AWS CERTIFIED CLOUD PRACTITIONER (CLF-C02)

COMPLETE CHEATSHEET

Created By Jaimin Vitthalpara

Table of Contents

SR No.	AWS Service / Topic	Page No.
1	Core Cloud Concept & Types of Cloud Services	3
3	Ec2 Pricing Models & instance Types	5
4	Amazon EBS Volume Types	8
5	Elastic Load Balancers (ELB)	9
6	Auto-Scaling	10
7	Amazon S3	12
8	Amazon Database	16
9	AWS Container Orchestration & Registry Services	20
10	Route 53	21
11	Amazon VPC	23
12	AWS Shared Responsibility Model	25
13	AWS Web Security	26
14	AWS AI/ML Services	29
15	AWS Multi-Account Management	31
16	AWS Cost Management	33
17	AWS Pricing Models for Different Services	34
18	AWS Identity Services & Authentication (IAM)	35
19	Other Remaining AWS Services	36
20	AWS Framework (CAF & WAF)	39
21	Business Value of AWS	42
22	7 R’s of AWS Migration	44
23	Developer Tools for Managing AWS Services	45

Core Cloud Concept

Term	Simple Explanation	Real-world Example
Elasticity	Auto increase or decrease resources as per demand	E-commerce site adds more servers on sale days
Scalability	Ability to handle growth (more users/data) smoothly	Start-up adds more EC2 instances as user base grows
High Availability	The system remains online and accessible even if parts fail	Two servers in different AZs keep website online
Durability	Data remains safe and intact over time	Files stored in Amazon S3 are never lost
Fault Tolerance	The system continues working even if some components fail	One server crashes, but app keeps running on others
Agility	Ability to adapt quickly to changes	Quickly launching test servers for a new feature
Resiliency	The system recovers quickly after a failure	Auto-restarting a failed EC2 instance
Reliability	Consistently performs its intended function	The app always returns correct data without crashing
Cost Efficiency	Pay only for what you use, avoid waste	Shutting down dev server at night to save money
Security	Protecting data, apps, and systems	IAM roles control who can access what in AWS

Types of Cloud Services

Types	Description
IaaS (Infrastructure as a Service)	● Provides Virtualized Computing Resources over the internet ● Users manage OS, runtime, apps ● **Examples:** ○ AWS EC2 ○ AWS VPC ○ AWS EBS ○ Azure VMs ○ Google Compute Engine ● **Use Case:** Full Control Over Infrastructure
PaaS (Platform as a Service)	● Provides platform to develop, run, and manage apps ● Handles OS, middleware, runtime ● **Examples:** ○ AWS Elastic Beanstalk ○ AWS Lambda ○ AWS Fargate ○ Heroku ○ Google App Engine ● **Use Case:** Focus on app development, not infrastructure
SaaS (Software as a Service)	● Delivers software over the internet ● No infrastructure / platform management needed by user ● **Examples:** ○ Gmail ○ DropBox ○ Salesforce ○ Microsoft 365 ○ Amazon Work Mail ○ Amazon Chime ● **Use Case:** End-users accessing apps via browser

Ec2 Pricing Models

Pricing Model	Key Features (Incl. Pricing / Discount)	Use Case	Easy-to-Remember Tips
On Demand	• Pay Per Second • No long-term commitment • 0% Discount (Most Expensive)	Short-term, Unpredictable Workload	**“Pay as you go”** like a taxi ride
Reserved Instances (RI)	• 1 or 3 year commitment • Up to 72% Discount	Predictable, Steady Use	**"Reservation = Hotel booking for long stays"**
Saving Plans	• Flexible 1 or 3 year commitment (Ec2, Lambda, Fargate) • Up to 72% Discount	Cost-saving with flexibility	**"Savings with flexibility"** like a gym membership
Spot Instances	• Bid On Unused Capacity • Can be interrupted (2-min warning) • Up to 90% Discount	Fault-tolerant / flexible tasks (batch processing, CI/CD)	**"Spot = auction seat–cheap, but may lose it anytime"**
Dedicated Host	• Full physical server dedicated • Supports BYOL (Bring Your Own License) • More Expensive	Compliance / Licensing needs full H/W control	**"Dedicated = Own the whole server"**
Dedicated Instances	• Hardware isolated but not full host control • Slightly more expensive	Isolation without full control	**"Instance = isolated ride, not full ownership"**
Capacity Reservation	• Reserve capacity in AZ • Pay only when running • No discount	Guarantee capacity (e.g., peak events)	**"Reserve your seat in advance."**

Ec2 Pricing Models Real-World Scenario Examples

Pricing Model	Real-World Scenario
On Demand	A startup is testing an app and needs flexibility - they use On-Demand EC2.
Reserved Instances (RI)	A company runs a web app 24/7 for 3 years - they buy RIs to save costs.
Savings Plans	A company wants to save money across EC2 and Lambda - they choose Compute Savings Plans.
Spot Instances	A data analytics team runs batch jobs overnight - they use Spot for big savings.
Dedicated Host	A bank needs to use a specific OS license tied to hardware - they go with Dedicated Hosts.
Dedicated Instances	A healthcare app needs a bit more isolation - they use Dedicated Instances.
Capacity Reservation	During peak shopping season, a retailer reserves capacity to ensure availability.

