"Key management services"

Overview

AWS Key Management Service (KMS) simplifies cryptographic key management for data encryption in the AWS cloud. It seamlessly integrates with AWS services, enabling encryption at rest and client-side encryption. KMS offers granular control through custom key policies, ensuring secure access. The decryption process enforces access control, supports key rotation, and provides audit trails for security. Scalable and reliable, AWS KMS is a robust solution for safeguarding sensitive information across services and meeting diverse client needs.

Types of Encryptions:

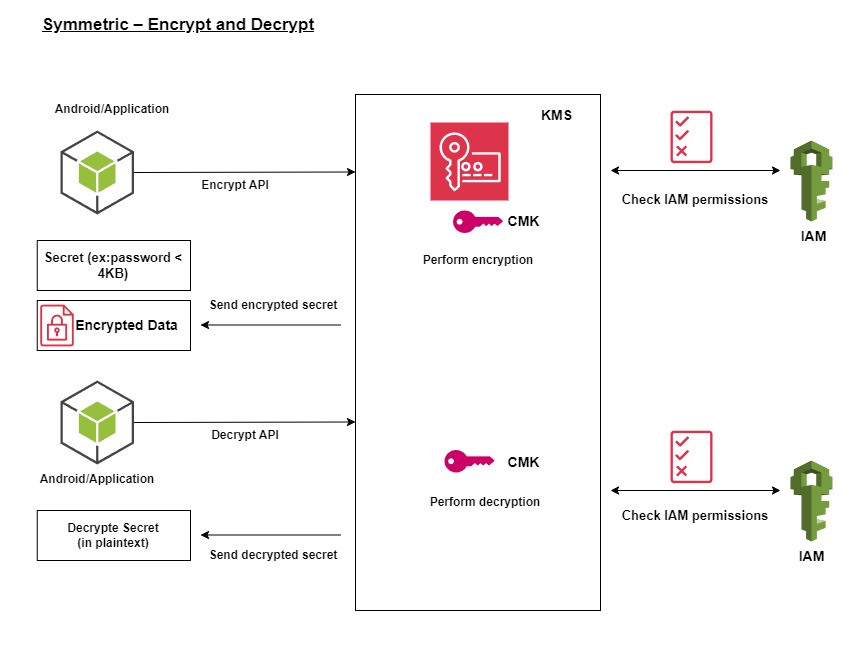
1. Symmetric Encryption:

* Key-based Encryption: Symmetric encryption uses a single secret key for both the encryption and decryption processes. The same key is used by both the sender and the recipient.
* Efficiency: Symmetric encryption is generally faster and computationally less intensive than asymmetric encryption, making it suitable for large volumes of data.

1. Asymmetric Encryption (Public Key Cryptography):

* Public and Private Keys: Asymmetric encryption involves a pair of keys – a public key for encryption and a private key for decryption. The public key can be freely shared, while the private key must be kept secret.
* Secure Key Exchange: Asymmetric encryption is often used for secure key exchange, digital signatures, and establishing secure communication channels.

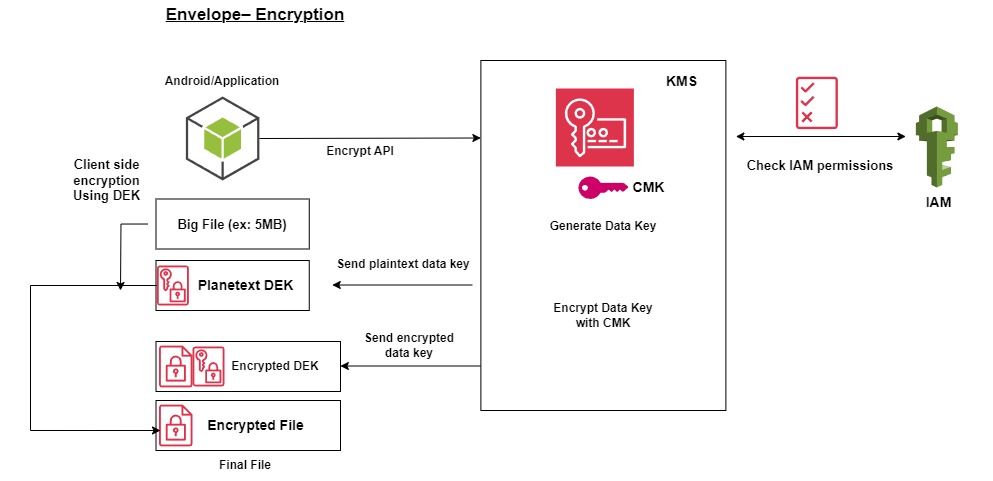
**Symmetric Encryption (for Asymmetric only key will change the API call remains same) :**



