 Storage Class	Use Case	Access Speed
S3 Standard	Frequently accessed data	Milliseconds
S3 Intelligent-Tiering	Data with unknown or changing access patterns	Milliseconds
S3 Standard-IA	Infrequently accessed data (lower storage price, retrieval fee)	Milliseconds
S3 One Zone-IA	Non-critical, infrequent data stored in one AZ	Milliseconds
S3 Glacier	Long-term archive (Flexible or Deep Archive)	Minutes to Hours

Block Storage: Amazon EBS and Instance Store Block storage provides dedicated storage volumes for EC2 instances, functioning like a physical hard drive.

- **Amazon EBS** (Elastic Block Store): Persistent storage that exists independently of the EC2 instance. If the instance is stopped, the data remains. It is used for databases and boot volumes.
- **Instance Store**: Ephemeral (temporary) storage physically attached to the host computer. It provides very high IOPS and low latency but **data is lost** if the instance is stopped or terminated.

File Storage: Amazon EFS and Amazon FSx File storage allows multiple compute resources to access a shared file system simultaneously.

- **Amazon EFS** (Elastic File System): A serverless, fully managed NFS (Network File System) primarily for Linux workloads. It scales automatically as files are added.
- **Amazon FSx**: Provides fully managed third-party file systems.
  - **FSx for Windows File Server**: Built on Windows Server for SMB-based applications.
  - **FSx for Lustre**: Designed for high-performance computing (HPC) and machine learning.

Hybrid and Data Protection Services

- **AWS Storage Gateway**: A hybrid cloud storage service that gives on-premises applications access to virtually unlimited cloud storage. It provides a **cached file system** where frequently accessed data is kept locally for low-latency access, while the rest is stored in Amazon S3.
- **AWS Backup**: A centralized service used to automate and manage backups across multiple AWS services (such as EBS, RDS, and EFS). It ensures compliance and provides a single place to configure backup policies and monitor activity.

Summary of Use Cases

- Use **Amazon S3** for web assets and long-term backups.
- Use **Amazon EBS** for database storage requiring persistence.
- Use **Instance Store** for temporary scratch space or caching.
- Use **Amazon EFS** for shared content management or home directories.
- Use **AWS Backup** to centralize data protection across the entire AWS environment.

AWS Artificial Intelligence, Machine Learning, and Analytics Services

AWS provides a comprehensive suite of services designed to help organizations process vast amounts of data and build intelligent applications. These services range from “pre-packaged” AI tools that require no machine learning expertise to powerful analytics engines that process data in real-time.

Artificial Intelligence (AI) and Machine Learning (ML)

AWS AI/ML services allow developers to add intelligence to applications or build custom models from scratch.

- **Amazon SageMaker AI**: A fully managed service that provides every developer and data scientist with the ability to build, train, and deploy machine learning (ML) models quickly. It removes the heavy lifting from each step of the ML process.
- **Amazon Lex**: A service for building conversational interfaces into any application using voice and text. It provides the deep learning functionalities of natural language understanding (NLU) and automatic speech recognition (ASR)—the same technology that powers Amazon Alexa.
- **Amazon Kendra**: An intelligent search service powered by machine learning. It allows users to search through unstructured data (like PDFs, Word docs, and FAQs) using natural language questions rather than just keywords.
- **Amazon Q**: A generative AI-powered assistant designed for work that can be tailored to your business. It helps users get answers, solve problems, generate content, and take actions using data and expertise found in your company’s systems.

Data Analytics Services

Analytics services help you transform raw data into meaningful insights. These services handle different stages of the data lifecycle, from ingestion to visualization.

- **Amazon Athena**: An interactive query service that makes it easy to analyze data in **Amazon S3** using standard SQL. It is serverless, so there is no infrastructure to manage, and you pay only for the queries you run.
- **Amazon Kinesis**: A platform for handling real-time data streams. It allows you to collect, process, and analyze data (such as website clickstreams or IoT telemetry) as it arrives, rather than waiting for batch processing.
- **AWS Glue**: A serverless data integration service that makes it easy to discover, prepare, and combine data for analytics. It is primarily used for **ETL** (Extract, Transform, and Load) jobs and

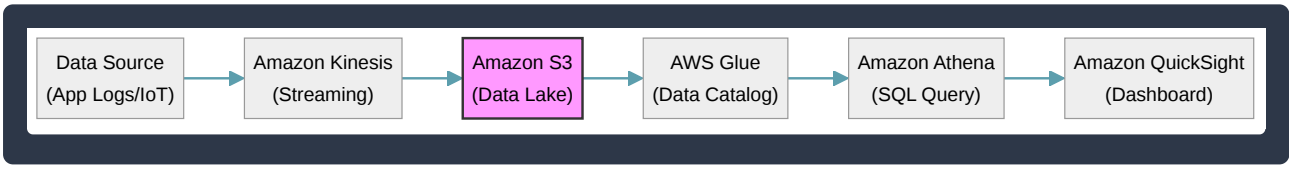
maintaining a **Data Catalog**.

- **Amazon QuickSight**: A cloud-powered Business Intelligence (BI) service. It allows you to create and publish interactive dashboards that include ML-powered insights and can be accessed from any device.

Service	Primary Use Case	Key Benefit
Amazon Athena	Querying log files stored in S3	No ETL required; use standard SQL
AWS Glue	Moving data between databases	Serverless ETL and data discovery
Amazon Kinesis	Processing real-time video or data streams	Immediate insights from live data
Amazon QuickSight	Creating executive dashboards	Fast, easy-to-use visualizations

Typical Data Analytics Workflow

The following diagram illustrates how these services often work together to move data from a source to a final visualization.



- **Example Use Case**: A retail company uses **Amazon Kinesis** to capture live website traffic. The data is stored in **Amazon S3**, where **AWS Glue** organizes it. Analysts then use **Amazon Athena** to run SQL queries on that data to find trends, and finally, they display those trends in **Amazon QuickSight** for the management team. AWS provides a wide array of specialized services designed to handle specific business workflows, application development, and end-user requirements. These services extend the cloud's utility beyond basic compute and storage into areas like messaging, IoT, and virtual desktops.

Messaging, Alerts, and Business Applications

AWS offers several services to facilitate communication between application components or directly to end users.

- **Amazon Simple Notification Service (SNS)**: A highly available, durable, secure, fully managed pub/sub messaging service. Use this for **push-based** notifications to many subscribers (e.g., sending an SMS or email alert when an alarm triggers).
- **Amazon Simple Queue Service (SQS)**: A fully managed message queuing service that enables you to decouple and scale microservices, distributed systems, and serverless applications. Use this for **pull-based** asynchronous processing.
- **Amazon Simple Email Service (SES)**: A cost-effective, flexible, and scalable email service that enables developers to send mail from within any application. Use this for marketing or transactional emails.

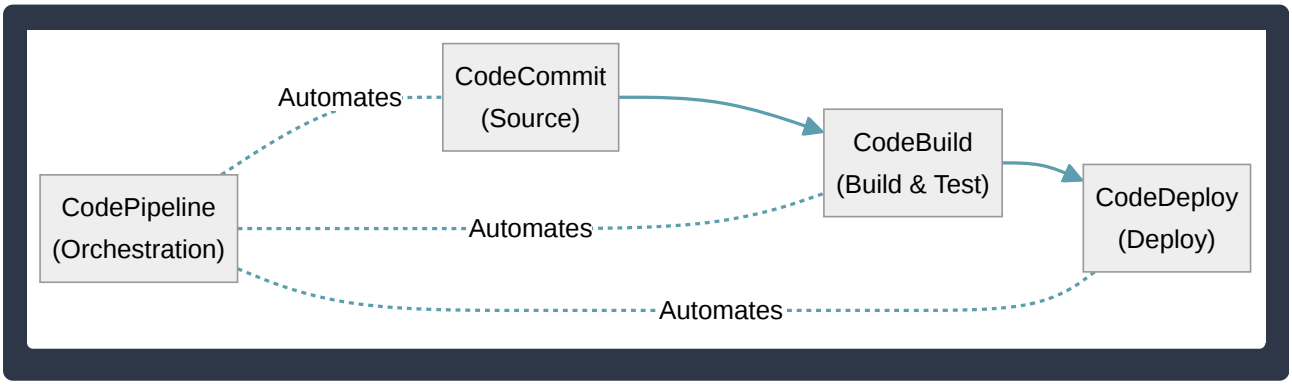
- **Amazon Connect**: An easy-to-use omnichannel cloud contact center that helps companies provide superior customer service at a lower cost.
- **Amazon Pinpoint**: A communication service used to engage customers by sending targeted marketing messages via email, SMS, push notifications, or voice.

Service	Primary Use Case	Communication Pattern
Amazon SNS	Alerts, notifications, fan-out	Push (One-to-Many)
Amazon SQS	Decoupling components, buffering	Pull (One-to-One)
Amazon SES	Marketing and transactional email	Bulk Email

Developer Tools and Troubleshooting

AWS provides a suite of tools to manage the application lifecycle, from writing code to deploying and monitoring it.

- **AWS Cloud9**: A cloud-based integrated development environment (IDE) that lets you write, run, and debug code with just a browser.
- **AWS CodeCommit**: A secure, highly scalable, managed source control service that hosts private Git repositories.
- **AWS CodeBuild**: A fully managed build service that compiles source code, runs tests, and produces software packages.
- **AWS CodeDeploy**: A service that automates code deployments to any instance, including Amazon EC2 and on-premises servers.
- **AWS CodePipeline**: A fully managed continuous delivery service that helps you automate your release pipelines for fast and reliable application and infrastructure updates.
- **AWS X-Ray**: A tool that helps developers analyze and debug distributed applications, such as those built using a microservices architecture.



End-User Computing and Mobile Services

These services focus on delivering application interfaces and environments to end users, regardless of their hardware.

- **Amazon WorkSpaces**: A managed, secure Desktop-as-a-Service (DaaS) solution. It provides users with a **full virtual desktop** (Windows or Linux) accessible from any supported device.
- **Amazon AppStream 2.0**: A fully managed non-persistent application streaming service. It allows users to access **specific desktop applications** through a web browser without installing them.
- **AWS Amplify**: A set of tools and features that lets web and mobile developers easily build, ship, and host full-stack applications on AWS.
- **AWS Device Farm**: An app testing service that lets you test and interact with your Android, iOS, and web apps against real, physical phones and tablets hosted by AWS.

IoT and Business Support

- **AWS IoT Core**: A managed cloud service that lets connected devices (sensors, appliances, etc.) easily and securely interact with cloud applications and other devices. It can support billions of devices and trillions of messages.
- **AWS Support Plans**: AWS offers four levels of support (Basic, Developer, Business, and Enterprise) to provide technical assistance and guidance.
- **AWS Trusted Advisor**: An online tool that provides real-time guidance to help you provision your resources following AWS best practices for cost optimization, security, and performance.

Domain 4: Billing, Pricing, and Support

AWS Pricing Models and Data Transfer Costs

AWS offers a variety of pricing models designed to provide flexibility and cost optimization based on workload requirements. Understanding these models is essential for managing cloud spend effectively.

Compute Purchasing Options

AWS provides several ways to pay for compute resources like Amazon EC2. Choosing the right model depends on the predictability and duration of the workload.

Purchasing Option	Best Use Case	Key Feature
On-Demand	Short-term, unpredictable workloads or application development.	No long-term commitment; pay by the second or hour.
Savings Plans	Consistent compute usage across EC2, Fargate, and Lambda.	Commitment to a consistent amount of usage (e.g., $10/hour) for 1 or 3 years.
Reserved Instances (RI)	Steady-state workloads with predictable usage.	Commitment to a specific instance configuration for 1 or 3 years.
Spot Instances	Fault-tolerant, flexible, or stateless applications (e.g., batch processing).	Use spare AWS capacity for up to a 90% discount; can be interrupted by AWS.

Reserved Instance Flexibility and AWS Organizations

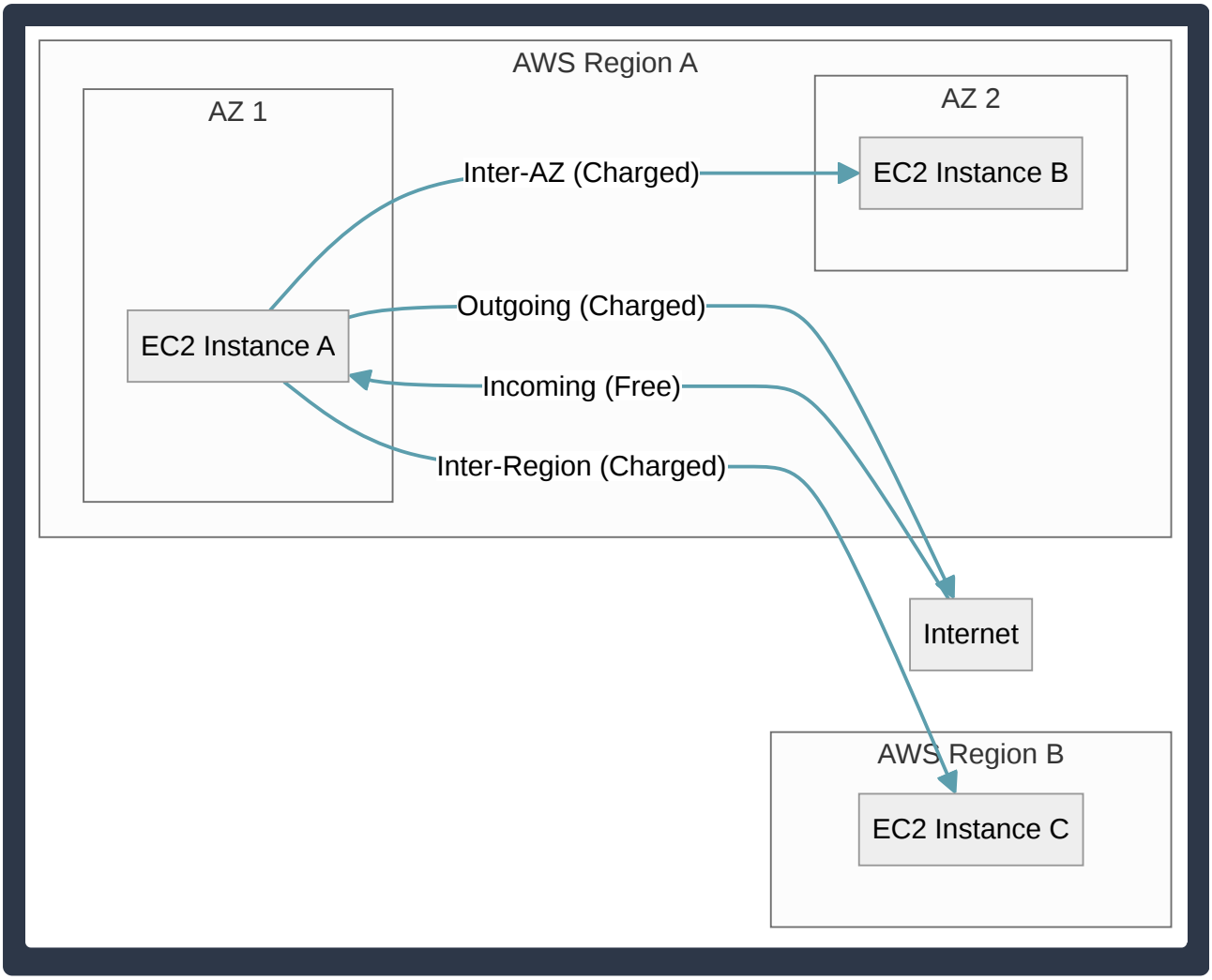
Reserved Instances (RIs) provide significant discounts compared to On-Demand pricing. They offer different levels of flexibility:

- **Standard RIs**: Offer the highest discount but are limited in how they can be modified.
- **Convertible RIs**: Allow you to change instance families, operating systems, or tenancies during the term.
- **Regional vs. Zonal**: Regional RIs provide “instance size flexibility,” meaning the discount applies to any size instance within the same family (e.g., a `t3.large` discount can apply to two `t3.medium` instances).

