AWS Solution Architect Associate

Version : C03

Domain 1

Task 1

Design Secure Access to AWS Resources

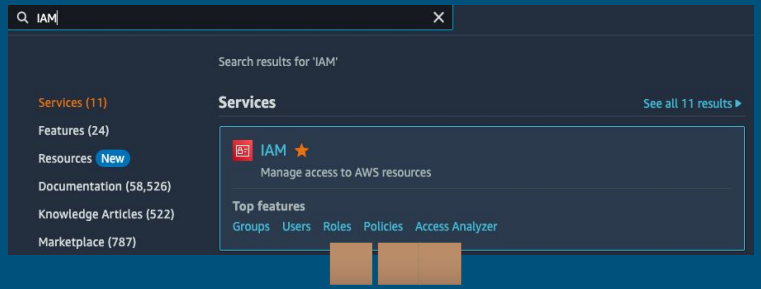
AWS Security Best Practices : IAM Users

- Require workloads to use temporary credentials with IAM roles to access AWS
- Require multi-factor authentication (MFA)
- Update access keys when needed for use cases that require long-term credentials
- Apply least-privilege permissions
- Use IAM Access Analyzer to generate least-privilege policies based on access activity
- Regularly review and remove unused users, roles, permissions, policies, and credentials
- Use conditions in IAM policies to further restrict access
- Use IAM Access Analyzer to validate your IAM policies , verify public and cross-account access to resources
- Establish permissions guardrails across multiple accounts
- Use permissions boundaries to delegate permissions management within an account

AWS Security Best Practices : Root User

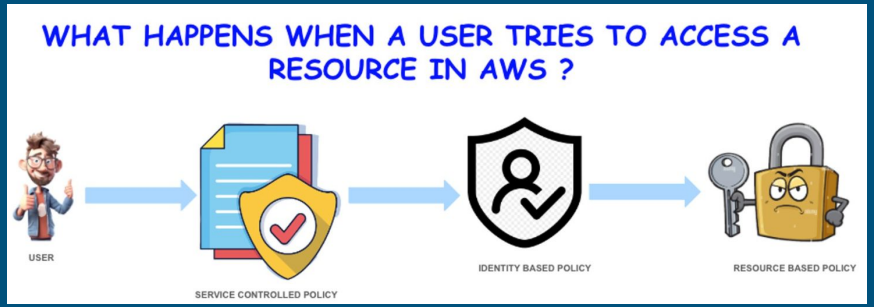
- Secure root user credentials to prevent unauthorized use
- Use a strong root user password to help protect access
- Use multi-factor authentication (MFA)
- Don't create access keys for the root user
- Use multi-person approval for root user sign-in wherever possible
- Use a group email address for root user credentials
- Restrict access to account recovery mechanisms
- Set preventative security controls in Organizations using a service control policy (SCP)
- Monitor access and usage
- Evaluate root user MFA compliance

IAM Identities

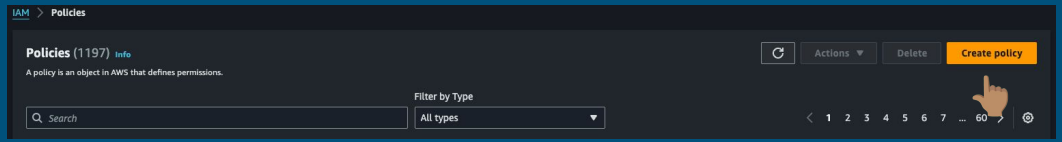


- **IAM Root User** :
  - Super User that has complete access to all AWS services and resources in the account
  - Accessed by signing in with the email address\password (Used while creating the account)
- **IAM User** :
  - Identity within AWS account that has specific permissions for a single person or application
  - Relying on temporary credentials instead of creating IAM users is recommended
- **IAM Group** :
  - Identity that specifies a collection of IAM users.
  - Can't be used to sign-in.
  - Can be used to specify permissions for multiple users at a time.
  - Make permissions easier to manage for large sets of users.
- **IAM Roles :**
  - Similar to an IAM user but isn't associated with a specific person
  - IAM role can be assumed in the AWS Management Console
  - Use case : Federated user access , Cross account access and Application running in EC2