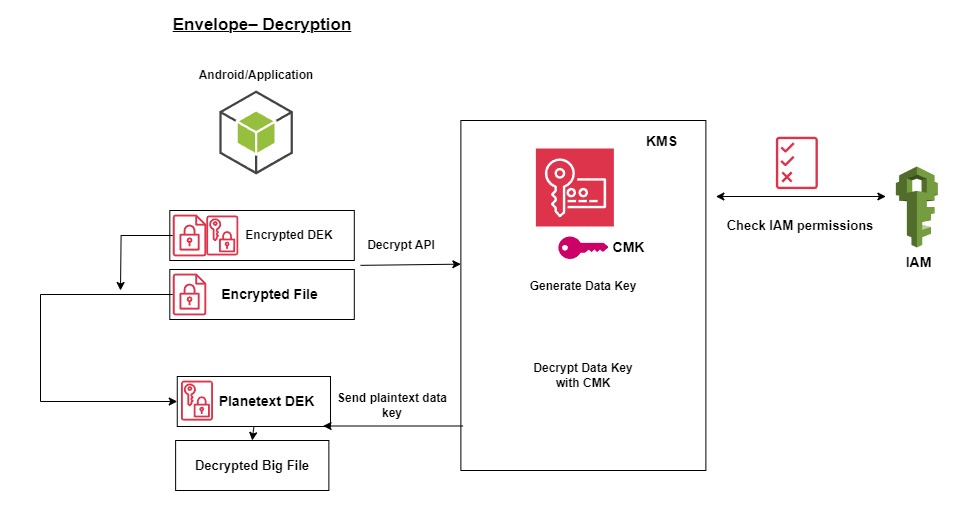
* Encrypt Data:
  + Call the Encrypt API of AWS KMS, providing the Key ID and plaintext data.
  + AWS KMS will return the ciphertext.
  + For Encryption: “**aws kms encrypt --key-id alias/AES-256-encryption-demo --plaintext fileb://ExampleSecretFile.txt --output text --query CiphertextBlob --region ap-south-1 > ExampleSecretFileEncrypted.base64**”
* Decrypt Data:
  + Use the Decrypt API of AWS KMS, passing the Key ID and the ciphertext.
  + AWS KMS will return the original plaintext data.
  + For Decript : “**aws kms decrypt --ciphertext-blob fileb://ExampleSecretFileEncrypted --output text --query Plaintext > ExampleFileDecrypted.base64 --region ap-south-1**”
* Key Rotation:
  + Optionally enable key rotation to automatically update the cryptographic material for the symmetric key.
* Key Policies:
  + The key policy associated with the symmetric key grants the necessary permissions for encryption and decryption.
* Integrate with Applications:
  + Integrate AWS KMS into your applications using AWS SDKs, CLI, or AWS Management Console for seamless encryption and decryption operations.

**Envelope Encryption with AWS KMS**

* Generate Data Key:
  + Use the GenerateDataKey API of AWS KMS to create a data key by providing the Key ID.
  + The returned data key consists of both a plaintext and an encrypted version.
* Encrypt Data:
  + Encrypt your sensitive data using the plaintext data key obtained from the previous step.
  + Store or transmit the encrypted data along with the ciphertext.
  + For Encryption: “**aws-encryption-cli --encrypt --input ExampleSecretFile.txt --wrapping-keys key="arn:aws:kms:ap-south-1:672261762453:key/6985a680-150b-4f17-a613-b31ac8eb64c9" --metadata-output metadata/metadata.txt --output output/**”



* Decrypt Data:
  + To decrypt, call the Decrypt API of AWS KMS, passing the encrypted data key and the Key ID.
  + AWS KMS will return the plaintext data key.
  + Use the plaintext data key to decrypt the sensitive data.
  + For “Decryption: “**aws-encryption-cli --decrypt --input output/ExampleSecretFile.txt.encrypted --wrapping-keys key="arn:aws:kms:ap-south-1:672261762453:key/6985a680-150b-4f17-a613-b31ac8eb64c9" --metadata-output metadata/metadata\_decript.txt --output output/ExampleSecretFile.txt.decrypted**”



* Key Rotation:
  + Optionally, if key rotation is enabled, periodically rotate the data key to enhance security.
* Key Policies for Envelope Encryption:
  + Ensure that the key policy associated with the master key grants the necessary permissions for generating data keys and decrypting.
* Integrate in Applications:
  + Modifying applications to incorporate the additional step of generating and managing data keys for envelope encryption.