Admissions & Support:
support@mockattempt.com
Phone / WhatsApp: +91 98765 43210
Course Code: AWSC • **Updated:** 04 Nov 2026
https://mockattempt.com/courses/aws-training

AWSC • CLOUD COMPUTING & AWS SOLUTIONS ARCHITECTURE **BEGINNER TO SENIOR AWS CLOUD ARCHITECT**

100% LIVE INDUSTRY EXPERT COHORT + INTERNSHIP LETTER

AWS Training

Master Amazon Web Services from Global Infrastructure, IAM Least-Privilege Policies & STS Roles to S3 Storage Classes, CloudFront OAC, EC2/EBS, ALB, Multi-AZ VPC/NAT, RDS, DynamoDB, Lambda, API Gateway & SQS — with Python Boto3 programs, 244 Topic Mock Tests (10,440 MCQs), and Spoken Voice + MockAttempt AI Interviews at ₹4,299 (Incl. 18% GST).

Duration: 15–17 Weeks • Live + Hands-On Projects (210 Hrs) • **Modules:** 16 • **Live Topics:** 42 • **Subtopics:** 336+ • **Labs & Tasks:** 84 • **Mocks / AI Vivas:** 244 / 244 • **Projects:** 16 Module + 1 Live Client Prod

All-Inclusive Direct Fee (Incl. 18% GST • Zero Hidden Charges): Total Payable: ₹4,299 (Net Tuition ₹3,643 + Included 18% GST ₹656) • **Split Option:** 2 × ₹2,150 Split • **Effective ROI:** ~₹20/Guided Hr (~₹102/Live Topic)

Faculty: 100% Active Working IT Professionals • **Live Batch Tracks (7:00 AM – 8:00 PM IST Flexible Slots):** Weekday Live Online Batch (Mon – Fri • 1 Hr/Day • 7:00 AM – 8:00 PM) (Monday to Friday (1 Hour Live Session / Day) • 07:00 AM – 08:00 PM IST (Students Can Choose Any 1-Hour Slot Based on Trainer Availability) • 18–20 Weeks Fast-Track) | Weekend Live Online Batch (Sat – Sun • 2 Hrs/Day • 7:00 AM – 8:00 PM) (Saturday & Sunday (2 Hours Live Session / Day) • 07:00 AM – 08:00 PM IST (Students Can Choose Any 2-Hour Slot Based on Trainer Availability) • 20–22 Weeks Weekend Track)

✓ **Trainer LeetCode & HackerRank Mentorship (Python 3.12 (Boto3 SDK & AWS Lambda)): Live trainer support to crack LeetCode & HackerRank problems, optimize complexity, and handle edge cases.**

Core Technologies & Engineering Stack: AWS IAM, STS & KMS • Amazon S3 & CloudFront CDN • Amazon EC2, EBS & ALB • Amazon VPC, Subnets & NAT Gateway • Amazon RDS Multi-AZ & DynamoDB • AWS Lambda, API Gateway & SQS • Python 3.12 & Boto3 SDK

1. PROGRAM MASTERY OUTCOMES & ELIGIBILITY

End-to-End Engineering Outcomes:

1. Design fault-tolerant Multi-AZ AWS architectures across Regions, Availability Zones, Edge Locations, and isolated VPC public/private subnets
2. Enforce least-privilege security using IAM JSON policies, STS AssumeRole temporary credentials, stateful Security Groups, and stateless NACLs
3. Optimize cloud storage, relational/NoSQL databases, and serverless event-driven pipelines across S3, CloudFront, RDS, DynamoDB, Lambda, API Gateway, and SQS
4. Clear 244 Timed Mock Exams (10,000 AWS MCQs) and 200 Spoken Voice + MockAttempt AI AWS Solutions Architect Interviews
5. Custom Multi-AZ VPC networking: public/private/isolated subnets, Route Tables, NAT Gateways, NACLs, and Security Groups.
6. Crack LeetCode and HackerRank problems in Python 3.12 (Boto3 SDK & AWS Lambda) with live trainer mentorship on algorithmic/query patterns, edge cases, and optimal complexity.
7. Zero-trust IAM governance: IAM Roles, STS temporary credentials, KMS encryption, and AWS Organizations SCPs.
8. Elastic compute & storage: ALB + EC2 Auto Scaling, ECS Fargate, Lambda serverless, S3 lifecycle tiers, and CloudFront.

Eligibility & Prerequisites: Basic Linux command-line, networking (IP/HTTP), and introductory Python familiarity — teaches AWS cloud architecture from scratch • Any laptop/PC with Python 3.10+, AWS CLI v2 / Boto3, and a Free-Tier AWS account (or LocalStack) • No prior AWS industry experience required — Module 1 starts from environment setup, core mental models, and plain-English fundamentals. • Ideal for B.Tech / BCA / MCA / B.Sc IT students, non-CS beginners, career switchers, and working developers upgrading to 2026 production standards. • Commitment to attend live Weekday or Weekend classes, complete daily hands-on lab checklists, and practice topic-wise Mock Tests & AI Interviews.

2. DETAILED ZERO-TO-EXPERT MODULE, TOPIC & SUBTOPIC SYLLABUS (16 MODULES • 42 LIVE TOPICS • 336+ SUBTOPICS)

Progressive Zero-to-Expert Mastery Stages: Modules 1–4: Stage 1: Day-1 Foundations & Core Mental Models (8 Topics / 64 Subtopics) • Modules 5–8: Stage 2: Applied Engineering, Data & API Workflows (8 Topics / 64 Subtopics) • Modules 9–12: Stage 3: Internals, Concurrency, Security & Performance (12 Topics / 96 Subtopics) • Modules 13–16: Stage 4: Production Architecture, Testing, CI/CD & Expert Mastery (14 Topics / 112 Subtopics)

MODULE 1 OF 16 • STAGE 1 • 11 LIVE + LAB HOURS • 2 TOPICS (16 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

Module 1: AWS Cloud Foundations, Shared Responsibility & Global Infrastructure

Understand on-demand cloud economics, the AWS Shared Responsibility Model, AWS Well-Architected pillars, and Multi-AZ / Multi-Region physical topology.

Objectives: ✓ AWS Cloud Computing Foundations — Shared Responsibility Model & Well-Architected Pillars — How AWS replaces upfront CapEx data centers with elastic OpEx cloud primitives governed by the Shared Responsibility Model and the 6 Well-Architected Framework pillars. • ✓ AWS Global Infrastructure — Regions, Availability Zones, Local Zones & Edge Locations — Deep dive into geographically isolated AWS Regions, redundant Availability Zones linked via low-latency private fiber (< 2 ms), and global Edge PoPs. • ✓ Build, verify, and ship Module Capstone: Multi-AZ AWS Well-Architected Readiness & Region Latency Evaluator.

1.1 AWS Cloud Computing Foundations — Shared Responsibility Model & Well-Architected Pillars

[50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: Shared Responsibility Model & CapEx vs OpEx cloud economics • Core Syntax, CLI & Production APIs: boto3.Session() + botocore.config.Config(retries={'mode': 'adaptive'}) • Deep-Dive Mechanics: Well-Architected 6 pillars & Boto3 Session and adaptive retries • Advanced Patterns: IaaS vs Managed Services & Elastic horizontal scaling • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: Shared Responsibility & Adaptive Boto3 Client Config Verifier • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for AWS Cloud Computing Foundations

Purpose: To understand and apply AWS Cloud Computing Foundations — Shared Responsibility Model & Well-Architected Pillars in real AWS systems — specifically: How AWS replaces upfr...

What Topic Is Next → Next Topic 1.2: AWS Global Infrastructure — Regions, Availability Zones, Local Zones & Edge Locations

1.2 AWS Global Infrastructure — Regions, Availability Zones, Local Zones & Edge Locations

[50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: AWS Region data sovereignty & Availability Zone fault domain • Core Syntax, CLI & Production APIs: ec2_client.describe_availability_zones(Filters=[{'Name': 'state', 'Values': ['available']}]) • Deep-Dive Mechanics: AZ ID cross-account mapping & Low-latency Metro fiber mesh • Advanced Patterns: Edge Locations and Regional Edge Caches & Multi-AZ high availability • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: Multi-AZ Quorum & AZ-ID Mapping Validator • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for AWS Global Infrastructure

Purpose: To understand and apply AWS Global Infrastructure — Regions, Availability Zones, Local Zones & Edge Locations in real AWS systems — specifically: Deep dive into geographi...

What Topic Is Next → Module 1 Capstone (Multi-AZ AWS Well-Architected Readiness & Region Latency Evaluator) → Next Topic 2.1: AWS IAM Architecture — Users, Groups, JSON Policies, Conditions & Explicit Deny Precedence

Module 1 Hands-On Capstone Project (100 Marks): Multi-AZ AWS Well-Architected Readiness & Region Latency Evaluator — Build a Python Boto3 architecture utility that audits workload placement across at least 3 Availability Zones in ap-south-1, maps Shared Responsibility controls per service tier, and flags s...