IAM Policy



- IAM policies define permissions for an action regardless of the method that you use to perform the operation
- Different Policy Types
  - **Identity Based Policy** - Identity-based policies grant permissions to an identity
  - **Resource Based Policy** - Resource-based policies grant permissions to the principal that is specified in the policy. Irrespective of if in the same account as the resource or in other
  - **Permission Boundary** - Maximum permissions that the identity-based policies can grant to an entity
  - **Service Control Policy** - Limit permissions that identity-based policies or resource-based policies grant to entities (users or roles)

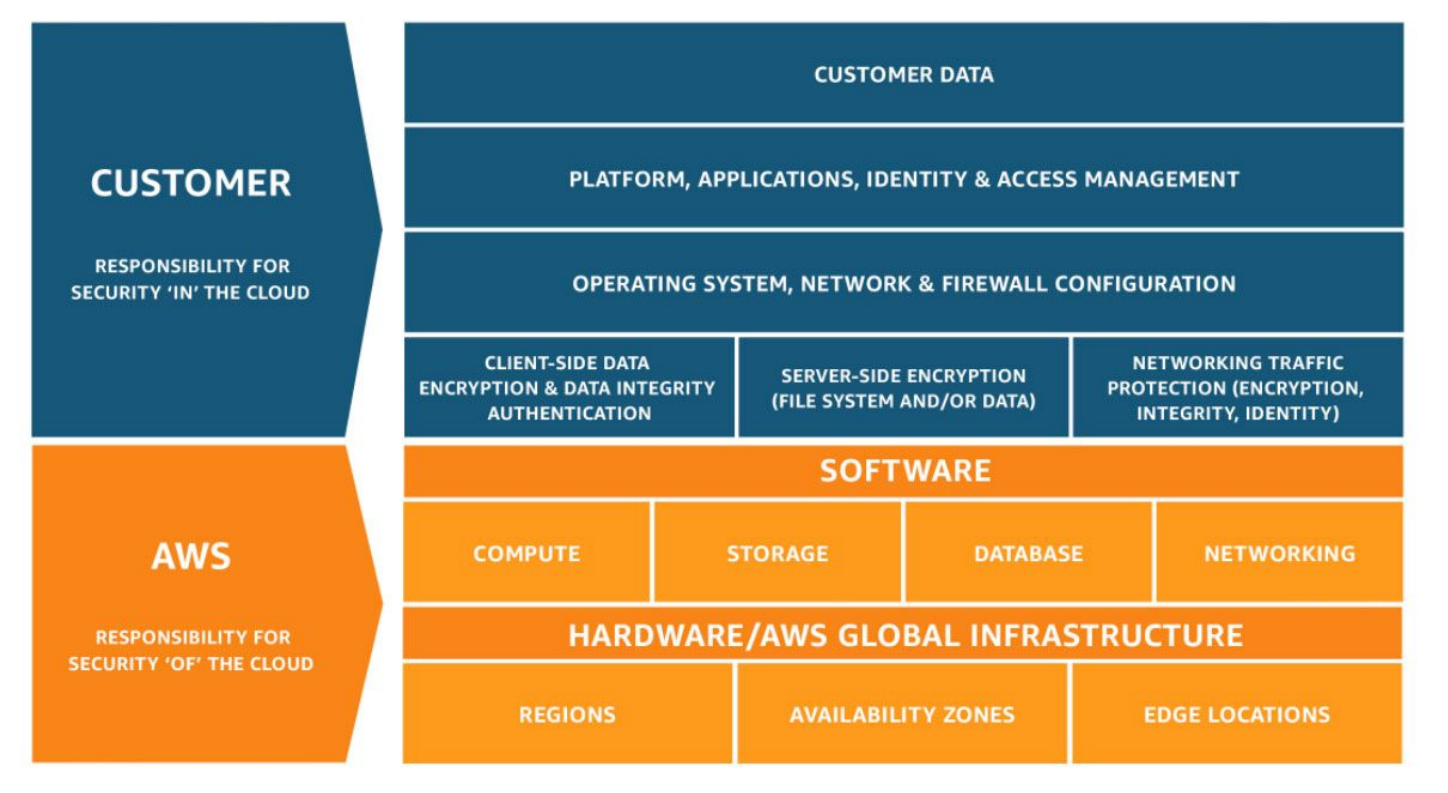


AWS Global Infrastructure

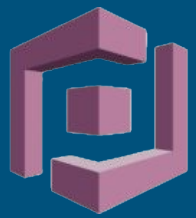
The AWS Cloud infrastructure is built around

- ➔ AWS Regions - A physical location in the world where we have multiple Availability Zones.
- ➔ Availability Zones(AZ's) -
  - ◆ Consist of one or more discrete data centers, each with redundant power, networking and connectivity, housed in separate facilities.
  - ◆ These AZ's offer the ability to operate production applications and databases that are more highly available, fault tolerant and scalable than would be possible from a single data center.

AWS Shared Responsibility Model



Role Based Access Control



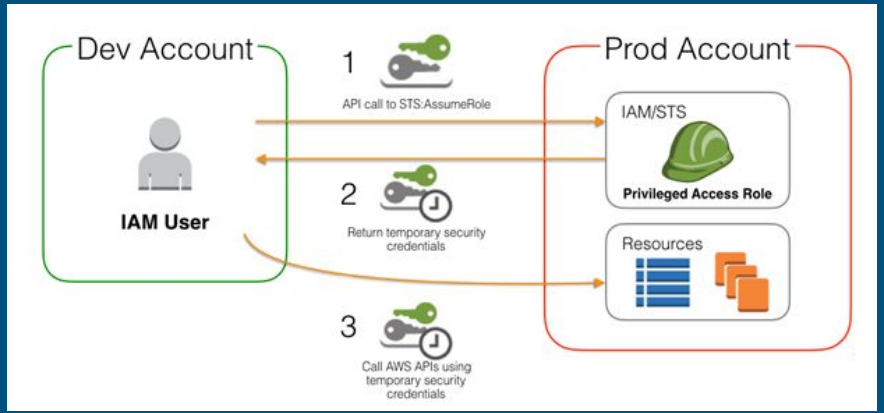
- Amazon Cognito identity pools assign authenticated users a set of temporary, limited-privilege credentials to access AWS resources.