Ec2 Instance Types

Instance Type	Instance Families	Focus On	Use Cases
General Purpose	t3, m5	Balance of Compute, memory & networking	● Web Servers ● Small Database ● Development Environments
Compute Optimized	c5, c6g	High compute capacity	● High-Performance Web Servers ● Gaming ● Machine Learning ● Batch Processing
Memory Optimized	r5, x1	Large memory capacity	● Large DB's ● In-memory caches ● Real-time big data analytics
Storage Optimized	i3, d2	High disk throughput & IOPS	● NoSQL DB ● Data Warehousing ● Distributed file systems
Accelerated Computing	p3, g4	GPU & FPGA Capabilities	● Machine Learning ● Graphics Processing ● High-Performance Computing (HPC)

Amazon EBS Volume Types

Volume Type	Storage Type	Best For	Use Cases
gp3	General Purpose SSD	Balanced Workloads	● Boot volumes, ● DB with Moderate I/O ● Dev / Test Workloads ● Web / App Servers ● Microservices
gp2 (Legacy)	General Purpose SSD	Legacy balanced workloads	Similar to gp3, but older systems
io2 / io2 Block Express	Provisioned IOPS SSD	Critical IOPS-intensive workloads	● SAP HANA ● Oracle ● high-performance SQL DB’s
st1	Throughput Optimized HDD	High-throughput, large data scans	● Data Lakes ● Log Processing ● Streaming
sc1	Cold HDD	Lowest-cost, infrequent access	● Backup ● Archival ● Rarely Accessed data
Standard	Magnetic (legacy)	Very old workloads (rare)	Legacy systems only

Elastic Load Balancers (ELB)

Load Balancer's Type	Layer Info.	Use Cases
ALB (Application Load Balancer)	Works at Layer 7 (Application layer)	● For HTTP/HTTPS traffic ● Supports Path-based & Host-based routing ● Ideal for web apps, microservices, containers ● Supports WebSocket, integrates with Lambda
NLB (Network Load Balancer)	Works at Layer 4 (Transport layer: TCP/UDP)	● For TCP, UDP & TLS traffic ● Ultra-low latency and high performance ● Handles millions of requests per second ● Supports static IPs ● Best for real-time apps like gaming and IoT
GWLB (Gateway Load Balancer)	Works at Layer 3 (Network layer: IP)	● Routes traffic to Third-party security tools ● Transparent to Source & Destination ● Ideal for firewalls, DDoS protection, packet inspection ● Used in security-focused architectures
CLB (Classic Load Balancer)	Works at Layer 4 & Layer 7	● Older generation load balancer ● Limited features, basic HTTP/TCP support ● No path-based routing or WebSocket support ● Only suitable for legacy applications

Auto-Scaling

Types	Description
Manual Scaling	● Manually add / remove instances ● Requires Human intervention ● Useful when traffic patterns are known
Scheduled Scaling	● Automatically scales at Pre-defined times ● Useful for predictable traffic patterns ● **Example:** Scale up / down during Business hours
Dynamic Scaling	● Adjusts instances based on metrics ● Uses CPU, memory, or request rate ● Ensures performance without manual work
Predictive Scaling	● Uses machine learning to predict demand ● Analyzes past traffic patterns ● Scales in advance to handle spikes

Server Scaling Techniques: Horizontal vs Vertical

Features	Horizontal Scaling	Vertical Scaling
Also Know As	Scale Out / Scale In	Scale Up / Scale Down
How it works	Adds More Instances / Servers	Increases Size of a Single Virtual Machine (VM's)
Flexibility	More Flexible, Scalable	Limited by instance type
Downtime	Usually No Downtime	May need Restart / Downtime
Performance Boost	Increases capacity & fault tolerance by distributing	Increases power of one server (CPU, RAM, etc.)
Example in AWS	Auto Scaling Group (ASG) Behind Load Balancer	Changing Insance Type t2.micro → t2.large Manually
Cost Efficiency	Pay for what you use	Can be wasteful at peak load
Best For	● Web Applications ● MicroServices ● Unpredictable Traffic	● DB's Servers ● Legacy Applications with Fixed load patterns

For Quick Reference

1) **Horizontal Scaling =** More Machines
2) **Vertical Scaling =** One Bigger Machine

Amazon S3 Storage Classes

S3 Storage Classes	Use Cases	Memory Trick to Remember	Min. Storage Duration (In Days)
S3 Standard	• For frequently accessed data • 99.99% availability, 11 9's durability • No retrieval fee	Standard = Used daily	None
S3 Intelligent-Tiering	• Auto-moves between frequent & infrequent • Great when access pattern is unknown • No impact on performance	Smart = Let AWS manage it	30
S3 Standard - IA	• For infrequently accessed, fast needed • Good for storing secondary backup copies of on-premises data • Less frequently but requires Rapid access when needed • Lower cost, but has retrieval fee	IA = Infrequent Access	30
S3 One Zone - IA	• Like Standard-IA but stored in 1 AZ • Cheaper, but no AZ redundancy	One Zone = Single AZ only	30
S3 Glacier Instant Retrieval	• Archival data with Instant Access • Cheaper than Standard-IA	Instant Glacier = Fast Cold	90
S3 Glacier Flexible Retrieval	• Archival storage, Retrieve in Minutes or Hours • Use for backups and long-term archives	Flexible = Options to pick	90
S3 Glacier Deep Archive	• Lowest-cost storage • Retrieval in ~12 hours	Deep = Deep Freeze	180
S3 Outposts	• For storing data **on-premises** • Same APIs as S3 in the cloud • Used where AWS Region is not an option	Outposts = S3 at your site	None

AWS Snow Family (Offline Data Transfer / Migration to Amazon S3)

