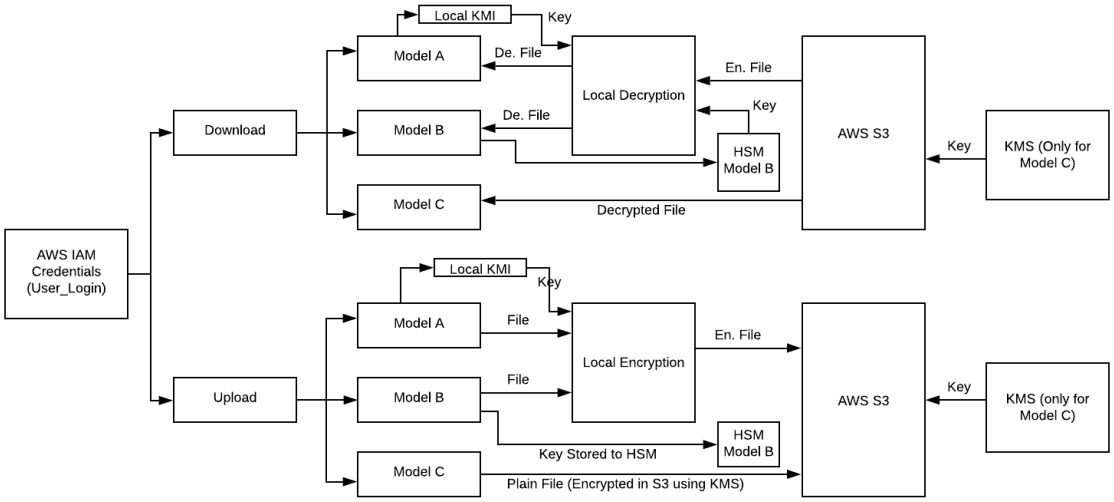
Implementation of Data encryption models in Cloud

Problem:

There are 3 different models of Data Encryption in cloud. To understand clearly and have practical knowledge of these models we have implemented these three models of Encryption into an Application. This application enables users to experience the three models of encryption, which helps them in understanding the models and also equip them to decide a model as per their need and infrastructure.

Conceptual Design:



Architecture of the project

Above figure depicts the architecture of our project. In short, this application uploads given files to S3 bucket selected by user according to process A or B or C selected by user. More details about Model A, Model B & Model C can be known from below document for three models.

Implementation description:

Application type: Standalone
Development Platform: Python
Libraries: tkinter (for UI), boto3 (AWS functionality using python), cryptogrpahy (for local encryption)

Modules:

Authentication: We are using AWS IAM credentials to authenticate the user. Application creates a session with given AWS credentials and checks if the given credentials are valid.
Local encryption (Model A & Model B): We are using Cryptography library available in python to generate an AES key randomly and Encrypt the given file.
Encryption & Decryption using KMS (Model C): Application uses KMS to encrypt and decrypt the file during upload and download respectively in Model C.
Local decryption (Model A & Model B): user provide the keys for Decrypting file downloaded from S3, Application uses Python cryptography library to decrypt the file with given key.

User guide:

System requirements: Any OS, minimum RAM requirements, Internet connection
Software requirements: Python 2.7 or later, AWS SDK installed, Pillow (Python Image library)

Steps for using the Application:

1. Copy the project folder into desired folder
2. Unzip the contents
3. Open terminal/command prompt and navigate to the unzipped project folder
4. Execute **python AWS_project.py**

Above steps allow the user to use the application in any system which satisfies given Software and System requirements.

Self-evaluation:

The main goal of the project is to develop the three models of encrypting data at rest-ranging from completely automated AWS encryption solutions to manual, client-side options. We have achieved this by designing our own encryption and decryption methods in Models A and B. In Model C, we were able to successfully make the connection to the AWS server-side encryption which then takes care of the encryption as well as the key storage and management.

Additional features:

- Added a functionality which allows user to create a new S3 bucket to upload files.

References:

1. COMS 559 Lecture Slides – (M4_s20_AWS_Security-4)

2. Securing Data at Rest with Encryption - Ken Beer and Ryan Holland
https://d0.awsstatic.com/whitepapers/aws-securing-data-at-rest-with-encryption.pdf

3. Getting Started with AWSCloudHSM
https://docs.aws.amazon.com/cloudhsm/latest/userguide/getting-started.html

4. Documentation of tkinter library
https://docs.python.org/3/library/tkinter.html

5. Documentation of Cryptography library
https://cryptography.io/en/latest/

6. Documentation of boto3
https://boto3.amazonaws.com/v1/documentation/api/latest/index.html