- The permissions for each user are controlled through IAM roles that are created.
- Rules can be defined to choose the role for each user based on claims in the user's ID token.
- Default role can be defined for authenticated users.
- A separate IAM role with limited permissions can also be defined for guest users who are not authenticated.
- Iam:PassRole permission can be granted to allow a user to set roles with permissions in excess of the user's existing permissions

AWS Control Tower

- AWS Control Tower offers a straightforward way to set up and govern an AWS multi-account environment, following prescriptive best practices.
- It is built on top of trusted and reliable AWS services including AWS Service Catalog, AWS IAM Identity Center and AWS Organizations
- It extends the capabilities of AWS Organizations.
- Uses the controltower.amazonaws.com service principal.
- AWS Control Tower can be incorporated with other AWS services into a solution that helps migrate the existing workloads to AWS

AWS STS(Security Token Service)

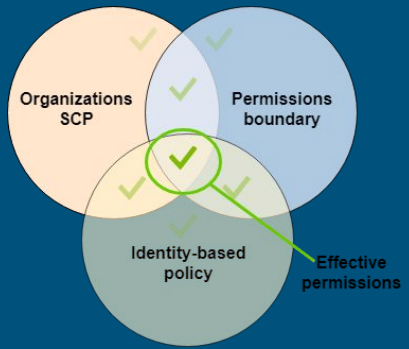
- ➔ A web service provided by AWS that enables to request temporary, limited-privilege credentials for users
- ➔ It is a global service and all requests go to a single endpoint at https://sts.amazonaws.com
- ➔ AWS recommends using Regional AWS STS endpoints instead of the global endpoint to reduce latency, build in redundancy, and increase session token validity
- ➔ This service has no quotas.



