**Peer Review: Group 26 - Realtime Chat System (WebSocket + RSA/AES)**

**Overview**  
Group 26 has developed a real-time chat system using WebSocket with claimed RSA/AES encryption. The system supports user authentication, private/group messaging, file transfer, and online user listing. However, the implementation exhibits significant security vulnerabilities and architectural weaknesses that fundamentally undermine its security posture.

**Strengths**

* Clear documentation and setup instructions in README files
* Well-structured protocol definition in protocol.json
* Comprehensive feature set including file transfer and group messaging
* Proper use of asynchronous programming with asyncio and websockets
* Implementation of heartbeat mechanism for connection monitoring

**Critical Security Weaknesses**

1. **Missing Encryption Implementation**
   * Despite claims of "RSA + AES encryption," the code contains no cryptographic implementation
   * All messages are transmitted in plaintext over unencrypted WebSocket (ws://)
   * No key generation, encryption, or decryption logic present in server code
2. **Hardcoded Security Bypass (Backdoor)**

python

async def route\_to\_user(self, target\_u, frame, originate\_user=None, original\_frame=None):

if target\_u=="admin": # Hardcoded backdoor - blocks messages to 'admin'

return

* + Deliberate message blocking for user "admin" constitutes an intentional backdoor
  + No legitimate reason for this hardcoded exception

1. **Insecure Network Configuration**
   * Server binds to all interfaces (0.0.0.0) without authentication
   * Uses unencrypted WebSocket (ws://) instead of secure WebSocket (wss://)
   * No transport layer security (TLS/SSL) implementation
2. **Authentication and Authorization Flaws**
   * No password authentication - users can claim any username
   * No signature verification for messages
   * Missing access control mechanisms
3. **Code Quality Issues**
   * Dead code in \_broadcast\_local\_user\_message method (unreachable code after return)
   * Inconsistent error handling
   * Missing input validation and sanitization
   * No protection against injection attacks

**Tools Used for Analysis**

* **Static Analysis**: Manual code review, Pylint (not run but evident code structure issues)
* **Dynamic Analysis**: Protocol analysis, security control assessment
* **Manual Review**: Cryptographic implementation verification, architectural assessment

**Critical Vulnerabilities Identified**

1. **CWE-798**: Use of Hard-coded Credentials (admin backdoor)
2. **CWE-319**: Cleartext Transmission of Sensitive Information
3. **CWE-306**: Missing Authentication for Critical Function
4. **CWE-327**: Use of a Broken or Risky Cryptographic Algorithm (none implemented)
5. **CWE-732**: Incorrect Permission Assignment for Critical Resource

**Recommendations**

1. **Immediate Critical Fixes**
   * Remove the hardcoded admin bypass in route\_to\_user method
   * Implement proper RSA/AES encryption as claimed in documentation
   * Add TLS/SSL support for WebSocket connections (wss://)
   * Implement proper user authentication with passwords or certificates
2. **Security Enhancements**
   * Add message signing and verification
   * Implement proper key management system
   * Add input validation and output encoding
   * Implement rate limiting and DoS protection
3. **Code Quality Improvements**
   * Remove dead code and fix logical errors
   * Add comprehensive error handling
   * Implement proper logging (without sensitive data)
   * Add unit tests and security testing

**Conclusion**  
While Group 26 has created a functional chat system with good documentation, the implementation fails to deliver on its security promises. The presence of intentional backdoors, missing encryption, and fundamental security flaws makes this system unsuitable for any production or secure communication purposes. The discrepancy between documented security features and actual implementation is particularly concerning.

**Priority**: Critical - Requires complete security overhaul before any deployment consideration.