**Triple DES Image Encryption and Decryption**

**Name: Ashish Mishra-21CSB0B05**

1. **Introduction**

Triple DES (Data Encryption Standard) is a symmetric-key encryption algorithm widely used for securing sensitive data. It's an enhancement of the original DES algorithm, which encrypts data in 64-bit blocks using a 56-bit key. Triple DES applies DES three times, employing two or three distinct keys for each encryption and decryption operation.

**Encryption Process:**

**Key Generation:** Triple DES requires two or three 56-bit keys, denoted as K1, K2, and optionally K3. If using two keys, K1 and K2 are used for encryption and decryption. If three keys are used, K1 encrypts, K2 decrypts, and K3 encrypts again.

**Initial Permutation (IP):** The input data (e.g., an image) is permuted according to a fixed table, providing some initial diffusion.

**Encryption Rounds (3 or 2):** Each round encrypts the data using the DES algorithm with one of the keys.a. Round 1: Data is encrypted using K1.b. Round 2: If using three keys, data is decrypted using K2. Otherwise, it's encrypted again using K1.c. Round 3: Data is encrypted using K3 if three keys are used. If only two keys are used, data is encrypted using K2.

**Final Permutation (FP):** After the last round, a final permutation is applied to the data, completing the encryption process.

**Decryption Process:**

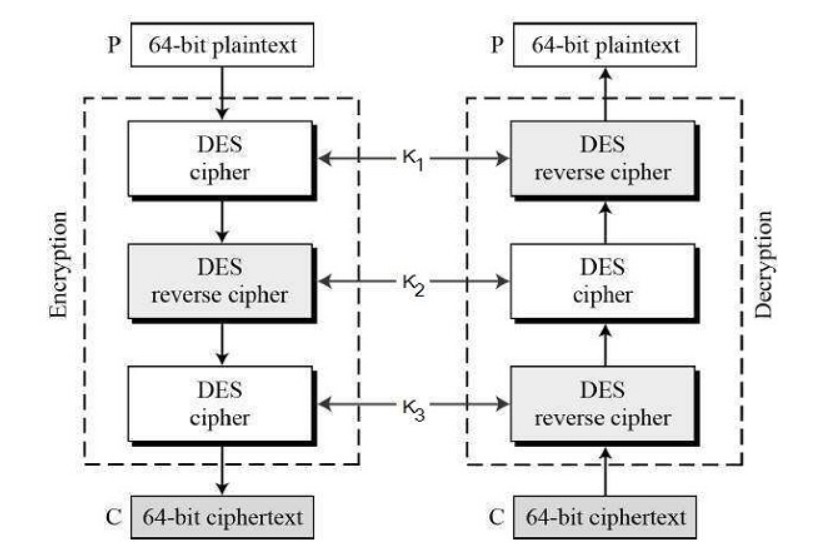
Decryption in Triple DES is essentially the reverse of encryption:

**Key Setup:** Same as encryption, keys are generated.

**Initial Permutation (IP):** The ciphertext (encrypted data) undergoes the initial permutation.

**Decryption Rounds (3 or 2):** The ciphertext is decrypted using the keys in reverse order of encryption.a. Round 1: Decrypt using K3 if three keys are used. Otherwise, encrypt using K2.b. Round 2: If three keys are used, encrypt using K2. Otherwise, decrypt using K1.c. Round 3: Decrypt using K1.

**Final Permutation (FP):** The data is permuted one final time, resulting in the original plaintext.



1. **Objectives**

* **Confidentiality:** The primary objective of encryption is to ensure that only authorized parties can access the sensitive information contained in the image. By encrypting the image using Triple DES, unauthorized users cannot view its contents without the appropriate decryption key.
* **Data Integrity:** Encryption helps maintain the integrity of the image data by preventing unauthorized modification or tampering during transmission or storage. Triple DES ensures that even if an attacker intercepts the image, they cannot alter it without detection.