In **AWS Organizations**, which allows for consolidated billing across multiple accounts, RI benefits are shared by default. If one account purchases an RI but does not use it, the discount is automatically applied to matching usage in any other account within the organization. This ensures maximum utilization of the reserved capacity.

Data Transfer Costs

Data transfer costs vary based on the source and destination of the traffic. Generally, data entering the AWS environment is free, while data leaving or moving between locations incurs costs.



- **Incoming Data Transfer**: Data transferred into AWS from the internet is typically free.
- **Outgoing Data Transfer**: Data transferred out to the internet is charged per GB (after the first 100 GB per month, which is free).
- **Inter-Region Transfer**: Moving data from one AWS Region to another (e.g., US East to EU West) always incurs a cost.
- **Intra-Region Transfer**: Data moving between Availability Zones (AZs) within the same Region is charged, whereas data moving within the same AZ is generally free.

Storage Pricing and Tiers

Storage pricing is primarily based on the amount of data stored per month, the storage tier, and data retrieval requests.

- **Amazon S3 Tiers**:
  - **S3 Standard**: High durability and availability for frequently accessed data.
  - **S3 Intelligent-Tiering**: Automatically moves data between tiers to save costs based on access patterns.
  - **S3 Standard-IA / One Zone-IA**: Lower cost for infrequently accessed data, but incurs a retrieval fee.

    - **S3 Glacier (Instant, Flexible, Deep Archive)**: Lowest cost for long-term archiving; retrieval times range from minutes to hours.
- **Amazon EBS**: Pricing is based on the volume type (e.g., **General Purpose SSD** `gp3` vs. **Provisioned IOPS** `io2`), the amount of storage provisioned (GB per month), and the performance (IOPS and throughput) requested.

Billing, Budget, and Cost Management Resources

Managing cloud spend requires a combination of planning, real-time monitoring, and historical analysis. AWS provides several tools to help users estimate costs before deployment, track spending against targets, and analyze usage patterns to identify savings opportunities.

AWS Planning and Analysis Tools

AWS offers distinct tools for different stages of the cloud journey, from initial estimation to ongoing visualization.

Tool	Primary Use Case	Key Capability
AWS Pricing Calculator	Pre-deployment planning	Estimate the cost of a solution before building it.
AWS Cost Explorer	Historical analysis and forecasting	Visualize and graph patterns in AWS spending over time.
AWS Budgets	Proactive monitoring and alerting	Set custom limits and receive notifications when costs exceed thresholds.

- **AWS Pricing Calculator**: This is a web-based tool used to create cost estimates for your specific use cases. You can model your solutions (e.g., specifying the number of `EC2` instances, `S3` storage volume, and data transfer) to see an estimated monthly bill.
- **AWS Cost Explorer**: This tool allows you to visualize your cost and usage data. It provides default reports (e.g., monthly costs by service) and allows you to filter data by attributes like Availability Zone, instance type, or tags. It also provides a **forecast** for the next 12 months based on your historical data.
- **AWS Budgets**: Unlike Cost Explorer, which is for analysis, **AWS Budgets** is for action. You can set budgets based on cost, usage, or Reserved Instance (RI) utilization. You can configure alerts via email or Amazon SNS to notify you when your actual or **forecasted** costs exceed your budget.

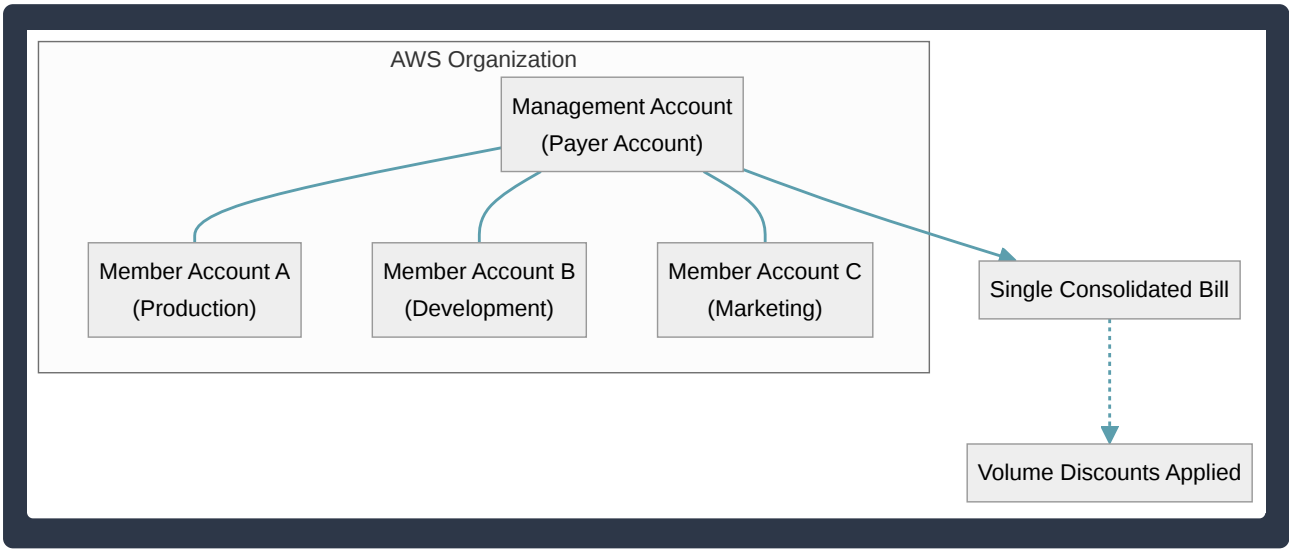
AWS Organizations and Consolidated Billing

AWS Organizations is an account management service that enables you to consolidate multiple AWS accounts into an organization that you create and centrally manage.

- **Consolidated Billing**: This feature treats all accounts in the organization as one for billing purposes. It provides a single payment method for the entire organization and allows you to see

a combined view of charges.

- **Volume Discounts**: By combining usage across all accounts, the organization can reach pricing tiers for services like `S3` or `EC2` data transfer faster, resulting in lower overall costs.
- **Cost Allocation**: Even with a single bill, the master account can track the spending of each member account individually.



Cost Allocation Tags and Reporting

To manage costs effectively at a granular level, AWS uses **Cost Allocation Tags**. These are key-value pairs attached to AWS resources (e.g., `Department: Finance` or `Project: Alpha`).

- **AWS-generated tags**: These are automatically applied by AWS (e.g., `createdBy`).
- **User-defined tags**: These are defined by the customer to track specific internal categories.
- **AWS Cost and Usage Report (CUR)**: This is the most comprehensive tool for cost data. It lists AWS usage for each service category used by an account and its users in hourly or daily line items. When cost allocation tags are activated, they appear in the CUR, allowing businesses to perform detailed “showback” or “chargeback” to specific departments. AWS provides a comprehensive ecosystem of documentation, support plans, and professional expertise to help customers build, migrate, and optimize their cloud environments.

AWS Technical Resources and Documentation

AWS offers several self-service platforms to help users find technical information and architectural best practices:

- **AWS Documentation**: The primary source for technical specifications, API references, and user guides for all AWS services.
- **AWS Whitepapers**: Deep-dive technical papers covering architecture, security, and compliance, often written by AWS engineers.
- **AWS Blogs**: Frequent updates on new feature launches, tutorials, and customer success stories.

- **AWS Prescriptive Guidance**: Provides time-tested strategies, guides, and patterns for cloud migration and modernization.
- **AWS Knowledge Center**: A repository of the most frequent “how-to” questions and technical issues encountered by customers.
- **AWS re:Post**: A community-driven, cloud-native Q&A service that replaces the legacy AWS Forums.

AWS Support Plans

AWS offers four primary paid support plans to meet different business needs. All customers receive **Basic Support** for free, which includes access to documentation, whitepapers, and support for billing/account inquiries.

Feature	Developer	Business	Enterprise On-Ramp	Enterprise
Use Case	Experimenting/Testing	Production workloads	Business-critical apps	Mission-critical apps
Tech Support	Email (Business hours)	24/7 Phone, Email, Chat	24/7 Phone, Email, Chat	24/7 Phone, Email, Chat
Response Time	< 24 hrs (General)	< 1 hr (Production down)	< 30 min (Critical)	< 15 min (Critical)
TAM	No	No	Pool of TAMs	**Dedicated TAM**
Concierge	No	No	Support Account Manager	**Support Concierge**

Monitoring and Optimization Tools

- **AWS Trusted Advisor**: An online tool that provides real-time guidance to help you provision resources following AWS best practices. It focuses on five categories: **Cost Optimization**, **Performance**, **Security**, **Fault Tolerance**, and **Service Limits**.
- **AWS Health Dashboard**: Provides alerts and remediation guidance when AWS is experiencing events that may impact you. It includes the **Service Health** (status of all services) and **Your Account Health** (events specific to your resources).
- **AWS Health API**: Allows Enterprise and Enterprise On-Ramp customers to programmatically integrate health data into their own management tools.
- **AWS Trust and Safety Team**: The specific team to contact if you suspect AWS resources are being used for abusive purposes (e.g., spam, port scanning, or hosting illegal content).

AWS Partners and Marketplace

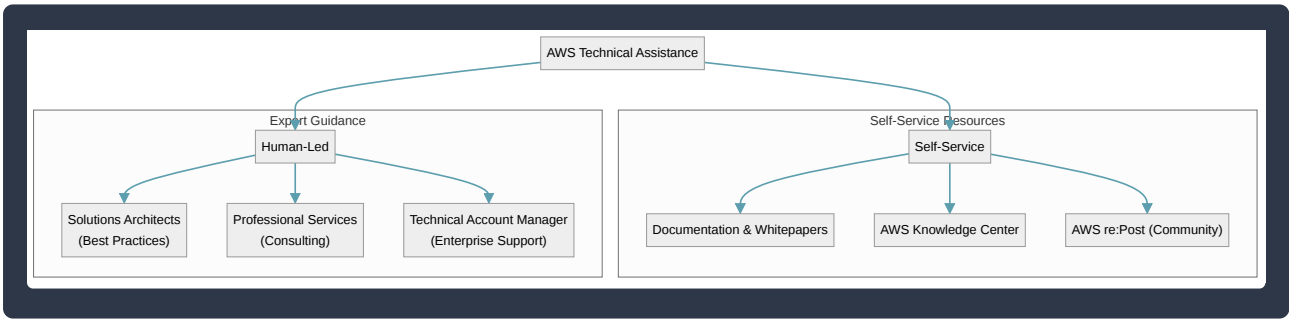
The **AWS Partner Network (APN)** consists of professional firms that help customers design, build, and manage their workloads.

- **Independent Software Vendors (ISVs)**: Partners that provide software that runs on or is integrated with AWS.
- **System Integrators (SIs)**: Partners that offer consulting and management services.
- **AWS Marketplace**: A digital catalog where customers can find, buy, and deploy third-party software. It simplifies **governance and entitlement** by centralizing software licensing and billing.
- **Partner Benefits**: Partners receive specialized training, certifications, access to partner-only events, and volume discounts to help grow their business.

Technical Assistance Options

When self-service resources are not enough, AWS provides expert human assistance:

- **AWS Solutions Architects (SAs)**: Technical experts who provide guidance on architectural best practices and help design cloud solutions.
- **AWS Professional Services (ProServe)**: A global team of experts that provides paid, project-based consulting to help customers achieve specific business outcomes.



In-Scope Services

In-Scope Services

Amazon Athena

Definition: This serverless, interactive query service allows users to analyze data stored in Amazon S3 using standard SQL. It requires no infrastructure management, as users simply point to their data, define a schema, and start querying while paying only for the data scanned.

Key Use Cases:

- Performing ad-hoc analysis on log files, such as VPC Flow Logs or AWS CloudTrail logs, stored in S3 buckets.
- Running one-off SQL queries against large datasets without the need to load them into a traditional relational database.
- Generating business reports and insights directly from unstructured or semi-structured data in a data lake.
- Integrating with Amazon QuickSight to create visualizations and dashboards based on raw data stored in S3.

Amazon EMR (Elastic MapReduce)

Definition: This managed cluster platform simplifies running big data frameworks, such as Apache Spark and Hadoop, to process and analyze vast amounts of data. It automates time-consuming tasks like capacity provisioning and cluster tuning, allowing users to focus on data analysis rather than infrastructure management.

Key Use Cases:

- Processing large-scale data sets using distributed computing frameworks.
- Performing extract, transform, and load (ETL) operations on massive data volumes.
- Running real-time streaming analytics from sources like Amazon Kinesis.
- Executing complex machine learning algorithms and genomic research simulations.
- Analyzing log files and clickstream data to gain business insights.

AWS Glue

Definition: This serverless data integration service simplifies the process of discovering, preparing, and combining data for analytics, machine learning, and application development. It automates the heavy lifting of extracting data from various sources, transforming it into a consistent format, and loading it into target data stores.

