AWS CERTIFIED SECURITY SPECIALTY

Neal Davis

DigitalCloud TRAINING

Table Of Contents

Welcome

These cheat sheets provide the key facts and scenarios that typically appear in the AWS Certified Security Specialty (SCS-C02) exam. We recommend reviewing these cheat sheets after having completed our on-demand training (video course and practice tests) as a final revision before sitting your exam.

Your Pathway to Success

• Step 1: Gain in-depth knowledge with our on-demand video course

• Step 2: Assess your exam readiness with practice exams / online exam simulator from Digital Cloud Training

• Step 3: Study these cheat sheets for quick review of the key facts

Through diligent study of these learning materials, you will be in the perfect position to ace your AWS Certified Security Specialty exam with confidence.

Wishing you the best for every step in your cloud journey!

Domain 1 - Threat Detection and Incident Response

VPC Flow Logs can be used to capture IP traffic and publish it to Amazon CloudWatch where metric filters can be used to search for specific event patterns such as connection attempts to a specific TCP port.

Automatic provisioning of a secure forensic environment can be achieved using AWS CloudFormation for infrastructure creation and AWS Step Functions to orchestrate the processes (using Lambda) required for forensic analysis.

EC2 instances with Elastic IPs must have the EIPs added to the ingress rules of security groups for connectivity using the public addresses across AZs.

You can collect memory dumps from EC2 instances that are unresponsive using the EC2Rescue CLI with the /offline mode and the device ID specified.

To securely share encrypted snapshots with a separate AWS account, share the snapshot, use a customer managed KMS key, and allow the Decrypt and CreateGrant actions for the target account in the key policy.

If SSH keys for EC2 instances are compromised they can be removed by connecting to each instance and replacing the public key information in the authorized_keys file.

If access key IDs and secret access keys are exposed you should invalidate any temporary security credentials and delete the access key IDs and secret access keys.

If many access key IDs have been exposed you can generate a credential report in each account in an organization to identify the users each access key ID belongs to and then rotate those keys.

If temporary security credentials issued by AWS STS are compromised you can revoke all active sessions for the IAM Role.

You can block DDoS attacks where the User-Agent field of the request header has a certain value by using a web ACL with a string match condition that matches that value.

AWS Secrets Manager helps you protect secrets needed to access your applications, services, and IT resources.

Secrets Manager offers secret rotation with built-in integration for Amazon RDS, Amazon Redshift, and Amazon DocumentDB.

AWS Secrets Manager encrypts secrets at rest using encryption keys that you own and store in AWS Key Management Service (KMS).

Secrets can be disabled to make them inaccessible. They can then be scheduled for deletion.

Temporary credentials issues by AWS STS expire after a specific duration.

If users inadvertently expose their credentials to an unauthorized third-party, that party has access for the duration of the session.

You can immediately revoke all permissions to the role's credentials issued before a certain point in time if you need to.

IAM attaches a policy named AWSRevokeOlderSessions to the role. The policy denies all access to users who assumed the role before the specified time.

AWS GuardDuty is an intelligent threat detection service that continuously monitors for malicious activity and delivers detailed security findings for visibility and remediation.

AWS GuardDuty monitors AWS accounts, workloads, and data in Amazon S3.

AWS GuardDuty detects account compromise, instance compromise, malicious reconnaissance, and bucket compromise.

Amazon Inspector is an automated security assessment service that helps improve the security and compliance of applications deployed on AWS.

Amazon Inspector automatically assesses applications for exposure, vulnerabilities, and deviations from best practices.

The AWS Security Incident Response Guide defines best practices for responding to security incidents. It includes the following aspects: Preparation, Operations, and Post-Incident Activity.

Domain 2 - Security Logging and Monitoring

Amazon CloudWatch is used for performance, logging, alarms, and Events.

Amazon EventBridge is based on Amazon CloudWatch Events.

The unified CloudWatch agent can be installed on EC2 instances to collect application log files and stream them to CloudWatch Logs.

The Unified CloudWatch Agent sends system-level metrics for EC2 and on-premises servers.

System-level metrics include memory and disk usage.