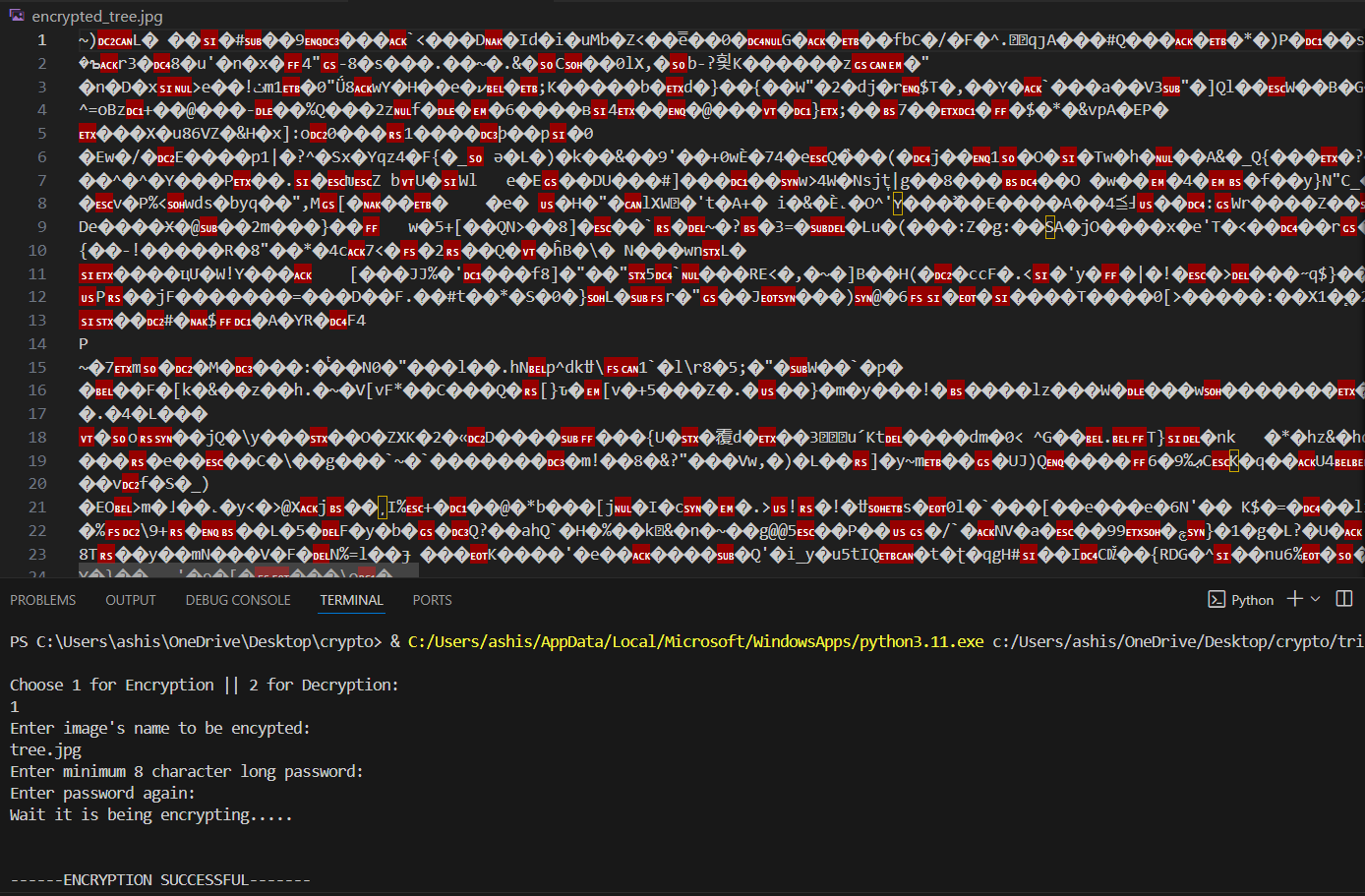
1. **Implementation and Result Analysis**

In this standard the encryption method is like the one in original DES but applied 3 times to increase the encryption level. Triple DES systems are significantly more secure than single DES, but these are clearly a much slower process than encryption using single DES.

The encryption function: Takes a file path as input, attempts to open an image file in binary mode at that path, and reads its contents. It then pads the image data with spaces until its length is divisible by 8, likely for alignment purposes in encryption algorithms. If any error occurs during the file loading process, it prints an error message indicating potential issues with the file's location, name spelling, or corruption, and exits the program. Overall, the function prepares an image file for encryption, ensuring it meets the requirements for certain encryption algorithms, but it does not perform encryption itself.

