Task 6: Introduction to Cryptography

Tools Used

Primary Tool: OpenSSL

Objective

The objective of this task is to understand cryptography fundamentals, perform encryption and hashing operations using OpenSSL, generate asymmetric keys, understand digital signatures, and learn real-world cryptography usage.

What is Cryptography

Cryptography is the practice of securing information by converting plaintext into an unreadable format using mathematical algorithms to ensure confidentiality, integrity, authentication, and non-repudiation.

Types of Encryption

Symmetric Encryption

Symmetric encryption uses a single secret key for both encryption and decryption. It is fast and commonly used for encrypting large amounts of data.

Asymmetric Encryption

Asymmetric encryption uses a pair of keys: a public key for encryption and a private key for decryption. It is commonly used for secure key exchange and authentication.

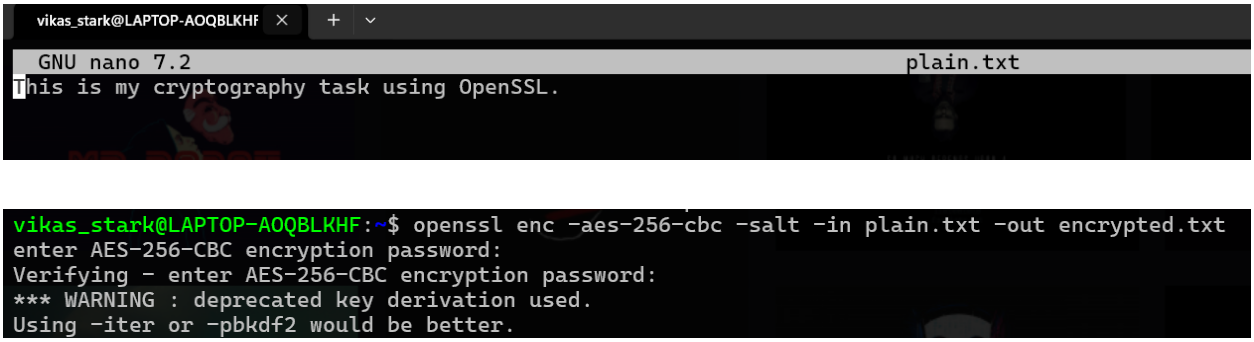
Symmetric Encryption Using AES (OpenSSL)

A file was encrypted using the AES algorithm with OpenSSL.

Command Used

openssl enc -aes-256-cbc -salt -in plain.txt -out encrypted.txt

Screenshot 1: AES File Encryption Using OpenSSL



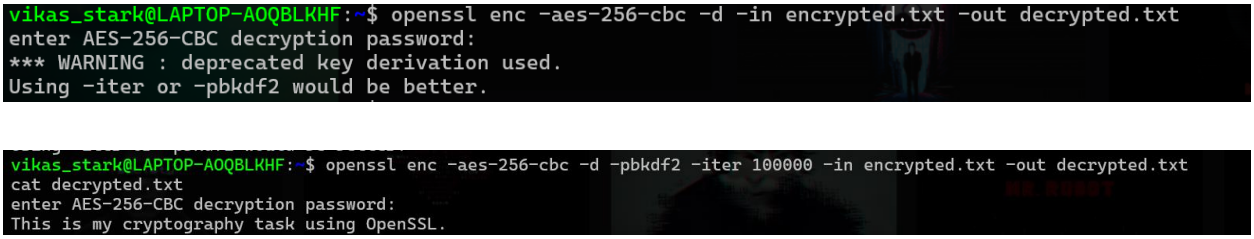
AES Decryption

Command Used

openssl enc -aes-256-cbc -d -in encrypted.txt -out decrypted.txt

---

Screenshot 2: AES File Decryption Output



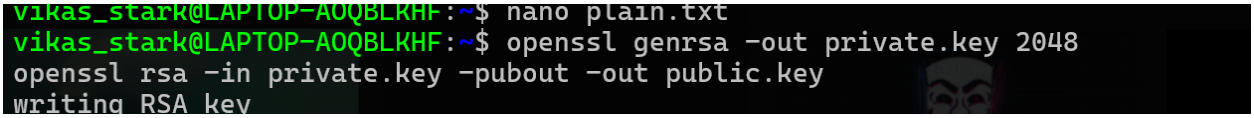
Asymmetric Encryption – RSA Key Generation

RSA public and private keys were generated using OpenSSL.

Command Used

openssl genrsa -out private.key 2048

openssl rsa -in private.key -pubout -out public.key



Screenshot 3: RSA Key Generation

```
vikas_stark@LAPTOP-AOQBLKHF:~$ openssl genrsa -out private.key 2048
openssl rsa -in private.key -pubout -out public.key
writing RSA key
```

Digital Signatures

A digital signature was created using a private key to verify data authenticity and integrity.

Command Used

openssl dgst -sha256 -sign private.key -out signature.bin plain.txt

Signature Verification

Command Used

openssl dgst -sha256 -verify public.key -signature signature.bin plain.txt

```
vikas_stark@LAPTOP-AOQBLKHF:~$ openssl dgst -sha256 -sign private.key -out signature.bin plain.txt
vikas_stark@LAPTOP-AOQBLKHF:~$
vikas_stark@LAPTOP-AOQBLKHF:~$ openssl dgst -sha256 -verify public.key -signature signature.bin plain.txt
Verified OK
```

Screenshot 4: Digital Signature Creation and Verification

Hashing and Integrity Verification

A cryptographic hash was generated to verify file integrity.

Command Used

openssl dgst -sha256 plain.txt

Screenshot 5: SHA-256 Hash Generation

```
vikas_stark@LAPTOP-AOQBLKHF:~$ openssl dgst -sha256 plain.txt
SHA2-256(plain.txt)= 8566f4c62d0238bc6c3c990ae486d738e31a2620eb7582d9458793a083bf2b8b
vikas_stark@LAPTOP-AOQBLKHF:~$
```

Comparison of Encryption Algorithms

AES is faster and efficient for bulk data encryption, while RSA is slower and mainly used for key exchange and authentication. Hashing algorithms are used for integrity verification rather than encryption.

Real-World Usage of Cryptography

Cryptography is used in HTTPS for secure web communication, VPNs for secure network connections, digital certificates for authentication, and password protection systems.

Summary

This task demonstrated practical cryptography operations using OpenSSL, including encryption, key generation, digital signatures, and hashing. It provided a clear understanding of cryptographic concepts and their real-world applications.

Final Outcome

Strong foundational knowledge of cryptography, encryption methods, digital signatures, and integrity verification.