MODULE 2 OF 16 • STAGE 1 • 12 LIVE + LAB HOURS • 2 TOPICS (16 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

Module 2: AWS Identity and Access Management (IAM), Policy Evaluation & STS Roles

Master IAM identities, JSON policy evaluation precedence, Permission Boundaries, MFA conditions, and short-lived STS AssumeRole credentials.

Objectives: ✓ AWS IAM Architecture — Users, Groups, JSON Policies, Conditions & Explicit Deny Precedence — How AWS IAM authenticates principals and evaluates identity-based, resource-based, and SCP JSON policy documents on every signed SigV4 API request. • ✓ AWS IAM Roles & STS — Trust Policies, Instance Profiles, OIDC Federation & AssumeRole — Replacing static long-lived secrets with dynamic AWS Security Token Service (STS) ASIA temporary credentials for EC2, Lambda, GitHub OIDC, and cross-account access. • ✓ Build, verify, and ship Module Capstone: Least-Privilege IAM Policy Linter & STS Temporary Session Broker.

2.1 AWS IAM Architecture — Users, Groups, JSON Policies, Conditions & Explicit Deny Precedence

[50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: Explicit Deny precedence & Implicit Deny default posture • Core Syntax, CLI & Production APIs: iam_client.simulate_principal_policy() + JSON Policy (Version: "2012-10-17") • Deep-Dive Mechanics: SigV4 request signing & Version 2012-10-17 JSON schema • Advanced Patterns: IAM Condition operators & Permissions Boundary intersection • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: AWS IAM Policy Precedence & Wildcard Auditor • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for AWS IAM Architecture

Purpose: To understand and apply AWS IAM Architecture — Users, Groups, JSON Policies, Conditions & Explicit Deny Precedence in real AWS systems — specifically: How AWS IAM authent...

What Topic Is Next → Next Topic 2.2: AWS IAM Roles & STS — Trust Policies, Instance Profiles, OIDC Federation & AssumeRole

2.2 AWS IAM Roles & STS — Trust Policies, Instance Profiles, OIDC Federation & AssumeRole

[50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: STS AssumeRole temporary tokens & ASIA vs AKIA access key prefixes • Core Syntax, CLI & Production APIs: sts_client.assume_role(RoleArn=..., RoleSessionName=..., DurationSeconds=3600) • Deep-Dive Mechanics: Role Trust Policy vs Permission Policy & EC2 Instance Profile and IMDSv2 • Advanced Patterns: Confused Deputy ExternalId guard & OIDC Web Identity Federation • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: STS Trust Policy & Confused-Deputy ExternalId Validator • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for AWS IAM Roles

Purpose: To understand and apply AWS IAM Roles & STS — Trust Policies, Instance Profiles, OIDC Federation & AssumeRole in real AWS systems — specifically: Replacing static long-li...

What Topic Is Next → Module 2 Capstone (Least-Privilege IAM Policy Linter & STS Temporary Session Broker) → Next Topic 3.1: Amazon S3 Internals — Flat Key-Object Namespace, Strong Consistency, Versioning & Presigned URLs

Module 2 Hands-On Capstone Project (100 Marks): Least-Privilege IAM Policy Linter & STS Temporary Session Broker — Create an IAM security auditor and STS credential helper in Python that flags wildcard Action/Resource grants, verifies MFA Condition blocks, and formats short-lived STS session credentials.

MODULE 3 OF 16 • STAGE 1 • 12 LIVE + LAB HOURS • 2 TOPICS (16 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

Module 3: Amazon S3 Object Storage, Strong Consistency & Storage Classes

Architect durable object lakes with Amazon S3, Block Public Access, Pre-Signed URLs, KMS SSE encryption, and tiered S3 Storage Classes.

Objectives: ✓ Amazon S3 Internals — Flat Key-Object Namespace, Strong Consistency, Versioning & Presigned URLs — How Amazon S3 delivers 99.999999999% (11 nines) durability across >= 3 AZs with strong read-after-write consistency, versioning, and presigned direct-to-S3 transfers. • ✓ Amazon S3 Storage Classes — Standard, Intelligent-Tiering, Standard-IA, One Zone-IA & Glacier — Selecting the optimal S3 Storage Class by balancing per-GB storage rate, minimum billable object size (128 KB), minimum storage duration (30/90/180 days), and retrieval latency. • ✓ Build, verify, and ship Module Capstone: Secure S3 Presigned Upload Service & Storage Tier Cost Calculator.

3.1 Amazon S3 Internals — Flat Key-Object Namespace, Strong Consistency, Versioning & Presigned URLs

[50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: 11-nines cross-AZ durability & Strong read-after-write consistency • Core Syntax, CLI & Production APIs: s3_client.put_object(..., ServerSideEncryption='aws:kms') + generate_presigned_url() • Deep-Dive Mechanics: Block Public Access four settings & Presigned URL offloaded uploads • Advanced Patterns: Multipart Upload above 100 MB & SSE-S3 vs SSE-KMS encryption • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: S3 Multipart Chunk Planner & Prefix Throughput Estimator • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for Amazon S3 Internals

Purpose: To understand and apply Amazon S3 Internals — Flat Key-Object Namespace, Strong Consistency, Versioning & Presigned URLs in real AWS systems — specifically: How Amazon S3...

What Topic Is Next → Next Topic 3.2: Amazon S3 Storage Classes — Standard, Intelligent-Tiering, Standard-IA, One Zone-IA & Glacier

3.2 Amazon S3 Storage Classes — Standard, Intelligent-Tiering, Standard-IA, One Zone-IA & Glacier

[50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: S3 Standard vs Standard-IA & S3 Intelligent-Tiering zero retrieval fee • Core Syntax, CLI & Production APIs: StorageClass='INTELLIGENT_TIERING' | 'STANDARD_IA' | 'GLACIER_IR' | 'DEEP_ARCHIVE' • Deep-Dive Mechanics: 128 KB minimum billable size rule & 30/90/180-day minimum duration charge • Advanced Patterns: Glacier Instant vs Flexible vs Deep Archive & One Zone-IA single-AZ trade-off • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: S3 Storage Class Advisor for Object Size & Retention Profile • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for Amazon S3 Storage Classes

Purpose: To understand and apply Amazon S3 Storage Classes — Standard, Intelligent-Tiering, Standard-IA, One Zone-IA & Glacier in real AWS systems — specifically: Selecting the op...

What Topic Is Next → Module 3 Capstone (Secure S3 Presigned Upload Service & Storage Tier Cost Calculator) → Next Topic 4.1: S3 Lifecycle Management — Waterfall Transitions, Noncurrent Version Expiry & Multipart Cleanup

Module 3 Hands-On Capstone Project (100 Marks): Secure S3 Presigned Upload Service & Storage Tier Cost Calculator — Implement a Python S3 storage utility that enforces SSE-KMS headers on uploads, computes optimal storage classes based on object access frequency and retention days, and prevents public ACL...

MODULE 4 OF 16 • STAGE 1 • 11 LIVE + LAB HOURS • 2 TOPICS (16 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

Module 4: S3 Lifecycle Automation & Global Edge Delivery with CloudFront CDN

Automate object tier transitions, abort incomplete multipart uploads, and accelerate global static/dynamic delivery securely via CloudFront Origin Access Control (OAC).

Objectives: ✓ S3 Lifecycle Management — Waterfall Transitions, Noncurrent Version Expiry & Multipart Cleanup — Automating zero-touch object tiering, compliance expiration, version pruning, and orphaned multipart chunk cleanup via S3 Lifecycle Rules. • ✓ Amazon CloudFront CDN — Edge Caching, Origin Access Control (OAC), Signed URLs & Origin Shield — Accelerating static assets, HLS video streams, and dynamic APIs across global Edge PoPs while shielding private S3 buckets via Origin Access Control (OAC). • ✓ Build, verify, and ship Module Capstone: Private S3 + CloudFront OAC Global Asset Delivery Blueprint.

4.1 S3 Lifecycle Management — Waterfall Transitions, Noncurrent Version Expiry & Multipart Cleanup

[50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: Lifecycle waterfall transition order & 30-day Standard-to-IA minimum • Core Syntax, CLI & Production APIs: s3_client.put_bucket_lifecycle_configuration(Bucket=..., LifecycleConfiguration={'Rules': [.... • Deep-Dive Mechanics: NoncurrentVersionExpiration pruning & Delete Marker cleanup • Advanced Patterns: AbortIncompleteMultipartUpload rule & Prefix and object-tag filtering • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: S3 Lifecycle Rule Validator & Orphaned Multipart Guard • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for S3 Lifecycle Management

Purpose: To understand and apply S3 Lifecycle Management — Waterfall Transitions, Noncurrent Version Expiry & Multipart Cleanup in real AWS systems — specifically: Automating zero...

