Secure Chat Application - Setup and Usage Instructions

This document provides detailed setup and usage instructions for the secure TLS-based chat application written in C.

=== PREREQUISITES ===

- A Linux-based environment with GCC and OpenSSL installed.
- Both server and client programs compiled with OpenSSL libraries.
- Users must share a pre-established HMAC secret ("s3cr3tkey" by default).

=== SETUP INSTRUCTIONS ===

1. Generate TLS Certificates (Server Side Only)

Run the following command to generate a self-signed certificate and private key:

```
openssl req -x509 -newkey rsa:2048 -keyout key.pem -out cert.pem -days 365 -nodes
```

Place both `cert.pem` and `key.pem` in the same directory as the server executable.

2. Compile the Programs

Use GCC to compile the programs with OpenSSL libraries:

```
gcc chat_server_secure.c -o chat_server_secure -lssl -lcrypto
gcc chat_client_secure.c -o chat_client_secure -lssl -lcrypto
```

3. Run the Server

Execute the server and provide a port number when prompted:

```
./chat_server_secure
```

4. Run the Client(s)

Each user must run the client and enter the following when prompted:

- Server IP address (e.g., 127.0.0.1 for localhost)
- Port number (must match server's port)
- A desired username

Example:

```
./chat_client_secure
```

=== SECURITY FEATURES ===

- TLS Encryption (via OpenSSL): Ensures data is encrypted in transit.
- Username Support: Identifies users during messaging.
- HMAC Validation: Authenticates message origin and integrity using SHA-256.
- Input Sanitization: Prevents terminal or injection exploits.
- Rate Limiting: Prevents flooding attacks by limiting messages per interval.

=== OPTIONAL IMPROVEMENTS ===

- Add TLS certificate verification on the client side for stronger identity checking.

- Limit file permissions on key.pem and cert.pem (e.g., chmod 600).

- Configure OpenSSL to use cipher suites with Perfect Forward Secrecy (PFS).

- Use certificate authorities (CA) to validate public keys rather than self-signed certs.

=== TROUBLESHOOTING ===

- Make sure ports are not blocked by firewalls.

- Ensure the OpenSSL library is installed: `sudo apt install libssl-dev`

- If client disconnects immediately, verify the server certificate and key.

For any further help, revisit the code comments or consult the OpenSSL documentation.