Portland State University Dec 3,2021

Internetworking Protocol

Internet Relay Chat Project

By: Manish Manchegowda,

Yaksh Santosh kumar patel,

Narasimha Nikhil Sai Polavarapu.

TABLE OF CONTENT

1. Introduction

This RFC represents a protocol in which multiple clients communicate with each other. There is a server which sends, receives messages from different users and there are users who can create rooms, join rooms, leave rooms and can also send personal messages to the other users. If a group of users are a part of a particular room, any message sent by one user is received by all the users in the room.

1.1 Server

Although the IRC defined in the RFC 1459 has multiple server architecture, to provide a common point for all the clients to connect, a single server is used as the pillar for this IRC. This is done to achieve simplicity.

1.2 Client

As mentioned above, this is a single server architecture. Therefore, all the users connect to a single server that relays messages. Client can have a name length up to 20 alpha numeric characters along with underscore which is used for the identification of the client.

1.3 Room Hall:

A user can either create a room or join a particular room. It is created by the user who joins the channel first. A room is typically a group where multiple users can communicate with each other. A user can join a room just by referring with its room name. If all the users in the room leaves, the room does not exist anymore.

The basic functionality of a room is that, when a user tries to join a room using the "$join room_name" command. If the room with the name mentioned is not available, then a new room is created with that name. Now this room is visible or available for all the other users to join. Other users can see the list of rooms available using the $list command.

2. Specification

2.1 Mode of Communication

This application supports One to One, One to Many, and One to All modes of communications through rooms and personal chats. When a user sends a message in the room, the server receives it and forwards to all the users who are part of that room excluding the sender. This resembles dynamic multicasting.

2.2 Character Codes

Each message may consist of any number of characters from the ASCII character set. Here the space acts as a delimiter.

2.3 Messages

The important parts of any message are the command's name, transaction ID and the payload. All these parts are separated by the delimiter which is the space. Here the transaction ID which is an unsigned integer increments by 1 whenever a message is sent by the client. However, it remains 0 for the server to client messages. This ID might be useful for tracing the client's steps.

2.4 Replies

Almost all the messages sent to the server generates a reply. Numeric reply being the most common. It consists of the transaction ID and status of the request. It represents a 0 for a successful request and a non-zero for any errors. This request cannot be generated by a client. Any such messages received by a server will generate an error response.

3. Message Infrastructure

3.1 Create room

Initially there are no rooms available. The first user to start this application can create a room using the "$join room_name" command. This created room will now be visible to all the other users to join and start communicating.

```
$join room1
room1 created
```

3.2 Join room

A user can see the list of rooms available to join using the "$list" command which displays all the rooms created by other users. User can join any of the listed rooms using the "$join room_name" command. We also get the list of people in that room.

```
$list
List of available roomdetails:
room1man
```

3.3 Leave room

A user can leave the room at any moment of the time using the "$leave room_name" command. The room will still be available as there are other users in the room. Otherwise, if there are no users in the room, the room does not exist anymore.

```
$leave room1
You left the room
```

4. IRC Concepts

4.1 One-to-one communication

Here the communication takes place only between two hosts. This might be in the form of a personal chat where the chat is not visible to any other client even if the sender and receiver are a part of the room or client's request to a server. The client first sends its message to the server which then redirects the message to the client that is supposed to receive that message. This is achieved by using the "$personal user_name" command

```
$personal man
[personal message] xyz:
```

4.2 One-to-many communication

This communication takes place when a room is created and multiple users join that room and start communicating. When a user of that room sends a message in that room, that message is viewed by all the other users who are also a part of that room.

4.3 One-to-all communication

A client can send a broadcast message that will be delivered to each and every client and server.

5. Error Handling

a) If a user tries to leave a room which he is not a part of, 'You are not part of any room' message will be displayed.

```
$leave room2
You are not part of any room
```

b) Suppose if a client crashes, then servers, rooms, and the client are notified.

```
['man']
exception occured  [WinError 10054] An existing connection was forcibly closed by the remote host
nick name is man2
room
```

c) Similarly, if the server crashes, the clients are logged out of the and will be notified with a "Server not responding" message.

```
$join room
room created
hello
[room] man: hello
Server not responding
```

d) User who is not a part of a particular room cannot send messages in that room.

e) A client cannot communicate with a client who does notexist.

```
$personal xyz
User not found
```

6. Conclusion & Future Work

The message sharing has been extended to single server and multiple clients either in the form of broadcast messages or private messages. The interaction between different rooms has been implemented. This application can be furthermore extended by implementing few security features and media sharing option. Security features includes eliminating the redundancy of the user's name and passwords for the rooms. Media sharing can be possible with the cryptographic transport protocols.