What Topic Is Next → Next Topic 4.2: Amazon CloudFront CDN — Edge Caching, Origin Access Control (OAC), Signed URLs & Origin Shield

4.2 Amazon CloudFront CDN — Edge Caching, Origin Access Control (OAC), Signed URLs & Origin Shield

[50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: Origin Access Control (OAC) & Cache Policy vs Origin Request Policy • Core Syntax, CLI & Production APIs: cloudfront_client.create_origin_access_control() + create_invalidation() • Deep-Dive Mechanics: Cache hit ratio optimization & CloudFront Signed URLs vs Signed Cookies • Advanced Patterns: Origin Shield caching layer & AWS WAF at the CloudFront edge • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: CloudFront OAC S3 Bucket Policy Scoper • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for Amazon CloudFront CDN

Purpose: To understand and apply Amazon CloudFront CDN — Edge Caching, Origin Access Control (OAC), Signed URLs & Origin Shield in real AWS systems — specifically: Accelerating st...

What Topic Is Next → Module 4 Capstone (Private S3 + CloudFront OAC Global Asset Delivery Blueprint) → Next Topic 5.1: Amazon EC2 Architecture — Nitro Instances, Purchasing Models, User Data & IMDSv2 Hardening

Module 4 Hands-On Capstone Project (100 Marks): Private S3 + CloudFront OAC Global Asset Delivery Blueprint — Generate a compliant S3 Lifecycle configuration (including incomplete multipart cleanup and noncurrent version expiration) paired with a CloudFront OAC bucket policy that blocks direct S3 UR...

MODULE 5 OF 16 • STAGE 2 • 12 LIVE + LAB HOURS • 2 TOPICS (16 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

Module 5: Elastic Compute Cloud (EC2), Purchasing Options & Elastic Block Store (EBS)

Size Nitro-based EC2 instance families, optimize On-Demand/Savings Plans/Spot purchasing, Harden IMDSv2 metadata, and provision gp3/io2 EBS volumes.

Objectives: ✓ Amazon EC2 Architecture — Nitro Instances, Purchasing Models, User Data & IMDSv2 Hardening — Provisioning virtual servers across General Purpose, Compute/Memory Optimized, and AWS Graviton ARM families while securing instance metadata via IMDSv2. • ✓ Amazon EBS Block Storage — gp3 vs io2 Volumes, AZ Affinity, Snapshots & KMS Encryption — Persistent network-attached block storage for EC2 with independent IOPS/throughput scaling on gp3, mission-critical 64k+ IOPS on io2, and incremental S3-backed snapshots. • ✓ Build, verify, and ship Module Capstone: EC2 Fleet Cost Optimizer & EBS gp3 Migration Planner.

5.1 Amazon EC2 Architecture — Nitro Instances, Purchasing Models, User Data & IMDSv2 Hardening

[50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: AWS Nitro hypervisor & On-Demand vs Savings Plans vs Spot • Core Syntax, CLI & Production APIs: ec2_client.run_instances(..., MetadataOptions={'HttpTokens': 'required'}) • Deep-Dive Mechanics: IMDSv2 session token protection & Instance Store ephemeral NVMe vs EBS • Advanced Patterns: User Data cloud-init bootstrap & Cluster vs Spread Placement Groups • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: EC2 Launch Template IMDSv2 & Fleet Blended Cost Calculator • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for Amazon EC2 Architecture

Purpose: To understand and apply Amazon EC2 Architecture — Nitro Instances, Purchasing Models, User Data & IMDSv2 Hardening in real AWS systems — specifically: Provisioning virtua...

What Topic Is Next → Next Topic 5.2: Amazon EBS Block Storage — gp3 vs io2 Volumes, AZ Affinity, Snapshots & KMS Encryption

5.2 Amazon EBS Block Storage — gp3 vs io2 Volumes, AZ Affinity, Snapshots & KMS Encryption

[50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: AZ-bound block volume scope & Gp3 decoupled IOPS and throughput • Core Syntax, CLI & Production APIs: ec2_client.create_volume(AvailabilityZone=..., VolumeType='gp3', Iops=6000, Encrypted=True) • Deep-Dive Mechanics: Io2 Block Express sub-ms durability & Incremental EBS Snapshots • Advanced Patterns: Fast Snapshot Restore (FSR) & DeleteOnTermination lifecycle flag • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: EBS gp3 Extra IOPS & Cross-AZ Attachment Guard • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for Amazon EBS Block Storage

Purpose: To understand and apply Amazon EBS Block Storage — gp3 vs io2 Volumes, AZ Affinity, Snapshots & KMS Encryption in real AWS systems — specifically: Persistent network-atta...

What Topic Is Next → Module 5 Capstone (EC2 Fleet Cost Optimizer & EBS gp3 Migration Planner) → Next Topic 6.1: AWS Security Groups — Stateful ENI Firewalls, Implicit Deny & SG-ID Chaining

Module 5 Hands-On Capstone Project (100 Marks): EC2 Fleet Cost Optimizer & EBS gp3 Migration Planner — Develop a Python capacity planner that splits baseline vs burst compute across Savings Plans and Spot Instances, enforces IMDSv2 launch template settings, and computes savings from migrating...

MODULE 6 OF 16 • STAGE 2 • 12 LIVE + LAB HOURS • 2 TOPICS (16 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

Module 6: Stateful Security Groups & Elastic Load Balancing (ALB / NLB)

Chain tier-to-tier Security Group references without brittle CIDRs and distribute Multi-AZ traffic cleanly via Application and Network Load Balancers.

Objectives: ✓ AWS Security Groups — Stateful ENI Firewalls, Implicit Deny & SG-ID Chaining — How Security Groups act as stateful virtual firewalls at the Elastic Network Interface (ENI) level supporting Allow-only rules and dynamic Security Group ID referencing. • ✓ Elastic Load Balancing — ALB vs NLB, Target Groups, Health Checks & X-Forwarded-For — Distributing Multi-AZ traffic across EC2 instances, ECS IP tasks, and Lambda functions via Layer-7 ALB host/path rules or Layer-4 static-IP NLB. • ✓ Build, verify, and ship Module Capstone: Zero-Trust 3-Tier Security Group Chain & ALB Target Health Verifier.

6.1 AWS Security Groups — Stateful ENI Firewalls, Implicit Deny & SG-ID Chaining

[50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: ENI-level stateful firewall & Allow-only rule evaluation • Core Syntax, CLI & Production APIs: ec2_client.authorize_security_group_ingress(GroupId=..., IpPermissions=[...]) • Deep-Dive Mechanics: Automatic return traffic tracking & Security Group ID chaining • Advanced Patterns: Default outbound allow-all vs lockdown & SSM Session Manager zero-inbound-port access • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: 3-Tier Security Group Chain Compliance Checker • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for AWS Security Groups

Purpose: To understand and apply AWS Security Groups — Stateful ENI Firewalls, Implicit Deny & SG-ID Chaining in real AWS systems — specifically: How Security Groups act as statef...

What Topic Is Next → Next Topic 6.2: Elastic Load Balancing — ALB vs NLB, Target Groups, Health Checks & X-Forwarded-For

6.2 Elastic Load Balancing — ALB vs NLB, Target Groups, Health Checks & X-Forwarded-For [50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: Layer-7 ALB vs Layer-4 NLB & Target Group Instance/IP/Lambda targets • Core Syntax, CLI & Production APIs: elbv2_client.create_load_balancer(Type='application') + create_target_group() • Deep-Dive Mechanics: X-Forwarded-For client IP preservation & Deregistration Delay connection draining • Advanced Patterns: Cross-Zone Load Balancing & Shallow vs deep health check endpoints • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: ALB Path-Based Listener Router & X-Forwarded-For Parser • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for Elastic Load Balancing

Purpose: To understand and apply Elastic Load Balancing — ALB vs NLB, Target Groups, Health Checks & X-Forwarded-For in real AWS systems — specifically: Distributing Multi-AZ traf...

What Topic Is Next → Module 6 Capstone (Zero-Trust 3-Tier Security Group Chain & ALB Target Health Verifier) → Next Topic 7.1: Amazon VPC Internals — CIDR Blocks, 5 Reserved Subnet IPs, Route Tables & Internet Gateways

Module 6 Hands-On Capstone Project (100 Marks): Zero-Trust 3-Tier Security Group Chain & ALB Target Health Verifier — Build an automated AWS security & load-balancing topology validator that verifies SG-ID chaining across ALB, App, and Database tiers and checks ALB Target Group deregistration delay and heal...