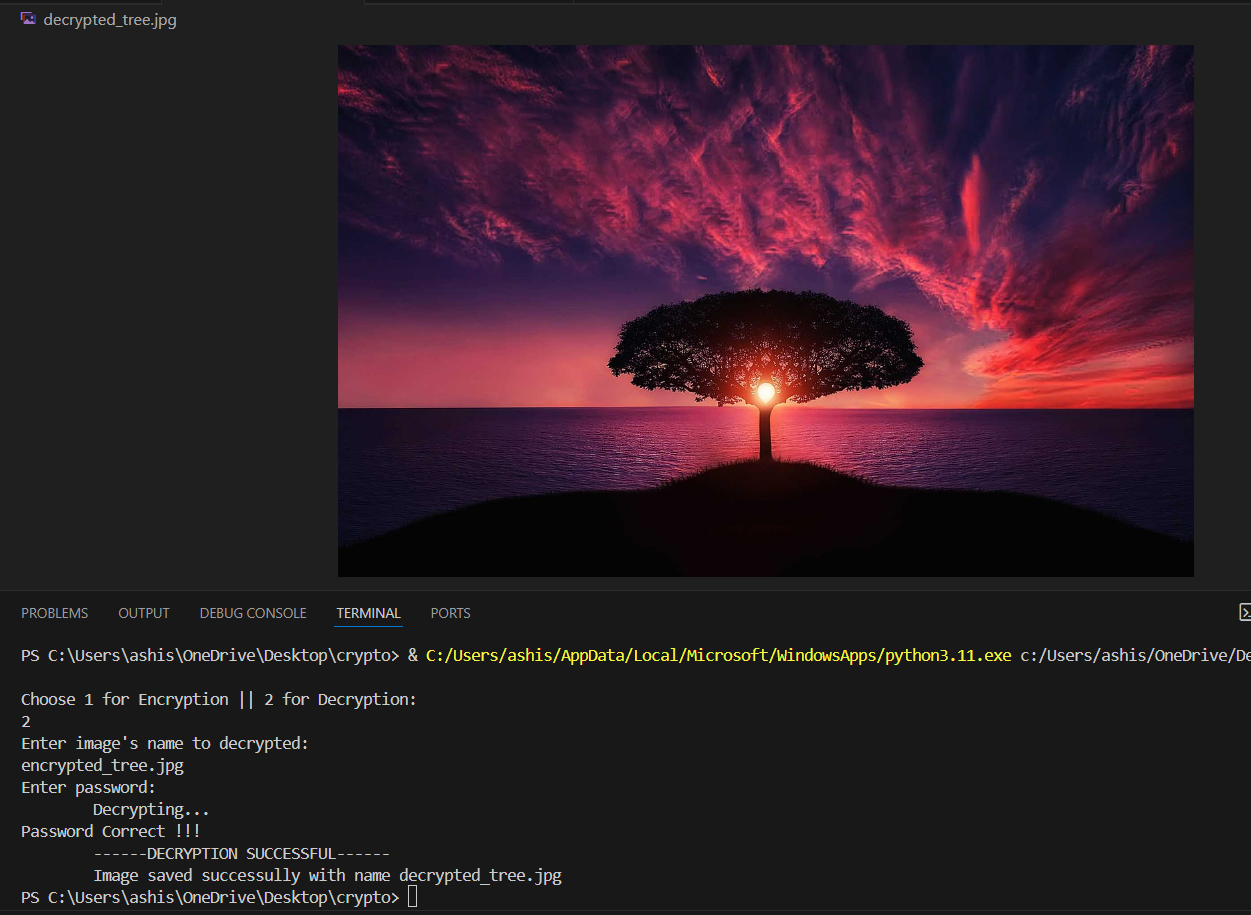
Hashing and Password Verification:The code snippet begins by creating a SHA256 hash of the original image data. Then, it prompts the user to input a password of at least 8 characters long, repeatedly asking until a valid length password is entered. It ensures the password confirmation matches the original password entered. If there's a mismatch, it prompts for password input again, ensuring both the original and confirmation passwords meet the minimum length requirement. This section of code is focused on preparing for encryption, generating a hash for integrity verification, and ensuring the user provides a sufficiently strong password for encryption, with robust validation and confirmation processes.

