

Raspberry Pi 5 Secure Chat Application

Group H

Nhat Hoang Ha

Linh Phung Lam

Sissy Corona

Ricardo Godinez

Professor

Mohamed El-Hadedy Aly



Outline

- Abstract
- Motivation
- Architecture
- Errors/Bugs
- Live Demo
- Future Work

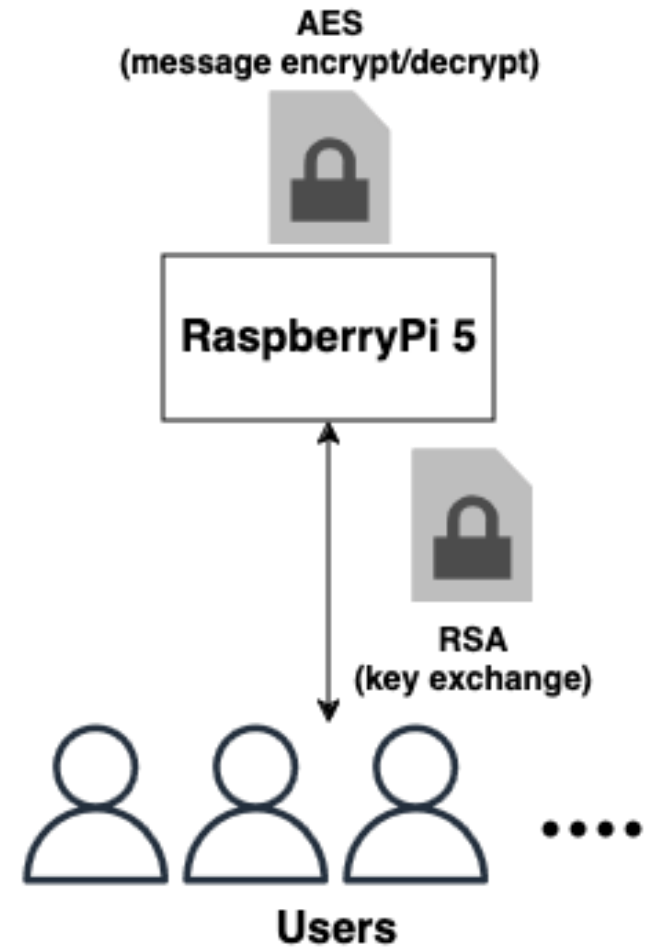


Motivation

- **Enhancing Skills in Cryptography:** Develop expertise in AES-256 encryption and ECC-based Diffie-Hellman key exchange for secure communication.
- **Real-World Problem Solving:** Address security challenges in industries like healthcare, finance, and defense with robust encryption.
- **Mastering Networking on Raspberry Pi:** Showcase scalability and resource optimization by hosting multiple secure clients on a Raspberry Pi.
- **Innovation and Technical Leadership:** Integrate cutting-edge cryptographic algorithms on a compact platform to highlight innovative thinking.
- **Career and Research Growth:** Build a foundation for advanced research or product development in cybersecurity and IoT systems.



