

CSE 489/589
Programming Assignment 1 Report
Text Chat Application

1 - Group and Contributions

- Name of member 1: Saurabh Tambolkar
 - UBITName: 50412968
 - Contributions – Wrote the files assignment1.c & global.h. Prepared the server side of the report.
- Name of member 2: Shantanu Kumar
 - UBITName: 50418500
 - Contributions – Wrote the file MainProc.c. Prepared the client side of the report.

2 - SHELL Functionality

[5.0] Application Startup

The application is started by executing the command given below –

```
/local/Fall_2021/<Insert-UB-IT-Name>/cse489589_assignment1/<Insert-UB-IT-Name>/assignment1 <'c'-or-'s'> <Port-Number>
```

The character 'c' signifies that the application has been started as a client whereas the character 's' signifies that the application has been started as a server. This is followed by the port number on which the application is running.

Once the application has been started, it is constantly waiting for input from the user, which it reads from the standard input and processes sequentially.

3 - Command for Server and Client

[0.0] AUTHOR

```
underground {/local/Fall_2021/skumar39/cse489589_assignment1/skumar39} > ./assignment1 s 5001
[AUTHOR]
[AUTHOR:SUCCESS]
I, skumar39, have read and understood the course academic integrity policy.
[AUTHOR:END]
```

The command AUTHOR states that we have understood and abide by the course academic integrity policy.

[5.0] IP

```
highgate {/local/Fall_2021/skumar39/cse489589_assignment1/skumar39} > ./assignment1 s 5001
[IP]
[IP:SUCCESS]
IP:128.205.36.33
[IP:END]
```

This command prints the IP address of the current process and is handled by the function get_localIP().

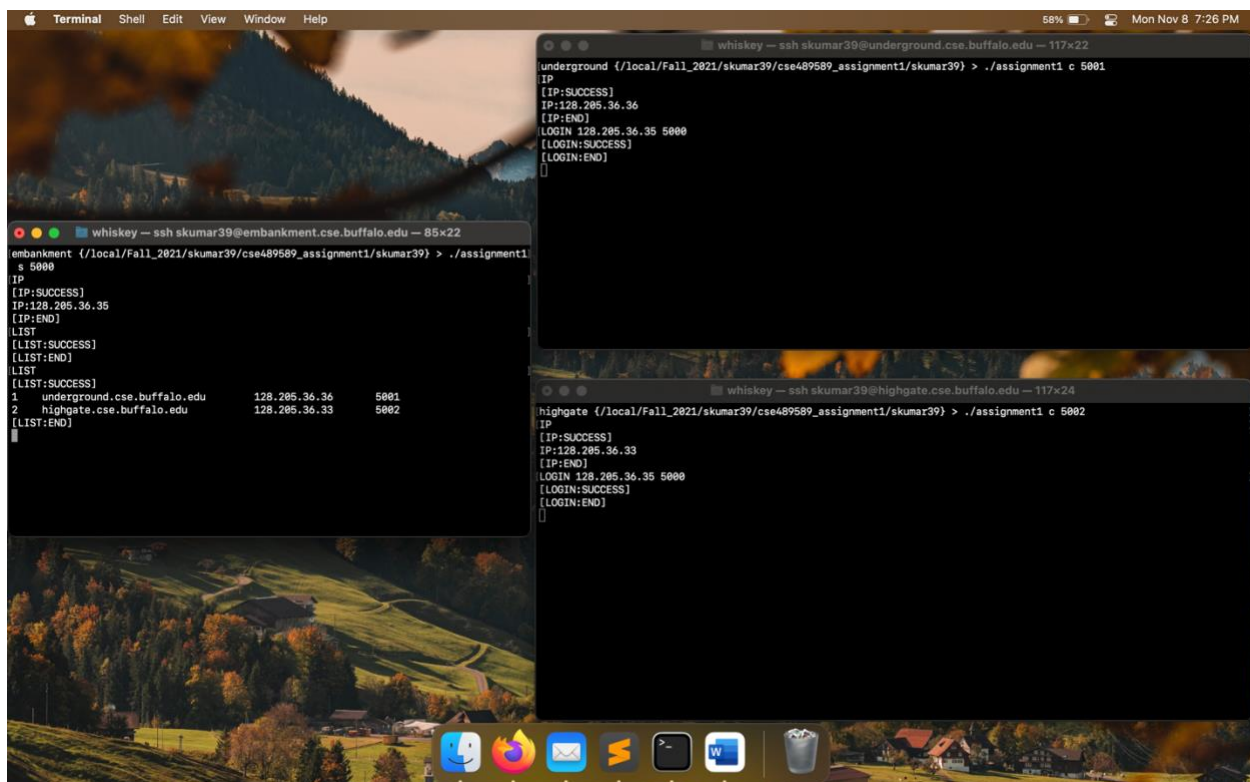
[2.5] PORT

```
[highgate {/local/Fall_2021/skumar39/cse489589_assignment1/skumar39} > ./assignment1 s 5001
[IP
[IP:SUCCESS]
IP:128.205.36.33
[IP:END]
[PORT
[PORT:SUCCESS]
PORT:5001
[PORT:END]
```