An EC2 instance in a private subnet running the unified CloudWatch agent can be configured to send logs to CloudWatch Logs securely via an interface VPC endpoint.

You can also send CloudWatch Logs to S3, Kinesis Data Streams, and Kinesis Data Firehose.

If Lambda fails to write logs to CloudWatch Logs check the function execution role for permissions.

AWS CloudTrail logs API activity for auditing purposes and you must create a trail to store logs indefinitely (90 days otherwise).

A CloudTrail trail can be configured in the management account of an AWS Organization with logging to a centralized S3 bucket. Administrators in child accounts cannot modify the trail.

Single region AWS CloudTrail trails can be reconfigured to apply to all regions and log API activity to a central Amazon S3 bucket.

To allow delivery of CloudTrail events to an S3 bucket you must ensure the S3 bucket policy grants CloudTrail the s3:PutObject permission and verify that the S3 bucket and prefix defined in CloudTrail exists.

To automatically notify about security group changes use AWS CloudTrail with a CloudWatch Logs log group and use a metric filter to match security group changes and SNS to send notifications.

AWS Config can be used with the managed rule cloudtrail-enabled to check that CloudTrail is enabled. Systems Manager automation can be configured for automatic remediation if the rule returns a non-compliant state.

You can run ad-hoc SQL queries on ELB access logs in an S3 bucket using Amazon Athena.

The restricted-ssh managed rule in AWS Config can be configured to identify security groups that allow unrestricted traffic on port 22 (SSH).

Amazon Kinesis can be used for ingesting real-time streaming data such as video, audio, application logs, website clickstreams, and IoT telemetry data for machine learning, analytics, and other applications.

Amazon OpenSearch can receive data from Kinesis and analyze that and store that data.

To enable Amazon Detective. you must have enabled Amazon GuardDuty more than 48 hours ago.

You can generate an alert if users add bucket policies by configuring an Amazon EventBridge rule that uses the “AWS API Call via CloudTrail” event source and the “s3:PutBucketPolicy” event pattern.

Domain 3 - Infrastructure Security

Use separate VPCs to isolate infrastructure, use multiple AZs for high availability, and use subnets to isolate the tiers of your application.

Control network traffic with Security Groups and Network ACLs. Security groups and stateful and NACLs are stateless.

Security groups support allow rules only, are applied at the ENI level, and all rules are evaluated. NACLs apply only to traffic **entering** / **exiting** the subnet and are applied in order until an allow or deny is reached.

You can audit security group changes by sending AWS API calls via CloudTrail to Amazon EventBridge and SNS for notifications.

Use private subnets for your instances if they should not be accessed directly from the internet and use VPC interface, gateway endpoints, and AWS PrivateLink to keep traffic private.

Use a NAT gateway or NAT instance for enabling internet access for EC2 instances in private subnets when using IPv4.

Outbound internet connectivity for instances using IPv6 can be enabled using an egress-only internet gateway which will only allow outbound and not inbound traffic.

A Lambda function can be configured to connect to DynamoDB using private IPs by configuring the function in a VPC and using a VPC endpoint for the DynamoDB table.

Lambda functions can be attached to VPCs by configuring the VPC and subnets and you can then use security groups.

For Lambda functions running in a VPC you will need a NAT gateway to reach the internet.

Amazon OpenSearch can be deployed in a VPC, and you will then have a private endpoint (cannot switch to public later).

Amazon Route 53, AWS Shield, and Elastic Load Balancer can be used to protect EC2 instances against layer 4 DDoS attacks.

Alias records can be used at the zone apex whereas CNAME records can only be used on subdomains.

Internet connectivity to an EC2 instance relies on correct configuration of security groups, network ACLs, route table entries for an internet gateway, and host-based firewall settings (if applicable).

EC2 instances can be scanned for known software vulnerabilities using Amazon Inspector with the “Common vulnerabilities and exposures” assessment and the “Center for Internet Security (CIS) Benchmarks” assessment.

You can improve security of the instance metadata service by requiring the use of IMDSv2 by setting the "--metadata-options HttpTokens" option to "required".

