

### **TCP Simple Broadcast Chat Server and Client:**

In this programming assignment SBCP(Simple Mail Broadcast Protocol) is used to communicate between server and client. SBCP defines a packet structure which can be used for different purposes: joining chat session, sending message, making a query to the server, like the list of online clients, type of message, username.

Whenever a client has to join the chat session, it sends a join request to the server. Server looks into the details of all the present clients. If a client with the same username is present in the system, the server denies the connection and sends a NACK. If the username is not present already, server accepts the join request and user is allowed to participate in the chat session. Server sends the list of clients present in the system. Clients have the capability to check if a user is online. When the client sees that another client has gone offline, it notifies the server. Upon receiving this information from the client, the server deletes the record of that user, freeing up the username. This makes the usernames reusable. Client uses I/O multiplexing to handle simultaneous send and receive.

Server works as the central agent, forwarding the message received from a client to everyone except the client who sent the message. Server only accepts the request from unknown users. When a user makes a request to join, the server checks if a client with the same username is already present. If not, the server adds it to the list of clients and sends it to everyone. If a user exits the chat session unceremoniously, the server detects that and removes that client from the list. Server uses I/O multiplexing to listen from the listening ports and connected clients simultaneously.

### **Test Cases:**

1. Normal operation of the chat client with three clients connected:

```
* server.cpp x Network Program: x
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ./echos 5678
Waiting for connections ....
Connection accepted from: 127.0.0.1, PORT:44864
Client ankit added!
Connection accepted from: 127.0.0.1, PORT:45573
Client mahesh added!
Connection accepted from: 127.0.0.1, PORT:46093
Client user added!
Received msg from ankit: hi
Client ankit Disconnected!
[]

Network Program: x
Last login: Fri Oct 13 17:04:45 2017 from 52.161.27.120
cabox@box-codeanywhere:~/workspace$ ls
Assignment 1 assignment pa2
cabox@box-codeanywhere:~/workspace$ cd pa2/pa2
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ./echo
Invalid usage. Try:
$ ./echo username server_ip server_port
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ./echo localhost 5678
Invalid usage. Try:
$ ./echo username server_ip server_port
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ./echo ankit localhost 5678
Client connecting to 127.0.0.1 ...
hi
^C
cabox@box-codeanywhere:~/workspace/pa2/pa2$

Network Program: x
Last login: Fri Oct 13 17:15:32 2017 from 52.161.27.120
cabox@box-codeanywhere:~/workspace$ cd pa2/pa2
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ./echo user localhost 5678
Client connecting to 127.0.0.1 ...
ankit : hi
[]

Network Program: x
Last login: Fri Oct 13 17:13:38 2017 from 52.161.27.120
cabox@box-codeanywhere:~/workspace$ cd pa2/pa2
cabox@box-codeanywhere:~/workspace/pa2/pa2$ echo mahesh localhost 5678
mahesh localhost 5678
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ./echo mahesh localhost 5678
Client connecting to 127.0.0.1 ...
ankit : hi
[]
```

## 2. Server rejects a client with a duplicate username:

```
* server.cpp x Network Program: x
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ./echos 5678
Waiting for connections ....
Connection accepted from: 127.0.0.1, PORT:44864
Client ankit added!
Connection accepted from: 127.0.0.1, PORT:45573
Client mahesh added!
Connection accepted from: 127.0.0.1, PORT:46093
Client user added!
Received msg from ankit: hi
Client ankit Disconnected!
Received msg from user: clear
Connection accepted from: 127.0.0.1, PORT:49604
Username already exists!
[]

Network Program: x
Last login: Fri Oct 13 17:15:32 2017 from 52.161.27.120
cabox@box-codeanywhere:~/workspace$ cd pa2/pa2
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ./echo user localhost 5678
Client connecting to 127.0.0.1 ...
ankit : hi
clear
[]

Network Program: x
Last login: Fri Oct 13 17:13:38 2017 from 52.161.27.120
cabox@box-codeanywhere:~/workspace$ cd pa2/pa2
cabox@box-codeanywhere:~/workspace/pa2/pa2$ echo mahesh localhost 5678
mahesh localhost 5678
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ./echo mahesh localhost 5678
Client connecting to 127.0.0.1 ...
ankit : hi
user8 : clear
[]
```