MODULE 7 OF 16 • STAGE 2 • 12 LIVE + LAB HOURS • 2 TOPICS (16 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

Module 7: Amazon VPC Networking, CIDR Subnetting, Gateways & VPC Endpoints

Design non-overlapping Multi-AZ VPC CIDRs, calculate AWS reserved subnet IPs, deploy per-AZ NAT Gateways, and eliminate NAT data charges using Gateway VPC Endpoints.

Objectives: ✓ Amazon VPC Internals — CIDR Blocks, 5 Reserved Subnet IPs, Route Tables & Internet Gateways — Building logically isolated software-defined cloud networks with non-overlapping RFC-1918 CIDRs, public/private route tables, VPC DNS (+2 IP), and Flow Logs. • ✓ AWS NAT Gateways & VPC Endpoints — Outbound-Only Private Egress, Multi-AZ HA & S3/DynamoDB Bypass — Enabling private-subnet outbound internet patching via managed NAT Gateways while slashing per-GB NAT fees with free Gateway VPC Endpoints for S3 and DynamoDB. • ✓ Build, verify, and ship Module Capstone: Multi-AZ VPC CIDR Calculator & NAT Cost-Elimination Planner.

7.1 Amazon VPC Internals — CIDR Blocks, 5 Reserved Subnet IPs, Route Tables & Internet Gateways [50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: RFC-1918 non-overlapping CIDR design & 5 AWS-reserved IPs per subnet • Core Syntax, CLI & Production APIs: ec2_client.create_vpc(CidrBlock='10.0.0.0/16') + create_subnet() + create_route_table() • Deep-Dive Mechanics: Public vs Private subnet route tables & Internet Gateway (IGW) 1:1 attachment • Advanced Patterns: VPC DNS resolver at base+2 IP & VPC Flow Logs ENI telemetry • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: VPC Subnet Usable Host & Reserved IP Inspector • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for Amazon VPC Internals

Purpose: To understand and apply Amazon VPC Internals — CIDR Blocks, 5 Reserved Subnet IPs, Route Tables & Internet Gateways in real AWS systems — specifically: Building logically...

What Topic Is Next → Next Topic 7.2: AWS NAT Gateways & VPC Endpoints — Outbound-Only Private Egress, Multi-AZ HA & S3/DynamoDB Bypass

7.2 AWS NAT Gateways & VPC Endpoints — Outbound-Only Private Egress, Multi-AZ HA & S3/DynamoDB Bypass [50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: NAT Gateway placement in Public Subnet & One NAT Gateway per AZ for fault isolation • Core Syntax, CLI & Production APIs: ec2_client.create_nat_gateway(SubnetId=public_subnet_id, AllocationId=eip_alloc) + create_vp... • Deep-Dive Mechanics: Outbound-only SNAT translation & Gateway VPC Endpoints (S3 & DynamoDB) • Advanced Patterns: Interface VPC Endpoints (AWS PrivateLink) & NAT per-GB data processing cost trap • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: Multi-AZ NAT Placement & S3 Gateway Endpoint Savings Calculator • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for AWS NAT Gateways

Purpose: To understand and apply AWS NAT Gateways & VPC Endpoints — Outbound-Only Private Egress, Multi-AZ HA & S3/DynamoDB Bypass in real AWS systems — specifically: Enabling pri...

What Topic Is Next → Module 7 Capstone (Multi-AZ VPC CIDR Calculator & NAT Cost-Elimination Planner) → Next Topic 8.1: Network ACL (NACL) vs Security Group — Stateless Subnet Filtering, Rule Ordering & Ephemeral Ports

Module 7 Hands-On Capstone Project (100 Marks): Multi-AZ VPC CIDR Calculator & NAT Cost-Elimination Planner — Write a Python networking tool using ipaddress that partitions a 10.20.0.0/16 VPC into 6 Multi-AZ subnets, computes exact usable host IPs (total minus 5 AWS-reserved addresses), and wires Ga...

MODULE 8 OF 16 • STAGE 2 • 12 LIVE + LAB HOURS • 2 TOPICS (16 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

Module 8: Stateless NACLs vs Security Groups & Managed Relational Databases (Amazon RDS)

Contrast stateless subnet Network ACLs (and ephemeral port ranges) against stateful Security Groups, and architect Multi-AZ Amazon RDS instances with Read Replicas.

Objectives: ✓ Network ACL (NACL) vs Security Group — Stateless Subnet Filtering, Rule Ordering & Ephemeral Ports — Comparing subnet-boundary stateless NACLs (ordered Allow/Deny rules + explicit ephemeral port rules) with ENI-boundary stateful Security Groups. • ✓ Amazon RDS Architecture — Multi-AZ Synchronous Failover, Read Replicas, PITR & RDS Proxy — Provisioning managed PostgreSQL/MySQL/Aurora databases in isolated DB Subnet Groups with automated Multi-AZ DNS failover, Point-in-Time Recovery, and RDS Proxy pooling. • ✓ Build, verify, and ship Module Capstone: Stateless NACL Simulator & Multi-AZ RDS High-Availability Blueprint.

8.1 Network ACL (NACL) vs Security Group — Stateless Subnet Filtering, Rule Ordering & Ephemeral Ports [50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: Stateless NACL vs stateful Security Group & Subnet boundary vs ENI boundary • Core Syntax, CLI & Production APIs: ec2_client.create_network_acl_entry(NetworkAclId=..., RuleNumber=100, RuleAction='deny', Egr... • Deep-Dive Mechanics: Ascending RuleNumber first-match wins & Ephemeral return ports 1024-65535 • Advanced Patterns: Explicit CIDR IP blacklisting via NACL & Custom NACL default deny-all posture • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: Stateless NACL Rule-Order & Ephemeral Return Packet Simulator • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for Network ACL (NACL) vs Security Group

Purpose: To understand and apply Network ACL (NACL) vs Security Group — Stateless Subnet Filtering, Rule Ordering & Ephemeral Ports in real AWS systems — specifically: Comparing s...

What Topic Is Next → Next Topic 8.2: Amazon RDS Architecture — Multi-AZ Synchronous Failover, Read Replicas, PITR & RDS Proxy

8.2 Amazon RDS Architecture — Multi-AZ Synchronous Failover, Read Replicas, PITR & RDS Proxy [50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: Multi-AZ synchronous standby failover & Asynchronous Read Replicas for SELECT scale • Core Syntax, CLI & Production APIs: rds_client.create_db_instance(..., MultiAZ=True, StorageEncrypted=True, PubliclyAccessible=F... • Deep-Dive Mechanics: Point-in-Time Recovery (PITR) transaction logs & Automated Backups vs manual DB Snapshots • Advanced Patterns: Amazon RDS Proxy connection pooling & PubliclyAccessible=False isolated subnet rule • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: Production RDS Multi-AZ Hardening Audit • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for Amazon RDS Architecture

Purpose: To understand and apply Amazon RDS Architecture — Multi-AZ Synchronous Failover, Read Replicas, PITR & RDS Proxy in real AWS systems — specifically: Provisioning managed...

What Topic Is Next → Module 8 Capstone (Stateless NACL Simulator & Multi-AZ RDS High-Availability Blueprint) → Next Topic 9.1: Amazon DynamoDB — Partition & Sort Keys, Query vs Scan, GSI/LSI, RCU/WCU & Streams

Module 8 Hands-On Capstone Project (100 Marks): Stateless NACL Simulator & Multi-AZ RDS High-Availability Blueprint — Implement a rule evaluator that tests numbered inbound/outbound stateless NACL rules (including ephemeral port return traffic) alongside an RDS Multi-AZ configuration validator.

MODULE 9 OF 16 • STAGE 3 • 12 LIVE + LAB HOURS • 2 TOPICS (16 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

Module 9: Serverless NoSQL & Compute — Amazon DynamoDB & AWS Lambda

Model single-digit-millisecond DynamoDB Partition/Sort keys without hot partitions and engineer fast, idempotent AWS Lambda functions with warm execution reuse.

Objectives: ✓ Amazon DynamoDB — Partition & Sort Keys, Query vs Scan, GSI/LSI, RCU/WCU & Streams — Delivering predictable single-digit millisecond latency at any scale using high-cardinality Partition Keys, B-tree Sort Keys, On-Demand/Provisioned capacity, and DynamoDB Streams. • ✓ AWS Lambda Internals — Firecracker Execution Lifecycle, Cold Starts, Concurrency & Idempotency — Event-driven serverless compute scaling from zero to thousands of concurrent microVMs with proportional memory-to-vCPU allocation (1,769 MB = 1 full vCPU) and max 15-minute execution. • ✓ Build, verify, and ship Module Capstone: Single-Table DynamoDB Capacity Planner & Idempotent AWS Lambda Processor.

