Securing User, Application, and Data Access

MANAGING PERMISSIONS · FEDERATING USERS · MULTI-ACCOUNT ACCESS · ENCRYPTING DATA AT REST · AWS SECURITY SERVICES

STUDY & REVIEW

01 MUST KNOW

if you remember five things, remember these

01 **Groups and ABAC scale permissions:** attach one policy to an **IAM group** and every member inherits it — build groups around **job functions**. To scale further, use **ABAC** (permissions based on **tags**) instead of **RBAC** (policies that list each resource): one attribute-driven policy covers new users and resources without edits.

02 **Explicit deny always wins; effective = intersection:** a request may be filtered by an identity-based policy, a group policy, a **permissions boundary**, and an **SCP**. It is allowed only if every applicable type allows it — and an **explicit deny in any one** overrides every allow.

03 **Federation = trust + temporary credentials:** an **identity provider (IdP)** authenticates the user; a **service provider (SP)** controls the resources. `AssumeRole` / `AssumeRoleWithSAML` on **AWS STS** returns **temporary, limited-privilege** credentials — no long-term IAM user needed for every person.

04 **Cognito: user pool authenticates, identity pool authorizes:** a **user pool** is a directory/IdP that signs users in and issues **JWT** tokens; an **identity pool** exchanges those tokens (via STS) for **temporary AWS credentials** so the app can reach services like Amazon S3 or DynamoDB.

05 **Encrypt data at rest with AWS KMS: KMS** creates and manages keys in FIPS 140-2 **HSMs**, and the keys **never leave KMS**. Integrated services (S3, EBS) use **symmetric** KMS keys only. **Envelope encryption** encrypts data with a data key, then wraps that data key under a KMS key.

02 KEY TERMS

TERM	DEFINITION
IAM group	A collection of IAM users; one policy grants all members the same access. A user can join multiple groups, groups cannot be nested, and groups have no credentials of their own
RBAC	Role-based access control — permissions defined by job function in policies that list individual resources; adding a resource means editing every affected policy
ABAC	Attribute-based access control — one policy grants access by matching attributes (tags) on the identity and the resource; scalable and fully auditable
Tag	Resource metadata as a key or key-value pair, applied to AWS resources and to IAM users and roles; also used for billing and filtered views
Identity federation	A system of trust in which an IdP authenticates users and an SP (such as AWS) controls access to its resources
AWS STS	Security Token Service — a web service (API) that issues temporary, limited-privilege credentials for IAM users, federated users, or applications
Identity broker	An intermediary proxy between an external IdP and the SP that requests temporary credentials from AWS STS on the user's behalf
IAM Identity Center	Successor to AWS Single Sign-On; centrally manages federated access and SSO across many AWS accounts and business applications
Cognito user pool	A user directory and standalone IdP for web and mobile apps; signs users in (or federates them) and issues JWT tokens
Cognito identity pool	Exchanges user-pool tokens for temporary AWS credentials (via STS) so users can access AWS services; also grants guest access

TERM	DEFINITION
Service control policy (SCP)	An AWS Organizations policy that sets the maximum permissions (a guardrail) for member accounts; it grants nothing on its own
Envelope encryption	Encrypting the data key under another key, wrapping keys in layers, and storing the encrypted key alongside the encrypted data

03 AMAZON COGNITO: AUTHENTICATE, THEN AUTHORIZE

user pool proves who · identity pool grants AWS access

01 User signs in through a **Cognito user pool** → 02 App receives **user pool tokens** (JWT) → 03 App exchanges tokens for AWS credentials via a **Cognito identity pool** → 04 App uses the **temporary AWS credentials** to reach S3, DynamoDB, and other services

04 SYMMETRIC VS. ASYMMETRIC ENCRYPTION

both work in transit or at rest

SYMMETRIC

one shared key encrypts and decrypts

- **Same key** to encrypt and decrypt; a shared secret
- Faster and efficient for **large amounts of data**
- Widely used; change the key frequently
- Use when speed and cost matter and data stays inside the network

