AES Encryption GUI Tool – CBC Mode

Project Overview

This project is a **Graphical User Interface (GUI)** application developed using **Python** and **Tkinter** to demonstrate **AES (Advanced Encryption Standard)** encryption and decryption with the help of the **PyCryptodome** library.

Features

- Real-time **AES encryption** and **decryption**
- User-friendly **Tkinter GUI** with modern layout
- Accepts custom AES keys of **16, 24, or 32 bytes**
- Supports **AES-128**, **AES-192**, and **AES-256**
- Uses **CBC (Cipher Block Chaining)** mode with random IV generation
- Ciphertext is displayed in **Base64** format
- Fully handles **PKCS7 padding** for proper block alignment
- Separate input/output sections for better understanding
- Simple to use for educational and learning purposes

What is AES?

AES (Advanced Encryption Standard) is a secure, symmetric encryption algorithm standardized by NIST. It encrypts data in fixed blocks of **128 bits**, using secret keys of 128, 192, or 256 bits.

AES is widely used in:

- **Wi-Fi security** (WPA2, WPA3)
- **Web encryption** (HTTPS via SSL/TLS)
- **VPN protocols** and **file encryption systems**
- **essaging apps** like WhatsApp, Signal
- **Secure disk and cloud storage**

Installation

To run the application, you need the PyCryptodome library. You can install it using:

```
bash

pip install pycryptodome
```

How to Run the Application

1. Ensure **Python 3.x** is installed on your system.
2. Open your terminal or IDE and run:

   ```
   bash

   python aes_gui.py
   ```
3. The GUI window will appear.
4. Enter your:
   - **Plaintext message** to be encrypted
   - **AES key** (must be 16/24/32 characters long)
5. Click the **"Encrypt"** button to generate the Base64 ciphertext.
6. Use the **"Decrypt"** button with the same key to view the decrypted message.