EC2 proxy instances and NAT instances must have source/destination checks disabled to be able to forward traffic.

All traffic sent to EC2 instances can be captured for inspection with an intrusion detection appliance by configuring VPC traffic mirroring with a network load balancer.

To connect securely to the CLI on EC2 instances the SSM agent can be used, and the connection made through Session Manager, bypassing SSH ports, and with access controlled by IAM user policy.

Systems Manager Patch Manager can be used to scan EC2 instances, identify vulnerable software, and install patches.

AWS Config can be used to check that EC2 instances have been launched from approved AMIs by using the rule "approved_ami_by_id".

If health checks are failing for EC2 instances behind an NLB check that the instance security groups, and subnet network ACLs allow traffic from the NLB IP addresses.

To ensure that TLS traffic to an ALB is secure even if the certificate private key is compromised you should create an HTTPS listener with a predefined security policy that supports forward secrecy (FS).

You can redirect HTTP connections to HTTPS on an ALB by creating both types of listener and a rule that redirects HTTP to HTTPS.

An origin access identity (OAI) can be used to restrict access to an S3 bucket so only CloudFront can access the bucket contents.

You can add a **custom header** in CloudFront origin settings to restrict access to a specific ALB with a conditional rule on the ALB looking for the header value.

You can use geo restriction in CloudFront to restrict users in specific locations from accessing the content.

To protect against DDoS attacks a static website can be migrated from EC2 to S3 behind a CloudFront distribution with an AWS WAF WebACL.

An AWS WAF web ACL can be associated with an ALB and configured with an IP set match rule statement with a block action to deny incoming requests from specific IP addresses or ranges of addresses.

AWS WAF can be configured with rate-based rules that protect against layer 7 DDoS attacks by putting a temporary block on requests from IP addresses that send excessive requests.

Encryption in transit can be added to an AWS Direct Connect (DX) connection by implementing a Virtual Private Gateway (VGW) and establishing an encrypted site-to-site VPN over the DX connection.

AWS Certificate Manager (ACM) can be used to create SSL/TLS certificates using public or private domain names and subdomains or wildcards.

Domain 4 - Identity and Access Management

IAM Roles are used for delegation and are assumed by IAM entities. IAM policies are attached to roles and define the permissions for the identities or resources they are associated with.

The trust policy associated with an IAM role determines who is allowed to assume the role.

Identity-based policies are JSON permissions policy documents that control what actions an identity can perform, on which resources, and under what conditions.

Identity-based policies can be applied to users, groups, and roles. Resource-based policies are JSON policy documents that you attach to a resource such as an Amazon S3 bucket.

AWS managed policies are standalone policies that can be attached to multiple users, groups, or roles. Customer managed policies allow you to create your own custom policy permissions to apply to users, groups, or roles.

Groups are collections of users. Users can be members of up to 10 groups and you can assign permissions to all members of the group by attaching IAM policies.

Access authorization to an S3 bucket depends on the union of all the IAM policies, S3 bucket policies, and S3 ACLs that apply.

Permissions boundaries can be configured to specify the maximum available permissions that an identity-based policy can grant to an IAM entity.

S3 bucket policies can be configured to only allow access from the ID of a VPC endpoint.

The condition element in an IAM policy can be used to restrict access based on group membership, IP address etc.

AWS Secrets Manager can be used for storing encrypted database connection strings and automatically rotating them for some AWS database services.

Automatic rotation of secrets is not available for Systems Manager Parameter Store or other database types in Secrets Manager. You would need to use Lambda to rotate the secrets.

To successfully issue the GetParameter API against secure string parameters in Systems Manager Parameter Store, the EC2 instance must have an IAM role assigned that has decrypt permissions on the KMS key and permissions to retrieve parameters.

To secure RDS database connection credentials store them in AWS Secrets Manager or Systems Manager Parameter Store in an encrypted state. Configure the EC2 instance role with credentials to retrieve the connection credentials.

Automatic rotation is available for Secrets Manager when using specific types of database such as RDS, DocumentDB and RedShift.