Key Use Cases:

- Automating ETL (Extract, Transform, and Load) pipelines to move data between sources like Amazon S3 and Amazon Redshift.
- Creating a centralized Data Catalog to store metadata about data assets across different AWS services for easy searching and querying.
- Using crawlers to automatically discover data schemas and update the metadata repository.
- Cleaning and normalizing disparate data sets to ensure high-quality input for business intelligence tools.
- Reducing operational overhead by managing data integration tasks without the need to provision or manage servers.

Amazon Kinesis

Definition: This managed service enables the collection, processing, and analysis of real-time, streaming data so you can respond instantly to new information. It handles high-volume data from thousands of sources, allowing for low-latency processing of information as it arrives.

Key Use Cases:

- Capturing and analyzing website clickstream data to understand user behavior in real-time.
- Ingesting and processing log files from application servers to monitor system health and security.
- Collecting telemetry data from IoT devices for immediate processing and storage.
- Feeding live data streams into Amazon S3, Redshift, or OpenSearch for downstream analytics and long-term storage.
- Processing high-frequency financial transactions or social media feeds for immediate sentiment analysis.

Amazon OpenSearch Service

Definition: This managed service simplifies the deployment, operation, and scaling of open-source search and analytics engines. It enables the ingestion, securing, and visualization of data in real-time for large-scale datasets.

Key Use Cases:

- Analyzing application and infrastructure logs to identify performance bottlenecks or errors.
- Implementing full-text search capabilities for e-commerce websites or document repositories.
- Monitoring security events in real-time to detect potential threats or unauthorized access.
- Visualizing complex data patterns and trends using integrated tools like OpenSearch Dashboards.

Amazon QuickSight

Definition: This is a fully managed, cloud-scale business intelligence (BI) service that enables the creation and publication of interactive dashboards. It provides machine learning-powered insights and scales automatically to support thousands of users without requiring manual infrastructure management.

Key Use Cases:

- Visualizing data from various AWS sources such as Amazon S3, Amazon RDS, and Amazon Redshift.
- Creating and sharing interactive dashboards with stakeholders to facilitate data-driven decision-making.
- Using natural language queries to ask questions about business data and receive instant visual responses.
- Identifying trends, outliers, and performing predictive forecasting using built-in machine learning capabilities.
- Embedding analytics and visualizations directly into third-party applications or portals.

Amazon Redshift

Definition: A fully managed, petabyte-scale data warehouse service designed for high-performance analysis of large datasets using standard SQL. It utilizes columnar storage and massively parallel processing to execute complex queries significantly faster than traditional relational databases.

Key Use Cases:

- Building enterprise data warehouses to centralize information from various business applications for unified reporting.
- Performing Online Analytical Processing (OLAP) to identify long-term trends and patterns in historical data.
- Connecting to Business Intelligence (BI) tools like Amazon QuickSight to generate complex visual dashboards.
- Analyzing structured and semi-structured data across a data lake without requiring manual data movement.
- Handling massive datasets where traditional operational databases (OLTP) would experience performance degradation.

Amazon EventBridge

Definition: A serverless event bus service that facilitates the connection between applications using data from a variety of sources. It delivers a stream of real-time data from your own applications, integrated SaaS applications, and AWS services to specific targets like AWS Lambda or Amazon SNS.

Key Use Cases:

- Building event-driven architectures to decouple microservices and improve scalability.
- Automating AWS service responses based on resource state changes, such as triggering a workflow when an S3 object is uploaded.
- Integrating third-party SaaS applications, like Zendesk or Shopify, directly into AWS environments.
- Scheduling recurring tasks or automated maintenance jobs using cron-like expressions.
- Routing events to multiple subscribers simultaneously to ensure data consistency across different systems.

Amazon Simple Notification Service (Amazon SNS)

Definition: This is a fully managed pub/sub messaging service that enables high-throughput, push-based communication between distributed systems and microservices. It allows for the decoupling of message producers from consumers, ensuring that messages are delivered reliably to multiple subscribers simultaneously.

Key Use Cases:

- Implementing a “fan-out” architecture where a single event triggers multiple downstream processes like AWS Lambda functions or SQS queues.
- Sending automated notifications to end-users via SMS, email, or mobile push notifications.
- Monitoring application health by alerting administrators when CloudWatch Alarms are triggered.
- Decoupling microservices to improve system scalability and fault tolerance by allowing components to operate independently.

Amazon Simple Queue Service (Amazon SQS)

Definition: This fully managed message queuing service enables the decoupling and scaling of microservices, distributed systems, and serverless applications. It allows components to communicate asynchronously by sending, storing, and receiving messages without requiring other services to be available or connected simultaneously.

Key Use Cases:

- Decoupling application components to ensure that a failure in one part of the system does not cause the entire application to crash.
- Buffering high-volume traffic spikes to prevent downstream services from becoming overwhelmed by sudden increases in requests.
- Managing asynchronous task processing where immediate responses are not required, such as background image processing or email delivery.
- Implementing First-In-First-Out (FIFO) logic to ensure messages are processed exactly once and in the precise order they were sent.

- Offloading heavy processing tasks from a web server to background worker instances to improve user interface responsiveness.

AWS Step Functions

Definition: This serverless orchestration service allows for the coordination of multiple AWS services into visual workflows. It manages the state, checkpoints, and restarts of complex business logic to ensure tasks are executed in order and as expected.

Key Use Cases:

- Orchestrating microservices by managing the flow of data between independent components.
- Automating Extract, Transform, and Load (ETL) processes to handle large-scale data processing.
- Implementing error handling and retry logic for distributed applications without writing custom code.
- Creating human-in-the-loop workflows that require manual approval before proceeding to the next step.
- Sequencing AWS Lambda functions to build complex, multi-step serverless applications.

Amazon Connect

Definition: This is an omnichannel, cloud-based contact center service that enables businesses to provide customer support through voice and chat. It offers a pay-as-you-go model and scales automatically to handle any volume of customer interactions without requiring physical infrastructure or complex hardware.

Key Use Cases:

- Building automated customer service workflows using interactive voice response (IVR) and AI-powered chatbots.
- Scaling support operations rapidly to accommodate seasonal spikes or unexpected surges in call and chat volume.
- Integrating with third-party CRM systems to provide agents with immediate access to customer history and data during live interactions.
- Leveraging built-in machine learning to perform real-time sentiment analysis and call transcriptions for quality assurance.

Amazon Simple Email Service (Amazon SES)

Definition: This cloud-based service provides a scalable and cost-effective platform for sending and receiving digital communications via email. It allows developers to integrate email functionality directly into their applications using an SMTP interface or AWS SDKs without maintaining their own mail servers.

Key Use Cases:

- Sending automated transactional messages such as order confirmations, shipping notifications, and password resets.
- Executing high-volume marketing campaigns and newsletters to reach a large customer base efficiently.
- Receiving incoming emails to trigger automated business processes or store content directly in Amazon S3.
- Monitoring email deliverability, bounce rates, and sender reputation through built-in analytics and dashboard tools.

AWS Budgets

Definition: This service allows users to set custom cost and usage thresholds that trigger notifications when actual or forecasted amounts exceed defined limits. It provides a proactive way to monitor cloud spend and resource consumption across an entire account or organization.

Key Use Cases:

- Receiving email or SNS alerts when monthly spending exceeds a specific dollar amount.
- Monitoring data transfer usage to ensure it stays within a free tier or allocated limit.
- Tracking Reserved Instance (RI) or Savings Plans utilization and coverage to ensure maximum return on investment.
- Automating specific actions, such as applying an IAM policy or stopping an instance, when a budget threshold is breached.
- Forecasting estimated costs for the remainder of the billing period based on current usage patterns.

AWS Cost and Usage Reports (CUR)

Definition: This tool provides the most comprehensive set of cost and usage data available, delivering granular metadata about services, pricing, and reservations to an Amazon S3 bucket. It allows for deep-dive analysis of spending by breaking down costs by the hour, day, or month, as well as by product or resource tags.

Key Use Cases:

- Analyzing detailed billing data using business intelligence tools like Amazon QuickSight or Amazon Athena.
- Tracking amortized costs and Reserved Instance (RI) utilization across an entire organization.
- Generating custom billing reports that include specific resource IDs and cost allocation tags for internal chargebacks.
- Exporting raw billing data to third-party applications for financial auditing and long-term compliance storage.

AWS Cost Explorer

Definition: This interface allows users to visualize, understand, and manage cloud spending and usage patterns over time. It provides high-level overviews or granular data to identify trends, cost drivers, and anomalies in billing.

Key Use Cases:

- Visualizing historical spending data for the past 12 months to identify which services consume the most budget.
- Forecasting future costs for the next 12 months based on current usage patterns to assist with financial planning.
- Filtering and grouping costs by specific attributes such as AWS Region, Availability Zone, or custom resource tags.
- Analyzing Reserved Instance (RI) and Savings Plans utilization and coverage to optimize commitment-based discounts.
- Identifying underutilized resources to improve cost-efficiency across the entire organization.

AWS Marketplace

Definition: This digital catalog allows customers to find, test, buy, and deploy software that runs on Amazon Web Services. It simplifies procurement by providing a curated selection of thousands of software listings from independent vendors, often featuring simplified licensing and consolidated billing through an existing AWS account.

Key Use Cases:

- Deploying pre-configured Amazon Machine Images (AMIs) for specialized security appliances, firewalls, or networking tools.
- Purchasing SaaS subscriptions and managing their costs directly through the AWS Billing console to avoid multiple vendor invoices.
- Accessing curated data sets for machine learning, research, and business intelligence projects.
- Utilizing professional services for consulting, implementation, or managed services from certified third-party partners.
- Implementing “Bring Your Own License” (BYOL) models to migrate existing software investments to the cloud environment.

AWS Batch

Definition: This fully managed service enables developers and data scientists to run hundreds of thousands of batch computing jobs efficiently. It automatically provisions the optimal quantity and type of compute resources based on the specific volume and requirements of the submitted tasks.

Key Use Cases:

- Processing large-scale datasets for financial services, such as risk analysis or fraud detection.

- Automating high-performance computing (HPC) workloads for scientific research and genomic sequencing.
- Managing Extract, Transform, Load (ETL) operations to prepare data for analytics or machine learning.
- Executing media transcoding tasks to convert video files into multiple formats simultaneously.
- Running periodic background tasks that require significant compute power but do not need to be processed in real-time.

Amazon EC2 (Elastic Compute Cloud)

Definition: This service provides resizable virtual servers in the cloud, allowing users to scale capacity up or down as computing requirements change. It offers complete control over the operating system and software stack, operating on a pay-as-you-go pricing model.

Key Use Cases:

- Hosting web applications and websites that require full administrative access to the underlying server.
- Running high-performance computing (HPC) workloads or data processing tasks that need specific CPU or GPU configurations.
- Developing and testing software in environments that mirror on-premises server setups.
- Deploying enterprise applications such as SAP, Oracle, or Microsoft SQL Server.
- Scaling compute resources automatically to meet fluctuating traffic demands through integration with Auto Scaling.

AWS Elastic Beanstalk

Definition: This Platform as a Service (PaaS) offering automates the deployment, management, and scaling of web applications and services. It handles the underlying infrastructure details—including capacity provisioning, load balancing, and health monitoring—while allowing users to retain full control over the AWS resources powering the application.

Key Use Cases:

- Rapidly deploying web applications written in popular languages like Java, Python, or Node.js without manually configuring servers.
- Automatically scaling application resources up or down based on traffic demands to maintain performance and optimize costs.
- Simplifying the management of environment configurations, such as integrated databases and load balancers, through a single interface.
- Running Docker containers in a managed environment that streamlines the container orchestration process for developers.

Amazon Lightsail

Definition: This service provides an easy-to-use virtual private server (VPS) platform that bundles compute, storage, and networking resources into a single, predictable monthly price. It is designed for users who need a simplified experience to launch and manage applications without the complexity of manually configuring individual AWS services.

Key Use Cases:

- Launching simple web applications or blogs using pre-configured stacks like WordPress, Magento, or LAMP.
- Hosting small-scale business software or development and testing environments.
- Deploying low-traffic websites that require a fixed, low-cost budget for easier financial planning.
- Managing simple databases and object storage through a streamlined, intuitive console interface.
- Creating quick proof-of-concept projects that can later be migrated to more complex AWS services if needed.

AWS Outposts

Definition: This service provides a hybrid cloud solution by delivering AWS-managed hardware and infrastructure to on-premises data centers or co-location spaces. It allows organizations to run native AWS services locally while maintaining a consistent connection to the nearest AWS Region for management and operations.

Key Use Cases:

- Applications requiring ultra-low latency to local end-users, factory equipment, or on-site medical imaging systems.
- Workloads that must remain on-premises due to strict data residency, sovereignty, or local regulatory requirements.
- Local data processing for high-volume datasets that are too expensive or time-consuming to migrate to the cloud.
- Modernizing legacy applications that need to stay physically close to on-premises databases or mainframe systems.
- Maintaining consistent developer tools and APIs across both cloud and on-premises environments.

Amazon Elastic Container Registry (Amazon ECR)

Definition: This managed service provides a secure, scalable location to store and manage container images. It eliminates the need to operate your own container repositories or worry about scaling the underlying infrastructure.

Key Use Cases:

- Storing Docker images for use with Amazon ECS, Amazon EKS, or AWS Lambda.

- Automating the software deployment workflow by integrating with CI/CD pipelines like AWS CodePipeline.
- Scanning container images for vulnerabilities to ensure security compliance before deployment.
- Sharing container software publicly or privately within an organization using fine-grained IAM permissions.
- Reducing latency by hosting images in the same AWS Region as your compute resources.

Amazon Elastic Container Service (Amazon ECS)

Definition: A fully managed container orchestration service designed to run, stop, and manage Docker containers on a cluster. It provides deep integration with the AWS ecosystem to offer a secure and scalable environment for containerized applications without requiring the user to manage complex orchestration software.

Key Use Cases:

- Deploying microservices architectures where individual components are isolated in containers for independent scaling.
- Running batch processing jobs that require rapid scaling of compute resources to handle large data volumes.
- Migrating legacy on-premises applications to the cloud by packaging them into portable container images.
- Managing hybrid cloud deployments by running containers across both local and AWS environments using consistent tooling.
- Integrating with AWS Fargate to run containers in a serverless environment, removing the need to manage underlying EC2 instances.

Amazon EKS (Amazon Elastic Kubernetes Service)

Definition: This managed service automates the deployment, management, and scaling of containerized applications using Kubernetes. It handles the operational complexity of maintaining the control plane and underlying infrastructure, ensuring high availability and security across multiple Availability Zones.