Snow Family Services	Features & Storage Capacity	Use case
AWS Snowcone	● 8 TB (HDD) or 14 TB (SSD) ● Secured & Lightweight (4.5 lbs) ● Battery powered	Small, portable data transfer & edge computing devices for remote locations
AWS Snowball Edge	● 80 TB Storage ● supports compute with Amazon Ec2 & Lambda ● Ideal for edge AI/ML	Large-scale data transfer & edge computing for pre-processing workloads
AWS Snowmobile	● 100 PB Storage in a **ruggedized shipping** Containers ● Suitable for **exabyte-scale**	Massive-scale petabyte-to-exabyte data migration for data centers to AWS

AWS Hybrid Storage Gateway

Snow Family Services	Description & Use case
File Gateway	● Provides File-based access (NFS/SMB) and stores files in S3 as Objects. ● Ideal for file Sharing & Backup solutions.
Volume Gateway (Cached & Stored)	● Presents cloud-backed ISCSI block storage. ● **Cached:** Frequently accessed Data is stored locally, with backups in S3. ● **Stored:** Entire Dataset is stored **on-premises**, async backups in S3.
Tape Gateway	● Replaces physical tape libraries with S3 and Glacier-based virtual tape storage. ● Used for Backups & Archiving.

Key Comparison: AWS S3 & Batch

Features	Amazon S3	AWS Batch
Purpose	Object storage service for Storing & Retrieving any amount of data.	Fully managed service for batch processing
Functionality	Stores data as objects in buckets.	Processes large-scale jobs in bulk.
Use case	Storing files, backups, media, logs, and application data.	Running ETL tasks, simulations, data processing, and batch computations.
Data Processing Capability	No direct processing, data must be accessed & processed externally (e.g., via Lambda).	Runs compute jobs on provided or dynamic resources like EC2 instances or Fargate.
Compute Resource Management	Not applicable, purely a storage service.	Manages compute resources dynamically (e.g., EC2, Spot Instances, Fargate)
Pricing Model	Pay for the storage used and data transfers.	Pay for the compute resources used.
Integration	Integrates with AWS services like Lambda, Batch, Glue, and Athena for workflows.	Integrates with S3, DynamoDB, CloudWatch & other AWS services for workflows.
Scalability	Scales storage automatically as needed.	Scales compute resources automatically based on job requirements.

Encryption in S3

Features	Server-Side Encryption (SSE)	Client-Side Encryption (CSE)
Who encrypts the data?	AWS (after upload).	You (before upload)
Where does encryption happen?	On AWS side	On your local system
Key management	AWS or Customer (via KMS/SSE-C)	You
Simplicity	Easier to implement	Requires more effort
Best for	Compliance, convenience	Max security, zero-trust setups.
Data visible to AWS?	Yes (temporarily before encryption) [AWS manages encryption but customers control access via IAM / KMS.]	No (encrypted before sending)

Server-side Encryption in S3

Features	Full form	Key Mgmt.	Description
SSE - S3	Server-Side Encryption with Amazon S3-Managed Keys	AWS manages everything	Simplest to use, Automatic encryption
SSE - KMS	Server-Side Encryption with AWS Key Management Service	AWS KMS + You	Offers key rotation, audit logging, and fine-grained access control
SSE - C	Server-Side Encryption with Customer-Provided Keys	You (provide your own key)	You manage and supply the encryption key in every request. AWS doesn't store it

Types of Database

Types	Description & Use Case
Relational DB (SQL Based)	• Stores data in structured tables with rows & columns. • Uses SQL for Querying • **Common DB:** MySQL, PostgreSQL, Amazon RDS
NoSQL DB	• Designed for flexible, unstructured or semi-structured data • Includes document, key-value, graph, and column-family DB's • **Common DB:** MongoDB, DynamoDB, Cassandra
In-Memory DB	• Stores data in RAM for ultra-fast access • Used for Caching, session mgmt., and Real-time analytics • **Common DB:** Redis, Memcached
Graph DB	• Stores relationships between Data points • Optimized for Traversing complex relationships • **Common DB:** Neo4j, Amazon Neptune
Data Warehouse	• Optimized for Analytical queries & Business intelligence • Supports OLAP workloads and large-scale reporting • **Common DB:** Amazon Redshift, Snowflake
Time-Series DB	• Specializes in Timestamped Data (e.g., IoT, logs, metrics) • Optimized for High Write throughput and Time-based Queries • **Common DB:** InfluxDB, Amazon Timestream

Types of NoSQL Database

Types	Description & Use Case
Document-based	● Stores data as documents (JSON, BSON) with flexible schemas. ● Ideal for Content Mgmt. & Catalogs (e.g., MongoDB, Couchbase).
Key-Value	● Simplest NoSQL type, it stores data as key-value pairs. ● Best for session Mgmt., Caching (e.g., DynamoDB, Redis).
Column-family	● Uses tables, rows & dynamic columns for Large-scale analytics. ● Suitable for Data Warehousing (e.g., Apache Cassandra, HBase).
Graph-based	● Designed for relationships and network-based data (nodes & edges). ● Useful for social networks and fraud detection (e.g., Neo4j, Amazon Neptune).