Fetches the port number on which the application is running.

This is handled by capturing the port number the user mentions when starting the application and then outputting it when this command is executed.

[10.0] LIST

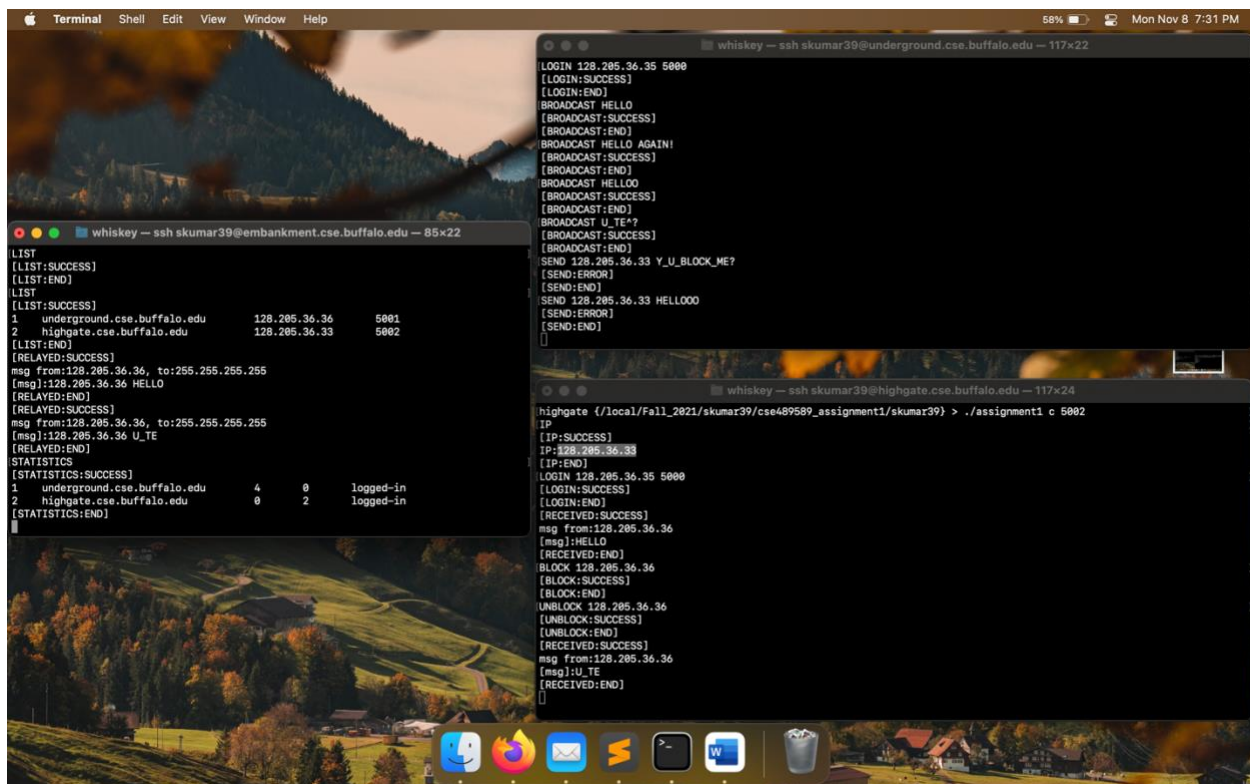


Displays the list of all the currently logged in clients along with their hostname, IP address and listening port number. The struct connection that is defined as a global structure is used to keep tab of all the current connections. When the command 'LIST' is executed, we iterate through all the elements of the structure array, find the ones whose current status is shown as logged in and print them along with their IP address and port number.