Key Use Cases:

- Running complex microservices architectures that require robust container orchestration and automated scaling.
- Migrating existing on-premises Kubernetes workloads to the AWS Cloud with minimal configuration changes.
- Managing hybrid cloud environments where applications must run consistently across local data centers and the cloud.
- Integrating containerized applications with native AWS security, networking, and monitoring services.

AWS Support

Definition: This offering provides a range of tiered plans that provide technical assistance, architectural guidance, and operational support to help customers manage their cloud infrastructure effectively. It includes access to documentation, community forums, and specialized experts depending on the selected tier.

Key Use Cases:

- Resolving technical issues or service outages through direct communication with cloud engineers.
- Optimizing infrastructure costs and performance using recommendations from the Trusted Advisor tool.
- Receiving proactive architectural guidance and reviews to ensure alignment with the Well-Architected Framework.
- Managing billing and account inquiries through a dedicated Concierge team for higher-tier plans.
- Gaining strategic business value and operational excellence via a Technical Account Manager (TAM) for enterprise-level accounts.

Amazon Aurora

Definition: This is a fully managed relational database engine that is compatible with MySQL and PostgreSQL. It provides the performance and availability of commercial-grade databases at a fraction of the cost by utilizing a distributed, fault-tolerant, and self-healing storage system.

Key Use Cases:

- Migrating existing MySQL or PostgreSQL workloads to a more scalable, cloud-native environment.
- Applications requiring high availability and automatic failover across multiple Availability Zones.
- Workloads with unpredictable traffic that benefit from the Serverless configuration to scale capacity up or down automatically.
- Enterprise-level applications that need high-speed performance and data durability with 6-way replication.

Amazon DocumentDB (with MongoDB compatibility)

Definition: This fully managed NoSQL database service is designed to store, query, and index JSON data at scale. It provides the durability and high availability required for mission-critical workloads by decoupling storage and compute layers.

Key Use Cases:

- Migrating existing MongoDB workloads to a managed AWS environment without changing application code.

- Managing content for web and mobile applications, such as user profiles, comments, and product catalogs.
- Building real-time big data applications that require high-throughput document processing.
- Storing semi-structured data that evolves frequently and requires a flexible schema.
- Scaling read-heavy workloads using up to 15 low-latency read replicas.

Amazon DynamoDB (NoSQL Database)

Definition: This is a fully managed, serverless NoSQL database service that provides fast, predictable performance with seamless scalability. It supports both key-value and document data structures, automatically handling hardware provisioning, setup, and configuration to ensure high availability.

Key Use Cases:

- Building mobile, web, and IoT applications that require single-digit millisecond latency at any scale.
- Storing session data for high-traffic websites to ensure consistent and reliable user experiences.
- Managing metadata for media files, digital assets, or large-scale product catalogs.
- Implementing serverless architectures by using event-driven triggers to initiate AWS Lambda functions.
- Developing global applications that require multi-region, multi-active data replication through Global Tables.

Amazon ElastiCache

Definition: This fully managed, in-memory data store and cache service provides sub-millisecond latency for high-performance applications. It automates common administrative tasks while supporting popular open-source engines like Redis and Memcached to improve the speed and scalability of web applications.

Key Use Cases:

- Offloading read-heavy workloads from relational databases to reduce latency and operational costs.
- Managing session state for web applications to ensure a seamless user experience across distributed server environments.
- Powering real-time applications such as gaming leaderboards, social media feeds, and live streaming analytics.
- Caching frequently accessed data, such as product catalogs or user profiles, to minimize direct queries to slower disk-based storage systems.

Amazon Neptune

Definition: A fully managed graph database service optimized for storing and navigating highly connected datasets with millisecond latency. It supports popular graph models like Property Graph and W3C's RDF, enabling the efficient processing of complex relationships that are difficult to manage in traditional relational databases.

Key Use Cases:

- Building social media feeds by mapping connections between users, followers, and shared content.
- Detecting fraud in real-time by identifying suspicious patterns and relationships between financial accounts or identities.
- Powering recommendation engines that suggest products or services based on complex user-item interactions.
- Creating knowledge graphs to link and query disparate data sources for research or enterprise data discovery.
- Mapping network security topologies to visualize and analyze dependencies and vulnerabilities within IT infrastructures.

Amazon RDS (Relational Database Service)

Definition: This managed service automates time-consuming administration tasks such as hardware provisioning, database setup, patching, and backups for relational engines. It allows users to focus on application development rather than the operational complexities of maintaining a database server.

Key Use Cases:

- Hosting traditional web applications that require structured data storage and complex SQL queries.
- Implementing high availability and failover support through Multi-AZ deployments to ensure business continuity.
- Scaling read performance for globally distributed applications by utilizing Read Replicas.
- Migrating existing on-premises databases—such as MySQL, PostgreSQL, Oracle, or SQL Server—to a cloud environment with minimal configuration changes.

AWS CLI (Amazon Web Services Command Line Interface)

Definition: This unified tool allows users to interact with and manage various cloud services through commands in a terminal or command-line shell. It provides a way to automate resource management and configuration through scripts, reducing the need for manual interaction with a graphical user interface.

Key Use Cases:

- Automating repetitive administrative tasks and resource provisioning using shell scripts.

- Managing multiple cloud services from a single interface without navigating the web console.
- Integrating cloud management tasks into continuous integration and continuous delivery (CI/CD) pipelines.
- Executing bulk operations, such as uploading large volumes of data to storage buckets or updating multiple instances simultaneously.

AWS CodeBuild

Definition: This fully managed continuous integration service compiles source code, runs tests, and produces software packages that are ready to deploy. It scales automatically and eliminates the need to provision, manage, or scale your own build servers.

Key Use Cases:

- Compiling source code into executable artifacts or software packages.
- Running automated unit tests to validate code changes before they move to the next stage of the pipeline.
- Building and packaging Docker images to be stored in container registries like Amazon ECR.
- Serving as the build and test component within a CI/CD pipeline to automate the software release process.
- Customizing build environments using pre-configured or custom Docker images to meet specific language or tool requirements.

AWS CodePipeline

Definition: A fully managed continuous delivery service that automates the release pipelines for fast and reliable application and infrastructure updates. It orchestrates the workflow of building, testing, and deploying code every time there is a change, based on defined release process models.

Key Use Cases:

- Automating the end-to-end software release process from source code to production environments.
- Integrating with other AWS developer tools like CodeCommit, CodeBuild, and CodeDeploy to create a unified workflow.
- Implementing rapid delivery of new features and bug fixes through continuous integration and continuous delivery (CI/CD) practices.
- Enforcing consistent quality gates by requiring manual approvals or automated tests before code moves to the next stage of the pipeline.
- Monitoring the status of software releases in real-time to identify and resolve bottlenecks in the development lifecycle.

AWS X-Ray

Definition: This service provides a complete view of requests as they travel through an application, allowing for the analysis and debugging of distributed systems and microservices. It collects data about the various components and services involved in a request to help visualize performance and identify the root cause of errors.

Key Use Cases:

- Visualizing application topology and service dependencies through an interactive service map.
- Identifying performance bottlenecks and high-latency areas within a specific request path.
- Troubleshooting errors and exceptions by pinpointing exactly where a failure occurred in a complex architecture.
- Analyzing the impact of downstream service performance on the overall end-user experience.
- Monitoring the health of microservices by tracking request success rates and response times.

Amazon AppStream 2.0

Definition: This fully managed service provides non-persistent desktop applications to users via a web browser, eliminating the need for local installations or high-end hardware. It streams applications from the AWS Cloud to any device, ensuring a consistent user experience regardless of the underlying operating system.

Key Use Cases:

- Delivering resource-intensive software like CAD, 3D modeling, or video editing to low-powered devices.
- Providing students with access to specific educational software for remote learning environments without requiring physical computer labs.
- Simplifying application management by centralizing updates and patches in the cloud rather than on individual endpoints.
- Offering instant-on software trials or demonstrations to potential customers without requiring complex local setups or downloads.
- Securing sensitive data by keeping application data on AWS infrastructure rather than on local user devices.

Amazon WorkSpaces

Definition: This managed, secure Desktop-as-a-Service (DaaS) solution provides users with persistent virtual cloud desktops accessible from any supported device. It eliminates the need to manage hardware inventory or complex on-premises Virtual Desktop Infrastructure (VDI) by hosting the operating system and applications entirely in the AWS Cloud.

Key Use Cases:

- Providing remote employees or contractors with secure access to corporate applications without shipping physical laptops.

- Standardizing desktop environments across a global workforce to ensure consistent security patches and software configurations.
- Scaling desktop resources up or down quickly to accommodate seasonal staffing or temporary project-based requirements.
- Enhancing data security by keeping sensitive information on AWS servers rather than storing it on vulnerable local end-user devices.
- Reducing capital expenditure by moving from purchasing hardware to a pay-as-you-go monthly or hourly subscription model.

Amazon WorkSpaces Secure Browser

Definition: A managed, enterprise-grade service that provides secure access to internal websites and public SaaS applications through a cloud-hosted web browser. It streams pixels to the user's local machine, ensuring that sensitive data never leaves the AWS Cloud or resides on the end-user device.

Key Use Cases:

- Providing remote employees or contractors with secure access to internal corporate web portals without requiring a VPN.
- Protecting the corporate network from web-based threats by isolating browsing activity in a disposable cloud container.
- Reducing data leakage risks by preventing users from downloading, copying, or printing sensitive information from web applications.
- Simplifying IT management by providing a low-cost alternative to full virtual desktops for users who only need web access.
- Ensuring compliance by centralizing browser security policies and monitoring web traffic within a controlled environment.

AWS Amplify

Definition: This is a comprehensive set of tools and services designed to help developers build, deploy, and host scalable full-stack web and mobile applications. It simplifies the process of connecting frontend code to backend cloud resources like authentication, storage, and databases through a unified framework.

Key Use Cases:

- Rapidly developing and deploying mobile or web applications with integrated backend services.
- Implementing secure user authentication and authorization using pre-built UI components.
- Hosting static websites or single-page applications (SPAs) with built-in CI/CD workflows and global content delivery.
- Managing application data and connecting to cloud-based APIs using GraphQL or REST.

- Storing and managing user-generated content, such as photos or videos, in a scalable cloud environment.

AWS AppSync

Definition: This managed service simplifies application development by providing a serverless GraphQL interface to securely query data from multiple sources. It enables developers to build applications that interact with data through a single endpoint, handling the heavy lifting of data fetching, manipulation, and real-time updates.

Key Use Cases:

- Building real-time collaborative applications, such as chat rooms or shared dashboards, where data must sync instantly across multiple users.
- Creating mobile or web applications that require offline data access and automatic synchronization once internet connectivity is restored.
- Aggregating data from various backend sources, including Amazon DynamoDB, AWS Lambda, or HTTP APIs, into a unified and efficient API.
- Implementing fine-grained access control and security for data-driven applications using AWS IAM, Amazon Cognito, or API keys.

AWS IoT Core (Internet of Things Core)

Definition: This managed cloud service enables connected devices to interact securely with cloud applications and other devices. It can support billions of devices and trillions of messages, processing and routing those messages to AWS endpoints and other devices reliably and securely.

Key Use Cases:

- Connecting and managing a fleet of smart home appliances to collect telemetry data and provide remote control capabilities.
- Implementing industrial automation by monitoring sensor data from manufacturing equipment in real-time to predict maintenance needs.
- Building asset tracking solutions to monitor the location, temperature, and health of shipping containers or commercial vehicles.
- Creating a secure communication channel between low-power edge devices and backend data processing services like Amazon S3, Amazon DynamoDB, or AWS Lambda.

Amazon Comprehend

Definition: This managed natural language processing (NLP) service uses machine learning to uncover valuable insights and connections within unstructured text. It allows users to process documents, social media feeds, or support tickets without requiring deep data science expertise.

Key Use Cases:

- Performing sentiment analysis to determine if customer feedback is positive, negative, or neutral.

- Identifying personally identifiable information (PII) within documents to ensure data privacy and regulatory compliance.
- Extracting key phrases, entities (such as people, places, or brands), and topics from large volumes of text.
- Automatically categorizing support tickets or news articles by theme to streamline routing and organization.
- Detecting the dominant language used in a specific document or dataset to facilitate global communication.

Amazon Kendra

Definition: This is an intelligent search service powered by machine learning that enables users to find information across various content repositories using natural language queries. It provides highly accurate answers by understanding the context and intent behind a user's question rather than just matching keywords.

Key Use Cases:

- Creating a centralized internal search engine for employees to find documents across SharePoint, S3, and ServiceNow.
- Enhancing customer support websites by allowing users to ask questions in plain English and receive direct answers from technical manuals.
- Improving research efficiency by indexing large volumes of unstructured data from multiple disparate sources.
- Implementing a "frequently asked questions" (FAQ) bot that retrieves specific answers from existing knowledge bases.

Amazon Lex

Definition: This fully managed service provides advanced deep learning functionalities of automatic speech recognition and natural language understanding to build conversational interfaces into applications. It enables developers to create sophisticated chatbots and virtual assistants that can process both voice and text inputs.

Key Use Cases:

- Building automated customer service chatbots to handle common inquiries and support tickets.
- Creating voice-activated virtual assistants for mobile devices or IoT hardware.
- Developing informational bots for tasks like checking weather, news, or account balances.
- Automating enterprise workflows such as booking appointments or processing employee leave requests.
- Integrating conversational AI into contact centers to provide self-service options for callers.

Amazon Polly

Definition: This managed service uses advanced deep learning technologies to synthesize natural-sounding human speech from written text. It allows developers to create applications that talk, supporting dozens of languages and a wide variety of lifelike voices in both standard and neural formats.

Key Use Cases:

- Enhancing accessibility by providing audio versions of text-based content for visually impaired users or those with reading disabilities.
- Powering automated voice response systems (IVR) for contact centers to provide a more natural and engaging customer experience.
- Creating narrated educational materials or e-learning modules to improve information retention through multi-modal learning.
- Enabling mobile applications to read news articles, blog posts, or messages aloud for users who are driving or multitasking.
- Generating high-quality voiceovers for videos and presentations without the need for professional recording equipment or voice actors.

Amazon Q

Definition: A generative artificial intelligence (AI) powered assistant designed to help users solve problems, generate content, and gain insights by interacting with data across an organization's systems and AWS resources. It provides tailored assistance based on the specific context of the user's business and technical environment while maintaining high standards for security and privacy.

Key Use Cases:

- Answering technical questions about AWS services, best practices, and architectural patterns.
- Analyzing business data to identify trends, generate reports, and provide data-driven insights.
- Assisting developers with code generation, debugging, and application optimization within the IDE.
- Summarizing long documents or meeting transcripts to improve employee productivity.
- Integrating with third-party enterprise applications to streamline workflows and automate routine tasks.