Host-based firewalls on EC2 instances may be used when complex rules are required to exceed the limits of security groups and network ACLs.

You can restrict that a KMS key is used only by specific services by using the kms:ViaService condition key and specifying the services that should be allowed to use the key.

To enforce that a role can only be assumed after authentication with MFA add an "aws:MultiFactorAuthPresent : true" condition to the role's trust policy.

You can identify the federated user that terminated an EC2 instance by searching CloudTrail for the TerminateInstances event and taking the IAM Role ARN and then looking for AssumeRoleWithSAML events that used that role.

AWS Managed Microsoft AD is a managed implementation of Microsoft Active Directory running on Windows Server 2012 R2. A HA pair of Windows Server 2012 Domain Controllers (DCs) are deployed in your VPC.

AD Connector is used to connect AWS to your on-premises Microsoft Active Directory. You can use it for sign-in to various AWS applications, seamlessly join Windows EC2 instances to the domain, and implement federated sign-in to the AWS management console.

When federating between a corporate IdP and IAM you can use IAM roles to manage access to AWS resources.

Federated sign in to AWS using ADFS and SAML can be used if you keep your identities in an on-premises directory such as Microsoft AD.

Password policies should be configured within Active Directory when using IAM federation.

Password policies can also be configured in IAM user pools and in IAM itself.

AWS SSO can be used for single sign on across many accounts using an existing on-premises identity provider.

Amazon Cognito is used for sign-in and sign-up for mobile applications. User pools are used for creating users or federating to social IdPs and identity pools are used for obtaining temporary, limited-privilege credentials for AWS services.

Domain 5 - Data Protection

You can create and manage symmetric and asymmetric encryption keys with AWS KMS, and they are protected by hardware security modules (HSMs).

KMS keys used to be known as “customer master keys” or CMKs and this terminology could still be used in the exam.

KMS keys can only encrypt data up to 4 KB in size and for anything larger you need to create data encryption keys.

AWS managed KMS keys are created, managed, and used on your behalf by an AWS service that is integrated with AWS KMS. You cannot manage these KMS keys, rotate them, or change their key policies.

Automatic rotation of KMS keys generates new key material every year (optional for customer managed keys) and is supported for symmetric keys with key material AWS KMS creates.

You cannot use automatic rotation in the following situations:

- You are using asymmetric KMS keys.
- You have KMS keys in custom key stores (AWS CloudHSM).
- You are using KMS keys with imported key material.

Manual rotation is creating a new KMS key with a different key ID . You must then update your applications with the new key ID. You can use an alias to represent a KMS key, so you don't need to modify your application code.

KMS key policies define management and usage permissions for KMS keys. Multiple policy statements can be combined to specify separate administrative and usage permissions.

When using AWS KMS customer managed keys with KMS key material you can enable automatic rotation for the key material every year. Automatic rotation of AWS managed KMS Keys is every 3 years and cannot be modified.

Temporary permissions to decrypt data encrypted with KMS keys can be provided using KMS grants. You must be granted permissions to use an AWS KMS key for encryption/decryption in the key policy.

If you require a managed service for encryption keys but must be able to immediately delete the keys use AWS KMS with imported key material and then use the DeleteImportedKeyMaterial API to remove the key material if necessary.

Cryptographic erasure within 15 minutes can be achieved using a KMS key with imported key material and deleting the key material if necessary.

An InvalidKeyId error when trying to use an AWS KMS key may indicate that the key is disabled.

AWS CloudHSM should be used if you need hardware based HSMs within a VPC.

If you need to import your own key material into KMS use a default key store and a customer managed KMS key.

RDS databases cannot be encrypted after creation and you must instead take a snapshot, encrypt a copy of the snapshot, and create a new encrypted DB from the encrypted snapshot.

A multi-tier web application with an RDS database can be secured using encrypted EBS volumes, encrypted DB volumes, TLS connections to the DB, and connection strings stored in AWS Secrets Manager (with automatic rotation).

RDS database instances are configured with an SSL/TLS certificate, and you can download the AWS-provided root certificates for all Regions or specific Regions.

To identify and alert on unencrypted databases use AWS Config with an SNS notification.

