Model A

In this model the Encryption method and the KMI (Key storage & Key Management) should be completely managed by the user.

The important security property of this model is that the user has full control over the Encryption method and Keys. The encryption method can be a combination of open-source tools, AWS SDKs, or third-party software and/or hardware. Physical location of the KMI and the encryption method can be outside of AWS or in an Amazon Elastic Compute Cloud (Amazon EC2) instance of the user. AWS has no access to the keys and cannot perform encryption or decryption on the user's behalf. The user is responsible for the proper storage, management, and use of keys to ensure the confidentiality, integrity, and availability of the data.

Features of this model:

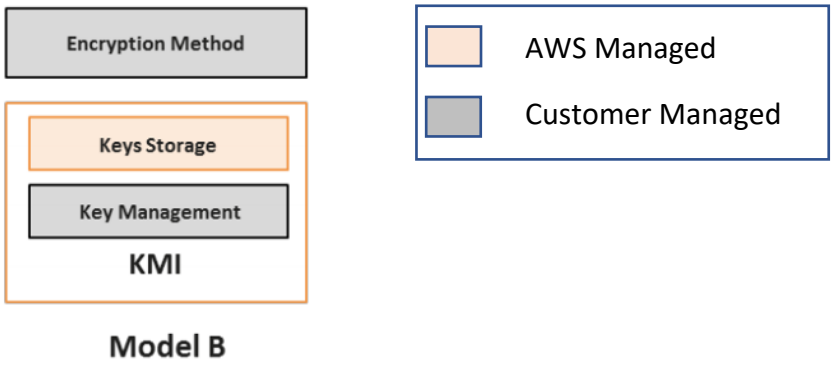
- Full control
- Highly customizable
- Needs more effort to build and maintain

In our implementation: Below are the details user need to provide

To Upload a file to S3: AWS IAM credentials with access to required Bucket, File for uploading, location to store Key.

To Download a file from S3: AWS IAM credentials with access to required Bucket, File name to be downloaded, Key to decrypt the file, location to save the downloaded file.

MODEL B:



Model B

In this model user still controls the Encryption method and Key management part of KMI. AWS provides the storage component of the KMI with a service called Cloud HSM.

This model is mostly similar to Model A, the only difference from Model A is that the keys are stored in AWS CloudHSM cluster rather than in a key storage system managed by user or third party. While the keys are stored in the AWS environment, they are inaccessible to any employee at AWS, this is because only the user has access to the cryptographic partitions within the dedicated HSM to use the keys. The AWS CloudHSM appliance has both physical and logical tamper detection and response mechanisms that trigger zeroization of the appliance. Zeroization erases the HSM's volatile memory where any keys in the process of being decrypted were stored and destroys the key that encrypts stored objects, effectively causing all keys on the HSM to be inaccessible and unrecoverable.

Features of this model:

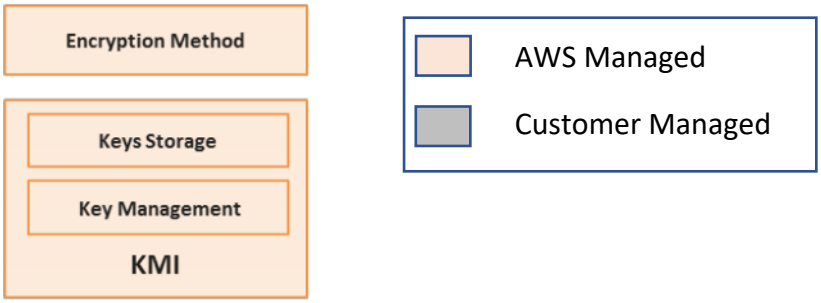
- Highly secure
- Moderately customizable
- Costly

In our implementation: Below are the details user need to provide

To Upload a file to S3: AWS IAM credentials with access to required Bucket, File for uploading, Cloud HSM cluster to upload the key file.

To Download a file from S3: AWS IAM credentials with access to required Bucket, File name to be downloaded, Key file name to decrypt the file, Cloud HSM cluster name to download the Key, location to save the downloaded file.

MODEL C:



Model C

In this model AWS controls the Encryption method and the entire KMI

AWS Key Management Service is a managed encryption service that lets the user to use keys to encrypt their data in AWS services and their applications. Master keys in AWS KMS are used in a fashion similar to the way master keys in an HSM are used. After master key are created, they are designed to never be exported from the service. Data can be sent into the service to be encrypted or decrypted under a specific master key under the user's account. This design gives the user centralized control over who can access the master keys to encrypt and decrypt data, and it gives user the ability to audit this access. AWS KMS is natively integrated with other AWS services including Amazon EBS, Amazon S3, and Amazon Redshift to simplify encryption of the data within those services. For applications that need to encrypt data, AWS KMS provides global availability, low latency, and a high level of durability for the keys.

Features of this model:

- Highly secure
- Limited customizability
- Ready to use service

In our implementation: Below are the details user need to provide

To Upload a file to S3: AWS IAM credentials with access to required Bucket, File for uploading.

To Download a file from S3: AWS IAM credentials with access to required Bucket, File name to be downloaded.