DOMAIN 4: SECURITY AND COMPLIANCE

Overview

- Weight: 16%
- Core Competency: Design secure access to AWS resources, implement encryption, enforce compliance, and audit access and activity using AWS security services.

4.1 Design Secure Access to AWS Resources

- IAM (Identity and Access Management)
  - Users: AWS accounts for people/services
  - Groups: Logical user collections for policy assignment
  - Roles: Temporary credentials used by apps, Lambda, EC2
  - Policies: JSON documents defining allow/deny rules
  - STS (Security Token Service): Generates temporary credentials for federated users
- IAM Best Practices
  - Use roles, not access keys, for EC2 or Lambda
  - Enforce MFA for users, especially root
  - Follow least privilege by granting minimal necessary permissions
  - Use IAM Access Analyzer to detect unintended access
- Resource-Based Policies vs Identity-Based Policies
  - Identity-Based: Attached to IAM users/roles/groups
  - Resource-Based: Attached directly to resources (e.g., S3, SNS, SQS)
- Cross-Account Access
  - IAM Role Assumption: Grant access from one AWS account to another
  - Resource Policies: Directly allow access via S3/SNS/SQS policies
  - Organizations SCPs: Org-wide guardrails that limit maximum permissions

4.2 Design Secure Workloads and Applications

- Encryption Types
  - At Rest: Encrypt data stored in S3, EBS, RDS, DynamoDB (SSE-S3, SSE-KMS, client-side)
  - In Transit: Use TLS (HTTPS, SSL) via ALB, CloudFront, API Gateway
  - Key Management: AWS KMS, customer-managed keys, HSM; supports envelope encryption and automatic 1-year key rotation
- Data Protection Services
  - S3: SSE-S3 (AES-256), SSE-KMS, client-side encryption
  - EBS: Encrypted on creation with KMS
  - RDS: Encrypted at rest and in transit
  - DynamoDB: Full encryption by default
  - Secrets Manager: Stores secrets, auto-rotates with Lambda
  - SSM Parameter Store: Stores encrypted strings (SecureString)
- VPC Security Controls
  - Security Groups: Stateful, instance-level firewall
  - NACLs: Stateless, subnet-level access control lists
  - VPC Flow Logs: Logs IP-level traffic in VPC
  - VPC Endpoints: Connect privately to AWS services without Internet
  - Block public access using:
  - S3 Block Public Access feature
  - IAM Conditions such as aws:SourceIp, aws:VpcSourceIp, aws:PrincipalArn

4.3 Design for Audit and Compliance

- AWS CloudTrail
  - Logs all API calls across AWS services
  - Event Types: Management events (default), Data events (S3, Lambda)
  - Multi-region Trails: Enable logging across regions
  - Org Trails: Apply trails to all accounts via AWS Organizations
  - Store logs in S3, optionally forward to CloudWatch Logs or OpenSearch for querying
- AWS Config
  - Tracks configuration changes and monitors resource configurations over time
  - Managed Rules: e.g., required-tags, s3-bucket-public-read-prohibited
  - Conformance Packs: Group rules into compliance packages
  - Use Config + Lambda for auto-remediation of non-compliant resources
- Security Monitoring Tools
  - Amazon GuardDuty: ML-based threat detection (IAM, S3, VPC logs)
  - AWS Security Hub: Centralized security posture dashboard
  - AWS Inspector: Scans EC2, ECR, Lambda for vulnerabilities
  - Amazon Macie: Discover and classify sensitive data in S3
  - CloudWatch Alarms: Alert on unusual activity or errors
- Compliance Support on AWS
  - Artifact: Access compliance documents (SOC, PCI, ISO)
  - Service Control Policies (SCPs): Organization-level permission guardrails
  - AWS Organizations: Group accounts with consolidated billing and governance

Common SAA-C03 Exam Scenarios (Domain 4)

- Secure EC2 access to S3 without hardcoding credentials: Assign IAM role to EC2 instance
- Prevent unauthorized access to public S3 bucket: Enable Block Public Access and bucket policy restrictions
- Log every S3 object access request: Enable CloudTrail Data Events for S3
- Enforce EC2 instance tagging with Environment: Use AWS Config managed rule required-tags
- Detect brute force or port scanning activity: Use Amazon GuardDuty
- Store database credentials securely for EC2 application: Use Secrets Manager or SSM Parameter Store (SecureString)

Summary Cheat Sheet

- Access Control: IAM, STS, SCPs
- Resource Isolation: VPC, Security Groups, NACLs
- Encryption: KMS, SSE-S3, SSE-KMS
- Auditing: CloudTrail, Config, Flow Logs
- Secrets Management: Secrets Manager, SSM
- Threat Detection: GuardDuty, Inspector, Macie
- Compliance: Security Hub, Artifact, Conformance Packs

Recommended AWS Resources

- AWS IAM Best Practices
- AWS Security Pillar (Well-Architected)
- CloudTrail User Guide
- AWS Config Rules Reference
- AWS Artifact

Presented with xmind AI