Task Statement 2: Implement Encryption by using AWS Services

Task 1: Use SSE-S3 and SSE-KMS for S3 Bucket

Goal: Understand the difference between server-side encryption with Amazon S3 managed keys and AWS KMS keys.

Steps:

- Create an S3 bucket and upload an object with SSE-S3 enabled.
- Upload another object using SSE-KMS with an AWS-managed key.
- Check encryption metadata in the S3 object properties.
- Enable default encryption at the bucket level for future uploads.

Concepts covered: SSE-S3, SSE-KMS, encryption at rest, AWS managed keys.

Task 2: Create and Use a Customer Managed Key (CMK)

Goal: Use AWS KMS to create and manage your own encryption keys.

Steps:

- Create a CMK in AWS KMS with key rotation enabled.
- Encrypt a small text string using the KMS Encrypt API via CLI.
- Decrypt the string using the KMS Decrypt API.
- Use the CMK to encrypt an S3 object.

Concepts covered: CMK, KMS APIs, key rotation, customer-managed keys.

Task 3: Encrypt and Decrypt with Envelope Encryption

Goal: Practice encrypting large files using envelope encryption.

Steps:

- Use AWS KMS to generate a data key.
- Use the plaintext key to encrypt a local file.
- Store the encrypted file and the encrypted data key.
- Decrypt the key via AWS KMS and decrypt the file.

Concepts covered: Envelope encryption, data keys, KMS GenerateDataKey.

Task 4: Use SSL/TLS in API Gateway

Goal: Ensure data in transit is encrypted using HTTPS endpoints.

Steps:

- Create a simple API using API Gateway.

- Deploy the API and test with HTTPS calls.
- Verify the TLS certificate used in the browser/dev tools.

Concepts covered: Encryption in transit, TLS, secure APIs.

Task 5: Generate and Use Self-Signed Certificates

Goal: Understand development-level certificate handling.

Steps:

- Use OpenSSL to generate a self-signed certificate and private key.
- Inspect certificate details using OpenSSL commands.
- Optionally, upload the certificate to ACM for use in CloudFront.

Concepts covered: OpenSSL, self-signed certs, ACM, certificate management.

Task 6: Share KMS Key Across Accounts

Goal: Allow another AWS account to use your KMS key.

Steps:

- Edit the key policy of a CMK to allow access from another account.
- Switch to the second account and test encryption/decryption.
- Review logs in CloudTrail for cross-account KMS usage.

Concepts covered: KMS cross-account access, key policies, CloudTrail.

Task 7: Explore Key Rotation and Permissions

Goal: Understand the security and governance around key rotation.

Steps:

- Enable and disable automatic rotation for a CMK.
- Create an IAM policy that restricts use of KMS keys by tag.
- Use CloudTrail to verify access attempts and key usage.

Concepts covered: Key rotation, IAM policies, tagging, CloudTrail logging.