## 3. Server allows a previously used username to be reused:

```
* server.cpp x Network Program x Network Program x
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ./echos 5678
Waiting for connections ....
Connection accepted from: 127.0.0.1, PORT:44864
Client ankit added!
Connection accepted from: 127.0.0.1, PORT:45573
Client mahesh added!
Connection accepted from: 127.0.0.1, PORT:46093
Client user added!
Received msg from ankit: hi
Client ankit Disconnected!
Received msg from user: clear
Connection accepted from: 127.0.0.1, PORT:49604
Username already exists!
Connection accepted from: 127.0.0.1, PORT:50472
Client ankit added!
[]

Network Program x Network Program x
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ./echo user localhost 5678
Client connecting to 127.0.0.1 ...
Error Code: -1
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ^C
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ./echo ankit localhost 5678
Client connecting to 127.0.0.1 ...
[]

Last login: Fri Oct 13 17:15:32 2017 from 52.161.27.120
cabox@box-codeanywhere:~/workspace$ cd pa2/pa2
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ./echo user localhost 5678
Client connecting to 127.0.0.1 ...
ankit : hi
clear
[]

Last login: Fri Oct 13 17:13:38 2017 from 52.161.27.120
cabox@box-codeanywhere:~/workspace$ cd pa2/pa2
cabox@box-codeanywhere:~/workspace/pa2/pa2$ echo mahesh localhost 5678
mahesh localhost 5678
cabox@box-codeanywhere:~/workspace/pa2/pa2$ ./echo mahesh localhost 5678
Client connecting to 127.0.0.1 ...
ankit : hi
userB : clear
[]
```

#### 4. Server rejects the client because it exceeds the maximum number of clients allowed:

```
Network Program x Network Program x Network Program x
cabox@box-codeanywhere:~/workspace/pa2$ ./echos localhost 5678 3
Waiting for connections ....
Connection accepted from: 127.0.0.1, PORT:43700
Client ankit added!
Connection accepted from: 127.0.0.1, PORT:43937
Client mahesh added!
Connection accepted from: 127.0.0.1, PORT:44132
Client user added!
[]

cabox@box-codeanywhere:~/workspace$ cd pa2
cabox@box-codeanywhere:~/workspace/pa2$ echo user1 localhost 5678
user1 localhost 5678
cabox@box-codeanywhere:~/workspace/pa2$ []

Network Program x Network Program x
Last login: Fri Oct 13 17:49:35 2017 from 52.161.27.120
cabox@box-codeanywhere:~/workspace$ cd pa2
cabox@box-codeanywhere:~/workspace/pa2$ ./echo ankit localhost 5678
Client connecting to 127.0.0.1 ...
Current User Count: 1
User mahesh is ONLINE
User userG is ONLINE
[]

cabox@box-codeanywhere:~/workspace/pa2$ ./echo user localhost 5678
Client connecting to 127.0.0.1 ...
Current User Count: 3
[]
```

#### 5. Separate test cases for the bonus features:

(a) IPv6 included

(b)

```
cabox@box-codeanywhere:~/workspace/pa2$ ./echos localhost 5678 3
Waiting for connections ....
Connection accepted from: 127.0.0.1, PORT:43700
Client ankit added!
Connection accepted from: 127.0.0.1, PORT:43937
Client mahesh added!
Connection accepted from: 127.0.0.1, PORT:44132
Client user added!
█

cabox@box-codeanywhere:~/workspace/pa2$ ./echo mahesh localhost 5678
Client connecting to 127.0.0.1 ...
Current User Count: 2
User user is ONLINE
█

cabox@box-codeanywhere:~/workspace$ cd pa2
cabox@box-codeanywhere:~/workspace/pa2$ ./echo ankit localhost 5678
Client connecting to 127.0.0.1 ...
Current User Count: 1
User mahesh is ONLINE
User userG is ONLINE
█

cabox@box-codeanywhere:~/workspace/pa2$ ./echo user localhost 5678
Client connecting to 127.0.0.1 ...
Current User Count: 3
█
```

Default - Codeanywhere - Goc

Connected