Triple DES Implementation and Saving the File:This code snippet initiates the encryption process using the Data Encryption Standard (DES) algorithm in Cipher Block Chaining (CBC) mode. It divides the encryption key into three parts and utilizes each part along with specific segments of the key for initialization vectors to encrypt the image data. Subsequently, it decrypts the second ciphertext using the second key portion. After encryption, it appends the SHA256 hash of the original image data to the encrypted bytes for integrity verification. Finally, it attempts to save the encrypted image file with the filename prefixed by "encrypted\_" in the same directory as the original image. If saving fails, it prompts for an alternate filename, and if that fails, it exits the program. This section of code encapsulates the encryption process, including key management, data encryption, hash appending, and file saving, with robust error handling for potential failures during encryption or file saving operations.



The Decryption Function: Initially, it attempts to open the encrypted image file specified by the input path in binary mode and reads its contents. Then, it prompts the user to input the decryption password for key authentication. Afterward, it extracts the SHA256 hash appended at the end of the encrypted data, separating it from the encrypted image data. Overall, this function serves as a preliminary step in the decryption process, focusing on reading encrypted data and preparing for key authentication and integrity verification.

Triple DES Implementation and Saving the File: The provided code segment initiates the decryption process for an encrypted image file using the Data Encryption Standard (DES) algorithm in Cipher Block Chaining (CBC) mode. It attempts to decrypt the encrypted image data using the decryption key obtained from the user input. The decryption is performed in three stages, each utilizing a portion of the decryption key and specific segments of the key for initialization vectors. After decryption, the SHA256 hash of the decrypted plaintext is computed. The code then compares this hash with the extracted hash from the encrypted data to verify the integrity of the decrypted image. If the hashes match, indicating successful decryption and integrity preservation, a success message is printed; otherwise, an incorrect password message is displayed, and the program exits. This section of code encapsulates the decryption process, including key usage, data decryption, and integrity verification, with error handling for potential decryption failures or incorrect passwords.



1. **Conclusion**

Triple DES offers a balance between security and compatibility, making it suitable for applications where both are essential. However, its use is declining in favor of more efficient and secure algorithms like AES. Still, it remains relevant in certain legacy systems and environments where upgrading is challenging.

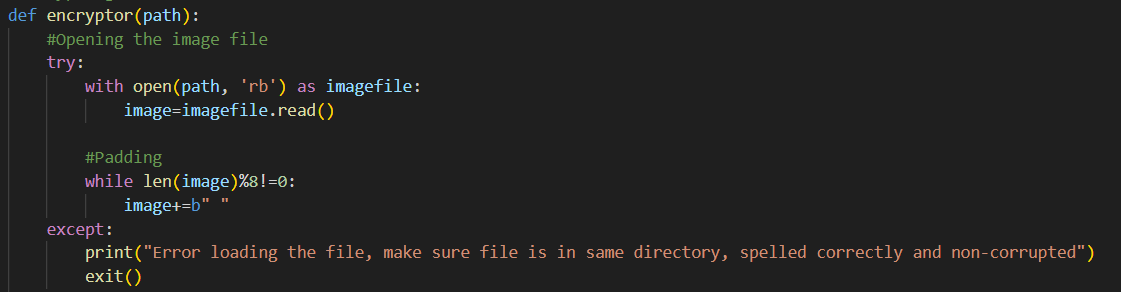
Triple DES algorithm to encrypt and decrypt the data. This supplies a better process of secure encryption and decryption. It is more secure and faster than double DES. As it has lengthy key chances of attacking the data is less. So, it supplies security in the storage and transmission of data.