AWS Data Storage, Database, and Analytics Services (Quick Reference Table)

AWS Service	Description & Use Case
Amazon RDS	● Fully managed Relational database service ● Supports MySQL, PostgreSQL, Oracle, Aurora, and SQL Server ● Manages Backup, Patching, Scaling & Replication ● Ideal for OLTP workloads with High Availability
Amazon Aurora	● MySQL/PostgreSQL-compatible engine with high performance ● 5x faster than MySQL, 3x faster than PostgreSQL ● Auto-scales to 128 TB, supports read replicas ● Best for Enterprise Apps needing High throughput
Amazon ElastiCache	● In-memory Caching service (Redis, Memcached) ● Speeds up apps by caching Frequently used data ● Good for Real-time apps like Gaming and Finance
Amazon DynamoDB	● Fully managed NoSQL DB (Key-value / Document models) ● Provides Single-digit millisecond latency & auto-scaling ● Ideal for IoT, gaming, mobile apps
Amazon Redshift	● Data warehouse for large-scale Analytics ● Uses columnar storage and MPP for fast queries ● Ideal for Business intelligence over Big datasets
Amazon EMR	● Managed Big Data processing (Hadoop, Spark, Presto) ● Used for ML, Data Transform & Log Analysis ● Cost-effective for processing Petabytes of Data
Amazon Athena	● Serverless SQL query service for data in S3 ● Used to analyze data in Amazon S3 using standard SQL ● Pay-per-scan, no infrastructure to manage ● Best for ad-hoc queries and log analytics
Amazon Quicksight	● BI tool for dashboards and reports ● Connects with RDS, Redshift, Athena, S3 ● Pay-per-session pricing model
Amazon DocumentDB	● Fully managed NoSQL document DB compatible with MongoDB ● Stores and queries JSON data ● Ideal for content management and catalogs

Amazon Neptune	● Managed Graph database ● Supports property graph and RDF/SPARQL queries ● Used in Social, Fraud Detection & recommendation systems
Amazon Timestream	● Time series database for IoT and ops monitoring ● Optimized for time-based data ● Ideal for metrics, real-time analytics
Amazon QLDB	● Immutable, cryptographically verifiable ledger DB ● Ensures data integrity ● Great for financial, regulatory, and supply chain logs
Amazon Managed Blockchain	● Managed service for Blockchain network creation ● Supports Hyperledger Fabric and Ethereum ● Used in contracts, supply chain, finance
Amazon Glue	● Serverless data integration service ● Automates ETL and data cataloging ● Prepares data for analytics and ML
Amazon DMS (Data Migration Service)	● Service for migrating databases to AWS ● Supports same/different engine migration ● Minimal downtime, supports ongoing replication

AWS Container Orchestration & Registry Services

Services	Description	Use Case
EKS	• Elastic Kubernetes Service • Managed Kubernetes by AWS • Lets you run Kubernetes apps without managing control plane	• When you need full Kubernetes control • Ideal for Microservices & Hybrid environments • Used by teams already using Kubernetes
ECS	• Elastic Container Service • Supports EC2 and Fargate launch types • Easier to use than EKS	• For simpler AWS-native container orchestration • Ideal for batch jobs, APIs, and small-scale services • Used when Kubernetes is not required • Best for AWS-integrated, simple container apps
Fargate	• Serverless compute engine for containers • Works with both ECS and EKS • You only pay for vCPU and memory used	• No need to manage EC2 instances • Ideal for small, event-driven, unpredictable workloads, automation, startups & cost optimization
ECR	• Elastic Container Registry • Fully managed and integrated with ECS/EKS/Fargate • Secure and scalable registry for pulling container images	• To store, manage and deploy Docker container images • Commonly used in CI/CD pipelines • Works with ECS, EKS, and Fargate easily

Trick to remember:

1) **EKS:** Managed Kubernetes service for running containerized apps.
2) **ECS:** Managed service for running Docker containers on AWS.
3) **ECR:** Container registry for storing Docker images.
4) **Fargate:** Serverless compute engine to run containers (works with ECS and EKS, no Ec2 management).

Routing Policies in Route 53

Types	Use Cases	Example
Simple Routing	● Default method ● Routes traffic to a single resource ● No health checks or rules	website hosted on one EC2
Weighted Routing	● Distributes traffic based on set weights (%) ● Useful for A/B testing or gradual deployments	70% to one server, 30% to another
Latency-based Routing	● Routes traffic to region with lowest latency ● Improves user experience globally	US users go to US-East, India users to Mumbai
Failover Routing	● Active-passive setup ● If primary fails (health check), traffic shifts to backup	If an EC2 instance fails in one AZ, switch to another
Geolocation Routing	● Routes based on user's geographic location ● Used for legal/regional content or custom experiences	Show different content in EU vs. US region.
Geoproximity Routing (via Traffic Flow)	● Routes based on proximity and bias ● Requires Route 53 Traffic Flow ● Can shift traffic weight between regions manually	Send more traffic to closer or stronger region
Multivalue Answer	● Returns multiple healthy records ● Includes health checks ● Acts like basic load balancing ● Improves user experience globally	Return 3 healthy IPs to the client.

Most common Record type in Route 53

Record Types	Description	Example
A Record	Maps a Domain name to an IPv4 address	example.com → 192.0.2.1
AAAA Record	Maps a Domain name to an IPv6 address	example.com → 2001:0db8::1
CNAME Record	Redirects one Domain name to another	www.example.com → example.com
MX Record	Specifies Mail servers for a domain.	Mail for example.com → mail.example.com
TXT Record	Stores Text information, often for verification and security	SPF or DKIM settings
SRV Record	Specifies the location of services like SIP or LDAP for a domain	_sip._tcp.example.com
PTR Record	Used for reverse DNS lookups, mapping IP addresses to domain names	192.0.2.1 → example.com
NS Record	Indicates Authoritative name servers for a domain	ns1.example.com, ns2.example.com
SOA Record	Contains administrative info about a domain, like the primary DNS server and contact details.	Primary DNS: ns1.example.com

