S3 Server Side Encryption

S3 can use three methods for Server Side Encryption

1. SSE-S3

This method uses S3 encryption keys

1. SSE-KMS

This method will use KMS keys

1. SSE-C

This method will use customer encryption keys

Customers can create their own encryption keys using the following command

* Openssl enc -aes-256-cbc -k secret -P

Use the key with aws cli commands to encrypt a file that is being uploaded to S3. This file will not be accessible in S3 since the key is not available in S3 and it resides on the user’s machine.

Aws s3 cp test.txt s3://new-bucket-am --sse-c --sse-c-key <key>

Encrypt files in S3 using customer encryption keys

1. Create an EC2 instance
2. Connect to the EC2 instance
3. Use the AWS config command to provide your private and secret keys to run aws commands
   * Aws config
4. Create a key using the openssl command
   * Openssl enc -aes-256-cbc -k secret –P
5. Copy key to a text file named key
   * Echo <key> > key
6. Create a text file using the following command
   * Echo test > test.txt
7. Use the AWS cli command to create bucket
   * Aws s3 mb s3://new-bucket-am
8. Use the aws cli command to copy test.txt to new bucket and encrypt the file before uploading
   * Aws s3 cp test.txt s3://new-bucket-am --sse-c --sse-c-key <key>
9. The file can only be opened if it’s downloaded to system again otherwise, while it’s in S3, the content is not accessible. The file can’t even be downloaded if the key is not available
   * Aws s3 cp s3://new-bucket-am/test.txt . --sse-c --sse-c-key <key>

Encrypt Files in S3 using SSE-S3 or SSE-KMS

1. When uploading files using the console, you have the option to use sse-s3 or sse-kms to encrypt your files
2. To use sse-kms, first create your kms keys. Then select the keys during the encryption process