Cloud Storage Encryption Review Report

1. Default Encryption Review (SSE-S3, SSE-KMS)

A sample file was uploaded to an AWS S3 bucket to evaluate default encryption. By default, AWS S3 supports encryption at rest using Server-Side Encryption (SSE). SSE-S3 encrypts each object with a unique key managed by AWS. Alternatively, SSE-KMS allows integration with AWS Key Management Service, offering audit logs and granular access control. Data in transit is encrypted using HTTPS (TLS).

2. Client-Side Encryption Implementation

Client-side encryption was implemented using OpenSSL. First, a 256-bit AES key was generated. Then, a sensitive file was encrypted using AES-256-CBC mode and uploaded to the S3 bucket. Decryption was tested locally using the same key and IV. This ensures data remains encrypted before it reaches the cloud provider, adding an extra security layer.

3. Access Control Testing

Bucket policies and IAM roles were configured to limit access to the encrypted file. Attempts to access the object from an unauthorized IAM user resulted in an AccessDenied error, verifying that encryption combined with strict access controls effectively safeguards the data.

4. Comparison: SSE-KMS vs Client-Side Encryption

Cloud-provided encryption (e.g., SSE-KMS) is easier to manage, integrates with AWS services, and logs key usage. However, it relies on AWS for key management. Client-side encryption gives full control over encryption keys but increases complexity and requires secure key distribution and storage. SSE is ideal for most use cases, while client-side encryption is better suited for highly regulated or extremely sensitive data.

5. Recommendation

Based on testing and analysis, it is recommended to use SSE-KMS for most sensitive data due to its balance between security and manageability. For extremely sensitive use cases, client-side encryption should be considered, provided key management and distribution are securely handled.