Jack Lin, Jared Ricks

Midterm Project

Objectives:

Confidentiality: Keep information private during communication.

- How: Use AES encryption in CBC mode. Establish a shared secret through Diffie-Hellman key exchange to encrypt and decrypt messages, limiting access to authorized parties.

Integrity: Prevent unauthorized changes to data during transmission.

- How: Apply SHA-256 hashing to the shared secret. This hash verifies the shared secret's integrity before encryption and decryption, detecting any tampering.

Key Management: Securely generate, exchange, store, and dispose of cryptographic keys.

- How: Utilize Diffie-Hellman for secure key exchange and SHA-256 hashing for consistent-size key representation, aiding secure storage and transmission.

Discussing Alternatives:

- *Confidentiality*: While AES is used, alternatives like Triple DES or different AES modes could be considered based on specific needs, performance, or compatibility.
- *Integrity*: SHA-256 is chosen, but other hash functions like SHA-3 or SHA-512 could be options depending on security requirements.
- *Key Management*: Diffie-Hellman is chosen, but alternatives include RSA, ECDH, or higher-level protocols like TLS/SSL based on system needs and performance considerations.

Decisions for Objectives:

AES Encryption:

- Choice: AES in CBC mode with a 128-bit key.
- Reasoning: AES is preferred over DES for better security, and a 128-bit key provides a good balance between security and performance.

Diffie-Hellman Key Exchange:

- Choice: Diffie-Hellman.
- Reasoning: Diffie-Hellman is preferred for key exchange due to its efficiency and ability to provide Perfect Forward Secrecy (PFS).

System Design Explanation:

- Key Exchange: Diffie-Hellman ensures secure key exchange, safeguarding the shared secret from eavesdroppers.
- Encryption (AES in CBC Mode): AES with a 128-bit key ensures confidentiality, and CBC mode secures block-based data.
- Hashing: SHA-256 provides a strong, fixed-size representation of the shared secret, ensuring its integrity.

- Initialization Vector (IV): A random IV enhances encryption security, shared between parties for decryption.
- Error Handling: The system incorporates error handling for robustness, printing error messages and exiting in case of failures.
- Output Display: The system provides informative output for better understanding and verification of cryptographic processes.

System Analysis:

Strengths:

- Diffie-Hellman provides PFS.
- AES with a 128-bit key is a secure choice.
- SHA-256 ensures strong key representation.
- CBC mode with a random IV enhances encryption security.

Weaknesses:

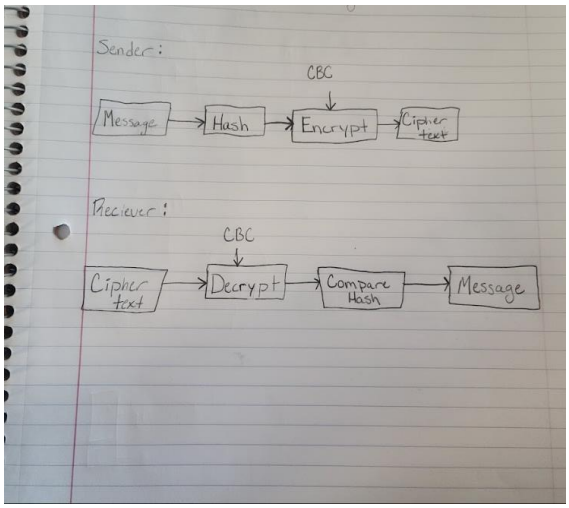
- Lack of advanced security features like party authentication.
- No protection against replay attacks or message tampering.
- Fixed key and block sizes may limit adaptability.

Considerations for Improvement:

- Integration with TLS/SSL for advanced security.
- Implementation of additional measures for authentication and integrity.
- Extension to handle variable key sizes for flexibility.

System Evaluation:

- Security: Achieves a reasonable level of security but lacks some advanced features.
- Usability: Clear output and relatively easy to understand but may need enhancements for complex scenarios.
- Efficiency: Strikes a balance between security and efficiency in key processes.
- Extensibility: Can be extended for additional security features and adaptability based on different requirements.


Sender:
CBC
Message
Hash
Encrypt
Cipher text
Reciever:
CBC
Cipher text
Decrypt
Compare Hash
Message