4 - Command/Event for Server

[5.0] STATISTICS

Displays a list of all the clients that have logged in at the server at any point of time along with the number of messages sent by them, the number of messages received by them and their current status. This is again handled by the struct connections which maintains a list of all the connections that have been established at any point of time. The variable `connections[i].msg_sent` & `connections[i].msg_received` keep a tab of the number of messages sent and received by a hostname respectively. If `connections[i].status` is equal to the value stored in `logged_in`, that particular host is logged in, otherwise not.



```
Terminal Shell Edit View Window Help 58% Mon Nov 8 7:31 PM

whiskey — ssh skumar39@embankment.cse.buffalo.edu — 85x22
[LIST:SUCCESS]
[LIST:END]
[LIST]
[LIST:SUCCESS]
1 underground.cse.buffalo.edu 128.205.36.36 5001
2 highgate.cse.buffalo.edu 128.205.36.33 5002
[LIST:END]
[RELAYED:SUCCESS]
msg from:128.205.36.36, to:255.255.255.255
[msg]:128.205.36.36 HELLO
[RELAYED:END]
[RELAYED:SUCCESS]
msg from:128.205.36.36, to:255.255.255.255
[msg]:128.205.36.36 U_TE
[RELAYED:END]
STATISTICS
[STATISTICS:SUCCESS]
1 underground.cse.buffalo.edu 4 0 logged-in
2 highgate.cse.buffalo.edu 0 2 logged-in
[STATISTICS:END]

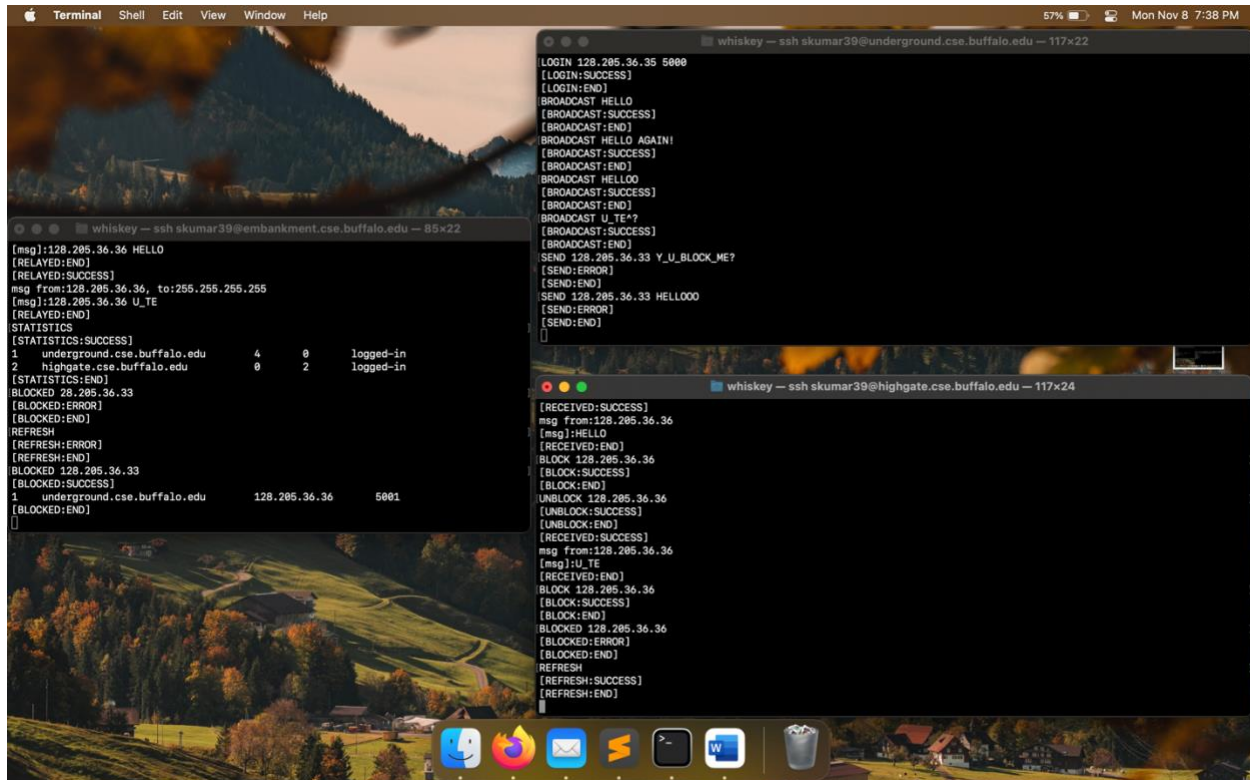
whiskey — ssh skumar39@underground.cse.buffalo.edu — 117x22
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
BROADCAST HELLO
[BROADCAST:SUCCESS]
[BROADCAST:END]
BROADCAST HELLO AGAIN!
[BROADCAST:SUCCESS]
[BROADCAST:END]
BROADCAST HELLOO
[BROADCAST:SUCCESS]
[BROADCAST:END]
BROADCAST U_TE?
[BROADCAST:SUCCESS]
[BROADCAST:END]
SEND 128.205.36.33 Y_U_BLOCK_ME?
[SEND:ERROR]
[SEND:END]
SEND 128.205.36.33 HELLOO
[SEND:ERROR]
[SEND:END]

whiskey — ssh skumar39@highgate.cse.buffalo.edu — 117x24
highgate {/local/Fall_2021/skumar39/cse489589_assignment1/skumar39} > ./assignment1 c 5002
[IP:SUCCESS]
IP:128.205.36.33
[IP:END]
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
[RECEIVED:SUCCESS]
msg from:128.205.36.36
[msg]:HELLO
[RECEIVED:END]
BLOCK 128.205.36.36
[BLOCK:SUCCESS]
[BLOCK:END]
UNBLOCK 128.205.36.36
[UNBLOCK:SUCCESS]
[UNBLOCK:END]
[RECEIVED:SUCCESS]
msg from:128.205.36.36
[msg]:U_TE
[RECEIVED:END]
```