Policy variables can be used to specify placeholders in a policy that are replaced with values from the context of the connection request. For example, you can replace the variable with the friendly name of the user making the connection.

The S3 Glacier vault lock operation can be aborted after the initiate vault lock operation.

To secure an S3 bucket with your own keys whilst not managing the encryption process, use server-side encryption with AWS KMS keys and supply your own key material.

Client-side encryption means the entire encryption and decryption takes place outside of AWS and you manage the keys. You can optionally use KMS keys.

Take the following actions to enforce encryption for an S3 bucket:

1. Add a condition to the S3 bucket policy that allows actions if the request meets the condition "aws:SecureTransport": "true".
2. Configure default encryption for the S3 bucket. Add a condition to the S3 bucket policy that denies PUT requests that don't include the "x-amz-server-side-encryption" header.

Amazon EBS encryption affects:

- Data at rest inside the volume.
- All data moving between the volume and the instance (in-transit).
- All snapshots created from the volume.
- All volumes created from those snapshots.

For Amazon EFS, encryption at-rest can be enabled when the file system is created but not after it has been created.

Encryption in-transit can be enabled for Amazon EFS file systems when mounting the file system using the TLS protocol.

The files, deployment packages, and environment variables in a Lambda function are encrypted at rest using an AWS KMS key. AWS recommend that you use Secrets Manager for storing sensitive data instead of environment variables.

To ensure that code written by a developer cannot be deployed to Lambda functions you can use AWS Signer and revoke all versions for the signing profile assigned to the developer.

To verify code integrity when deploying code to Lambda functions use AWS Signer and use IAM policies to enforce that developers can only create functions that have code signing enabled.

Use a VPC endpoint for Kinesis Data Streams to keep the connection private between the VPC and the stream.

Data in Amazon DynamoDB tables is encrypted at rest, and you can choose to select an AWS KMS key or use the default key. Clients connect using published APIs over TLS.

Amazon RDS databases can be encrypted when creating the database. This affects DB storage, backups, read replicas, and snapshots. You cannot enable/disable encryption after creating the database.

All data flowing across AWS Regions over the AWS global network is automatically encrypted at the physical layer before it leaves AWS secured facilities. All traffic between AZs is also encrypted.

Traffic between instances may be encrypted in some circumstances. The instances must use a supported instance type and be within the same Region and VPC (or in a peered VPC.)

Domain 6 - Management and Security Governance

AWS Organizations enables centralized management and governance of multiple AWS accounts, and you can apply service control policies (SCPs), tag policies, and use consolidated billing.

SCPs affect only member accounts in the organization. They have no effect on users or roles in the management account.

SCPs attached to member accounts control permissions for all users including the root user.

An explicit deny in an SCP cannot be overridden with an allow statement anywhere beneath it in the hierarchy.

The only way to get around the inheritance of SCP deny statements is to ensure that the SCP is not applied to any OUs with accounts you do not want to apply the deny statement to.

You can use the condition key aws:PrincipalOrgID in policies to require all principals accessing the resource to be from an account in an AWS Organization.

Use * as a principal and aws:PrincipalOrgID as a condition key to restrict to users within the organization.

You should restrict and limit access to the management account only to those admin users who need rights to make changes to the organization.

Use the management account only for tasks that *require* the management account.

Because SCPs do not apply to the management account AWS recommend that you avoid deploying workloads to the organization's management account.

AWS recommend that you use an SCP to restrict what the root user in member accounts can do. Use the "aws:PrincipalArn": "arn:aws:iam::*:root" in the policy statement.

AWS Control Tower can be used for deploying and managing large numbers of AWS accounts. Control Tower creates a well-architected multi-account baseline known as a landing zone that is based on best practices.

With Control Tower you have preventive guardrails and detective guardrails. Preventive guardrails are based on SCPs and disallow API actions.

Control Tower detective guardrails are implemented using AWS Config rules and Lambda functions and monitor and govern compliance.

According to the well-architected framework, AWS recommends a multi-account environment. This provides security, isolation, centralized billing, and centralized governance.

