Security Overview of Secure Chat

Assumptions

- Secure Environment: It is assumed that the operating environment is secure, and the underlying hardware and OS are not compromised.
- Reliable Cryptographic Libraries: The GMP and OpenSSL libraries used in the application are assumed to be up-to-date and free from vulnerabilities at the time of deployment.
- Informed Users: Users are assumed to be aware of basic security practices, such as keeping their private keys secure and verifying the identity of the server and client during connections.

Claims

- Confidentiality: All messages exchanged between the client and server are encrypted using AES-256-CBC, ensuring that eavesdroppers cannot read the content of the messages. The encryption keys are securely generated using the Diffie-Hellman (DH) key exchange protocol.

- Authentication: The DH key exchange mechanism ensures that both parties can authenticate each other. The public keys are exchanged and verified, ensuring the identities of the parties involved in the communication.
- Integrity: HMAC-SHA256 is used to verify the integrity of messages. This ensures that the transmitted data is protected against alterations during transit.

Limitations

- Endpoint Security: The application does not directly address endpoint security. If the client or server systems are compromised, the security of the entire communication channel could be jeopardized.
- Key Management: Mismanagement of DH keys, such as failure to securely initialize, store, or shred keys, can lead to vulnerabilities.
- Physical Security: The application assumes the physical security of the hosting servers and client machines. If compromised, unauthorized access to the communication can occur.

- Forward Secrecy: The code does not explicitly ensure forward secrecy, which would protect past sessions against future private key compromises.
- Denial of Service (DoS): The application does not include mechanisms to protect against DoS attacks, which could make the service unavailable to legitimate users.

Implementation Details

Key Exchange and Generation

The application uses the Diffie-Hellman (DH) key exchange protocol to securely generate a shared secret key between the client and server. The DH parameters (prime numbers and generator) are initialized from a file (params), and the keys are securely exchanged and verified.

Encryption and Decryption

Messages are encrypted using AES-256-CBC to ensure confidentiality. The encryption key is derived from the DH key exchange process. Each message is

encrypted before being sent over the network and decrypted upon receipt.

Message Integrity

HMAC-SHA256 is used to verify the integrity of messages. A hash-based message authentication code (HMAC) is calculated for each message using a secret key derived from the shared DH key. This ensures that any tampering of the message during transit is detected.