AWS Federated Access

- A federated identity is a user that can access secure AWS account resources with external identities.
- External identities can come from a corporate identity store (such as LDAP or Windows Active Directory) or from a third party (such as Login in with Amazon, Facebook, or Google).
- 2 AWS services can be used to federate workforce in AWS accounts and business applications
  - AWS IAM Identity Center
  - AWS Identity and Access Management
- Federation support can be added to the customer-facing web and mobile applications using Amazon Cognito. It helps to add user sign-up, sign-in and access control to mobile and web apps quickly and easily. It scales to millions of users and supports sign-in with social identity providers such as Apple, Facebook, Google and Amazon , and enterprise identity providers via SAML 2.0.

Permissions boundaries and Guardrails



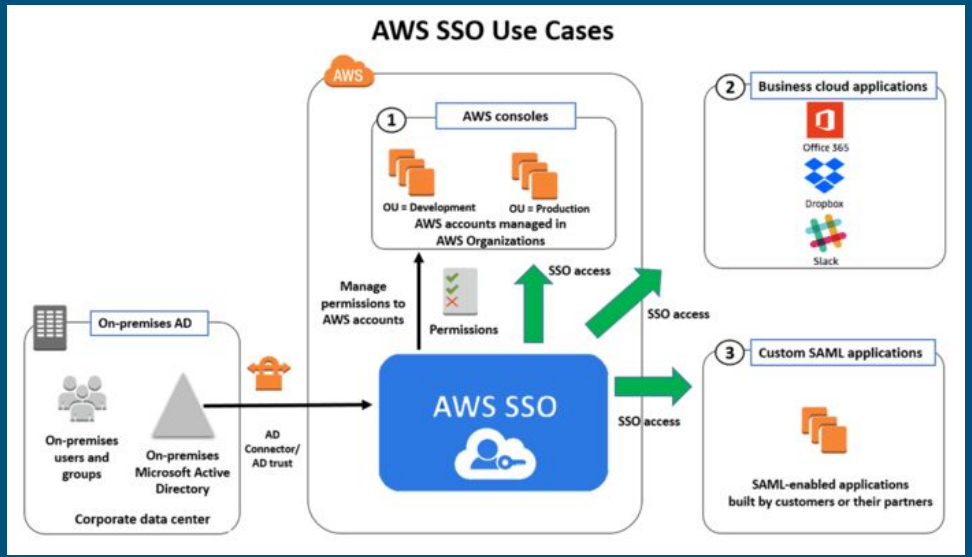
- ➔ Permissions boundary allows to set the maximum permissions that an identity-based policy can grant to an IAM entity.
- ➔ An entity's permissions boundary allows it to perform only the actions that are allowed by both its identity-based policies and its permissions boundaries.
- ➔ Permission guardrail refers to a mechanism or strategy used to enforce or manage permissions within an AWS environment