Here in the terminal on the left we have the server whereas the two terminals on the right represent two clients. Both the clients then connect to the server and exchange messages as shown above. We then execute the command `STATISTICS` on the server which then displays the clients and the number of messages sent and received by each.

[7.0] BLOCKED <client-ip> + Exception Handling

Displays a list of all the clients blocked by the client with a particular IP address. The struct connections contains an array blockedIPs that maintains a list of all the IP address that have been blocked by that client. We simply iterate through this list and print all the elements in it.



```
Terminal Shell Edit View Window Help 57% Mon Nov 8 7:38 PM

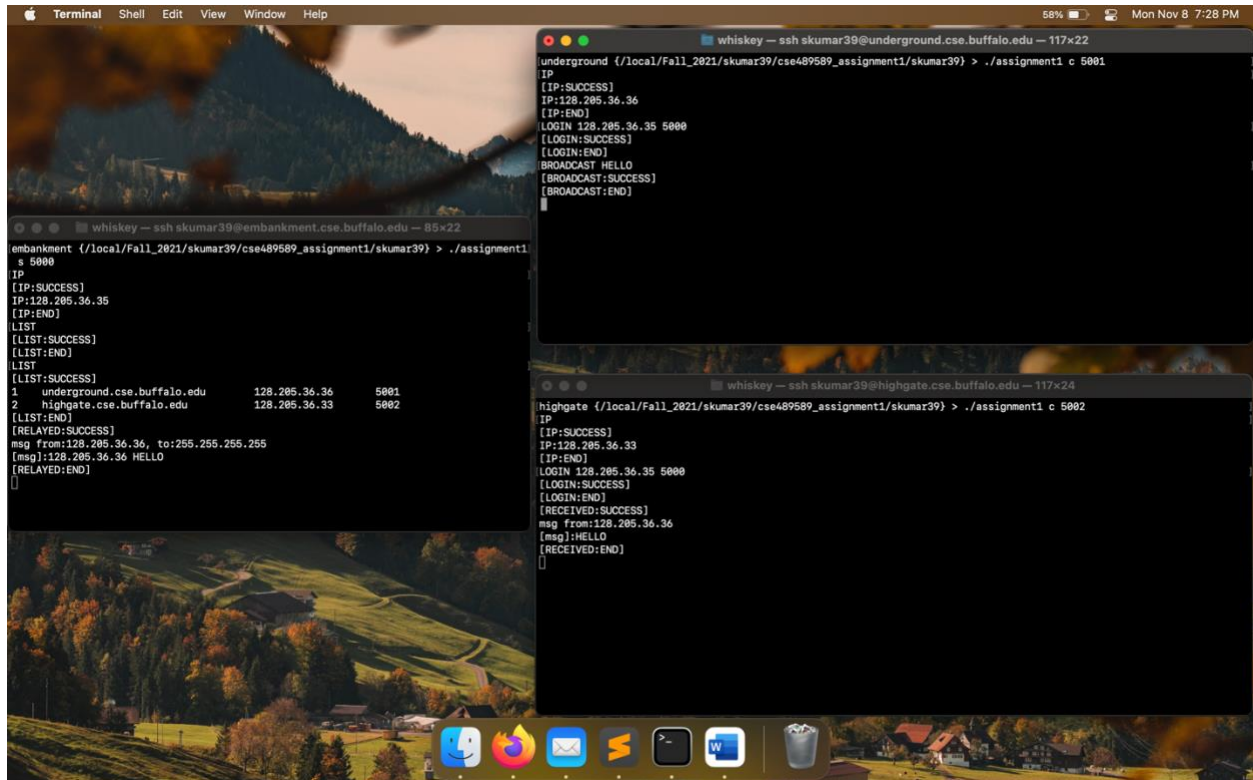
whiskey — ssh skumar39@embankment.cse.buffalo.edu — 85x22
[msg]:128.205.36.36 HELLO
[RELAYED:END]
[RELAYED:SUCCESS]
msg from:128.205.36.36, to:255.255.255.255
[msg]:128.205.36.36 U_TE
[RELAYED:END]
STATISTICS
[STATISTICS:SUCCESS]
1 underground.cse.buffalo.edu 4 0 logged-in
2 highgate.cse.buffalo.edu 0 2 logged-in
[STATISTICS:END]
BLOCKED 28.205.36.33
[BLOCKED:ERROR]
[BLOCKED:END]
REFRESH
[REFRESH:ERROR]
