**Hands-On with AWS Key Management Service (KMS)**

AWS Key Management Service (KMS) enables you to create and manage cryptographic keys and control their use across AWS services and applications.

**Objective**

1. Understand how to create and manage Customer Managed Keys (CMKs).
2. Use KMS for encrypting and decrypting data with an S3 bucket.
3. Integrate KMS with AWS services like EC2.

**Part 1: Create a Customer Managed Key (CMK)**

1. **Access the AWS KMS Console**:
   * Navigate to the **AWS Key Management Service (KMS)** console.
2. **Create a New Key**:
   * Select **Create Key**.
   * Choose **Symmetric Key** (used for encrypting and decrypting data).
   * Click **Next**.
3. **Configure Key Usage Permissions**:
   * Name your key (e.g., MyCMKKey).
   * Add a description for clarity.
   * Set key administrators (IAM users or roles who can manage this key).
4. **Add Key Usage Permissions**:
   * Define the IAM users or roles who can encrypt and decrypt data with this key.
5. **Complete the Key Creation**:
   * Review the configurations.
   * Click **Finish** to create the key.

**Part 2: Encrypt and Decrypt Data with AWS KMS**

1. **Install AWS CLI**:
   * Ensure AWS CLI is installed and configured on your machine:

bash

Copy code

aws configure

1. **Encrypt Data**:
   * Create a plain text file (e.g., mydata.txt):

bash

Copy code

echo "Sensitive data to encrypt" > mydata.txt

* + Encrypt the file using your KMS key:

bash

Copy code

aws kms encrypt \

--key-id alias/MyCMKKey \

--plaintext fileb://mydata.txt \

--output text \

--query CiphertextBlob > encrypted.txt

* + This creates an encrypted file, encrypted.txt.

1. **Decrypt Data**:
   * Use KMS to decrypt the data:

bash

Copy code

aws kms decrypt \

--ciphertext-blob fileb://encrypted.txt \

--output text \

--query Plaintext | base64 --decode

**Part 3: Enable S3 Bucket Encryption with AWS KMS**

1. **Create an S3 Bucket**:
   * Navigate to the **S3** console.
   * Create a bucket (e.g., my-kms-encrypted-bucket).
2. **Enable Default Encryption**:
   * In the **Bucket Settings**, enable **Default Encryption**.
   * Choose **AWS KMS** and select your created key (MyCMKKey).
3. **Upload and Test File Encryption**:
   * Upload a file to the S3 bucket.
   * AWS automatically encrypts the file using the KMS key.
   * Verify encryption by selecting the file in S3 and checking the **Encryption** section.

**Part 4: Use KMS with EC2**

1. **Launch an EC2 Instance**:
   * Open the **EC2** console and launch an instance.
2. **Attach an Encrypted Volume**:
   * Navigate to the **Elastic Block Store (EBS)** section.
   * Create a new volume:
     + Enable encryption and select your KMS key (MyCMKKey).
   * Attach the encrypted volume to your EC2 instance.
3. **Verify Encryption**:
   * SSH into the EC2 instance and check the attached volume:

bash

Copy code

lsblk

* + Ensure the volume is mounted and data is accessible.

**Part 5: Rotate the Key**

1. **Enable Key Rotation**:
   * Go to the **KMS Console**.
   * Select your key (MyCMKKey).
   * Enable automatic key rotation.
2. **Verify Rotation**:
   * AWS KMS will automatically create a new version of the key every 365 days (if enabled).

**Summary**

This hands-on covers:

1. Creating and managing a Customer Managed Key (CMK).
2. Encrypting and decrypting data.
3. Integrating KMS with S3 and EC2.
4. Enabling key rotation.