9.1 Amazon DynamoDB — Partition & Sort Keys, Query vs Scan, GSI/LSI, RCU/WCU & Streams [50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: Partition Key hash + Sort Key range & Query vs full-table Scan trap • Core Syntax, CLI & Production APIs: table.query(KeyConditionExpression=Key('pk').eq(...) & Key('sk').begins_with(...)) • Deep-Dive Mechanics: Hot partition 3000 RCU / 1000 WCU cap & RCU 4KB and WCU 1KB capacity math • Advanced Patterns: Global Secondary Index (GSI) vs LSI & DynamoDB Streams + TTL expiration • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: DynamoDB RCU & WCU Provisioned Capacity Calculator • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for Amazon DynamoDB

Purpose: To understand and apply Amazon DynamoDB — Partition & Sort Keys, Query vs Scan, GSI/LSI, RCU/WCU & Streams in real AWS systems — specifically: Delivering predictable sing...

What Topic Is Next → Next Topic 9.2: AWS Lambda Internals — Firecracker Execution Lifecycle, Cold Starts, Concurrency & Idempotency

9.2 AWS Lambda Internals — Firecracker Execution Lifecycle, Cold Starts, Concurrency & Idempotency [50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: Firecracker microVM init vs invoke phase & Module-scope SDK client warm reuse • Core Syntax, CLI & Production APIs: def lambda_handler(event: dict, context) -> dict: ... # SDK init at global module scope • Deep-Dive Mechanics: 1769 MB = 1 full vCPU scaling rule & 900-second (15-min) max timeout • Advanced Patterns: Reserved vs Provisioned Concurrency & At-least-once async retry + DLQ/Destinations • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: Warm-Container Lambda Handler with Idempotency Guard • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for AWS Lambda Internals

Purpose: To understand and apply AWS Lambda Internals — Firecracker Execution Lifecycle, Cold Starts, Concurrency & Idempotency in real AWS systems — specifically: Event-driven se...

What Topic Is Next → Module 9 Capstone (Single-Table DynamoDB Capacity Planner & Idempotent AWS Lambda Processor) → Next Topic 10.1: Amazon API Gateway — HTTP vs REST vs WebSocket APIs, Authorizers, Throttling & 29s Limit

Module 9 Hands-On Capstone Project (100 Marks): Single-Table DynamoDB Capacity Planner & Idempotent AWS Lambda Processor — Build a DynamoDB RCU/WCU estimator paired with an AWS Lambda handler simulator that caches SDK clients at module scope and enforces conditional PutItem idempotency.

MODULE 10 OF 16 • STAGE 3 • 12 LIVE + LAB HOURS • 2 TOPICS (16 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

Module 10: Managed Front-Door APIs & Decoupled Messaging — API Gateway & Amazon SQS

Expose JWT/Cognito-protected serverless endpoints via Amazon API Gateway and buffer traffic spikes reliably with Amazon SQS Standard/FIFO queues, Long Polling, and Dead-Letter Queues.

Objectives: ✓ Amazon API Gateway — HTTP vs REST vs WebSocket APIs, Authorizers, Throttling & 29s Limit — Securing and throttling serverless APIs at scale using Cognito/JWT Authorizers, Usage Plans, Token-Bucket Burst Limits, and Lambda Proxy Integration. • ✓ Amazon SQS — Standard vs FIFO Queues, Visibility Timeout, Long Polling & Dead-Letter Queues — Decoupling producer spikes from downstream workers with SQS Standard (unlimited throughput, at-least-once) and FIFO (.fifo, MessageGroupId ordering, deduplication), Long Polling, and DLQs. • ✓ Build, verify, and ship Module Capstone: API Gateway Async Ingestion + SQS Partial-Batch Resilient Worker.

10.1 Amazon API Gateway — HTTP vs REST vs WebSocket APIs, Authorizers, Throttling & 29s Limit [50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: HTTP API vs REST API vs WebSocket API & Lambda Proxy Integration response contract • Core Syntax, CLI & Production APIs: apigwv2_client.create_api(ProtocolType='HTTP') + Lambda Proxy Response {statusCode, headers,... • Deep-Dive Mechanics: Cognito and JWT Lambda Authorizers & Token-Bucket rate and burst throttling • Advanced Patterns: 29-second integration timeout rule & Async 202 Accepted + SQS offload pattern • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: API Gateway Lambda Proxy Formatter & Timeout Offload Guard • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for Amazon API Gateway

Purpose: To understand and apply Amazon API Gateway — HTTP vs REST vs WebSocket APIs, Authorizers, Throttling & 29s Limit in real AWS systems — specifically: Securing and throttli...

What Topic Is Next → Next Topic 10.2: Amazon SQS — Standard vs FIFO Queues, Visibility Timeout, Long Polling & Dead-Letter Queues

10.2 Amazon SQS — Standard vs FIFO Queues, Visibility Timeout, Long Polling & Dead-Letter Queues [50 mins • Assignment (100M) • Hands-On Lab (50M) • 10 Mock Sets • 10 AI Vivas]

Subtopics Covered: SQS Standard vs FIFO (.fifo) ordering & Visibility Timeout >= 6x worker timeout • Core Syntax, CLI & Production APIs: sqs_client.receive_message(QueueUrl=..., WaitTimeSeconds=20, MaxNumberOfMessages=10) • Deep-Dive Mechanics: Long Polling WaitTimeSeconds=20 & Dead-Letter Queue (DLQ) maxReceiveCount • Advanced Patterns: ReportBatchItemFailures partial ack & 256 KB payload cap + S3 Extended Client • Production Failure Mode & Pitfall Guard • Hands-On Live Coding Lab: SQS Partial-Batch Failure Handler & Visibility Timeout Check • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for Amazon SQS

Purpose: To understand and apply Amazon SQS — Standard vs FIFO Queues, Visibility Timeout, Long Polling & Dead-Letter Queues in real AWS systems — specifically: Decoupling produce...

What Topic Is Next → Module 10 Capstone (API Gateway Async Ingestion + SQS Partial-Batch Resilient Worker) → Next Topic 11.1: Remote Terraform State in Encrypted S3 + DynamoDB State Locking

Module 10 Hands-On Capstone Project (100 Marks): API Gateway Async Ingestion + SQS Partial-Batch Resilient Worker — Architect an async order-processing pipeline that validates API Gateway Lambda Proxy responses, checks SQS VisibilityTimeout >= 6x consumer timeout, and returns batchItemFailures so poison m...

MODULE 11 OF 16 • STAGE 3 • 16 LIVE + LAB HOURS • 4 TOPICS (32 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

Infrastructure as Code on AWS: Terraform / CloudFormation & Multi-Env State

Production-grade AWS Training engineering module covering Infrastructure as Code on AWS: Terraform / CloudFormation & Multi-Env State with hands-on labs and capstone build.

Objectives: ✓ Remote Terraform State in Encrypted S3 + DynamoDB State Locking — Purpose: Prevent two engineers or CI pipelines from corrupting cloud state simultaneously. • ✓ Reusable VPC, ALB & RDS Modules Across Dev/Stage/Prod Workspaces — Purpose: Provision identical, auditable AWS environments in minutes with zero console click-ops. • ✓ IAM OIDC GitHub Actions Federation (Zero Long-Lived AWS Access Keys) — Purpose: Deploy to AWS from CI using short-lived STS tokens instead of leak-prone static keys. • ✓ Drift Detection, Policy-as-Code Guardrails & Tagging Enforcement — Purpose: Block unencrypted buckets or public security groups automatically before apply.

11.1 Remote Terraform State in Encrypted S3 + DynamoDB State Locking [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of Edge • Integration with Reusable VPC, ALB & RDS Modules Across Dev/Stage/Prod Workspaces • Live Debugging, Telemetry & Technical Interview Trade • What Is Remote Terraform State in Encrypted S3 + DynamoDB State Locking & Why Industry Teams Use It • In AWS Training, Remote Terraform State in Encrypted S3 + DynamoDB State Locking exists for one clear real • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply Remote Terraform State in Encrypted S3 + DynamoDB State Locking in real AWS systems — specifically: Prevent two engineers or CI pipelines from cor...

What Topic Is Next → Next Topic 11.2: Reusable VPC, ALB & RDS Modules Across Dev/Stage/Prod Workspaces

11.2 Reusable VPC, ALB & RDS Modules Across Dev/Stage/Prod Workspaces [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of ALB • Integration with IAM OIDC GitHub Actions Federation (Zero Long • Live Debugging, Telemetry & Technical Interview Trade • What Is Reusable VPC, ALB & RDS Modules Across Dev/Stage/Prod Workspaces & Why Industry Teams Use It • In AWS Training, Reusable VPC, ALB & RDS Modules Across Dev/Stage/Prod Workspaces exists for one clear real • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply Reusable VPC, ALB & RDS Modules Across Dev/Stage/Prod Workspaces in real AWS systems — specifically: Provision identical, auditable AWS environmen...