[REFRESH:END]
BLOCKED 128.205.36.33
[BLOCKED:SUCCESS]
1 underground.cse.buffalo.edu 128.205.36.36 5001
[BLOCKED:END]

whiskey — ssh skumar39@underground.cse.buffalo.edu — 117x22
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
BROADCAST HELLO
[BROADCAST:SUCCESS]
[BROADCAST:END]
BROADCAST HELLO AGAIN!
[BROADCAST:SUCCESS]
[BROADCAST:END]
BROADCAST HELLOO
[BROADCAST:SUCCESS]
[BROADCAST:END]
BROADCAST U_TE*?
[BROADCAST:SUCCESS]
[BROADCAST:END]
SEND 128.205.36.33 Y_U_BLOCK_ME?
[SEND:ERROR]
[SEND:END]
SEND 128.205.36.33 HELLOOO
[SEND:ERROR]
[SEND:END]

whiskey — ssh skumar39@highgate.cse.buffalo.edu — 117x24
[RECEIVED:SUCCESS]
msg from:128.205.36.36
[msg]:HELLO
[RECEIVED:END]
BLOCK 128.205.36.36
[BLOCK:SUCCESS]
[BLOCK:END]
UNBLOCK 128.205.36.36
[UNBLOCK:SUCCESS]
[UNBLOCK:END]
[RECEIVED:SUCCESS]
msg from:128.205.36.36
[msg]:U_TE
[RECEIVED:END]
BLOCK 128.205.36.36
[BLOCK:SUCCESS]
[BLOCK:END]
BLOCKED 128.205.36.36
[BLOCKED:ERROR]
[BLOCKED:END]
REFRESH
[REFRESH:SUCCESS]
[REFRESH:END]
```

Again, the server is on the left and the clients on the right. The client on the bottom right blocks the client on the top left which is then evident when the server on the left runs the BLOCKED command.

