Linux-Chat

Thomas Tallentire

Lewis Scott

State Transition Diagrams

# Server STD



# Client STD

## 

State Descriptions

# **Client**

### Setup UI

* Initialize all UI elements and link them to the code

### Idle

* Wait for UI event to occur

### Process UI Event

* Determine which UI event occurred and take the proper action
* If the send button was clicked, call send
* If the encryption settings were changed, update encryption

### Send

* Send message typed in message box over open TCP connection

### Update Encryption

* Set the encryption keys to the new keys

### Display Message

* Add the given message to the list box

### Receive

* Loop on reading from the socket descriptor
* When information arrives, call process message

### Process Message

* Switch on the first character of the message to determine what to do with it.
* CHAT – Display the message
* ADD – Add User
* DISCONNECT – Remove User
* CHANGE – Rename User

### Add User

* Enter a new user into the array and update the users table

### Remove User

* Using the position given, set the corresponding name to null

### Rename User

* Change the users name to the one given.

# **Server**

### Daemonize

* Create a new session for the process

### Listen

* Listen for incoming connections
* Listen for messages from users

### Connection

* Add new socket to the list of socket descriptors

### Notify Clients

* Loop through all other clients and send CONNECTION messages

### Read

* Interpret the message and take the corresponding action

### Command

* Take the action and notify users if necessary

### Chat

* Send chat message to all clients except for the one it came from

### Disconnect

* Null out user on server
* Notify clients

Pseudocode

# Server:

# Client:

Init()

{

Initialize UI elements

Connect receive thread signals to UI slots

Start receive thread

}

setSocket()

{

Create and connect socket

}

Receive()

{

Loop forever on read

Switch on message\_type

Add new user connection

Display chat

Apply name change

Remove user

Shut down socket

}

Send()

{

Send given data over the socket

}

Display()

{

Update UI listview with new text

}

Encrypt()

{

Encrypt or decrypt given message using provided keys

}