What Topic Is Next → Next Topic 11.3: IAM OIDC GitHub Actions Federation (Zero Long-Lived AWS Access Keys)

11.3 IAM OIDC GitHub Actions Federation (Zero Long-Lived AWS Access Keys) [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of Edge • Integration with Drift Detection, Policy • Live Debugging, Telemetry & Technical Interview Trade • What Is IAM OIDC GitHub Actions Federation (Zero Long • In AWS Training, IAM OIDC GitHub Actions Federation (Zero Long • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply IAM OIDC GitHub Actions Federation (Zero Long-Lived AWS Access Keys) in real AWS systems — specifically: Deploy to AWS from CI using short-lived S...

What Topic Is Next → Next Topic 11.4: Drift Detection, Policy-as-Code Guardrails & Tagging Enforcement

11.4 Drift Detection, Policy-as-Code Guardrails & Tagging Enforcement [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of Policy • Integration with Remote Terraform State in Encrypted S3 + DynamoDB State Locking • Live Debugging, Telemetry & Technical Interview Trade • What Is Drift Detection, Policy • In AWS Training, Drift Detection, Policy • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply Drift Detection, Policy-as-Code Guardrails & Tagging Enforcement in real AWS systems — specifically: Block unencrypted buckets or public security...

What Topic Is Next → Module 11 Capstone (Modular Terraform AWS VPC + Compute + RDS Blueprint) → Next Topic 12.1: Amazon ECR Image Scanning & Serverless ECS Fargate Task Definitions

Module 11 Hands-On Capstone Project (100 Marks): Modular Terraform AWS VPC + Compute + RDS Blueprint — Design, implement, test, and document 'Modular Terraform AWS VPC + Compute + RDS Blueprint' applying every production concept from Module 11 (Infrastructure as Code on AWS: Terraform / Cloud...

MODULE 12 OF 16 • STAGE 3 • 16 LIVE + LAB HOURS • 4 TOPICS (32 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

AWS Containers & Serverless Event-Driven Architecture (ECS, Lambda, SQS)

Production-grade AWS Training engineering module covering AWS Containers & Serverless Event-Driven Architecture (ECS, Lambda, SQS) with hands-on labs and capstone build.

Objectives: ✓ Amazon ECR Image Scanning & Serverless ECS Fargate Task Definitions — Purpose: Run production containers without patching EC2 hosts, using per-task IAM roles. • ✓ ALB Target Group Health Checks & ECS Service Auto Scaling — Purpose: Scale container tasks seamlessly on CPU/request spikes with zero dropped connections. • ✓ AWS Lambda Cold-Start Tuning, API Gateway & DynamoDB Single-Table Patterns — Purpose: Build pay-per-request serverless APIs with sub-10ms DynamoDB lookups. • ✓ Decoupled Microservices with SQS Dead-Letter Queues, SNS & EventBridge — Purpose: Absorb 10x traffic bursts reliably and isolate failed events for safe replay.

12.1 Amazon ECR Image Scanning & Serverless ECS Fargate Task Definitions [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of Serverless ECS Fargate Task Definitions • Integration with ALB Target Group Health Checks & ECS Service Auto Scaling • Live Debugging, Telemetry & Technical Interview Trade • What Is Amazon ECR Image Scanning & Serverless ECS Fargate Task Definitions & Why Industry Teams Use It • In AWS Training, Amazon ECR Image Scanning & Serverless ECS Fargate Task Definitions exists for one clear real • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply Amazon ECR Image Scanning & Serverless ECS Fargate Task Definitions in real AWS systems — specifically: Run production containers without patching...

What Topic Is Next → Next Topic 12.2: ALB Target Group Health Checks & ECS Service Auto Scaling

12.2 ALB Target Group Health Checks & ECS Service Auto Scaling [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of ECS Service Auto Scaling • Integration with AWS Lambda Cold • Live Debugging, Telemetry & Technical Interview Trade • What Is ALB Target Group Health Checks & ECS Service Auto Scaling & Why Industry Teams Use It • In AWS Training, ALB Target Group Health Checks & ECS Service Auto Scaling exists for one clear real • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply ALB Target Group Health Checks & ECS Service Auto Scaling in real AWS systems — specifically: Scale container tasks seamlessly on CPU/request spik...

What Topic Is Next → Next Topic 12.3: AWS Lambda Cold-Start Tuning, API Gateway & DynamoDB Single-Table Patterns

12.3 AWS Lambda Cold-Start Tuning, API Gateway & DynamoDB Single-Table Patterns [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of API Gateway • Integration with Decoupled Microservices with SQS Dead • Live Debugging, Telemetry & Technical Interview Trade • What Is AWS Lambda Cold • In AWS Training, AWS Lambda Cold • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply AWS Lambda Cold-Start Tuning, API Gateway & DynamoDB Single-Table Patterns in real AWS systems — specifically: Build pay-per-request serverless AP...

What Topic Is Next → Next Topic 12.4: Decoupled Microservices with SQS Dead-Letter Queues, SNS & EventBridge

12.4 Decoupled Microservices with SQS Dead-Letter Queues, SNS & EventBridge [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of SNS • Integration with Amazon ECR Image Scanning & Serverless ECS Fargate Task Definitions • Live Debugging, Telemetry & Technical Interview Trade • What Is Decoupled Microservices with SQS Dead • In AWS Training, Decoupled Microservices with SQS Dead • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply Decoupled Microservices with SQS Dead-Letter Queues, SNS & EventBridge in real AWS systems — specifically: Absorb 10x traffic bursts reliably and...

What Topic Is Next → Module 12 Capstone (Event-Driven Order Processing Pipeline on ECS Fargate & SQS) → Next Topic 13.1: IAM Permission Boundaries, SCPs & Least-Privilege Policy Analysis

Module 12 Hands-On Capstone Project (100 Marks): Event-Driven Order Processing Pipeline on ECS Fargate & SQS — Design, implement, test, and document 'Event-Driven Order Processing Pipeline on ECS Fargate & SQS' applying every production concept from Module 12 (AWS Containers & Serverless Event-Driven...

MODULE 13 OF 16 • STAGE 4 • 16 LIVE + LAB HOURS • 4 TOPICS (32 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

AWS Enterprise Security, KMS Encryption, Secrets Manager & WAF

Production-grade AWS Training engineering module covering AWS Enterprise Security, KMS Encryption, Secrets Manager & WAF with hands-on labs and capstone build.

Objectives: ✓ IAM Permission Boundaries, SCPs & Least-Privilege Policy Analysis — Purpose: Contain blast radius across multi-account AWS Organizations workloads. • ✓ Envelope Encryption with AWS KMS & Automatic Secrets Manager Rotation — Purpose: Rotate RDS credentials and API keys automatically without redeploying application code. • ✓ AWS WAF Managed Rules, Shield DDoS Protection & VPC Endpoints (PrivateLink) — Purpose: Block SQLi/XSS bot traffic at CloudFront edge and keep S3/Dynamo traffic off public internet. • ✓ CloudTrail Audit Trails, GuardDuty Threat Detection & AWS Config Rules — Purpose: Detect anomalous API calls and compliance drift within minutes.

13.1 IAM Permission Boundaries, SCPs & Least-Privilege Policy Analysis

[65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of SCPs • Integration with Envelope Encryption with AWS KMS & Automatic Secrets Manager Rotation • Live Debugging, Telemetry & Technical Interview Trade • What Is IAM Permission Boundaries, SCPs & Least • In AWS Training, IAM Permission Boundaries, SCPs & Least • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply IAM Permission Boundaries, SCPs & Least-Privilege Policy Analysis in real AWS systems — specifically: Contain blast radius across multi-account AW...

What Topic Is Next → Next Topic 13.2: Envelope Encryption with AWS KMS & Automatic Secrets Manager Rotation

13.2 Envelope Encryption with AWS KMS & Automatic Secrets Manager Rotation

[65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of Automatic Secrets Manager Rotation • Integration with AWS WAF Managed Rules, Shield DDoS Protection & VPC Endpoints (PrivateLink) • Live Debugging, Telemetry & Technical Interview Trade • What Is Envelope Encryption with AWS KMS & Automatic Secrets Manager Rotation & Why Industry Teams Use It • In AWS Training, Envelope Encryption with AWS KMS & Automatic Secrets Manager Rotation exists for one clear real • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply Envelope Encryption with AWS KMS & Automatic Secrets Manager Rotation in real AWS systems — specifically: Rotate RDS credentials and API keys auto...