[EVENT]: Message Relayed



The screenshot shows a macOS desktop with a scenic wallpaper of a mountain valley. Three terminal windows are open, each displaying network traffic logs from a program named 'assignment1'. The top-left window is titled 'embankment' and shows a list of connections. The top-right window is titled 'underground' and shows a single connection log. The bottom-right window is titled 'highgate' and shows a single connection log. The system status bar at the top right indicates 58% battery and the date/time 'Mon Nov 8 7:28 PM'.

```
Terminal Shell Edit View Window Help 58% Mon Nov 8 7:28 PM

embankment (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5000
IP
[IP:SUCCESS]
IP:128.205.36.35
[IP:END]
LIST
[LIST:SUCCESS]
[LIST:END]
LIST
[LIST:SUCCESS]
1 underground.cse.buffalo.edu 128.205.36.36 5001
2 highgate.cse.buffalo.edu 128.205.36.33 5002
[LIST:END]
[RELAYED:SUCCESS]
msg from:128.205.36.36, to:255.255.255.255
[msg]:128.205.36.36 HELLO
[RELAYED:END]

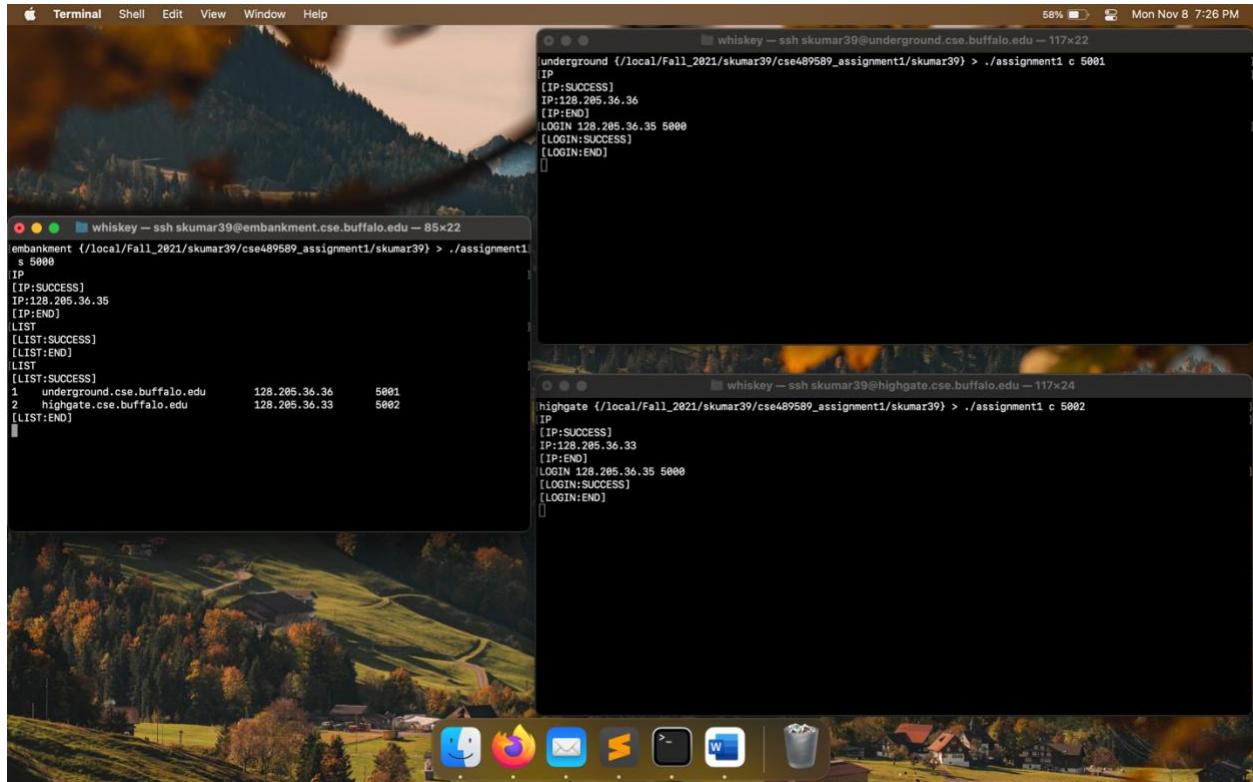
underground (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5001
IP
[IP:SUCCESS]
IP:128.205.36.36
[IP:END]
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
BROADCAST HELLO
[BROADCAST:SUCCESS]
[BROADCAST:END]

highgate (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5002
IP
[IP:SUCCESS]
IP:128.205.36.33
[IP:END]
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
[RECEIVED:SUCCESS]
msg from:128.205.36.36
[msg]:HELLO
[RECEIVED:END]
```

Message relayed.