Amazon VPC Networking Components

Types	Use Cases
VPC	- Known as a Virtual Private Network - Isolated section to launch AWS resources - Customizable IP range, subnets, route tables, etc.
Subnet	- Segment within a VPC to group resources - **Public subnet:** Accessible from internet - **Private subnet:** Internal access only
Internet Gateway (IGW)	- Enables internet access for public subnets - Must be attached to the VPC
NAT Gateway / NAT Instance	- Allows private subnets to access internet outbound only - NAT Gateway is managed, scalable, and preferred
Route Table	- Contains rules to direct traffic within the VPC - Each subnet is associated with a route table
Security Group	- Acts as a virtual firewall at the **instance** level - Controls **inbound and outbound traffic** only Allow - **Stateful:** return traffic is automatically allowed
Network ACL (NACL)	- Firewall at the **subnet** level - Allow & Deny inbound/outbound traffic - **Stateless:** return traffic must be explicitly allowed
VPC Peering	- Connects two VPCs to communicate using private IPs - Works across accounts and regions
Transit Gateway (TGW)	- Central hub to connect multiple VPCs and on-prem networks - Supports transitive routing and scalable network design - Ideal for large, multi-VPC, multi-account architectures
VPC Endpoints	- Connects VPC privately to AWS services without internet - Two types: Interface (ENI) and Gateway (S3, DynamoDB)
Site-to-site VPN	- A VPN connection between an on-premises network and AWS VPC - Uses IPSec VPN for secure communication over the internet - Provides encrypted traffic between AWS and on-prem networks

Direct Connect	● Dedicated network connection between on-prem and AWS ● Offers more reliable, consistent performance than VPN ● Used for high-bandwidth, low-latency connections ● Ideal for hybrid cloud setups, reducing data transfer costs
DHCP Options Set	● Customizes DNS settings within VPC ● Used to set domain name and DNS servers
Elastic IP	● Static public IP address ● Can be associated with EC2 in a public subnet

AWS Shared Responsibility Model (IMP)

Responsibility	AWS (Security OF the cloud)	Customer (Security IN the cloud)
Physical Security	Protects Data centers, Hardware & Networking	N / A
Compute (Ec2, Lambda etc.)	Provides infrastructure & virtualization	Configures Security settings, Patches OS & Software
Storage (S3, EBS etc.)	Ensures Durability & Availability	Manages Data access, Encryption & Backups
Database (RDS, DynamoDB)	Maintains service uptime and patching	Manages Database access, Encryption & Backups
Networking (VPC, ELB, Route 53)	Provides Secure network infrastructure	Configures security groups, ACLs & VPNs
IAM & Access Control	Provides IAM framework and tools	Manages IAM roles, policies, MFA & least privilege
Application Security	Ensures that AWS services are patched	Secures applications, APIs & Authentications
Compliance & Auditing	Meets ISO, SOC, PCI, GDPR Standards	Ensures compliance with Regulatory requirements

AWS Web Security Services: WAF vs Shield

Features	AWS WAF (Web Application Firewall)	AWS Shield
Type of Threat	Web attacks (SQLi, XSS)	DDoS attacks (traffic overload)
Cost	Paid (based on rules)	Standard (Free), Advanced (Paid)
AWS Services Used	CloudFront, ALB, API Gateway	CloudFront, Route 53, ELB, EC2
Custom Rules	Yes, user-defined rules	No (automated protection)
Attack Type	Blocks bad requests	Absorbs & mitigates traffic
Use Case	Protect apps from SQLi, XSS, bots, rate limiting, country or IP blocking & header filtering	Defend against large-scale DDoS attacks on public-facing services (e.g., CloudFront, Route 53)

AWS Security and Compliance Services

Service / Concept	Description
Shared Responsibility Model	• AWS manages security **of** the cloud (infra, hardware, etc.) • Customer manages security **in** the cloud (data, IAM, apps)
AWS WAF & Shield	• **WAF:** Web Application Firewall to block malicious HTTP/S traffic • **Shield:** DDoS protection (Standard = default, Advanced = paid tier)
AWS Network Firewall	• Managed firewall for VPC traffic control • Filters outbound/inbound traffic at the subnet level
AWS Firewall Manager	• Centralized management for firewall rules (WAF, Shield, etc.) • Applies policies across accounts in AWS Organizations
Penetration Testing	• Customers can test certain services with prior approval • AWS has a list of services allowed without permission (EC2, RDS, etc.)
AWS KMS & CloudHSM	• **KMS:** Managed key service for encryption • **CloudHSM:** Hardware-backed key management for compliance-heavy needs
AWS Certificate Manager (ACM)	• Manages and provisions SSL/TLS certificates • Used to secure websites and APIs
Secrets Manager	• Securely stores and rotates sensitive data (API keys, DB passwords) • Offers built-in rotation, audit logging
AWS Artifact	• Portal to access AWS compliance reports, certifications, agreements • Use AWS Artifact to access official HIPAA compliance reports & certificates • Useful for audits and legal reviews
Amazon GuardDuty	• Threat detection service for accounts and workloads • Detects anomalies, suspicious API calls, IPs, etc.
Amazon Inspector	• Scans EC2, Lambda, and container images for vulnerabilities • Identifies software flaws and CVEs
AWS Config	• Tracks configuration changes in AWS resources • Helps enforce compliance using rules and history • Use AWS Config to monitor and enforce HIPAA-compliant configurations.

