# Day 21

# “CLOUD SECURITY”

## Implementing AWS Keys Management Services:

**Why do we need encryption?**

We need to encrypt data to protect its confidentiality and integrity, ensuring that only authorized users can access or modify it. Encryption transforms readable data into an unreadable format, making it useless to attackers or unauthorized parties who might intercept or steal it. This is especially important for sensitive information such as personal details, financial records, or business secrets, helping prevent data breaches, comply with regulations, and maintain trust.

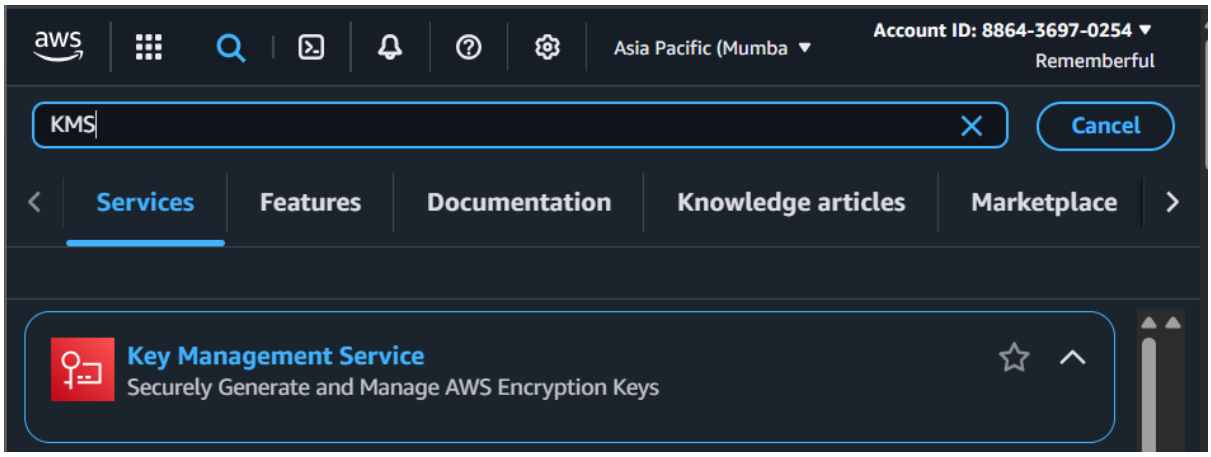
**Two main methods to implement encryption at rest:**

- **Client-Side Encryption:** Data is encrypted on the user’s device (client) before it is sent to the server. This means only the client holds the encryption keys, so even the server storing the data cannot read it unless the client shares the key.
    - You encrypt your data and manage your own keys.
    - **Use KMS** if required
- **Server-Side Encryption:** Data is sent to the server in plain text, and the server encrypts it before storing. The server manages the encryption keys and handles both encryption and decryption when needed.
    - **AWS encrypts your data** and manages the key for you
    - Most AWS services like S3, EBS, Redshift provide server-side Encryption using KMS behind the scenes

## What is AWS KMS?

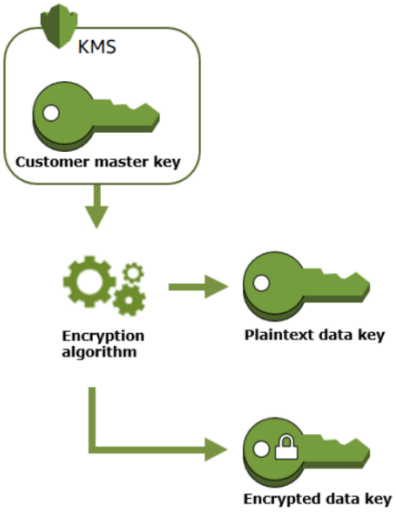
**AWS** Key Management Service (KMS) is a fully managed service that makes it easy to create, control, and manage encryption keys used to encrypt your data. It helps you securely generate, store, and control cryptographic keys used across AWS services and your applications, ensuring data protection and compliance. AWS KMS integrates with many AWS services for seamless encryption and supports key rotation, access control via IAM policies, and detailed logging for auditing.

Where can we find this KMS?



**What is CMK?**

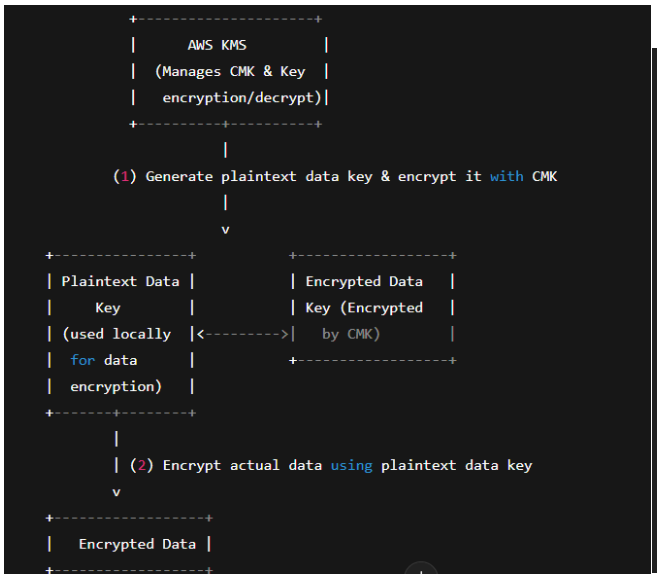
Customer Master Key (CMK) is a logical representation of a master key in AWS Key Management Service (KMS). It is the primary encryption key you create, manage, and use to encrypt and decrypt data encryption keys (DEKs) or directly encrypt small amounts of data. CMKs can be either customer-managed (created and controlled by you) or AWS-managed (managed by AWS). They include metadata, key material, usage policies, and permissions that control how and by whom the key can be used.




