Steganography

05 February 2025 19:00

Secure Image Steganography with Python

Overview

This Image-Based Steganography Tool allows users to securely conceal messages within images using AES encryption and LSB (Least Significant Bit) steganography. It features an intuitive Streamlit-based interface for seamless encoding and decoding.

Features

- **AES Encryption**: Encrypts messages using AES-CBC mode before embedding them in images.
- **LSB Steganography**: Hides encrypted messages at the pixel level.
- **Image Upload & Download**: Supports image upload, encoding/decoding, and downloading of encrypted images.
- **User-Friendly Interface**: Built with Streamlit for an interactive experience.

Installation

1. Clone the repository:
   git clone https://github.com/chamu1999-Devi/Stagnography.git
   Install dependencies:
   pip install streamlit opencv-python numpy pycryptodome
2. Run the application:
   streamlit run stego.py

Usage

Encoding a Message

1. Upload a PNG image.
2. Enter a secret message.
3. Provide a passcode for encryption.
4. Click **Encode & Save Image**.
5. Download the encrypted image.

Decoding a Message

1. Upload the encrypted image.
2. Enter the correct passcode.
3. Click **Decode Message**.
4. View the decrypted message.

Screenshots

Encode Message

Decode Message

Security Considerations

- AES encryption ensures message security.
- Messages are only retrievable with the correct passcode.
- Store images securely to prevent unauthorized access.

Technologies Used

- **Python**
- **Streamlit**
- **OpenCV**

- **PyCryptodome** 🔑

🤝 Contributing

Contributions are welcome! Feel free to fork this repository and submit a pull request.

🌐 Connect with Me

- 📧 Email
- 💼 LinkedIn