Securing the Cloud - Lucia Brown

My cloud security with encryption and decryption project consists of two Python files – one containing the secure cloud functionality (secure.py) and a test script with three test cases. The test cases ensure that specified users are able to decrypt files available on the server; that newly added users can also access the functionality; and that removed users can no longer decrypt.

Secure.py Overview:

generate_user_cert: This function generates public and private keys for the users using RSA with a key size of 2048. The generated files are in the format {username}_public.pem and {username}_private.pem.

load_public_key_from_cert: This function extracts the user's public key from their generated public.pem file

encrypt_aes_key: This function encrypts the AES key, using a public RSA key.

decrypt_aes_key: This function decrypts the AES key, using a private RSA key.

encrypt_file: This function generates an AES-256 key, reads in plaintext and uses the AES key to encrypt the plaintext. The encrypted ciphertext is written as a document in the format {filepath}.enc

upload_to_cloud: This function uploads the sample file to both GoogleDrive and Dropbox in parallel. The sample.txt.key and sample.txt.enc are uploaded.

decrypt_file: This function takes in the path of the encoded file, a username and the user's prviate key path. It looks for the encrypted file at the desired file path, checks if the user can decrypt the file and if they can, their private key is used for decryption.

add_user_to_group: This function takes in a username and a person's public key as parameters and adds the person to the user group of those who are able to decrypt the files from the cloud. It loads the public key and encrypts the AES key for the

remove_user_from_group: This function takes in a username and searches for that username in the user group. If the user is found, they and their key will be removed from the user group and will lose access to decryption.

Test_Script.py Overview:

There are three test cases in this script.

Test 1: This test adds Alice and Bob to the user group and ensures that they are able to decrypt the file while securely stored on the cloud.

Test 2: After the user group is initialised with just Alice and Bob, Charlie is added to the user group. This test ensures that Charlie is also able to decrypt and access the file.

Test 3: Bob is removed from the user group and should no longer be able to decrypt the file.