A tag is a *key-value pair* applied to a resource to hold metadata about that resource. Each tag is a label consisting of a key and an optional value.

Tags can be beneficial for several uses including cost management, governance, compliance, and operational needs.

A tag policy applied in AWS Organizations defines the values that are acceptable for a tag key on specific resource types.

AWS CloudFormation is a popular infrastructure as Code (IaC) tool that allows you to manage and provision AWS resources predictably and repeatedly.

CloudFormation templates can be stored in a version control system (like Git) to track changes, understand when and why changes were made, and revert to previous versions if necessary.

When using IaC, you should handle sensitive data such as passwords, API keys, etc., securely. AWS Secrets Manager or AWS Parameter Store can be used for this purpose.

Template Hardening is a process to ensure that your CloudFormation templates adhere to best practices for security, reliability, performance, and cost optimization.

AWS provides a service called AWS CloudFormation Guard, which allows you to define rules for your AWS resources and prevents deployments that violate those rules.

CloudFormation has a feature called Drift Detection, which can identify and report configuration changes that were made outside CloudFormation to a stack and its resources.

Amazon Macie is a data security service that discovers sensitive data by using machine learning and pattern matching.

Macie provides visibility into data security risks and enables automated protection against those risks.

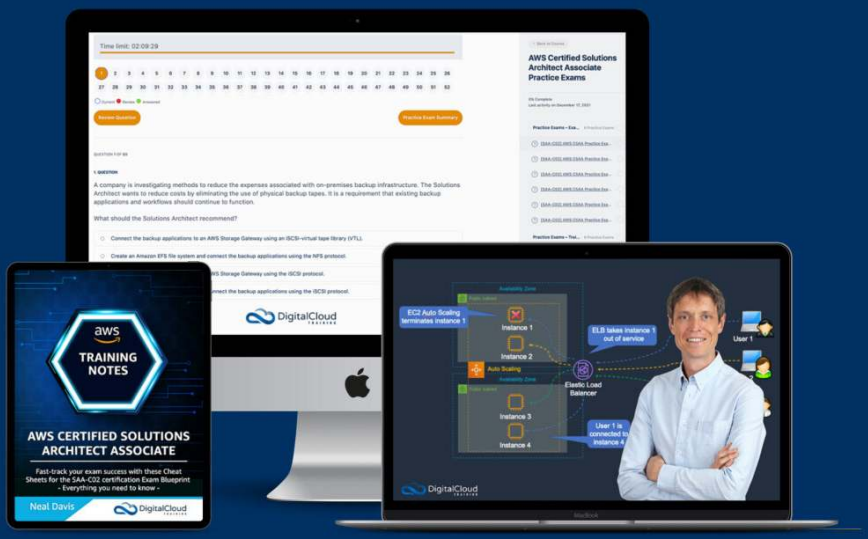
Macie provides you with an inventory of your S3 buckets, and automatically evaluates and monitors the buckets for security and access control.

Macie provides managed identifiers for credentials, financial information, and personally identifiable information (PII).

AWS Service Catalog allows organizations to standardize and manage approved AWS resources and services centrally, ensuring deployments adhere to company guidelines and compliance requirements.

With role-based access controls, Service Catalog users can only access authorized services, promoting cost management by preventing redundant resource provisioning.

AWS Service Catalog integrates seamlessly with other AWS services and provides a self-service portal, making it easier for users to browse, provision, and manage approved services within their set permissions.



DigitalCloud
TRAINING

About Digital Cloud Training

Digital Cloud Training was created to help students achieve their career goals through high-quality AWS certification training resources. We provide a variety of certification training resources for Amazon Web Services (AWS) certifications that represent a higher quality standard than is otherwise available in the market.

Our popular AWS Certification exam preparation resources include instructor-led Video Courses, Hands-on Challenge Labs, in-depth Training Notes, Exam-Cram lessons for quick revision, Quizzes to test your knowledge and exam-difficulty Practice Exams to assess your exam readiness.

Join the AWS Community of over 750,000 happy students that are currently enrolled in Digital Cloud Training courses.

Visit digitalcloud.training for more information