What Topic Is Next → Next Topic 13.3: AWS WAF Managed Rules, Shield DDoS Protection & VPC Endpoints (PrivateLink)

13.3 AWS WAF Managed Rules, Shield DDoS Protection & VPC Endpoints (PrivateLink)

[65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of Shield DDoS Protection • Integration with CloudTrail Audit Trails, GuardDuty Threat Detection & AWS Config Rules • Live Debugging, Telemetry & Technical Interview Trade • What Is AWS WAF Managed Rules, Shield DDoS Protection & VPC Endpoints (PrivateLink) & Why Industry Teams Use It • In AWS Training, AWS WAF Managed Rules, Shield DDoS Protection & VPC Endpoints (PrivateLink) exists for one clear r... • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply AWS WAF Managed Rules, Shield DDoS Protection & VPC Endpoints (PrivateLink) in real AWS systems — specifically: Block SQLi/XSS bot traffic at Clou...

What Topic Is Next → Next Topic 13.4: CloudTrail Audit Trails, GuardDuty Threat Detection & AWS Config Rules

13.4 CloudTrail Audit Trails, GuardDuty Threat Detection & AWS Config Rules

[65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of GuardDuty Threat Detection • Integration with IAM Permission Boundaries, SCPs & Least • Live Debugging, Telemetry & Technical Interview Trade • What Is CloudTrail Audit Trails, GuardDuty Threat Detection & AWS Config Rules & Why Industry Teams Use It • In AWS Training, CloudTrail Audit Trails, GuardDuty Threat Detection & AWS Config Rules exists for one clear real • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply CloudTrail Audit Trails, GuardDuty Threat Detection & AWS Config Rules in real AWS systems — specifically: Detect anomalous API calls and complian...

What Topic Is Next → Module 13 Capstone (SOC2-Ready Hardened AWS Security & Secrets Architecture) → Next Topic 14.1: RTO vs RPO Tiers: Backup/Restore, Pilot Light, Warm Standby & Active-Active

Module 13 Hands-On Capstone Project (100 Marks): SOC2-Ready Hardened AWS Security & Secrets Architecture — Design, implement, test, and document 'SOC2-Ready Hardened AWS Security & Secrets Architecture' applying every production concept from Module 13 (AWS Enterprise Security, KMS Encryption, Sec...

MODULE 14 OF 16 • STAGE 4 • 16 LIVE + LAB HOURS • 4 TOPICS (32 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

Multi-Region Disaster Recovery, Observability & FinOps Cost Optimization

Production-grade AWS Training engineering module covering Multi-Region Disaster Recovery, Observability & FinOps Cost Optimization with hands-on labs and capstone build.

Objectives: ✓ RTO vs RPO Tiers: Backup/Restore, Pilot Light, Warm Standby & Active-Active — Purpose: Match business continuity SLAs to exact Route 53 failover and RDS cross-region designs. • ✓ CloudWatch Composite Alarms, Logs Insights, X-Ray Tracing & Dashboards — Purpose: Pinpoint cross-service AWS latency bottlenecks and page on-call teams before outages spread. • ✓ S3 Intelligent-Tiering, CloudFront Origin Shield & RDS Read Replicas — Purpose: Slash egress bandwidth bills and database CPU load simultaneously. • ✓ AWS Budgets, Cost Anomaly Alerts, Savings Plans & Right-Sizing Audits — Purpose: Cut monthly AWS cloud spend by 30–50% without sacrificing reliability.

14.1 RTO vs RPO Tiers: Backup/Restore, Pilot Light, Warm Standby & Active-Active [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of Pilot Light • Integration with CloudWatch Composite Alarms, Logs Insights, X • Live Debugging, Telemetry & Technical Interview Trade • What Is RTO vs RPO Tiers • In AWS Training, RTO vs RPO Tiers • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply RTO vs RPO Tiers: Backup/Restore, Pilot Light, Warm Standby & Active-Active in real AWS systems — specifically: Match business continuity SLAs to...

What Topic Is Next → Next Topic 14.2: CloudWatch Composite Alarms, Logs Insights, X-Ray Tracing & Dashboards

14.2 CloudWatch Composite Alarms, Logs Insights, X-Ray Tracing & Dashboards [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of Logs Insights • Integration with S3 Intelligent • Live Debugging, Telemetry & Technical Interview Trade • What Is CloudWatch Composite Alarms, Logs Insights, X • In AWS Training, CloudWatch Composite Alarms, Logs Insights, X • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply CloudWatch Composite Alarms, Logs Insights, X-Ray Tracing & Dashboards in real AWS systems — specifically: Pinpoint cross-service AWS latency bott...

What Topic Is Next → Next Topic 14.3: S3 Intelligent-Tiering, CloudFront Origin Shield & RDS Read Replicas

14.3 S3 Intelligent-Tiering, CloudFront Origin Shield & RDS Read Replicas [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of CloudFront Origin Shield • Integration with AWS Budgets, Cost Anomaly Alerts, Savings Plans & Right • Live Debugging, Telemetry & Technical Interview Trade • What Is S3 Intelligent • In AWS Training, S3 Intelligent • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply S3 Intelligent-Tiering, CloudFront Origin Shield & RDS Read Replicas in real AWS systems — specifically: Slash egress bandwidth bills and database...

What Topic Is Next → Next Topic 14.4: AWS Budgets, Cost Anomaly Alerts, Savings Plans & Right-Sizing Audits

14.4 AWS Budgets, Cost Anomaly Alerts, Savings Plans & Right-Sizing Audits [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Plain • Production Implementation of Cost Anomaly Alerts • Integration with RTO vs RPO Tiers • Live Debugging, Telemetry & Technical Interview Trade • What Is AWS Budgets, Cost Anomaly Alerts, Savings Plans & Right • In AWS Training, AWS Budgets, Cost Anomaly Alerts, Savings Plans & Right • Even if you come from a non • Runnable Code Implementation (HCL)

Purpose: To understand and apply AWS Budgets, Cost Anomaly Alerts, Savings Plans & Right-Sizing Audits in real AWS systems — specifically: Cut monthly AWS cloud spend by 30–50% wi...

What Topic Is Next → Module 14 Capstone (Multi-AZ/Region Failover & FinOps Cost-Governed Cloud Stack) → Next Topic 15.1: AWS Organizations, Service Control Policies (SCPs), IAM Permission Boundaries & OIDC

Module 14 Hands-On Capstone Project (100 Marks): Multi-AZ/Region Failover & FinOps Cost-Governed Cloud Stack — Design, implement, test, and document 'Multi-AZ/Region Failover & FinOps Cost-Governed Cloud Stack' applying every production concept from Module 14 (Multi-Region Disaster Recovery, Observab...

MODULE 15 OF 16 • STAGE 4 • 14 LIVE + LAB HOURS • 3 TOPICS (24 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

AWS Cloud Security Governance, KMS Encryption & Cost FinOps Architecture

Self-explanatory list-and-bullet production mastery module covering AWS Cloud Security Governance, KMS Encryption & Cost FinOps Architecture in AWS Training, paired with hands-on labs, capstone repository, Topic Mock Tests, and Spoken MockAttempt AI Viva.

Objectives: ✓ AWS Organizations, Service Control Policies (SCPs), IAM Permission Boundaries & OIDC — Purpose: Enforce multi-account security guardrails and keyless CI/CD deployments without static access keys. • ✓ AWS KMS Customer-Managed Keys, Secrets Manager Rotation & WAF + Shield Protection — Purpose: Encrypt S3, EBS, and RDS data at rest and block SQLi/bot layer-7 attacks at CloudFront/ALB. • ✓ AWS Cost Explorer, Budgets, Savings Plans, S3 Intelligent-Tiering & Right-Sizing — Purpose: Prevent surprise cloud bills with automated anomaly alerts and storage/compute lifecycle rules. • ✓ Build, verify, and document 'Least-Privilege AWS Organization SCP, KMS Envelope Encryption & Budget Guard' with clean Git commits and edge-case protection. • ✓ Clear 100% of Topic Timed Mock Tests and Spoken Voice + MockAttempt AI Technical Interviews.

15.1 AWS Organizations, Service Control Policies (SCPs), IAM Permission Boundaries & OIDC [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Security Hygiene & Data Safety Rules for AWS Organizations, Service Control Policies (SCPs), IAM Permission Boundar... • Performance Optimization, Complexity & Latency Impact of AWS Organizations, Service Control Policies (SCPs), IAM Pe... • Live Debugging, Stack • Runnable Code Implementation (PYTHON) • Hands • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for AWS Organizations • Production Hardening & Security: IAM least-privilege OIDC roles, KMS encryption, Security Groups & WAF

Purpose: To understand and apply AWS Organizations, Service Control Policies (SCPs), IAM Permission Boundaries & OIDC in real AWS systems — specifically: Enforce multi-account sec...