1. **Learning Outcomes**

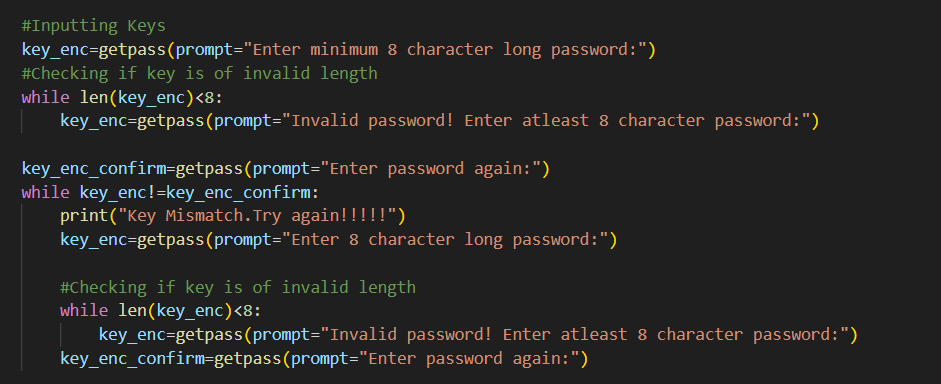
* Understanding of Encryption Concepts: Learners will gain a solid understanding of symmetric-key encryption concepts, including block ciphers, key generation, encryption rounds, and decryption processes.
* Knowledge of Triple DES Algorithm: Students will learn the specific details of the Triple DES algorithm, including its key length, encryption rounds, and permutation steps.
* Security Awareness: By studying Triple DES encryption, learners will become more aware of the importance of data security and the potential vulnerabilities associated with unencrypted data transmission and storage.
* Legacy System Integration: For students working in environments with legacy systems, learning about Triple DES provides insights into integrating encryption solutions with older hardware and software, ensuring data security without the need for extensive system upgrades.
* Compliance Knowledge: Learners will become familiar with data protection and privacy regulations that mandate the encryption of sensitive data, gaining insights into compliance requirements and best practices for securing image data.

