

---

## **\*\*User's Guide - VersaDecryptor Version 3\*\***

The User's Guide provides detailed instructions on using VersaDecryptor to decrypt data and retrieve Bitcoin address balances.

### **\*\*1. Decrypting Data\*\***

To decrypt data encrypted with various encryption algorithms, follow these steps:

1. Run the VersaDecryptor script using the command: ``python VersaDecryptor.py``.
2. The program will prompt you for the necessary information based on the encryption algorithm and scenario.
3. Provide the required information such as the encryption key, initialization vector (IV), and ciphertext.
4. If you have a passphrase, Bitcoin address, and ciphertext, you can enter them to derive the private key and IV for decryption.
5. The program will decrypt the data and display the decrypted result.

### **\*\*2. Retrieving Bitcoin Address Balances\*\***

VersaDecryptor allows you to retrieve and display the balances of Bitcoin addresses. Here's how to do it:

1. Run the VersaDecryptor script using the command: ``python VersaDecryptor.py``.
2. Select the option to retrieve Bitcoin address balances.
3. Enter the Bitcoin address for which you want to retrieve the balance.
4. The program will fetch the balance using a blockchain explorer API and display the result.

### **\*\*3. Decrypting Files\*\***

VersaDecryptor supports decrypting entire files. To decrypt a file, follow these steps:

1. Run the VersaDecryptor script using the command: ``python VersaDecryptor.py``.
2. Select the option to decrypt files.
3. Enter the path to the file you want to decrypt.
4. The program will decrypt the file and provide the decrypted result.

#### **\*\*4. Multi-threaded Decryption\*\***

VersaDecryptor offers multi-threading support for parallel processing of multiple decryption tasks. To decrypt multiple ciphertexts simultaneously, follow these steps:

1. Run the VersaDecryptor script using the command: ``python VersaDecryptor.py``.
2. Select the option for multi-threaded decryption.
3. Provide the necessary information for each ciphertext you want to decrypt.
4. The program will decrypt the ciphertexts concurrently, improving overall performance.

#### **\*\*5. Key Management\*\***

VersaDecryptor includes key management features for generating, importing, and exporting keys. Here's how to use these features:

1. Run the VersaDecryptor script using the command: ``python VersaDecryptor.py``.
2. Select the key management option (generate, import, or export).
3. Follow the prompts to generate a new key, import an existing key, or export a key to a file.

#### **\*\*6. Advanced Encryption Modes\*\***

VersaDecryptor supports advanced encryption modes like GCM (Galois/Counter Mode) and CTR (Counter) mode. To use these modes, follow these steps:

1. Run the VersaDecryptor script using the command: ``python VersaDecryptor.py``.
2. Select the option for advanced encryption modes.
3. Provide the necessary information for decryption using the selected mode.
4. The program will perform the decryption using the advanced encryption mode.

#### **\*\*7. Decrypting Other Cryptocurrencies\*\***

In addition to Bitcoin, VersaDecryptor can handle other cryptocurrencies like Ethereum, Litecoin, and Ripple. To decrypt data related to these cryptocurrencies, follow these steps:

1. Run the VersaDecryptor script using the command: ``python VersaDecryptor.py``.
2. Select the option for decrypting other cryptocurrencies.
3. Choose the specific cryptocurrency you want to decrypt (Ethereum, Litecoin, or Ripple).

4. Provide the necessary information for decryption, such as private keys or ciphertexts.
5. The program will decrypt the data and display the decrypted result

.

Note: Make sure to provide accurate and valid information for decryption, as incorrect or invalid inputs may lead to incorrect results or errors.

### **\*\*Placeholder Code Explanations:\*\***

Throughout the VersaDecryptor script, you may encounter instances of placeholder code. These sections are marked with comments and are intended to be replaced with your custom implementation based on your specific requirements. Make sure to review these sections and replace the placeholder code with your desired functionality.

For any further assistance or questions, please refer to the project's documentation or consult the README file on the project's GitHub repository.

---