IAM Access Analyzer

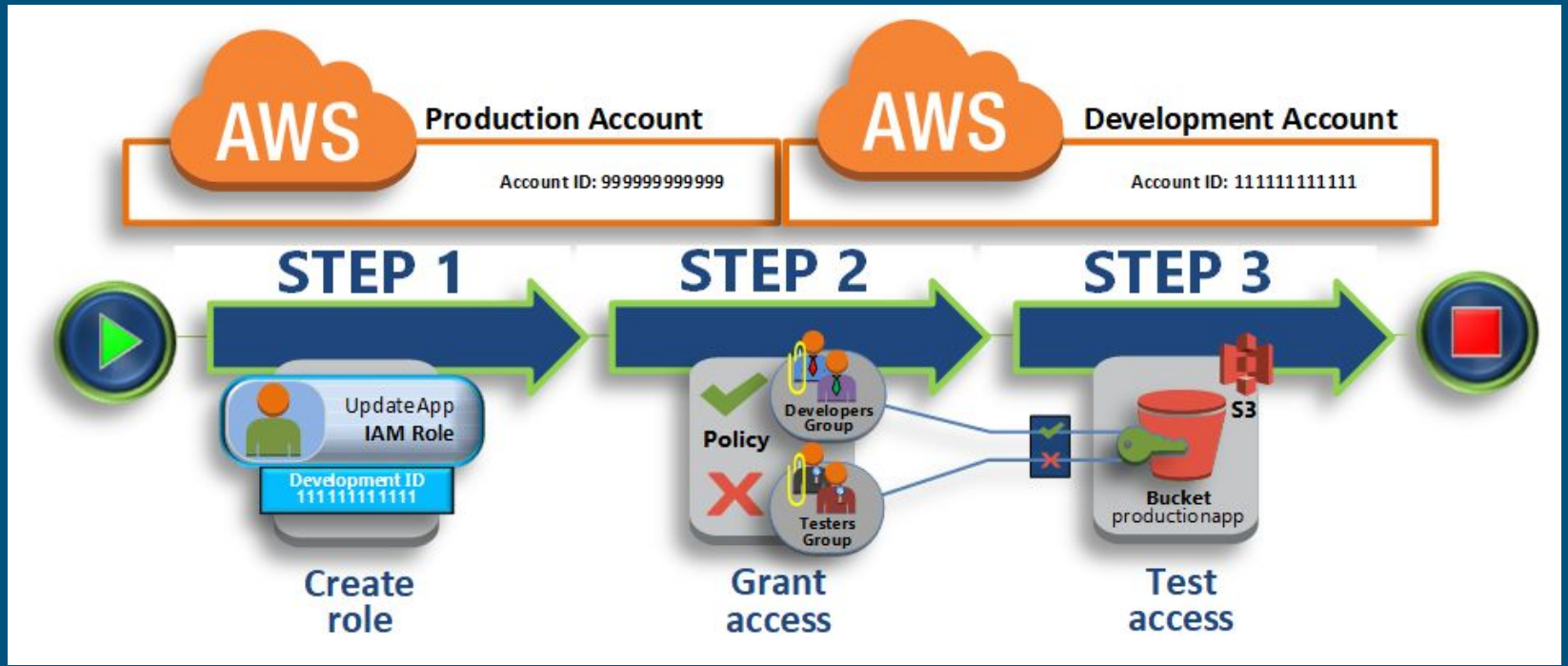
- AWS IAM Access Analyzer helps to set, verify and refine the IAM policies by providing a suite of capabilities.
- Its features include findings for external and unused access, basic and custom policy checks for validating policies and policy generation to generate fine-grained policies.
- To start using IAM Access Analyzer to identify external or unused access, analyzer has to be created first.
  - External access analyzers help identify potential risks of accessing resources by enabling you to identify any resource policies that grant access to an external principal
  - Unused access analyzers help identify potential identity access risks by enabling you to identify unused IAM roles, unused access keys, unused console passwords, and IAM principals with unused service and action-level permissions

AWS Single SignON (AWS-SSO)

- ➔ AWS IAM Identity Center is a cloud authentication solution that allows organizations to securely create or connect their workforce identities and manage their access centrally across AWS accounts and applications.
- ➔ User identities can be created or imported from external identity providers such as Azure.



Cross-account access to resources



AWS Solution Architect Associate

Version : C03

Domain 1

Task 1

The END