1. **Source Code**

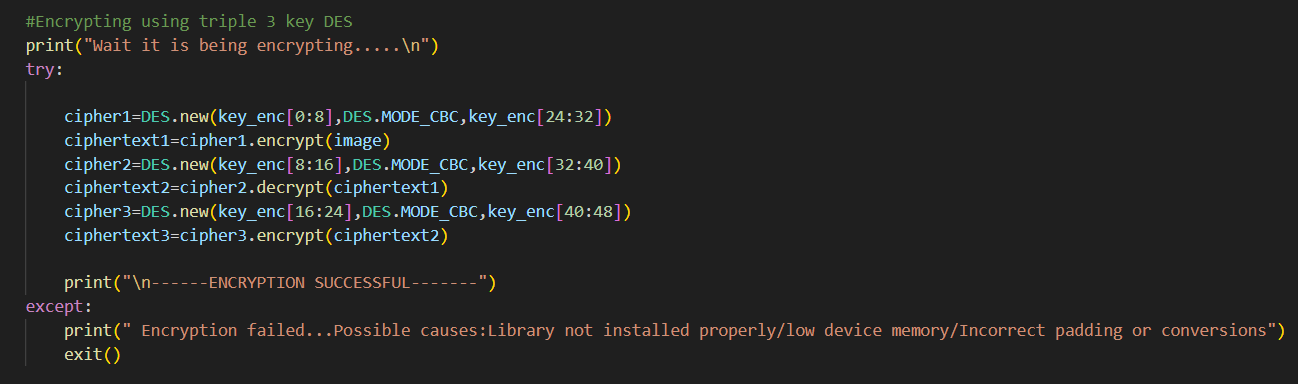
*i.*Encryption Function



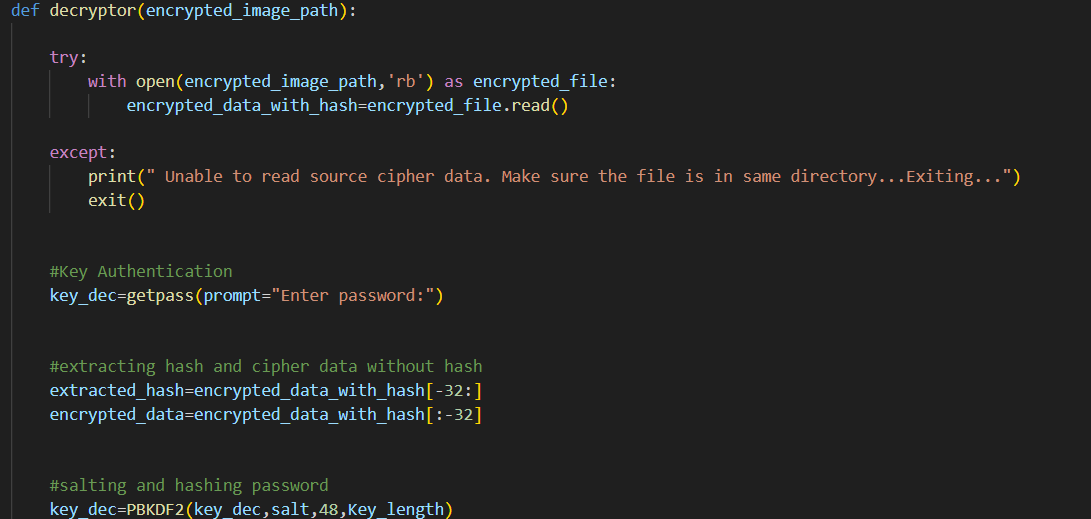
*ii.* Inputting Keys



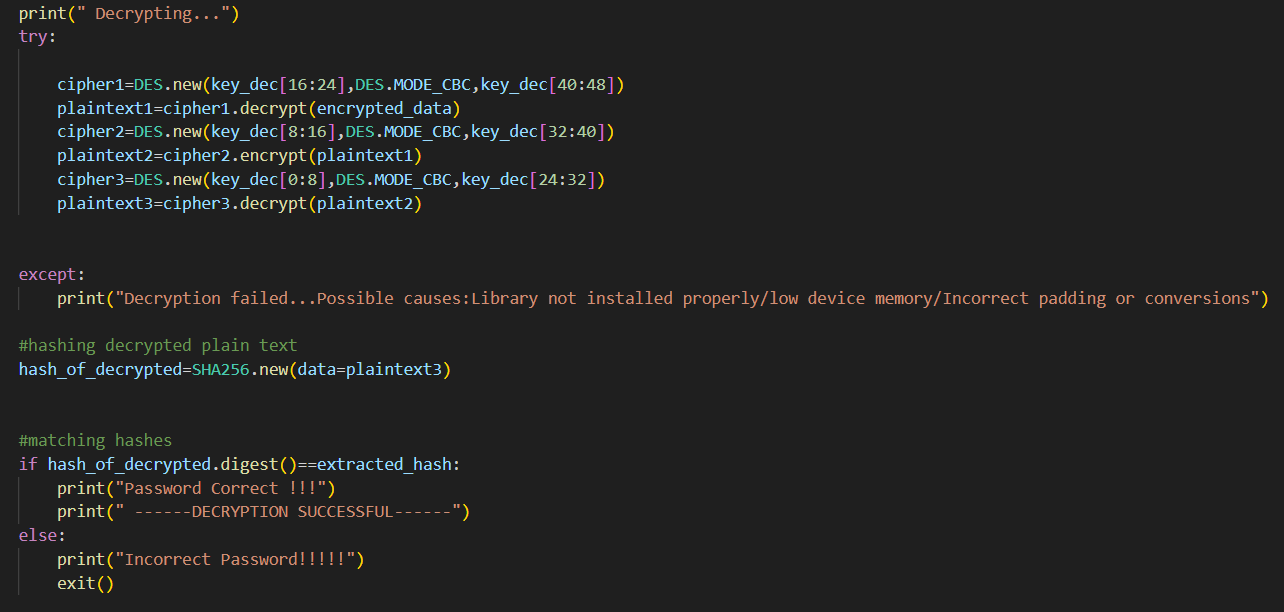
*iii.* Encryption Using 3-DES



*iv.* Decryption Function



*v.* Decrypting Using 3-DES



1. **References**
2. Aamer Nadeem, “A Performance Comparison of Data Encryption Algorithm,” IEEE 2005.
3. Feistel, Cryptography and Computer Privacy, Scientific American, Volume: 28, No.5, 1973.
4. Book: Cryptography and Network Security.