Amazon Macie	- Uses ML to detect sensitive data (like PII) in S3 - Useful for GDPR/Compliance audits
AWS Security Hub	- Central dashboard to monitor multiple security services - Integrates with Guard Duty, Inspector, Macie, etc.
Amazon Detective	- Investigates security findings using logs and visualizations - Helps analyze root cause of Guard Duty alerts
AWS Abuse	- Handles Abuse reports (spam, port scanning, DoS from AWS IPs) - Public-facing response team for AWS IP misuse
Root User Privileges	- Full access to all services and billing - Should be avoided for daily use - Enable MFA and secure credentials
IAM Access Analyzer	- Identifies resources shared outside your account - Helps detect unintended public or cross-account access

AWS AI/ML Services

Service	Description
Recognition	● Image and video analysis ● DO NOT RESIZE THE IMAGE ● Detects faces, objects, unsafe content, celebrities, etc.
Transcribe	● Converts Speech to Text using automatic speech recognition (ASR) ● Used for transcription of audio/video files
Polly	● Text-to-speech (TTS) service ● Converts text into natural-sounding speech ● Supports multiple languages and voices
Lex	● Build conversational interfaces (Chatbots) ● Powers Alexa and integrates with Lambda ● Includes speech-to-text and natural language understanding
Amazon Connect (cloud call center)	● Cloud-based contact center service ● Includes ML-based speech analytics, chatbots, and call routing
Translate	● Neural machine translation service ● Real-time language translation for websites, apps, or docs
Comprehend	● Natural language processing (NLP) ● Extracts key phrases, sentiment, entities, and language
SageMaker	● End-to-end ML platform ● Build, train, and deploy ML models at scale ● Supports notebooks, autoML, and built-in algorithms
Forecast	● Time series forecasting using ML ● Predicts future demand, revenue, or usage patterns ● Based on the same tech used at Amazon.com
Kendra	● Intelligent search engine ● Searches across documents, wikis, FAQs using natural language
Personalize	● Real-time Personalization & Recommendation engine ● Used for suggesting products, movies, content, etc

Textract	● Extracts text and data from scanned documents and forms ● Uses OCR and ML to detect key-value pairs and tables

AWS Multi-Account Mgmt. and Governance Services

Service	Description
AWS Organizations	● Manage multiple AWS accounts from a single place ● Enables consolidated billing and policy-based control ● Organizes accounts into Organizational Units (OUs)
Service Control Policies (SCPs)	● Apply permission boundaries across AWS accounts in the organization ● Can deny or allow actions regardless of individual IAM policies ● Does not grant permissions, only restricts
Consolidated Billing	● Combines billing for multiple accounts under one payer account ● Shares volume discounts and savings plans across accounts ● Helps simplify payment and cost tracking
AWS Control Tower	● Pre-configures and automates setup of secure multi-account AWS environments. ● Uses Guardrails (SCPs + AWS Config rules) for governance ● Best for setting up a governed landing zone quickly
AWS Service Catalog	● Lets admins create and manage approved AWS service templates ● Ensures consistent deployment of apps, services, and stacks ● Helps enforce compliance and cost controls
AWS Compute Optimizer	● Recommends optimal compute resources based on Usage patterns ● Supports EC2, Lambda, EBS, and Auto Scaling groups ● Helps reduce costs and improve performance

AWS Account Mgmt. Summary

Best Practices	Description
Use Multi-Factor Authentication (MFA)	Enable MFA for all AWS accounts, especially root and privileged IAM users.
Use IAM Roles Instead of Root User	● Avoid using the root user for day-to-day tasks ● Create IAM roles with necessary permissions
Create Least-Privilege IAM Policies	Grant only the permissions needed for users and roles to perform their tasks.
Use IAM Groups	Assign users to IAM groups with appropriate policies for consistent management
Enable CloudTrail & Monitor Logs	Enable CloudTrail to log API calls and monitor logs for suspicious activities
Review and Rotate Access Keys Regularly	Regularly rotate IAM access keys to maintain security
Use AWS Organizations for Account Mgmt.	Manage multiple AWS accounts using AWS Organizations to set up organizational units
Use AWS Config for Resource Compliance	Enable AWS Config to track and ensure compliance with internal and external policies
Protect Access to Console and API	Apply IP restrictions and secure protocols (HTTPS) for API access
Monitor Billing & Usage	Use AWS Budgets and Cost Explorer to monitor costs and set alarms for unexpected spikes
Implement Guardrails with AWS Control Tower	Use AWS Control Tower to set up and manage multi-account environments with best practices
Back Up Critical Data	Regularly back up critical data to Amazon S3 and RDS to ensure disaster recovery

AWS Cost Mgmt. and Billing Tools

Service / Tool	Description
AWS Cost Explorer	● Visual tool to view and analyze AWS spending over time ● Helps you filter by service, account, region, etc. ● Useful for identifying trends and usage patterns
AWS Budgets	● Set custom budgets for cost, usage, RI/SP utilization ● Sends alerts via email or SNS when thresholds are crossed ● More proactive than Cost Explorer
AWS Cost and Usage Report (CUR)	● Most Detailed billing report available ● Provides line-item level data for all AWS usage and costs ● Used for external BI tools and deep cost analysis
AWS Pricing Calculator	● Estimates costs before you launch services ● Helps plan architecture within budget ● Great for proof of concept or cost comparison ● Also called “Estimating Costs in the Cloud”
AWS Cost Anomaly Detection	● Monitors for unusual spend patterns using ML ● Sends alerts when anomalies are detected ● Helps prevent unexpected billing spikes
AWS Cost Categories	● Lets you Group and Label accounts or usage types logically ● Organize costs by business unit, project, or team ● Works with Budgets, Cost Explorer, and CUR
AWS Trusted Advisor	● Provides real-time best practice checks ● Includes cost optimization, security, and fault tolerance ● Some checks are free, more available with Business/Enterprise support
AWS Organizations (Consolidated Billing)	● Combines billing across multiple accounts ● Shares volume discounts and savings plans ● Simplifies invoice management

