**Basik Marketing**

**Define identity security best practices on how to access the customer environment by leveraging IAM**

**Purpose**

The purpose of this Standard Operating Procedure (SOP) is to define a standard approach for accessing customer-owned AWS accounts. This SOP covers both AWS Management Console access and programmatic access using the AWS Command Line Interface or other custom tools. It also outlines when and how to use temporary credentials such as IAM roles and emphasizes the utilization of the customer's existing enterprise user identities and their credentials for accessing AWS services through Identity Federation or migrating to AWS Managed Active Directory.

**Procedure**

**AWS Management Console Access**

To access the AWS Management Console, follow these steps:

a. Navigate to the AWS Management Console login page.

b. Enter the appropriate username and password for the customer's enterprise user identity.

c. Complete any additional multi-factor authentication (MFA) requirements if enabled.

d. Upon successful authentication, the user will have access to the AWS Management Console.

**Programmatic Access**

**AWS Command Line Interface (CLI)**

a. Install the AWS CLI on the local machine if not already installed.

b. Configure the AWS CLI with the appropriate customer-owned AWS account credentials using the following command: aws configure.

c. Provide the AWS Access Key ID and Secret Access Key associated with the customer's IAM user or IAM role.

d. Set the desired default region and output format for the AWS CLI.

e. Once configured, the AWS CLI can be used to interact with AWS services programmatically.

**Temporary Credentials and IAM Roles**

Temporary credentials should be used when accessing AWS resources programmatically to minimize the risk of long-lived credentials being compromised.

**To use temporary credentials:**

a. Define IAM roles with the necessary permissions for accessing specific AWS services.

b. Assign the IAM roles to the appropriate customer's enterprise user identities.

c. Generate temporary credentials using AWS Security Token Service (STS) APIs or AWS CLI commands, based on the assigned IAM roles.

d. Utilize the temporary credentials for programmatic access to AWS services.

**Identity Federation or AWS Managed Active Directory**

Leverage the customer's existing enterprise user identities and their credentials to access AWS services.

Implement Identity Federation to establish trust between the customer's identity provider and AWS. This allows users to authenticate using their enterprise credentials.

Alternatively, migrate to AWS Managed Active Directory for centralized user management and authentication within AWS.

Additional Considerations

* Ensure that access to customer-owned AWS accounts follows the principle of least privilege, granting only the necessary permissions required for specific roles or tasks.
* Regularly review and audit access logs, IAM policies, and user permissions to identify and address any security vulnerabilities or unauthorized access.
* Implement multi-factor authentication (MFA) for all user accounts accessing the AWS Management Console.
* Provide training and guidance to customer personnel on secure access practices and AWS security best practices.
* Continuously monitor and stay informed about AWS service updates, security advisories, and best practices to ensure ongoing security and compliance.

**Implementing identity federation in AWS involves several steps. Here are five key steps to follow:**

**Set up an Identity Provider (IdP):**

Choose an Identity Provider (IdP) that supports Security Assertion Markup Language (SAML) 2.0 or OpenID Connect (OIDC) protocols.

Configure the IdP with the necessary settings, such as creating an application or service provider (SP) entity and configuring user attributes.

**Create an IAM role in AWS:**

Define an IAM role that will be assumed by federated users.

Assign appropriate permissions to the IAM role based on the desired level of access and privileges for federated users.

**Configure AWS as a Service Provider (SP):**

In the AWS Management Console, navigate to the IAM service and access the "Identity Providers" page.

Configure AWS as an SP by providing the necessary metadata from the IdP.

Set the attribute mapping between the IdP and AWS IAM.

**Establish a trust relationship:**

Establish a trust relationship between AWS and the IdP by exchanging metadata or certificates.

Validate and confirm the trust relationship between AWS and the IdP.

**Test and verify federation:**

Test the federation by initiating a login request from the IdP.

Validate that the federated user is able to assume the IAM role and access the desired AWS resources.

Perform thorough testing to ensure the federation is working as expected, including different scenarios and user roles.