Amazon Rekognition

Definition: This fully managed machine learning service automates image and video analysis using deep learning technology. It allows developers to add computer vision capabilities to applications without requiring expertise in machine learning models or infrastructure.

Key Use Cases:

- Identifying objects, people, text, scenes, and activities within image or video files.

- Performing facial analysis to detect attributes like gender, age range, and emotions.
- Implementing automated content moderation to identify inappropriate, unsafe, or offensive material.
- Verifying user identities by comparing live photos against images on file for security and access control.
- Recognizing well-known celebrities in media libraries for automated metadata tagging and search.

Amazon SageMaker AI (Artificial Intelligence)

Definition: This fully managed service provides a comprehensive suite of tools to build, train, and deploy machine learning models at scale. It removes the heavy lifting from each step of the machine learning process, allowing developers to create high-quality models without managing the underlying infrastructure.

Key Use Cases:

- Building models using integrated development environments like Jupyter Notebooks.
- Training machine learning models efficiently using high-performance AWS compute infrastructure.
- Deploying trained models into production environments with one-click hosting and scaling.
- Automating the machine learning workflow through built-in algorithms and automated model tuning.
- Monitoring model performance over time to ensure accuracy and detect data drift.

Amazon Textract

Definition: This machine learning service automatically extracts text, handwriting, and structured data from scanned documents, PDFs, and images. It goes beyond simple optical character recognition (OCR) by identifying the relationship between data points in forms and tables to maintain the original context.

Key Use Cases:

- Automating document processing workflows for high-volume paperwork like invoices, receipts, and medical records.
- Extracting structured data from complex tables without requiring manual configuration or custom code.
- Processing identity documents, such as passports and driver’s licenses, for automated verification and onboarding.
- Converting physical archives into searchable digital databases to improve information accessibility and compliance.
- Integrating extracted data into downstream applications or databases for further analysis and reporting.

Amazon Transcribe

Definition: This fully managed machine learning service utilizes automatic speech recognition (ASR) to convert spoken language into accurate text. It enables the processing of both real-time audio streams and batch files, providing features like speaker identification, timestamping, and multi-language support.

Key Use Cases:

- Generating closed captions and subtitles for video content to enhance accessibility and global reach.
- Transcribing customer service call recordings to perform sentiment analysis and quality monitoring.
- Creating searchable text archives from recorded meetings, interviews, or legal proceedings.
- Automatically redacting sensitive personally identifiable information (PII) from transcripts to ensure data privacy and compliance.
- Improving documentation efficiency by allowing users to dictate notes that are converted into digital text.

Amazon Translate

Definition: This neural machine translation service uses deep learning models to deliver fast, high-quality, and affordable language translation between supported languages. It enables the localization of content for international users while maintaining a natural-sounding flow across diverse document formats and communication channels.

Key Use Cases:

- Localizing website and application content to reach a global audience in their native languages.
- Enabling real-time communication in chat and messaging platforms by translating user-generated text instantly.
- Processing large volumes of unstructured text data or documents for cross-lingual analysis and discovery.
- Automating the translation of customer support tickets to provide efficient service across different geographic regions.
- Integrating with other AWS services to create automated workflows, such as translating transcribed audio from Amazon Transcribe.

AWS Auto Scaling

Definition: This service provides a unified interface to automatically adjust the capacity of multiple resources to maintain steady performance at the lowest possible cost. It monitors applications and scales resources up or down based on predefined conditions or real-time demand.

Key Use Cases:

- Managing capacity for Amazon EC2 instances, ECS tasks, and DynamoDB tables from a single location.
- Maintaining application availability by ensuring the correct number of resources are running to handle current traffic.
- Optimizing costs by automatically terminating underutilized resources during periods of low demand.
- Responding to unpredictable traffic spikes by quickly provisioning additional capacity to prevent performance degradation.
- Implementing predictive scaling to forecast future traffic and schedule capacity changes in advance.

AWS CloudFormation

Definition: This service allows you to model, provision, and manage AWS and third-party resources by treating infrastructure as code. It uses declarative templates formatted in JSON or YAML to automatically create and configure a collection of resources in a repeatable and predictable manner.

Key Use Cases:

- Automating the deployment of entire technology stacks to ensure consistency across development, test, and production environments.
- Managing a group of related resources as a single unit, known as a “stack,” to simplify updates and deletions.
- Replicating infrastructure across multiple AWS Regions quickly and reliably without manual configuration.
- Implementing version control for infrastructure by storing templates in a repository, allowing for easy auditing and rollbacks.

AWS CloudTrail

Definition: This service enables governance, compliance, operational auditing, and risk auditing of an AWS account by logging every action taken by a user, role, or service. It provides a detailed history of API calls made via the Management Console, SDKs, and command line tools.

Key Use Cases:

- Compliance auditing to meet internal policies and regulatory standards by providing a complete history of account activity.
- Security analysis to detect unauthorized access or unusual behavior by identifying which user or role performed a specific action.
- Operational troubleshooting to identify recent changes in the environment that may have caused performance issues or outages.

- Resource lifecycle tracking to monitor when resources were created, modified, or deleted across the infrastructure.

Amazon CloudWatch

Definition: This monitoring and observability service collects data in the form of logs, metrics, and events to provide a unified view of AWS resources, applications, and services. It enables real-time monitoring of performance, resource utilization, and operational health across the entire infrastructure.

Key Use Cases:

- Monitoring infrastructure metrics like CPU utilization, disk I/O, and network traffic for EC2 instances.
- Setting up billing alarms to receive notifications when estimated AWS charges exceed a specific dollar threshold.
- Centralizing and analyzing log files from various sources, such as AWS Lambda or Amazon EC2, to troubleshoot application issues.
- Creating visual dashboards to track the health and performance of multiple services in a single, customizable view.
- Triggering automated actions, such as scaling resources or stopping instances, based on predefined metric thresholds.

AWS Compute Optimizer

Definition: This service leverages machine learning to analyze historical utilization metrics and provide actionable recommendations for right-sizing resources. It helps ensure workloads run on the most cost-effective and high-performing configurations by comparing current usage against hundreds of thousands of cloud workloads.

Key Use Cases:

- Identifying over-provisioned EC2 instances to reduce monthly cloud spend without impacting application performance.
- Detecting under-provisioned resources that require additional CPU or memory to eliminate performance bottlenecks.
- Optimizing Amazon EBS volume configurations by analyzing IOPS and throughput requirements.
- Evaluating AWS Lambda function memory settings to improve execution speed and lower operational costs.
- Reviewing Amazon ECS on AWS Fargate task configurations to ensure resource allocations match actual workload demands.

AWS Config

Definition: This service provides a detailed view of the configuration of resources in an AWS account, including how they were configured in the past and how they relate to one another. It continuously monitors and records resource changes to simplify compliance auditing, security analysis, and change management.

Key Use Cases:

- Tracking resource configuration changes over time to maintain a historical record for auditing purposes.
- Evaluating resource settings against internal guidelines or industry best practices using pre-defined or custom rules.
- Visualizing relationships between resources to understand how a change to one component might impact others.
- Automating the remediation of non-compliant resources to ensure the environment stays within a desired security posture.
- Simplifying troubleshooting by reviewing the state of a resource at a specific point in time.

AWS Control Tower

Definition: This service automates the setup of a secure, multi-account environment based on AWS best practices, establishing a governed “landing zone” in minutes. It provides centralized management for security, operations, and compliance across an entire organization by orchestrating other services like AWS Organizations and IAM Identity Center.

Key Use Cases:

- Provisioning new AWS accounts quickly using a standardized “Account Factory” to ensure consistent configurations and security baselines.
- Implementing mandatory or optional guardrails to prevent resource deployment that violates organizational security policies.
- Centralizing logging and auditing across multiple accounts to maintain a clear, immutable trail of activity for compliance requirements.
- Monitoring the real-time compliance status of an entire multi-account environment through a single, high-level visual dashboard.
- Automating the application of governance rules to existing accounts when they are moved into a managed organizational unit.

AWS Health Dashboard

Definition: This tool provides a comprehensive view into the performance and availability of AWS services, offering both a general status of all services and a personalized view of the specific resources used by an account. It serves as the central location for alerts regarding service interruptions, upcoming maintenance, and security notifications.

Key Use Cases:

- Monitoring the real-time operational status of global AWS services across all regions.
- Reviewing personalized alerts and notifications specifically affecting your active AWS resources.
- Planning for scheduled infrastructure changes or hardware maintenance that may impact running instances.
- Investigating historical service availability data to correlate past performance issues with AWS outages.
- Automating responses to service events using Amazon EventBridge to minimize downtime.

AWS License Manager

Definition: A management service that simplifies the process of tracking and controlling software licenses from third-party vendors across AWS and on-premises environments. It provides a centralized dashboard to ensure compliance with licensing agreements and helps prevent costly overages by automating usage limits.

Key Use Cases:

- Managing Bring Your Own License (BYOL) models for software such as Microsoft Windows Server, SQL Server, and Oracle database engines.
- Automating the discovery of managed and unmanaged software installed on EC2 instances or on-premises servers.
- Setting hard or soft limits to prevent developers from launching instances that exceed the available license count.
- Consolidating license tracking for multi-account environments through seamless integration with AWS Organizations.
- Generating detailed compliance reports for internal audits or vendor requests to demonstrate adherence to licensing terms.

AWS Management Console

Definition: This web-based graphical user interface (GUI) provides a centralized portal to access and manage nearly all aspects of an account and its associated cloud resources. It allows users to interact with services through a browser without needing to write code or use complex command-line tools.

Key Use Cases:

- Performing manual administrative tasks such as launching EC2 instances or creating S3 buckets through guided wizards.
- Monitoring account activity, viewing billing dashboards, and managing cost allocation tags.
- Searching for specific services, features, or documentation using the integrated global search bar.

- Accessing the AWS CloudShell directly from the browser for quick command-line interactions.
- Managing user permissions, security credentials, and multi-factor authentication (MFA) within Identity and Access Management (IAM).

AWS Organizations

Definition: This account management service allows for the central governance and consolidation of multiple AWS accounts into a single entity. It provides tools to automate account creation, group accounts into organizational units (OUs), and apply security policies across the entire environment.

Key Use Cases:

- Consolidating billing to receive a single invoice and take advantage of volume discounts across all linked accounts.
- Implementing Service Control Policies (SCPs) to centrally restrict specific AWS services or actions at the account or OU level.
- Automating the creation and management of new accounts for different departments, projects, or development stages.
- Simplifying the sharing of resources and centralizing logging and security monitoring across the entire enterprise.
- Organizing accounts into a hierarchy to mirror business structures and apply granular governance.

AWS Service Catalog

Definition: This management tool allows organizations to create, organize, and govern a curated list of approved IT services for deployment on AWS. It enables central administrators to ensure compliance and security standards while providing end-users with a self-service portal to quickly launch pre-configured resources.

Key Use Cases:

- Standardizing commonly deployed resources to ensure consistent configurations across different teams or departments.
- Implementing granular access control by restricting which users can view and launch specific products or versions.
- Managing the lifecycle of IT services by providing versioning and updates for existing product portfolios.
- Maintaining compliance and cost control by limiting resource options to only those that meet corporate policy and budget requirements.

Service Quotas

Definition: This central location allows users to view and manage their maximum resource limits across various AWS services from a single dashboard. It provides visibility into current usage levels and facilitates the process of requesting increases to prevent service interruptions.

Key Use Cases:

- Monitoring resource consumption to ensure applications do not hit hard limits unexpectedly.
- Requesting increases for specific service limits, such as the number of EC2 instances or VPCs allowed in a region.
- Configuring Amazon CloudWatch alarms to receive notifications when usage reaches a specific percentage of a quota.
- Reviewing default AWS service limits to plan for future architectural scaling and resource requirements.

AWS Systems Manager (SSM)

Definition: This operations hub provides a centralized interface to view and control infrastructure across AWS and on-premises environments. It simplifies resource management by automating routine maintenance tasks and providing secure access to instances without requiring SSH keys or bastion hosts.

Key Use Cases:

- Automating the process of patching operating systems and software across large fleets of EC2 instances.
- Securely storing and managing configuration data, connection strings, and passwords using Parameter Store.
- Executing remote commands or scripts across multiple instances simultaneously to ensure consistency.
- Maintaining software compliance by tracking resource inventory and ensuring instances remain in a defined state.
- Providing secure, audited shell access to instances through a browser-based interface rather than traditional SSH/RDP.

AWS Trusted Advisor

Definition: This online tool provides real-time guidance to help provision resources following AWS best practices. It inspects your AWS environment and makes actionable recommendations across five specific categories to improve efficiency, security, and reliability.

Key Use Cases:

- Identifying idle or underutilized resources to reduce monthly cloud spend and optimize costs.
- Improving security posture by identifying open ports, missing MFA on root accounts, or publicly accessible S3 buckets.
- Enhancing application performance by checking for overutilized instances or misconfigured storage volumes.
- Increasing availability through fault tolerance checks, such as verifying multi-AZ deployments and backup monitoring.

- Monitoring service quotas to ensure you do not hit resource limits during critical scaling events.

AWS Well-Architected Tool

Definition: This service provides a consistent process for reviewing cloud architectures against established best practices. It helps users measure their workloads against the six pillars of the Well-Architected Framework to identify critical risks and receive actionable remediation guidance.

Key Use Cases:

- Evaluating existing workloads to ensure they align with operational excellence, security, reliability, performance efficiency, cost optimization, and sustainability.
- Identifying high-risk architectural flaws and obtaining a prioritized list of improvements to strengthen infrastructure.
- Documenting architectural decisions and tracking the progress of workload improvements over time.
- Generating structured reports to demonstrate compliance with cloud best practices during internal or external audits.
- Standardizing the review process across multiple teams to ensure consistent design patterns throughout an organization.

AWS Application Discovery Service

Definition: This tool automates the process of gathering information about on-premises data centers to help plan migration projects. It collects server configuration data, usage metrics, and network dependencies to provide a comprehensive view of the existing infrastructure.

Key Use Cases:

- Identifying server dependencies and mapping application stacks before moving to the cloud.
- Estimating the Total Cost of Ownership (TCO) for running workloads on AWS based on actual resource utilization.
- Exporting collected data into AWS Migration Hub to track the progress of a large-scale migration.
- Discovering inventory in virtualized environments without installing software on every host using the agentless collector.
- Providing data for AWS Migration Strategy (7 Rs) decision-making by analyzing server performance and behavior.