ASYMMETRIC

a public/private key pair

- **Public key** encrypts, **private key** decrypts
- More secure but **slower** (longer keys, complex math)
- Supports **non-repudiation** — the sender cannot deny sending
- Use when sharing outside the org or the key cannot be shared

05 AWS KMS KEY TYPES

the primary resource in AWS KMS

KEY TYPE	MANAGED BY	REMEMBER
Customer managed key	You	You create, own, and manage it; full control over policies and rotation
AWS managed key	An AWS service, on your behalf	Lives in your account; created and used for you by an integrated service
AWS owned key	An AWS service, across accounts	Not in your account; protects your resources but you do not see or manage it
Data key (symmetric)	You, outside KMS	Returned to you to encrypt data locally, including large data and other keys
Data key pair (asymmetric)	You, outside KMS	Public + private key pair; the private key never leaves KMS unencrypted

06 PERMISSIONS BOUNDARY VS. ORGANIZATIONS SCP

neither one grants permissions

ASPECT	PERMISSIONS BOUNDARY	SCP
Applies to	One IAM entity (a user or role)	All member accounts of an organization or OU
Sets	Max permissions an identity-based policy can grant that entity	Max permissions for every account member
Attaches to	The IAM user or role	The root, an OU, or an account — not a role
Typical use	Scope which services a user or role may access	Deny a set of services across the organization

07 EXAM TRAPS

where the knowledge check gets you

- × “IAM groups can be nested inside other groups.” — No. A user can belong to several groups, but a group cannot contain another group; groups also have no credentials and cannot call AWS directly.
- × “An allow always beats a deny.” — Backwards. An **explicit deny** overrides any allow. Zhang's user policy that denies Kinesis wins over the group policy that allows it.
- × “An SCP grants permissions to member accounts.” — No. An SCP grants **nothing**; it is a guardrail that caps the maximum. You still need IAM identity-based policies to actually grant access.
- × “A permissions boundary and an SCP are the same thing.” — A boundary applies to a single IAM **entity**; an SCP applies to whole **accounts** in an org or OU. Neither grants permissions.
- × “You can attach an SCP to an IAM role, or create one in a member account.” — No. SCPs attach only to the **root, an OU, or an account**, and are managed from the organization's management account.
- × “A Cognito user pool lets the app reach AWS services directly.” — A user pool only authenticates and issues tokens. You need an **identity pool** to exchange those tokens for temporary AWS credentials.
- × “KMS can encrypt S3 or EBS data with an asymmetric key.” — AWS services integrated with KMS use **symmetric encryption keys only**; they do not support asymmetric KMS keys.
- × “STS credentials are long-lived like an access key.” — They are **temporary**, lasting from minutes to a few hours, and stop working once they expire.

08 SELF-QUIZ

answers are upside-down below

Q1 What single mechanism lets you grant the same permissions to many users organized by job function?

Q2 Which access-control strategy uses tags so one policy scales without listing each resource — RBAC or ABAC?

Q3 A user's own policy denies Kinesis while their group allows it. Can they use Kinesis, and why?

Q4 In identity federation, which party authenticates the user and which controls access to resources?

Q5 Which AWS service issues temporary, limited-privilege credentials when a role is assumed?

Q6 In Amazon Cognito, which component authenticates users and which grants temporary AWS credentials?

Q7 To which three targets can an SCP be attached?

Q8 Does an SCP grant permissions, and what must it be paired with?

Q9 How does envelope encryption protect data?

Q10 AWS services integrated with AWS KMS support which kind of KMS key only?

ANSWER KEY (rotate the page)

Q1 an IAM group **Q2** ABAC (attribute-based access control) **Q3** No — an explicit deny overrides the allow **Q4** the IdP authenticates; the SP controls resource access **Q5** AWS STS (Security Token Service) **Q6** the user pool authenticates; the identity pool grants AWS credentials **Q7** the root, an OU, or an account **Q8** No — it only sets a maximum; pair it with IAM identity-based policies that grant access **Q9** encrypt the data with a data key, then encrypt that data key under another (KMS) key **Q10** symmetric encryption keys