AWS Pricing Models for Different Services

AWS Service’s	Pricing Model	Details
Ec2	On-Demand, RIs, Spot, Savings	Pay per sec/min. Reserved & Spot = Discounts
S3	Pay-as-You-Go	Based on storage class & data retrieval
Lambda	Pay-per-Execution	Billed on requests & execution time (GB-sec)
RDS	On-Demand, Reserved Instances	Pay per hour, RIs offer savings
DynamoDB	Pay-per-Request, On-Demand	Charged per read/write request or capacity
VPC	Free (Basic)	NAT Gateway & VPN incur costs
Cloudfront	Pay-as-You-Go	Based on data transfer & requests
EBS	Pay-as-You-Go	Per GB/month + I/O operations
ELB	Pay-as-You-Go	Billed per LCU (Load Capacity Unit)
SNS & SQS	Pay-per-Request	Based on API calls & data transfer
Step Function	Pay-per-Transition	Billed per workflow state transition
Route 53	Pay-per-Hosted-Zone, Queries	Billed for hosted zones & DNS queries
Glue	Pay-per-Second	Charged per Data Processing Unit (DPU)
Redshift	On-Demand, Reserved Instances	Pay per hour for clusters. RIs save costs
Fargate	Pay-per-CPU/Memory Usage	Based on allocated CPU & memory for containers
Secret Manager	Pay per Secret	Charges per active secret & API requests
KMS	Pay per API request	Charges for key creation, usage, API calls

AWS Identity Services & Authentication Tools

Service / Tool	Description
AWS Security Token Service (STS)	• Provides Temporary Security Credentials for IAM or federated users • Useful for cross-account access, identity federation, CLI sessions • Credentials are short-term and automatically expire
Amazon Cognito	• Manages user sign-up, sign-in, and access for web & mobile apps • Supports social login (Google, Facebook), SAML, and custom IDPs • Issues JWT Tokens for Authentication
AWS Directory Service	• Provides Microsoft Active Directory in the AWS Cloud • Used for integrating with on-prem AD or running AD-dependent apps • Supports user authentication for Windows workloads
AWS IAM Identity Center (formerly AWS SSO)	• Centralized Access Mgmt. across multiple AWS accounts • Integrates with corporate directories (like AD, Okta) • Assign users/groups access to accounts, roles, and apps

Other Remaining AWS Services

1) Compute and Application Services

Service	Description
Amazon Workspace	Managed Desktop computing service for secure, scalable, and flexible access to Windows and Linux desktops
Appstream 2.0	Fully managed application streaming service for delivering desktop applications to users on any device
Appsync	Managed GraphQL service for building scalable, real-time, and offline-enabled mobile and web applications
Amplify	Development platform for building, deploying, and managing scalable mobile and web applications
Wave length	extends AWS infrastructure to the edge of 5G networks, allowing developers to build ultra-low latency applications by deploying compute and storage closer to mobile devices.
Step Function	Serverless orchestration service for coordinating the components of distributed applications and microservices

2) IoT & Device Services

Service	Description
AWS IoT Core	Managed cloud service for securely connecting, managing, and analyzing data from IoT devices
AWS Device Farm	Cloud-based testing service for testing and debugging mobile and web applications on real devices

3) Media & Storage Services

Service	Description
AWS Elastic Transcoder	Media transcoding service for converting media files into different formats
AWS DataSync	Data transfer service for securely and efficiently transferring data between on-premises storage and AWS storage services

4) Migration & Transfer Services

Service	Description
AWS Migration Hub	Central location for tracking the progress of application migrations.
AWS Application Discovery Service	Service for discovering and inventorying on-premises applications
AWS Application Migration Service	Service for migrating on-premises applications to AWS
AWS Migration Evaluator	Tool for evaluating the complexity and cost of migrating on-premises applications to AWS
Cloud Migration Strategies - The 7Rs	Framework for migrating applications to the cloud, including ● Rehost ● Refactor ● Revise ● Rebuild ● Replace ● Retire ● Retain

5) Backup and Disaster Recovery Services

Service	Description
AWS Backup	Fully managed Backup service for protecting data across AWS services
Disaster Recovery Strategies	Framework for Designing and Implementing disaster recovery plans
AWS Elastic Disaster Recovery (DRS)	Service for recovering and restoring EC2 instances and data volumes in the event of a disaster

6) Simulation and Testing Services

Service	Description
AWS Fault Injection Simulator	Service for Testing the resilience and reliability of applications by simulating real-world faults and failures

7) Space and Satellite Services

Service	Description
AWS Ground Station	Service for controlling satellite communications, downlinking data, and processing satellite data in the cloud

8) Marketing and Analytics Services

Service	Description
Amazon Pinpoint	Service for creating and managing targeted Marketing campaigns across multiple channels

AWS Cloud Adoption Framework (CAF)

The AWS Cloud Adoption Framework (CAF) helps organizations Plan their cloud adoption journey by identifying gaps and creating workstreams based on business and technical perspectives

Purpose of CAF

- Provides a structured approach to cloud transformation
- Aligns cloud strategies with business goals
- Identifies capabilities needed for successful cloud adoption

Perspective	Who's Involved?	Focus Area
Business	● Business managers ● Finance ● Strategy teams	Define business goals and benefits from cloud adoption
People	● HR ● Organizational change managers	● Skills ● Roles ● Training ● Change Mgmt.
Governance	● Risk managers ● Finance ● Compliance	● Budgeting ● Cloud Policy ● Compliance ● License Mgmt.
Platform	● Solution Architects ● Infrastructure teams	● Cloud infrastructure ● Services ● Automation
Security	● Security teams ● Risk compliance	● Identity ● Access Mgmt. ● Threat detection ● Data protection
Operations	● IT Operation ● Support teams	● Monitoring ● Incident Mgmt. ● Reliability ● Disaster recovery

AWS Well-Architected Framework (WAF)

The AWS Well-Architected Framework helps you Design & Review your Cloud Architecture based on best practices in Six key pillars.