What Topic Is Next → Next Topic 15.2: AWS KMS Customer-Managed Keys, Secrets Manager Rotation & WAF + Shield Protection

15.2 AWS KMS Customer-Managed Keys, Secrets Manager Rotation & WAF + Shield Protection [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Security Hygiene & Data Safety Rules for AWS KMS Customer • Performance Optimization, Complexity & Latency Impact of AWS KMS Customer • Live Debugging, Stack • Runnable Code Implementation (PYTHON) • Hands • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for AWS KMS Customer-Managed Keys • Production Hardening & Security: IAM least-privilege OIDC roles, KMS encryption, Security Groups & WAF

Purpose: To understand and apply AWS KMS Customer-Managed Keys, Secrets Manager Rotation & WAF + Shield Protection in real AWS systems — specifically: Encrypt S3, EBS, and RDS dat...

What Topic Is Next → Next Topic 15.3: AWS Cost Explorer, Budgets, Savings Plans, S3 Intelligent-Tiering & Right-Sizing

15.3 AWS Cost Explorer, Budgets, Savings Plans, S3 Intelligent-Tiering & Right-Sizing [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Security Hygiene & Data Safety Rules for AWS Cost Explorer, Budgets, Savings Plans, S3 Intelligent • Performance Optimization, Complexity & Latency Impact of AWS Cost Explorer, Budgets, Savings Plans, S3 Intelligent • Live Debugging, Stack • Runnable Code Implementation (PYTHON) • Hands • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for AWS Cost Explorer • Production Hardening & Security: IAM least-privilege OIDC roles, KMS encryption, Security Groups & WAF

Purpose: To understand and apply AWS Cost Explorer, Budgets, Savings Plans, S3 Intelligent-Tiering & Right-Sizing in real AWS systems — specifically: Prevent surprise cloud bills...

What Topic Is Next → Module 15 Capstone (Least-Privilege AWS Organization SCP, KMS Envelope Encryption & Budget Guard) → Next Topic 16.1: Disaster Recovery Tiers: Backup & Restore, Pilot Light, Warm Standby & Multi-Site Active-Active

Module 15 Hands-On Capstone Project (100 Marks): Least-Privilege AWS Organization SCP, KMS Envelope Encryption & Budget Guard — Architect, implement, test, and document 'Least-Privilege AWS Organization SCP, KMS Envelope Encryption & Budget Guard' applying every bullet-wise engineering concept from Module 15 (AWS Clo...

MODULE 16 OF 16 • STAGE 4 • 14 LIVE + LAB HOURS • 3 TOPICS (24 SUBTOPICS) • 1 CAPSTONE PROJECT (100 MARKS)

AWS Well-Architected Disaster Recovery, Observability & SAA-C03 Scenarios

Self-explanatory list-and-bullet production mastery module covering AWS Well-Architected Disaster Recovery, Observability & SAA-C03 Scenarios in AWS Training, paired with hands-on labs, capstone repository, Topic Mock Tests, and Spoken MockAttempt AI Viva.

Objectives: ✓ Disaster Recovery Tiers: Backup & Restore, Pilot Light, Warm Standby & Multi-Site Active-Active — Purpose: Match exact RTO and RPO business targets to cost-effective AWS multi-region architectures. • ✓ Centralized Observability with CloudWatch Embedded Metrics, X-Ray Tracing & CloudTrail — Purpose: Correlate API latency, Lambda cold starts, and IAM audit trails during live incident triage. • ✓ AWS Solutions Architect Scenario Walkthroughs & Production Cutover Checklists — Purpose: Solve real SAA-C03 architecture trade-off questions across networking, storage, and serverless. • ✓ Build, verify, and document 'Multi-Region Failover Architecture Blueprint & CloudWatch Alarm Runbook' with clean Git commits and edge-case protection. • ✓ Clear 100% of Topic Timed Mock Tests and Spoken Voice + MockAttempt AI Technical Interviews.

16.1 Disaster Recovery Tiers: Backup & Restore, Pilot Light, Warm Standby & Multi-Site Active-Active [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Security Hygiene & Data Safety Rules for Disaster Recovery Tiers • Performance Optimization, Complexity & Latency Impact of Disaster Recovery Tiers • Live Debugging, Stack • Runnable Code Implementation (PYTHON) • Hands • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for Disaster Recovery Tiers • Production Hardening & Security: IAM least-privilege OIDC roles, KMS encryption, Security Groups & WAF

Purpose: To understand and apply Disaster Recovery Tiers: Backup & Restore, Pilot Light, Warm Standby & Multi-Site Active-Active in real AWS systems — specifically: Match exact RT...

What Topic Is Next → Next Topic 16.2: Centralized Observability with CloudWatch Embedded Metrics, X-Ray Tracing & CloudTrail

16.2 Centralized Observability with CloudWatch Embedded Metrics, X-Ray Tracing & CloudTrail [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Security Hygiene & Data Safety Rules for Centralized Observability with CloudWatch Embedded Metrics, X • Performance Optimization, Complexity & Latency Impact of Centralized Observability with CloudWatch Embedded Metrics... • Live Debugging, Stack • Runnable Code Implementation (PYTHON) • Hands • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for Centralized Observability with CloudWatch Embedded M... • Production Hardening & Security: IAM least-privilege OIDC roles, KMS encryption, Security Groups & WAF

Purpose: To understand and apply Centralized Observability with CloudWatch Embedded Metrics, X-Ray Tracing & CloudTrail in real AWS systems — specifically: Correlate API latency,...

What Topic Is Next → Next Topic 16.3: AWS Solutions Architect Scenario Walkthroughs & Production Cutover Checklists

16.3 AWS Solutions Architect Scenario Walkthroughs & Production Cutover Checklists [65 mins • Assignment (100M) • Hands-On Lab (50M) • 2 Mock Sets • 2 AI Vivas]

Subtopics Covered: Security Hygiene & Data Safety Rules for AWS Solutions Architect Scenario Walkthroughs & Production Cutover Checkli... • Performance Optimization, Complexity & Latency Impact of AWS Solutions Architect Scenario Walkthroughs & Production... • Live Debugging, Stack • Runnable Code Implementation (PYTHON) • Hands • Idiomatic AWSC Implementation: Well-Architected IaC (Terraform), private subnets, auto-scaling & S3/CloudFron... • AWS Multi-AZ VPC, IAM STS, ALB/ECS/EKS, RDS/DynamoDB & EventBridge Execution & Memory Lifecycle for AWS Solutions Architect Scenario Walkthroughs • Production Hardening & Security: IAM least-privilege OIDC roles, KMS encryption, Security Groups & WAF

Purpose: To understand and apply AWS Solutions Architect Scenario Walkthroughs & Production Cutover Checklists in real AWS systems — specifically: Solve real SAA-C03 architecture...

What Topic Is Next → Module 16 Capstone → End-of-Course Live Client Project: CloudShield Multi-AZ Auto-Scaling SaaS Infrastructure & Disaster Recovery Platform

Module 16 Hands-On Capstone Project (100 Marks): Multi-Region Failover Architecture Blueprint & CloudWatch Alarm Runbook — Architect, implement, test, and document 'Multi-Region Failover Architecture Blueprint & CloudWatch Alarm Runbook' applying every bullet-wise engineering concept from Module 16 (AWS Well-Arc...

3. LIVE CLIENT PRODUCTION PROJECT & CAREER SCOPE

End-of-Course Live Client Production Project + Official Internship Confirmation Letter

Project: CloudShield Multi-AZ Auto-Scaling SaaS Infrastructure & Disaster Recovery Platform • **Confirmed Role:** AWS Cloud & Solutions Architecture Intern

Domain & Production Infra: Cloud-Native FinTech & SaaS Client Infrastructure (AWS VPC + ALB + EC2 Auto Scaling + RDS Multi-AZ + S3 + CloudFront + IAM)

- Designed custom multi-tier AWS VPC architecture across public/private subnets, NAT Gateways, and least-privilege IAM roles.
- Provisioned Application Load Balancers, Auto Scaling EC2/ECS workloads, and encrypted Multi-AZ RDS databases.
- Configured S3 lifecycle policies, CloudFront HTTPS edge distribution, Route 53 DNS failover, and CloudWatch alarms.
- Validated production readiness, cost governance budgets, and live client cutover on AWS cloud infrastructure.

Target Job Roles: AWS Cloud Engineer, Associate Solutions Architect, Cloud Operations (CloudOps) Engineer, Junior Site Reliability Engineer (SRE)

Realistic Annual CTC Scope: **Fresher / Entry-Level (0–1 Yr):** ₹5.0 LPA – ₹9.5 LPA **Mid-Level AWS Cloud / DevOps Engineer (1–3 Yrs):** ₹10.5 LPA – ₹21.0 LPA **AWS Solutions Architect / Principal SRE (3–6+ Yrs):** ₹22.0 LPA – ₹44.0+ LPA

MockAttempt Academy Pvt. Ltd. • Email: support@mockattempt.com • Phone/WhatsApp: +91 98765 43210 • Web: https://mockattempt.com/courses/aws-training
Try 2 Free Topic-Wise Timed Topic Mock Tests & Spoken MockAttempt AI Interviews daily on every topic above before enrolling.