AWS Application Migration Service (MGN)

Definition: This primary migration service simplifies and accelerates the process of moving physical, virtual, or cloud-based servers to the AWS Cloud. It utilizes continuous, block-level data replication to ensure that applications run natively on AWS without requiring architectural changes.

Key Use Cases:

- Performing “lift-and-shift” migrations (rehosting) to move large numbers of servers quickly.
- Migrating on-premises physical servers or virtual machines (VMware, Hyper-V) to Amazon EC2.
- Moving workloads from other cloud providers to AWS with minimal downtime.
- Automating the conversion of source servers to run natively on AWS infrastructure.
- Minimizing cutover windows by keeping source and target environments synchronized until the final transition.

AWS Database Migration Service (AWS DMS)

Definition: A managed migration service that helps move relational databases, non-relational databases, and other types of data stores to the AWS Cloud. It ensures the source database remains fully operational during the transfer, minimizing downtime for applications that rely on the data.

Key Use Cases:

- Migrating on-premises databases to Amazon RDS or Amazon Aurora.
- Moving data between different database engines, such as switching from Oracle to Amazon Aurora (heterogeneous migration).
- Consolidating multiple databases into a single target database.
- Replicating data continuously to a data warehouse like Amazon Redshift for real-time analytics.
- Synchronizing data across distributed systems to maintain data consistency.

Migration Evaluator

Definition: This complimentary service provides a data-driven business case for migrating to the cloud by analyzing on-premises resource utilization and inventory. It helps organizations visualize the financial impact of migration by comparing current infrastructure costs against projected AWS expenditures.

Key Use Cases:

- Creating a comprehensive business case for stakeholders to justify cloud migration costs and potential savings.
- Identifying the most cost-effective AWS instance types based on actual CPU, memory, and storage usage rather than provisioned capacity.
- Analyzing existing software licensing, such as Windows or SQL Server, to determine the best licensing strategy on AWS.
- Discovering on-premises assets and dependencies to plan a phased migration strategy with high accuracy.
- Comparing the Total Cost of Ownership (TCO) between maintaining local data centers and moving to a cloud-based model.

AWS Migration Hub

Definition: A centralized dashboard that provides a single location to track the progress of application migrations across multiple AWS and partner solutions. It offers visibility into the status of individual components and helps manage the entire migration lifecycle from discovery to modernization.

Key Use Cases:

- Monitoring the status of migrations involving multiple tools like AWS Application Migration Service (MGN) and AWS Database Migration Service (DMS).
- Discovering on-premises server inventory and mapping dependencies to plan effective migration strategies.
- Grouping related servers and databases into specific applications to track their migration progress as a single unit.
- Accessing right-sizing recommendations for EC2 instances based on collected utilization data from on-premises environments.
- Streamlining the refactoring of legacy applications into microservices using integrated modernization features.

AWS Schema Conversion Tool (AWS SCT)

Definition: This standalone software application automates the conversion of source database schemas and code objects to a format compatible with a different target engine. It identifies potential manual effort required by highlighting incompatible features and providing guidance for remediation during heterogeneous migrations.

Key Use Cases:

- Converting a commercial database schema (such as Oracle or SQL Server) to an open-source engine like Amazon Aurora or PostgreSQL.
- Analyzing existing database schemas to generate an assessment report that predicts the complexity and feasibility of a migration.
- Migrating data warehouse schemas from legacy platforms like Teradata or Netezza to Amazon Redshift.
- Scanning application source code to identify and convert embedded SQL statements that are specific to the source database engine.

AWS Snow Family

Definition: A collection of physical hardware devices that facilitate the movement of massive amounts of data into and out of the AWS Cloud, while also providing compute capabilities in remote or disconnected environments. These ruggedized devices help overcome challenges like high network costs, long transfer times, and limited bandwidth by using physical transport instead of the public internet.

Key Use Cases:

- Migrating terabytes or petabytes of data from on-premises locations to S3 when internet bandwidth is insufficient or unreliable.
- Performing local data processing and edge computing in environments with little to no connectivity, such as ships, oil rigs, or remote mines.
- Collecting and pre-processing large datasets at the source before shipping the physical device back to AWS for data ingestion.
- Providing temporary storage and compute power for disaster recovery or tactical field operations where a permanent data center is unavailable.

Amazon API Gateway (Application Programming Interface Gateway)

Definition: This fully managed service allows developers to create, publish, maintain, monitor, and secure RESTful and WebSocket APIs at any scale. It acts as a “front door” for applications to access data, business logic, or functionality from backend services like AWS Lambda or EC2.

Key Use Cases:

- Creating a serverless entry point for AWS Lambda functions to build scalable microservices.
- Managing and throttling high volumes of API requests to prevent backend service overloads.
- Implementing security layers such as API keys, OAuth, or AWS IAM to control access to backend resources.
- Providing a unified endpoint for multiple backend services, including on-premises servers or third-party applications.
- Enabling real-time, two-way communication for chat apps or dashboards using WebSocket APIs.

Amazon CloudFront

Definition: A global content delivery network (CDN) service that accelerates the distribution of static and dynamic web content to users by caching it at edge locations. It reduces latency by ensuring requests are routed to the nearest point of presence, significantly improving the end-user experience and reducing the load on origin servers.

Key Use Cases:

- Distributing high-volume static assets like images, stylesheets, and JavaScript files to decrease page load times.
- Delivering live or on-demand video streams with low latency and high throughput to a global audience.
- Enhancing web application security by integrating with AWS Shield and AWS WAF to mitigate DDoS attacks at the edge.
- Providing secure, encrypted access to private content using signed URLs or signed cookies.

- Serving dynamic content or APIs by optimizing the network path between the user and the backend infrastructure.

AWS Direct Connect

Definition: This service establishes a dedicated, private network connection between an on-premises data center and AWS. By bypassing the public internet, it provides more consistent network performance, increased bandwidth, and reduced latency for data transfers.

Key Use Cases:

- Transferring large datasets between local environments and the cloud to reduce bandwidth costs.
- Supporting real-time applications that require low latency and high throughput.
- Enhancing security by keeping sensitive traffic off the public internet.
- Creating a hybrid cloud architecture that integrates existing infrastructure with AWS resources.
- Providing a more reliable connection than standard internet-based VPNs for mission-critical workloads.

AWS Global Accelerator

Definition: This networking service improves the availability and performance of applications by directing user traffic through the AWS global network infrastructure. It provides static IP addresses that act as a fixed entry point to application endpoints, such as Application Load Balancers or EC2 instances, across multiple AWS Regions.

Key Use Cases:

- Reducing latency for global users by routing traffic over the optimized AWS private network instead of the public internet.
- Simplifying IP management by providing fixed entry points that do not change even if the underlying backend resources are updated.
- Improving application availability through automatic health checks that reroute traffic to healthy endpoints in different regions.
- Protecting applications from distributed denial of service (DDoS) attacks via integration with AWS Shield.
- Managing blue/green deployments or A/B testing by using traffic dials to shift traffic between different AWS Regions.

AWS PrivateLink

Definition: This networking technology provides private connectivity between Virtual Private Clouds (VPCs), AWS services, and on-premises applications without exposing data to the public internet. It simplifies network architecture by using interface endpoints to route traffic securely over the Amazon network backbone.

Key Use Cases:

- Accessing AWS managed services from within a private subnet without requiring an Internet Gateway or NAT Gateway.
- Connecting to third-party SaaS applications hosted on AWS while ensuring the traffic remains entirely within the AWS network.
- Sharing internal services or microservices across different AWS accounts and VPCs within a large organization.
- Meeting strict regulatory compliance requirements by preventing sensitive data from ever traversing the public internet.

Amazon Route 53

Definition: This is a highly available and scalable Domain Name System (DNS) web service designed to route end users to internet applications by translating human-readable names into numeric IP addresses. It effectively connects user requests to infrastructure running in AWS—such as EC2 instances, Elastic Load Balancers, or S3 buckets—as well as infrastructure outside of AWS.

Key Use Cases:

- Registering new domain names or transferring existing ones to be managed within the AWS ecosystem.
- Routing internet traffic to resources based on specific criteria like geographic location, network latency, or weighted distributions.
- Performing automated health checks on endpoints to ensure traffic is only directed to healthy, functioning resources.
- Implementing DNS failover to improve application availability by automatically redirecting users to a secondary site during an outage.

AWS Transit Gateway

Definition: This service acts as a central hub that connects multiple Virtual Private Clouds (VPCs) and on-premises networks through a single gateway. It simplifies network topology by replacing complex peering relationships with a centralized management point for data routing across an entire AWS organization.

Key Use Cases:

- Connecting hundreds or thousands of VPCs across different AWS accounts within a single region.
- Consolidating edge connectivity for on-premises data centers via VPN or AWS Direct Connect.
- Reducing operational overhead by eliminating the need for a “mesh” of individual VPC peering connections.
- Managing cross-account network traffic through a centralized point of control and monitoring to improve security and visibility.

Amazon VPC (Virtual Private Cloud)

Definition: This service provides a logically isolated section of the AWS Cloud where you can launch resources in a virtual network that you define. It offers complete control over the networking environment, including the selection of IP address ranges, creation of subnets, and configuration of route tables and network gateways.

Key Use Cases:

- Hosting multi-tier web applications by separating public-facing resources from private backend databases.
- Connecting on-premises infrastructure to the cloud using a hardware VPN or AWS Direct Connect.
- Enhancing security by using Security Groups and Network Access Control Lists (NACLs) to filter inbound and outbound traffic.
- Creating a hybrid cloud environment that allows seamless communication between local data centers and AWS resources.
- Providing secure, private connectivity to AWS services without using the public internet via VPC Endpoints.

AWS VPN (Virtual Private Network)

Definition: A managed service that establishes secure, encrypted tunnels between an on-premises network or individual device and the AWS global network. It leverages the public internet to provide a cost-effective way to extend private infrastructure into the cloud while maintaining data privacy.

Key Use Cases:

- Connecting a corporate data center to an Amazon VPC via a Site-to-Site connection for hybrid cloud architectures.
- Enabling remote employees to securely access AWS resources from their mobile devices or laptops using a Client VPN.
- Providing a quick-to-deploy, temporary backup connection for AWS Direct Connect circuits.
- Securing data transit between branch offices and cloud-hosted applications without the need for dedicated physical lines.

AWS Site-to-Site VPN (Virtual Private Network)

Definition: This managed service establishes a secure, encrypted connection between a remote network and Amazon Virtual Private Clouds (VPCs) over the public internet. It utilizes IPsec tunnels to ensure data confidentiality and integrity during transit between on-premises data centers and the cloud.

Key Use Cases:

- Extending an existing on-premises data center into the AWS Cloud to create a hybrid architecture.

- Providing a cost-effective and quick-to-deploy alternative to dedicated physical connections like AWS Direct Connect.
- Securing data communication between branch offices and centralized AWS resources.
- Serving as a backup connectivity option for primary network links to ensure high availability.

AWS Client VPN

Definition: A managed client-based VPN service that enables secure access to AWS resources and on-premises networks from any location. It functions as a scalable, high-availability solution that allows remote users to connect to a Virtual Private Cloud (VPC) using an OpenVPN-based client.

Key Use Cases:

- Providing remote employees with secure access to internal applications and development environments hosted in the cloud.
- Enabling administrators to perform maintenance on EC2 instances or databases within private subnets without exposing them to the public internet.
- Integrating with existing authentication systems like Microsoft Active Directory or SAML-based identity providers for centralized user management.
- Scaling connection capacity automatically based on the number of active users without needing to manage physical hardware or virtual appliances.

AWS Artifact

Definition: This self-service portal provides on-demand access to AWS security and compliance reports and select online agreements. It serves as a central repository for documentation that demonstrates how the infrastructure meets various global, regional, and industry-specific compliance standards.

Key Use Cases:

- Downloading Service Organization Control (SOC) reports and Payment Card Industry (PCI) documentation to provide to external auditors.
- Reviewing, accepting, and managing Business Associate Addendums (BAA) for HIPAA compliance requirements.
- Validating that the cloud infrastructure adheres to ISO certifications and other regulatory frameworks.
- Supporting internal audit and compliance assessments by providing official third-party verification of security controls.
- Accessing the AWS Privacy Notice and other legal documents related to data protection and governance.

AWS Audit Manager

Definition: This service automates the continuous collection of evidence to assess how an organization's cloud usage aligns with industry standards and regulations. It simplifies the process of mapping cloud resources to specific compliance controls, significantly reducing the manual effort required for risk assessments.

Key Use Cases:

- Preparing for external audits by generating pre-formatted reports based on frameworks like PCI DSS, GDPR, or HIPAA.
- Continuously monitoring environments to ensure ongoing compliance with internal governance policies and security best practices.
- Automating the gathering of evidence from multiple accounts and services into a centralized, audit-ready location.
- Managing internal risk assessments to identify and remediate gaps in security controls before they become formal compliance issues.

ACM (AWS Certificate Manager)

Definition: This service simplifies the process of provisioning, managing, and deploying public and private SSL/TLS certificates. It automates the renewal of certificates to ensure that websites and applications remain secure without manual intervention.

Key Use Cases:

- Securing network communications by encrypting data in transit for web applications.
- Deploying SSL/TLS certificates directly to integrated resources like Elastic Load Balancing (ELB) and Amazon CloudFront.
- Managing the lifecycle of certificates, including automated renewal and domain validation.
- Creating and managing private certificates for internal resources within a private network.

AWS CloudHSM (Cloud Hardware Security Module)

Definition: This service provides dedicated, single-tenant hardware security modules within the AWS Cloud to protect encryption keys and perform cryptographic operations. It ensures that only the customer has access to the keys, meeting the highest levels of regulatory and compliance requirements for data security.

Key Use Cases:

- Meeting corporate or regulatory compliance mandates that require FIPS 140-2 Level 3 validated hardware.
- Offloading SSL/TLS processing for web servers to reduce the computational burden on application instances.
- Protecting private keys for an Issuing Certificate Authority (CA) to secure internal public key infrastructure.

- Performing high-throughput cryptographic operations that require dedicated hardware performance and low latency.
- Managing encryption keys independently of AWS for applications that require strict separation of duties.

Amazon Cognito

Definition: This service provides customer identity and access management (CIAM) for web and mobile applications, allowing developers to add user sign-up, sign-in, and access control features. It supports scaling to millions of users and integrates with social identity providers like Google or Facebook, as well as enterprise identity providers via SAML 2.0 or OpenID Connect.