Purpose of WAF

- Helps identify design flaws
- Provides a structured approach to assess architecture
- Ensures your workloads are secure, high-performing, resilient, and efficient

Six Pillars of WAF	Focus Area	Key AWS Service
Operational Excellence	Run workloads efficiently, monitor, and improve systems and operations over time	• CloudWatch • CloudTrail • Config • Systems Manager
Security	Protecting data, systems & assets through identity, detection, and response controls	• IAM & KMS • CloudTrail • GuardDuty • AWS WAF • AWS Shield
Reliability	Ensure workloads recover from failure and meet demand consistently	• Route 53 • ELB • Auto Scaling • RDS Multi-AZ • AWS Backup
Performance Efficiency	Use resources efficiently, scale based on demand, and adopt modern architectures	• Lambda • CloudFront • S3 • Aurora • Auto Scaling
Cost Optimization	Avoid unnecessary costs, use right pricing Avoid unnecessary costs, use right pricing	• Cost Explorer • Budgets • Trusted Advisor • S3 Lifecycle
Sustainability	Minimizing environmental impact and using	• Graviton

	energy-efficient and optimized workloads	• Lambda • Fargate • S3 Lifecycle • Compute Optimizer • CloudWatch • Carbon Footprint Tool

TOOLS Provided by AWS

Six Pillars of WAF	Focus Area
AWS Well-Architected Tool	Free tool in the AWS Console to review workloads using WAF principles
AWS Trusted Advisor	Helps with best practice checks (especially for Cost, Security, etc.)

Business Value of AWS

AWS delivers Business value through cost savings, agility, scalability, global reach, security, and innovation

In the table below, we discuss the **Key Benefits:**

Categories	Description	Example / Value
Cost Saving	● Pay-as-you-go pricing ● No upfront capital expenses	Avoid buying Physical servers, you just have to pay for what you use
Agility	Quickly deploy, test, and scale new ideas and applications	Launch a global app in hours instead of months
Governance	● Risk managers ● Finance ● Compliance	● Budgeting ● Cloud Policy ● Compliance ● License Mgmt.
Platform	● Solution Architects ● Infrastructure teams	● Cloud infrastructure ● Services ● Automation
Security	● Security teams ● Risk compliance	● Identity ● Access Mgmt. ● Threat detection ● Data protection
Operations	● IT Operation ● Support teams	● Monitoring ● Incident Mgmt. ● Reliability ● Disaster recovery

In the table below, we discuss the **Financial Benefits:**

Model	Description
CAPEX to OPEX	Avoid capital expenses (CAPEX); shift to operational (OPEX)
TCO Reduction	Total Cost of Ownership can reduce by up to 70% with AWS
Economies of Scale	AWS passes on savings from large-scale infrastructure

Tip For the Exam: If you see options like “reduces upfront costs”, “increases agility”, “global reach”, or “enables innovation” they all refer to the business value of AWS.

7 R’s of AWS Migration (IMP)

Strategy	What It Means	Example	Easy Way to Remember
Rehost	Move apps to AWS without changes	Move a VM from on-prem to EC2 as it is	Pick it up and move it (Lift & shift)
Replatform	Move to AWS with some optimizations and no code changes	Move database to RDS instead of EC2	Lift, tweak, shift
Repurchase	Switch to a new product, often SaaS	Replace on-prem CRM with Salesforce	Buy instead of build
Refactor / Re-Architect	Redesign the app for the cloud	Break monolith into microservices on ECS	Change the engine, not the car
Retire	Remove apps you no longer need	Shut down a legacy reporting server	Let it go
Retain	Keep the app on-prem for now	Delay migration for regulatory reasons	Keep it where it is
Relocate	Move entire virtual environments to AWS without app-level changes	VMware Cloud on AWS	Move the data center as it is

AWS CI/CD Services

Tool	Category	Purpose	Key Features
AWS CodeCommit	Source Control	Fully managed Git-based repository service	• Secure private Git repositories • Encrypted at rest & in transit • Integrates with IAM
AWS CodeBuild	Build Service	Fully managed build and test service	• Compiles source code • Runs tests • Produces build artifacts
AWS CodeDeploy	Deployment	Automates application deployments	• Deploys to EC2, Lambda, ECS • Supports rolling & blue/green deployments
AWS CodePipeline	CI/CD Orchestration	Automates end-to-end CI/CD pipelines	• Connects CodeCommit, CodeBuild, CodeDeploy • Visual workflow

Easy Memory Trick:

[C → B → D → P]

- CodeCommit → Store code
- CodeBuild → Build & test
- CodeDeploy → Deploy application
- CodePipeline → Automate everything

Developer Tools for Managing AWS Services

Tool	Category	Purpose	Key Features
AWS Software Developer Kit (SDK)	Development Tool	Access AWS services using language-specific APIs	- Supports multiple languages (Java, Python, JS, etc.) - Handles authentication, retries, serialization - Simplifies coding
AWS Management Console	Web Interface	Manage AWS services via a web-based UI	- Browser-based - Visual dashboards - Easy service discovery
AWS Command Line Interface (CLI)	Command-Line Tool	Manage AWS services using commands and scripts	- Unified command tool - Automates tasks via scripts - Works across services
Integrated Development Environments (IDE)	Development Environment	Write, test, and debug code efficiently	- Code editor, debugger, build tools - Example: AWS Cloud9