**LAB15 AI based symmetric key encryption & decryption**

Class: M04 Name: Truong Dang Truc Lam Students ID: B2111933

**Check system environment**

| Resource | Details |
| --- | --- |
| Client | OS :  Browser Type and version :  Language Type and version : |
| Web Server | OS :  Web Server : Type and version  Web Application : Type of Web Application  Language : Type and version |

**Choose one practice model among three ones**

[Exercise type A]

**Survey the process of traditional encryption & decryption full process**

Traditional encryption, also known as symmetric-key encryption, involves a single key to both encrypt and decrypt data. Here's a simplified breakdown of the process:

**1. Plaintext:** This is the original, readable message or data that needs to be protected.

**2. Encryption Key:** A secret key, shared between the sender and the receiver. It's used to scramble the plaintext.

**3. Encryption Algorithm:** A mathematical function that applies the encryption key to the plaintext, transforming it into ciphertext. Common algorithms include:

* + **Caesar Cipher:** Shifts each letter of the plaintext by a fixed number of positions in the alphabet.
  + **Vigenère Cipher:** A polyalphabetic substitution cipher that uses a keyword to encrypt the plaintext.
  + **DES (Data Encryption Standard):** A block cipher that operates on 64-bit blocks of data.
  + **AES (Advanced Encryption Standard):** A more secure block cipher that operates on 128-, 192-, or 256-bit blocks of data.

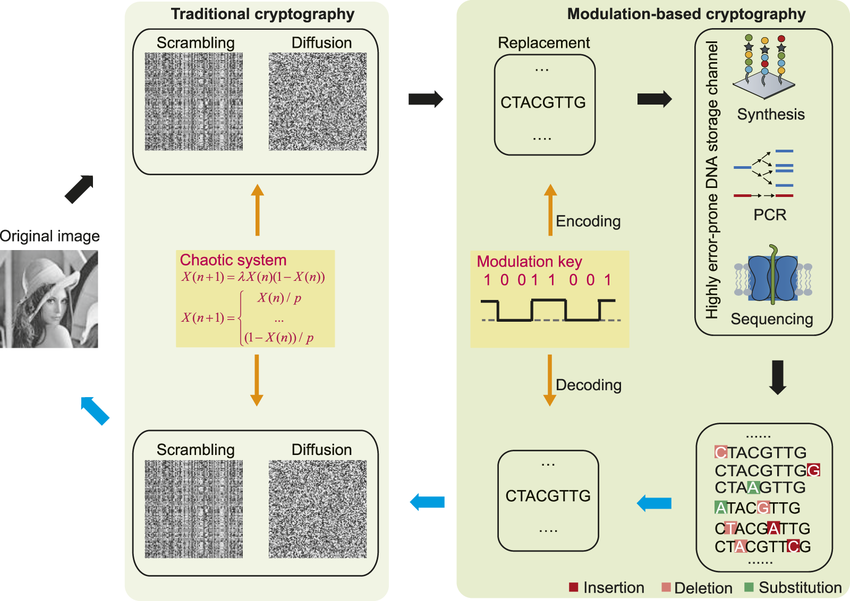
**4. Ciphertext:** The scrambled, unreadable version of the plaintext, resulting from the encryption process.

**5. Decryption Key:** The same secret key used for encryption.

**6. Decryption Algorithm:** The reverse process of the encryption algorithm, which applies the decryption key to the ciphertext to recover the original plaintext.

**The Process:**

1. **Encryption:** The sender applies the encryption algorithm to the plaintext using the secret key. This transforms the plaintext into ciphertext.
2. **Transmission:** The ciphertext is transmitted to the intended receiver.
3. **Decryption:** The receiver applies the decryption algorithm to the ciphertext using the same secret key. This transforms the ciphertext back into the original plaintext.

****

Traditional encryption and decryption process

[Exercise type B]

**Survey ai based symmetric key encryption & decryption coding process**

**Data Preparation:**

* Collect and preprocess the data to be encrypted or decrypted.
* Convert data into a suitable format for the chosen encryption algorithm.

**Key Generation:**

* Implement an AI-based key generation algorithm, such as a neural network or a random number generator.
* Ensure the generated key is secure and meets the desired cryptographic strength.

**Encryption:**

* Select a symmetric key encryption algorithm (e.g., AES, DES).
* Implement the encryption algorithm using the generated key and the prepared data.
* Output the encrypted ciphertext.

**Decryption:**

* Use the same key to decrypt the ciphertext.
* Implement the decryption algorithm to recover the original plaintext.

**Security Analysis:**

* Integrate AI-based security analysis tools to monitor the system for threats and vulnerabilities.
* Implement anomaly detection and adversarial attack detection mechanisms.

[Exercise type C]

**Code ai based symmetric key encryption & decryption**

Plain text : “I am studying ai based symmetric key encryption & decryption coding process”

from cryptography.fernet import Fernet

def generate\_key():

"""Generates a random key for encryption and decryption."""

key = Fernet.generate\_key()

return key

def encrypt\_message(message, key):

"""Encrypts a message using the provided key."""

f = Fernet(key)

encrypted\_message = f.encrypt(message.encode())

return encrypted\_message

def decrypt\_message(encrypted\_message, key):

"""Decrypts an encrypted message using the provided key."""

f = Fernet(key)

decrypted\_message = f.decrypt(encrypted\_message).decode()

return decrypted\_message

if \_\_name\_\_ == "\_\_main\_\_":

plain\_text = "I am studying ai based symmetric key encryption & decryption coding process"

# Generate a key

key = generate\_key()

# Encrypt the message

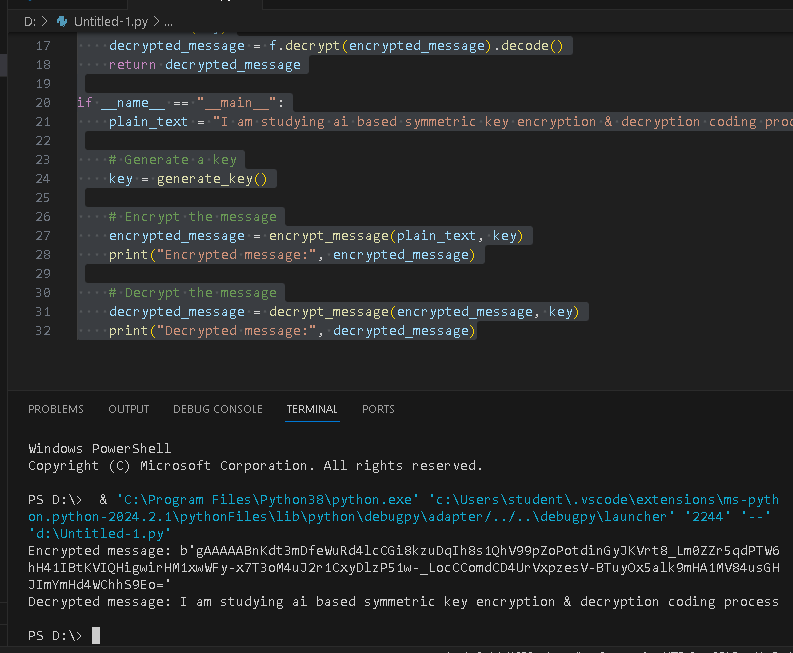
encrypted\_message = encrypt\_message(plain\_text, key)

print("Encrypted message:", encrypted\_message)

# Decrypt the message

decrypted\_message = decrypt\_message(encrypted\_message, key)

print("Decrypted message:", decrypted\_message)



Code result