Key Use Cases:

- Implementing a secure, managed user directory (User Pools) to handle registration, authentication, and account recovery.
- Granting users temporary, limited-privilege AWS credentials (Identity Pools) to access backend resources like Amazon S3 buckets or DynamoDB tables.
- Enabling “Social Login” so customers can sign into a custom application using their existing social media or Amazon accounts.
- Providing built-in security features such as multi-factor authentication (MFA) and encryption of data at rest and in transit to protect user profiles.

Amazon Detective

Definition: This security service simplifies the investigation process by automatically collecting and analyzing log data from multiple AWS sources to identify the root cause of potential security issues. It uses machine learning and graph theory to create visual representations of relationships between resources, users, and IP addresses over time.

Key Use Cases:

- Investigating security findings flagged by Amazon GuardDuty to understand the scope and impact of a potential threat.
- Conducting root cause analysis for suspicious activities or unauthorized access attempts within an AWS environment.
- Visualizing complex relationships and historical patterns of resource behavior to identify anomalies.
- Streamlining incident response by providing a unified view of security data without requiring manual log aggregation or complex data modeling.

AWS Directory Service

Definition: This managed service enables the connection of AWS resources to an existing on-premises Microsoft Active Directory or the creation of a new directory in the cloud. It simplifies the

management of users, groups, and permissions across cloud-based applications and infrastructure without the operational overhead of maintaining domain controllers.

Key Use Cases:

- Enabling Single Sign-On (SSO) for users to access AWS applications like Amazon WorkSpaces or Amazon QuickSight using their existing corporate credentials.
- Joining Amazon EC2 instances to a domain to manage Windows workloads using standard Group Policy Objects (GPOs).
- Extending an on-premises directory to the cloud to provide seamless identity management for hybrid environments.
- Providing a standalone directory for small-scale applications that require basic Active Directory features through Simple AD.
- Integrating with AWS IAM Identity Center to manage administrative access to multiple AWS accounts.

AWS Firewall Manager

Definition: A security management service that provides central configuration and deployment of firewall rules across all accounts and resources within an AWS Organization. It ensures that both new and existing resources automatically comply with a mandatory set of security policies from a single administrative account.

Key Use Cases:

- Centrally managing AWS WAF rules across multiple web applications and accounts to ensure consistent web traffic filtering.
- Enforcing common VPC security group policies to prevent overly permissive traffic across an entire organization.
- Deploying AWS Network Firewall protections across multiple VPCs to inspect and control traffic at the network level.
- Automating the protection of new resources as they are created, ensuring they immediately inherit the organization’s security posture.
- Managing Amazon Route 53 Resolver DNS Firewall policies to block malicious domain requests across all accounts.

Amazon GuardDuty

Definition: This managed threat detection service continuously monitors AWS accounts and workloads for malicious activity and unauthorized behavior. It utilizes machine learning, anomaly detection, and integrated threat intelligence feeds to identify potential security risks such as compromised credentials or communication with known malicious IP addresses.

Key Use Cases:

- Detecting crypto-mining activity or malware on EC2 instances and container workloads.

- Identifying unusual API calls or login patterns that suggest account compromise or credential leakage.
- Monitoring for unauthorized access, suspicious data discovery, or public exposure within S3 buckets.
- Detecting communication with known malicious command-and-control (C2) servers.
- Automating security responses and remediation by integrating findings with AWS Lambda or Amazon EventBridge.

AWS Identity and Access Management (IAM)

Definition: This service provides centralized control over authentication and authorization for resources within an account. It enables the creation of users, groups, and roles with specific permissions to ensure the principle of least privilege is maintained across the cloud environment.

Key Use Cases:

- Managing individual users and their security credentials, such as passwords or access keys.
- Assigning permissions to groups of users to simplify access management for specific departments or job functions.
- Granting temporary access to applications or services through roles without sharing long-term security credentials.
- Enforcing Multi-Factor Authentication (MFA) to add an extra layer of protection for account logins.
- Defining fine-grained access policies to specify exactly which actions can be performed on specific resources.

AWS IAM Identity Center (Successor to AWS Single Sign-On)

Definition: This service provides a central location to manage user access to multiple AWS accounts and business applications. It allows users to sign in once with their existing corporate credentials to access all assigned resources through a single, unified web portal.

Key Use Cases:

- Managing multi-account access within an AWS Organization from a single administrative point.
- Integrating with external identity providers such as Microsoft Active Directory, Okta, or Google Workspace to synchronize users and groups.
- Providing a simplified login experience for employees to access their assigned AWS accounts and third-party SaaS applications.
- Implementing fine-grained permissions across an entire organization using centralized permission sets.
- Auditing user access and sign-in activity across the cloud environment through integration with AWS CloudTrail.

Amazon Inspector

Definition: This automated vulnerability management service continually scans AWS workloads for software vulnerabilities and unintended network exposure. It provides a centralized view of security findings across multiple accounts to help prioritize remediation efforts and improve the overall security posture.

Key Use Cases:

- Identifying Common Vulnerabilities and Exposures (CVEs) within Amazon EC2 instances.
- Scanning container images stored in Amazon Elastic Container Registry (ECR) for known security flaws before deployment.
- Analyzing AWS Lambda functions and layers for code-level vulnerabilities and security risks.
- Detecting unintended network accessibility to resources to ensure compliance with internal security policies.
- Automating security assessments throughout the software development lifecycle to catch and remediate issues early.

AWS Key Management Service (AWS KMS)

Definition: A managed service that enables the creation, management, and control of cryptographic keys used to protect data across various applications and integrated cloud services. It utilizes hardware security modules (HSMs) to ensure the security and integrity of the encryption keys while providing a centralized console for administration.

Key Use Cases:

- Encrypting data at rest within storage and database services such as Amazon S3, Amazon EBS, and Amazon RDS.
- Managing Customer Master Keys (CMKs) to define granular access permissions for specific encrypted datasets.
- Automating the periodic rotation of cryptographic keys to meet regulatory compliance and security best practices.
- Integrating with AWS CloudTrail to audit and log all key usage for security monitoring, compliance reporting, and forensic analysis.
- Digitally signing data or verifying signatures to ensure the authenticity and integrity of digital communications.

Amazon Macie

Definition: This fully managed data security and data privacy service uses machine learning and pattern matching to automatically discover, monitor, and protect sensitive data. It provides continuous visibility into data security risks by scanning objects stored within Amazon S3 buckets.

Key Use Cases:

- Identifying Personally Identifiable Information (PII) such as names, addresses, or credit card numbers to ensure regulatory compliance with GDPR or HIPAA.
- Monitoring S3 buckets for security risks, such as public accessibility or unencrypted data storage.
- Automating the discovery of sensitive data at scale across an entire AWS organization.
- Generating security findings that can be integrated with Amazon EventBridge to trigger automated remediation workflows.
- Providing a dashboard of data sensitivity and security posture for storage environments.

AWS Resource Access Manager (AWS RAM)

Definition: This service enables the secure sharing of specific AWS resources across multiple accounts or within an AWS Organization. It eliminates the need to create duplicate resources in every account, reducing operational overhead and costs while maintaining centralized control and visibility.

Key Use Cases:

- Sharing VPC subnets to allow multiple accounts to deploy application resources into a common, centrally managed network.
- Distributing Transit Gateways to interconnect virtual private clouds and on-premises networks across an entire organization.
- Providing access to Route 53 Resolver rules to ensure consistent DNS resolution across a multi-account environment.
- Sharing AWS License Manager configurations to track and manage software license usage across different business units.
- Granting access to Capacity Reservations to ensure that multiple accounts can utilize reserved compute power when needed.

AWS Secrets Manager

Definition: This service provides a secure, centralized repository for managing sensitive information such as database credentials, API keys, and OAuth tokens. It enables users to replace hardcoded credentials in application code with a programmatic call to retrieve the secret, significantly enhancing an organization’s security posture.

Key Use Cases:

- Automating the rotation of database passwords on a scheduled basis without requiring application downtime or manual intervention.
- Managing and protecting third-party API keys and service credentials used by distributed applications.
- Integrating with AWS Identity and Access Management (IAM) to enforce fine-grained access control, ensuring only authorized users and services can retrieve specific secrets.

- Securing sensitive configuration data for workloads running on Amazon EC2, AWS Lambda, or containerized environments.
- Monitoring and auditing secret usage through integration with AWS CloudTrail to meet compliance requirements.

AWS Security Hub

Definition: This cloud security posture management service provides a comprehensive view of security alerts and compliance status across multiple AWS accounts. It aggregates, organizes, and prioritizes security findings from various AWS services and third-party partner products into a single, centralized dashboard.

Key Use Cases:

- Centralizing security findings from integrated services like Amazon GuardDuty, Amazon Inspector, and Amazon Macie.
- Running automated, continuous configuration checks against industry standards and best practices, such as the CIS AWS Foundations Benchmark.
- Monitoring compliance status across an entire AWS organization to identify and remediate resource misconfigurations.
- Consolidating security data into a standardized format to simplify analysis and accelerate incident response efforts.
- Providing a high-level overview of the security health of an environment to help stakeholders prioritize remediation tasks.

AWS Shield

Definition: This managed service provides comprehensive protection against Distributed Denial of Service (DDoS) attacks for applications running on the cloud platform. It offers always-on monitoring and automatic inline mitigations to minimize application downtime and latency caused by malicious traffic.

Key Use Cases:

- Defending against common, frequent Layer 3 and Layer 4 infrastructure attacks automatically at no additional cost using the Standard tier.
- Implementing higher-level protection for critical applications against sophisticated volumetric or state-exhaustion attacks via the Advanced tier.
- Gaining 24/7 access to specialized response teams for real-time assistance and manual mitigation during active security events.
- Protecting specific resources such as Amazon CloudFront distributions, Amazon Route 53 hosted zones, and Elastic Load Balancers.
- Providing financial protection against scaling charges resulting from DDoS-related usage spikes on protected resources.

AWS WAF (Web Application Firewall)

Definition: This security service monitors and filters HTTP and HTTPS requests that are forwarded to protected web application resources. It provides control over how traffic reaches applications by using customizable security rules to block malicious traffic patterns and unauthorized bots.

Key Use Cases:

- Protecting web applications from common exploits like SQL injection (SQLi) and Cross-Site Scripting (XSS).
- Implementing rate-based rules to prevent brute-force login attempts and application-layer DDoS attacks.
- Filtering incoming traffic based on specific geographic locations, IP addresses, or custom HTTP headers.
- Securing resources deployed on Amazon CloudFront, Application Load Balancer, Amazon API Gateway, or AWS AppSync.
- Utilizing Managed Rulesets to quickly deploy protections against evolving security threats without writing custom code.

AWS Fargate

Definition: This serverless compute engine allows for the execution of containerized applications without the need to manage, provision, or scale the underlying virtual machine infrastructure. It integrates with Amazon Elastic Container Service (ECS) and Amazon Elastic Kubernetes Service (EKS) to handle resource allocation automatically, ensuring you only pay for the resources used by your tasks.

Key Use Cases:

- Deploying microservices architectures where managing individual server instances would create unnecessary operational overhead.
- Running batch processing jobs or periodic tasks that require temporary compute resources that scale based on demand.
- Simplifying the security posture of containerized workloads by providing workload isolation at the pod or task level.
- Migrating legacy applications to a containerized environment while minimizing time spent on infrastructure maintenance and OS patching.

AWS Lambda

Definition: This serverless, event-driven compute service allows for the execution of code without the need to provision or manage underlying infrastructure. It automatically scales applications by running code in response to specific triggers and charges only for the compute time consumed.

Key Use Cases:

- Automating administrative tasks such as scheduled backups or resource tagging.

- Processing data in real-time immediately after files are uploaded to Amazon S3 buckets.
- Building backend logic for web and mobile applications when integrated with Amazon API Gateway.
- Transforming and loading data into databases or data warehouses through automated ETL processes.
- Handling real-time stream processing for IoT devices or website clickstream analysis.

AWS Backup

Definition: A fully managed, policy-based service that centralizes and automates data protection across multiple cloud and on-premises resources. It provides a unified dashboard to configure backup policies, monitor activity, and ensure data availability for various storage, database, and compute services.

Key Use Cases:

- Automating scheduled backups for Amazon EBS volumes, RDS databases, and DynamoDB tables to reduce manual administrative overhead.
- Implementing cross-Region and cross-account backup strategies to enhance disaster recovery and protect against accidental deletion.
- Enforcing data retention policies to meet organizational or regulatory compliance requirements through automated lifecycle management.
- Centralizing the management of backup activities across diverse AWS services from a single console to simplify auditing and reporting.

Amazon Elastic Block Store (Amazon EBS)

Definition: This service provides high-performance, scalable block storage volumes designed for use with Amazon EC2 instances. It functions as a persistent virtual hard drive that remains available even if the associated compute instance is stopped or terminated.

Key Use Cases:

- Hosting operating system boot volumes for EC2 instances.
- Running relational or NoSQL databases that require consistent, low-latency performance.
- Storing data for enterprise applications like ERP or CRM systems.
- Creating point-in-time backups of data using snapshots for disaster recovery.
- Providing high-availability storage within a single Availability Zone.

Amazon EFS (Amazon Elastic File System)

Definition: This service provides a serverless, fully managed network file system (NFS) that automatically scales storage capacity as files are added or removed. It allows thousands of compute instances, including Amazon EC2 and on-premises servers, to access shared data concurrently across multiple Availability Zones.

Key Use Cases:

- Providing a common data source for web serving and content management systems that require shared access for multiple instances.
- Supporting big data analytics and media processing workflows that need high throughput and parallel access to large datasets.
- Serving as a persistent storage layer for containerized applications running on Amazon ECS, EKS, or AWS Fargate.
- Simplifying lift-and-shift migrations for enterprise applications that rely on traditional file-based storage structures rather than object storage.

AWS Elastic Disaster Recovery (DRS)

Definition: This service minimizes downtime and data loss by continuously replicating source servers into a low-cost staging area in the AWS Cloud. It enables rapid, automated recovery of physical, virtual, and cloud-based infrastructure during unexpected outages or cyberattacks.

Key Use Cases:

- Protecting on-premises workloads by replicating them to an AWS Region for failover.
- Increasing resilience for cloud-based applications by setting up cross-Region or cross-Availability Zone disaster recovery.
- Recovering from ransomware attacks by using point-in-time snapshots to restore servers to a clean, previous state.
- Reducing operational costs by replacing expensive, idle secondary data centers with a pay-as-you-go cloud recovery site.