In short:

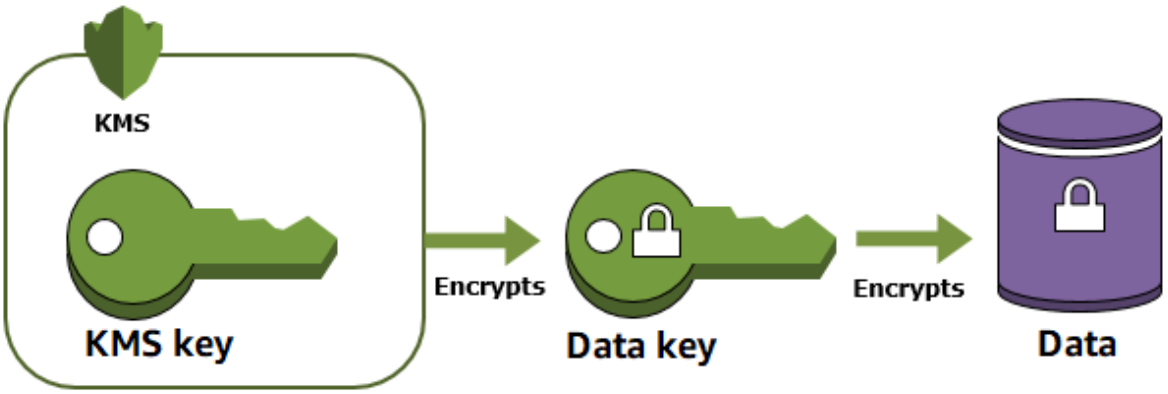
- **KMS** is the service platform.
- **CMK** is the specific encryption key that KMS handles to encrypt, decrypt, and control access to data.

So, CMKs live inside KMS and are used by KMS to perform encryption operations securely.



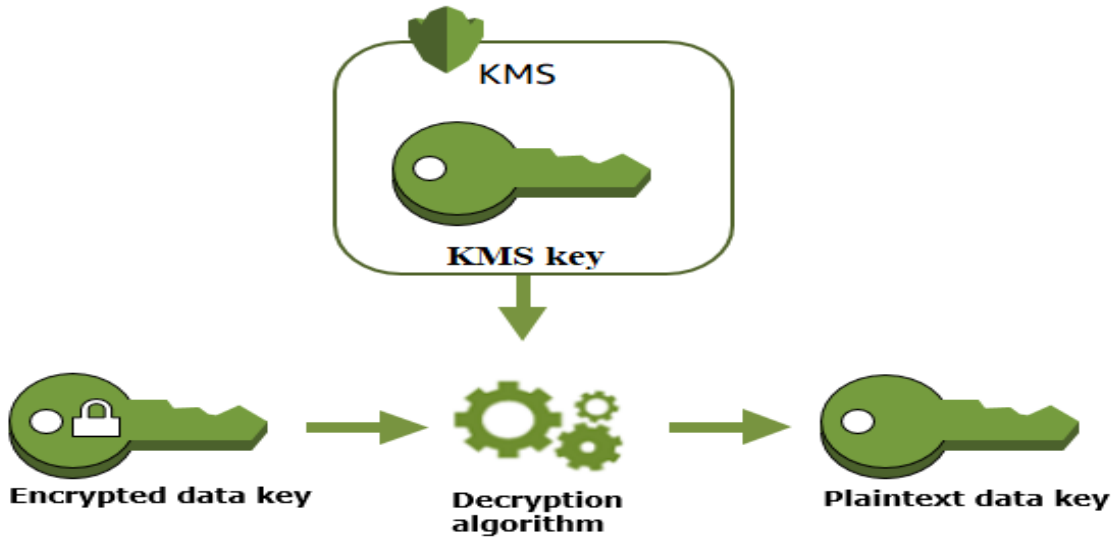

**How to Encrypt data?**

1. KMS generates a plaintext data key and returns it *along with* an encrypted data key (encrypted by the CMK).
2. The plaintext data key is used locally to encrypt the actual data.
3. The encrypted data and encrypted data key are stored together.
4. When decrypting, the encrypted data key is sent back to KMS, which uses the CMK to decrypt it, returning the plaintext data key.
5. The plaintext data key then decrypts the actual data.




**How to decrypt data?**

1. The encrypted data key is retrieved from storage along with the encrypted data.
2. This encrypted data key is sent to AWS KMS, which uses the CMK to decrypt it and returns the plaintext data key.
3. The plaintext data key is then used locally to decrypt the actual data, restoring it to its original form.




--The End--