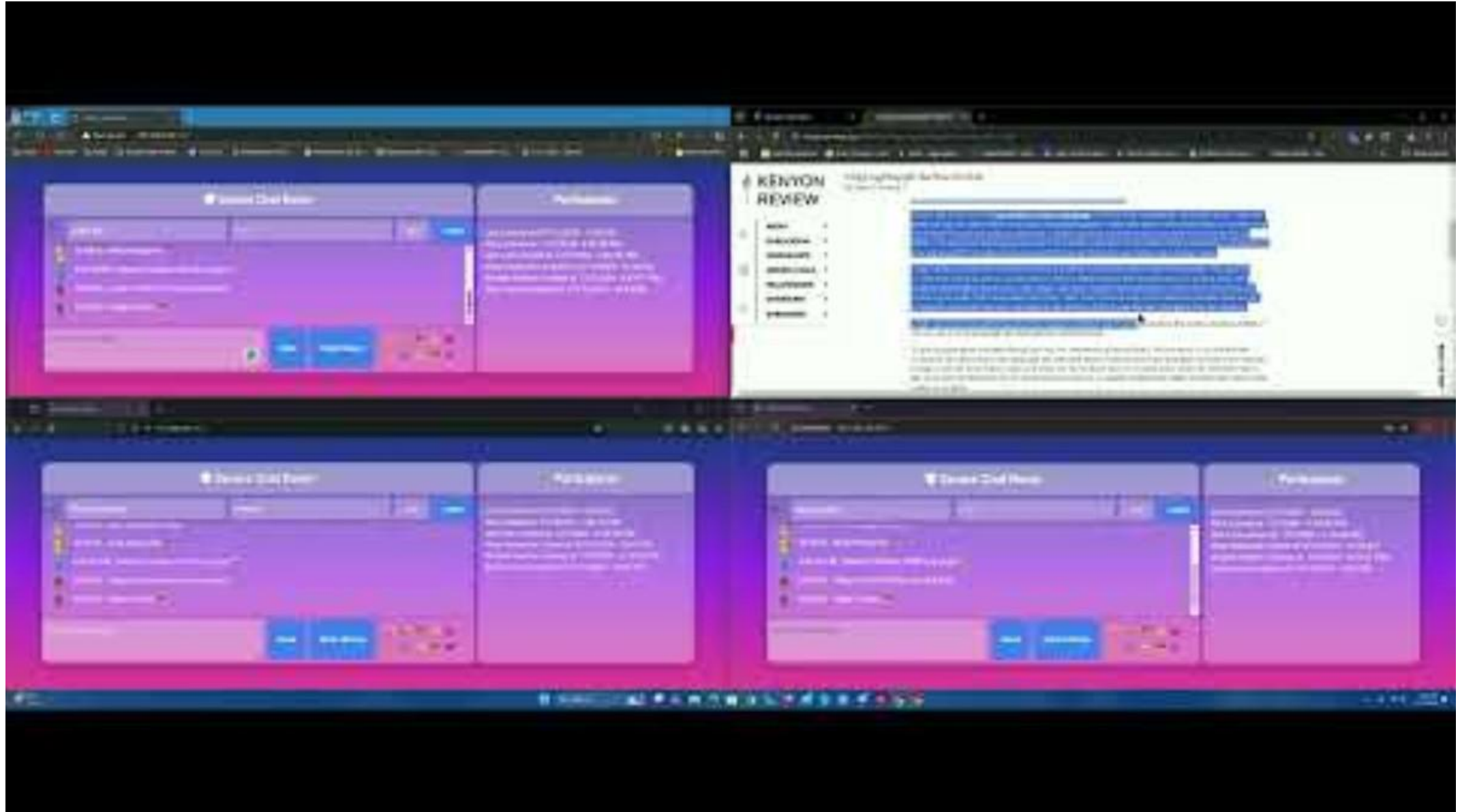
Architecture








Errors

Errors	Causes
Incorrect AES Key Management	Using a static or hardcoded AES key, which poses a security risk.
Misuse of AES Modes	AES requires an Initialization Vector (IV) for modes like CBC, but the IV was missing in some cases, leading to decryption errors.
Empty Ciphertext	Decryption failed when the ciphertext was empty or improperly encoded. Occurred during system messages (e.g., login/logout notifications) that were mistakenly encrypted.
Base64 Encoding/Decoding Issues	Improper handling of encoded IV and ciphertext resulted in decryption failures.
Ciphertext Length Mismatch	Ciphertext length was not a multiple of the AES block size, causing decryption to fail.
Global Key Use	Reused the same AES key for all users/sessions, which reduced security.
Key Size Compatibility	Using RSA keys that were too small (e.g., 512 bits), making them insecure.
Incorrect Key Import/Export	Encountered format-related errors during public/private key export/import.
Performance Overhead	RSA is computationally expensive for large data.
Incorrect Key Pair Association	Mismatched private and public keys between the server and client, leading to decryption errors.
Padding Scheme Errors	RSA decryption failed due to mismatched padding schemes.
Public Key Distribution Issues	Failures in transmitting the public key from the server to the client.
WebSocket Configuration Issues	The server fell back to HTTP polling due to missing WebSocket support (simple-websocket or eventlet).

Demo



Future Work

-  **File Sharing:**
 - Enable secure file transmission with AES encryption.
-  **Image Transmission:**
 - Support sending and viewing encrypted images in chat.
-  **Audio Recording:**
 - Record and send 1-minute encrypted audio clips.
-  **Enhanced Privacy:**
 - Apply robust security for media transmissions.
-  **Performance Optimization:**
 - Improve encryption and decryption efficiency for larger files.