Amazon FSx (File System)

Definition: This service provides fully managed, high-performance file systems that are compatible with popular file systems like Windows File Server, Lustre, NetApp ONTAP, and OpenZFS. It allows users to leverage the rich feature sets and performance of these systems without the administrative overhead of managing hardware or software.

Key Use Cases:

- Migrating legacy Windows-based applications that require shared file storage and Active Directory integration.
- Powering high-performance computing (HPC) workloads, such as video rendering or financial modeling, using the Lustre file system.
- Providing shared storage for machine learning training sets that require high throughput and low latency.
- Implementing enterprise-grade data management features like snapshots, cloning, and replication for NetApp-based environments.

Amazon S3 (Simple Storage Service)

Definition: This object storage service provides industry-leading scalability, data availability, security, and performance for virtually any type of data. It organizes data into buckets and objects, allowing for high durability and accessibility via the internet or private connections.

Key Use Cases:

- Hosting static websites by serving HTML, CSS, and image files directly to users without a server.
- Storing and retrieving backup files, snapshots, and archives for disaster recovery.
- Acting as a centralized data lake for big data analytics and machine learning workflows.
- Distributing media files, software installers, and large datasets globally.
- Managing long-term data retention using cost-effective storage classes like S3 Glacier for compliance.

Amazon S3 Glacier (Simple Storage Service Glacier)

Definition: This service provides secure, durable, and extremely low-cost storage classes designed for data archiving and long-term backup. It is optimized for data that is infrequently accessed and where retrieval times ranging from minutes to several hours are acceptable.

Key Use Cases:

- Storing regulatory compliance records that must be retained for several years to meet legal requirements.
- Archiving large sets of digital media assets that are no longer in active production but must be preserved.
- Maintaining long-term backups of enterprise data as a cost-effective alternative to maintaining on-premises tape libraries.
- Preserving historical healthcare records or scientific research data that requires high durability but rare access.

AWS Storage Gateway

Definition: This hybrid cloud service connects an on-premises software appliance with cloud-based storage to provide seamless integration between local environments and the AWS infrastructure. It allows local applications to use AWS storage services like Amazon S3, Amazon EBS, and Amazon S3 Glacier through standard storage protocols.

Key Use Cases:

- Moving backups to the cloud to reduce on-premises storage costs and physical hardware footprint.
- Providing low-latency local access to frequently used data while automatically storing the bulk of the data in Amazon S3.

- Replacing physical tape libraries with virtual tapes in the cloud for long-term data archiving and regulatory compliance.
- Facilitating disaster recovery by mirroring local data to AWS for quick restoration during a site outage.
- Migrating on-premises data to AWS for processing by cloud-native services like Amazon Athena or Amazon Rekognition.

Key Concepts

Technologies and Concepts

APIs (Application Programming Interfaces)

Definition: These are sets of protocols, routines, and tools that allow different software applications to communicate and exchange data with one another. They act as an intermediary layer that processes requests and ensures the seamless functioning of enterprise systems. In a cloud environment, these interfaces serve as the primary method for interacting with services, enabling programmatic access to infrastructure, storage, and management tools.

- Provide a standardized way for developers to request services.
- Abstract the underlying complexity of the hardware or software implementation.
- Enable automation by allowing scripts and code to manage cloud resources.

Benefits of migrating to the AWS Cloud (Amazon Web Services)

Definition: This paradigm shift allows organizations to transition from heavy upfront capital investments to a pay-as-you-go model, optimizing financial efficiency. It provides near-instant access to vast computing resources, enabling rapid innovation and global reach without the burden of managing physical infrastructure.

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

AWS Cloud Adoption Framework (AWS CAF)

Definition: This methodology provides structured guidance to help organizations develop and execute a comprehensive plan for their digital transformation. It organizes best practices into six distinct perspectives to ensure all functional areas of an enterprise are addressed:

- Business, People, and Governance (focusing on business capabilities)
- Platform, Security, and Operations (focusing on technical capabilities) By identifying specific gaps in existing skills and processes, it enables the creation of a roadmap that aligns technical efforts with organizational goals.

AWS Compliance

Definition: This framework ensures that the underlying infrastructure meets various global, regional, and industry-specific security standards and regulations. It operates under the shared responsibility model, where the provider manages the security and adherence of the physical hardware and software layers. Users can access on-demand reports and certifications to verify that the environment aligns with legal and regulatory requirements.

- Includes certifications like ISO 27001, SOC 1/2/3, and PCI DSS.
- Provides tools like AWS Artifact for downloading audit reports.
- Supports adherence to regional laws such as GDPR or HIPAA.

Compute

This category of cloud resources provides the processing power, memory, and logic required to run applications and process data. It encompasses a variety of delivery models that allow for the execution of code across virtualized hardware or managed environments. These services are designed to be scalable and elastic, providing the necessary infrastructure to handle varying workloads without the need for physical hardware management.

- Virtual machines
- Containerized environments
- Serverless functions

Cost management

Definition: This practice involves the continuous process of planning, monitoring, and controlling cloud-related expenses to ensure financial predictability and efficiency. It utilizes various tools to provide visibility into resource consumption, allowing organizations to forecast future spending and set budget thresholds. By analyzing billing data, businesses can align their cloud investments with operational goals while maintaining fiscal accountability.

Databases

Definition: These are organized collections of structured or unstructured data stored and accessed electronically. In a cloud environment, these systems are often provided as managed services that automate administrative tasks such as hardware provisioning, setup, patching, and backups. They are categorized based on data models, including:

- Relational systems for structured data using SQL
- Non-relational (NoSQL) systems for flexible schemas
- In-memory stores for high-speed data retrieval
- Specialized engines for graph, document, or time-series data

Amazon EC2 (Elastic Compute Cloud) Instance Types

Definition: These represent various pricing models and purchasing options for virtual server capacity. They allow users to balance cost-efficiency with performance requirements based on workload predictability and duration.

- On-Demand: Fixed-rate capacity billed by the second or hour with no long-term commitment.
- Reserved Instances: Discounted pricing offered in exchange for a one- or three-year commitment to a specific instance configuration.
- Spot Instances: Unused capacity available at significant discounts that can be reclaimed by AWS with short notice.
- Savings Plans: Reduced rates provided for a commitment to a consistent amount of hourly compute usage.

AWS Global Infrastructure (Amazon Web Services Global Infrastructure)

Definition: This physical framework consists of geographically dispersed locations where cloud services are hosted and delivered to users. It is organized into several key components:

- Regions: Physical locations around the world where data centers are clustered.
- Availability Zones: One or more discrete data centers within a specific area, each with redundant power, networking, and connectivity.
- Local Zones: Extensions that place compute, storage, and database services closer to large population centers.
- Edge Locations: Sites used by content delivery networks to cache data closer to end users for lower latency.

Infrastructure as Code (IaC)

Definition: This methodology involves managing and provisioning computing resources through machine-readable definition files rather than manual hardware configuration or interactive tools. It treats environment setups as software, allowing for version control, consistency, and repeatability across different deployment stages. By utilizing descriptive scripts, teams can automate the creation of complex architectures while ensuring:

- Reduced human error
- Elimination of configuration drift
- Rapid, standardized deployments

AWS Knowledge Center

Definition: This online repository contains a collection of the most frequently asked questions and technical articles provided by AWS Support. It serves as a self-service resource where users can find authoritative answers and step-by-step instructions for common technical challenges. The content is curated by experts to help customers understand and resolve issues independently.

- Features articles and videos created by support engineers.

- Organized by service and category for easy navigation.
- Provides verified solutions to common configuration and billing questions.

Machine learning

Definition: This field of artificial intelligence focuses on developing algorithms and statistical models that allow computer systems to perform specific tasks without relying on explicit instructions. These systems analyze large datasets to identify complex patterns, improve their performance over time through experience, and generate data-driven predictions. Within cloud environments, this technology leverages high-performance computing and specialized hardware to automate the training, tuning, and deployment of predictive models at scale.

Management and governance

Definition: This domain encompasses the tools and frameworks used to maintain control over cloud environments while enabling organizational agility. It focuses on providing visibility into resource configurations, tracking changes, and ensuring that infrastructure adheres to specific compliance and security standards. These processes allow for the automated oversight of large-scale deployments to minimize risk and manage costs effectively.

- Centralized resource monitoring
- Automated policy enforcement
- Audit logging and activity tracking
- Budgeting and cost management

Migration and data transfer

Definition: This category encompasses the methodologies and services used to move digital assets, such as applications, databases, and large-scale storage, from on-premises environments or other clouds into the AWS ecosystem. It includes physical hardware for offline transport and network-based tools for continuous synchronization or one-time shifts. These solutions ensure minimal downtime and maintain data integrity during the transition to cloud-based infrastructure. Key services include:

- AWS Application Migration Service (MGN)
- AWS Database Migration Service (DMS)
- AWS Snow Family
- AWS DataSync

Network services

Definition: These cloud-based solutions provide the underlying infrastructure necessary to connect resources within a virtual environment and to external endpoints. They facilitate secure communication, traffic routing, and domain name resolution across a global infrastructure. This category encompasses:

- Virtual private clouds and subnets

- Domain Name System (DNS) management
- Content delivery and edge caching
- Load balancing and traffic distribution
- Dedicated physical connectivity options

AWS Partner Network (APN)

Definition: This global community consists of professional services firms and software vendors that leverage cloud infrastructure to build, market, and sell customer solutions. It provides members with technical, marketing, and go-to-market support to help organizations accelerate their digital transformation. The ecosystem is categorized into two primary tracks:

- Services Partners: Includes consulting, professional, managed, and value-added resale services.
- Technology Partners: Includes independent software vendors (ISVs), SaaS, PaaS, and hardware providers.

AWS Prescriptive Guidance

Definition: This resource provides a library of time-tested strategies, guides, and patterns developed by experts and partners to accelerate cloud migration, modernization, and optimization. It offers structured methodologies for implementing cloud solutions based on real-world experience and best practices. The content is organized into three distinct categories:

- High-level strategies for business and technical objectives
- Detailed guides for specific technologies or methodologies
- Reusable patterns for common architectural tasks

AWS Pricing Calculator

Definition: This web-based planning tool allows users to create estimates for their planned infrastructure costs on the platform. It provides a transparent breakdown of monthly and annual expenses based on specific service configurations and usage parameters. Users can model various architectural scenarios and organize these projections into groups to reflect different business units or projects.

AWS Professional Services

Definition: This global team of experts provides specialized guidance and technical assistance to help organizations achieve specific business outcomes. They work alongside internal teams and partners to supplement existing skills with deep cloud expertise and best practices. The primary focus is on accelerating cloud adoption and ensuring successful enterprise-level transitions through strategic planning and execution.

AWS re:Post

Definition: This is a community-driven, crowdsourced Q&A service that provides technical guidance and knowledge sharing for cloud practitioners. It features expert-reviewed content contributed by community members, partners, and AWS employees to ensure accuracy and reliability. The platform utilizes a reputation-based system to highlight trusted contributors and provides a centralized location for finding verified answers to complex technical challenges.

AWS SDKs (Software Development Kits)

Definition: These collections of libraries and tools allow developers to interact with cloud resources using various programming languages. They simplify the process of making API calls by handling low-level tasks such as:

- Authentication and request signing
- Automated retry logic
- Error handling
- Data serialization

By providing language-specific abstractions, these tools enable programmatic management of infrastructure and services directly within application code.

Security

Definition: This fundamental pillar of cloud operations focuses on protecting information, systems, and assets while delivering business value through risk assessment and mitigation strategies. It involves establishing a robust framework for managing identities, maintaining visibility into environment changes, and safeguarding data across the entire infrastructure. Key components include:

- Implementing strong identity foundations and access controls
- Maintaining traceability through logging and monitoring
- Applying protection at all layers of the infrastructure
- Automating responses to potential incidents
- Protecting data at rest and in transit through encryption

AWS Security Blog

Definition: This official online publication provides technical deep dives, announcements, and expert guidance focused on security, identity, and compliance within the cloud environment. It serves as a primary source for staying updated on the latest security features, regulatory requirements, and architectural best practices. Content is authored by engineers and architects to help organizations maintain a robust security posture.

AWS shared responsibility model

Definition: This framework delineates the specific security and compliance obligations managed by the cloud provider versus those managed by the customer. It ensures operational clarity by dividing tasks into two distinct categories:

- Security of the cloud: The provider manages the global infrastructure, including hardware, software, networking, and physical facilities.
- Security in the cloud: The customer is responsible for protecting their data, managing user identities, and configuring guest operating systems or firewalls.

AWS Solutions Architects

Definition: These professionals are responsible for designing and implementing scalable, resilient, and cost-effective systems within the cloud environment. They utilize the Well-Architected Framework to ensure infrastructure meets security, performance, and reliability standards. These experts translate complex business requirements into technical blueprints, guiding the selection of appropriate compute, storage, and networking services.

Storage

This fundamental cloud service category provides the infrastructure necessary to persist, protect, and retrieve digital data across a distributed network. It encompasses various architectures designed to manage specific data structures, including:

- Object-based systems for unstructured data
- Block-level volumes for high-performance applications
- File-based systems for shared access These services ensure high durability and availability while allowing capacity to scale dynamically based on demand without the need to manage physical hardware.

AWS (Amazon Web Services) Support Center

Definition: This centralized hub within the Management Console serves as the primary interface for managing technical support cases and accessing expert assistance. It provides a unified location to monitor active inquiries, review historical correspondence, and access specialized resources like the Knowledge Center. Access levels and communication channels within this portal are determined by the specific support plan associated with the account.

- Dashboard for case status and communication history
- Integration point for Trusted Advisor and Health Dashboard
- Gateway to technical documentation and community forums

AWS Support plans

Definition: These tiered subscription models provide technical assistance, guidance, and tools to help customers manage their cloud infrastructure effectively. Each level offers escalating degrees

of access to cloud support engineers, response time guarantees, and architectural guidance based on the complexity of the environment.

- Basic: Included for all customers, providing access to whitepapers, documentation, and support for billing or account issues.
- Developer: Offers email access to support during business hours for non-production workloads.
- Business: Provides 24/7 phone, email, and chat access for production environments and full Trusted Advisor checks.
- Enterprise: Includes a dedicated Technical Account Manager (TAM) and the fastest response times for business-critical systems.

AWS Well-Architected Framework

Definition: This set of best practices provides a consistent approach for evaluating cloud architectures and implementing scalable designs. It offers a structured methodology to compare workloads against established design principles and identifies specific areas for improvement. The framework is organized into six distinct pillars:

- Operational Excellence
- Security
- Reliability
- Performance Efficiency
- Cost Optimization
- Sustainability