5 - Command/Event for Client

[17.0] LOGIN <server-ip> <server-port> + Exception Handling



The screenshot shows a macOS desktop with three terminal windows. The top-left window is titled 'whiskey — ssh skumar39@embankment.cse.buffalo.edu — 85x22' and shows the command `./assignment1 s 5000` and its output. The top-right window is titled 'whiskey — ssh skumar39@underground.cse.buffalo.edu — 117x22' and shows the command `./assignment1 c 5001` and its output. The bottom-right window is titled 'whiskey — ssh skumar39@highgate.cse.buffalo.edu — 117x24' and shows the command `./assignment1 c 5002` and its output. The desktop background is a scenic view of a valley with green fields and trees. The dock at the bottom contains icons for various applications including Finder, Firefox, Mail, and a terminal icon.

```
embankment (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 s 5000
IP
[IP:SUCCESS]
IP:128.205.36.35
[IP:END]
LIST
[LIST:SUCCESS]
[LIST:END]
LIST
[LIST:SUCCESS]
1 underground.cse.buffalo.edu 128.205.36.36 5001
2 highgate.cse.buffalo.edu 128.205.36.33 5002
[LIST:END]
```

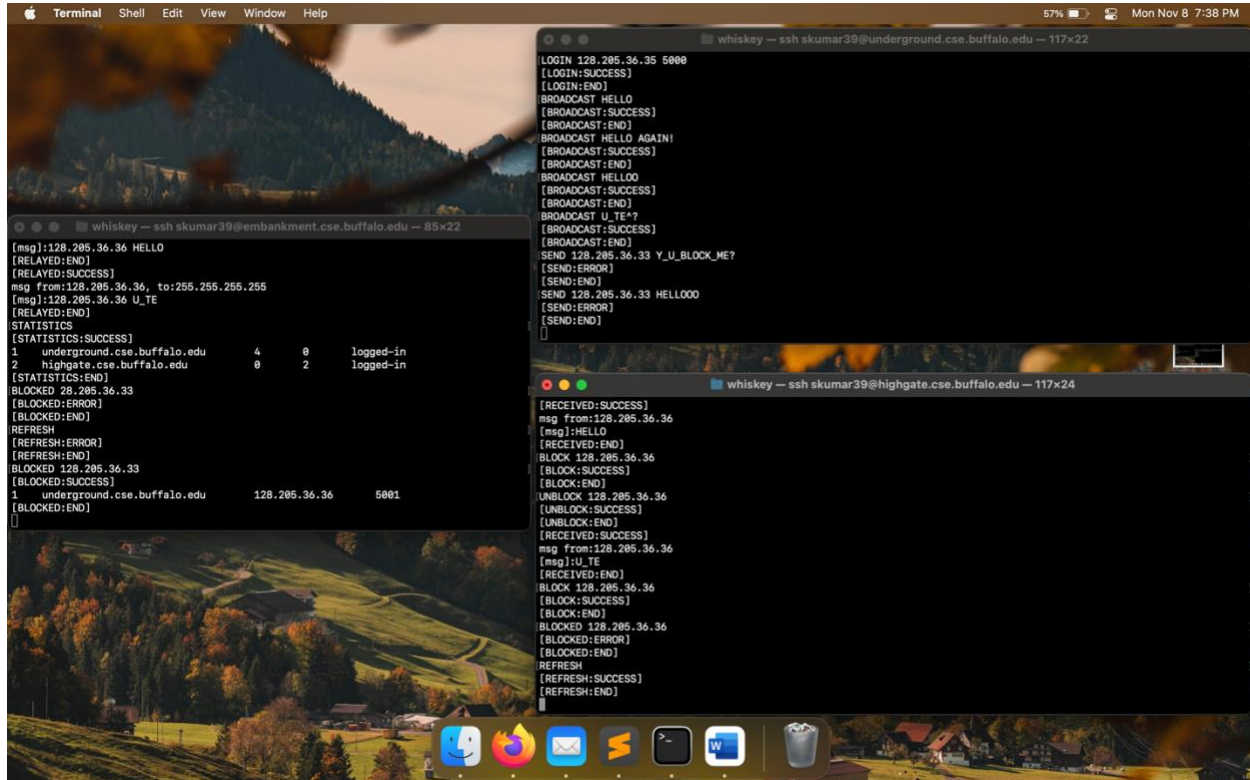
```
underground (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5001
IP
[IP:SUCCESS]
IP:128.205.36.36
[IP:END]
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
```

```
highgate (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5002
IP
[IP:SUCCESS]
IP:128.205.36.33
[IP:END]
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
```

This command is used by a client to login to the server located a particular IP address.

[5.0] REFRESH

Fetches the list of all the currently logged in clients from the server. We iterate through the struct connections, check the status of all the clients, those whose status is currently equal to `logged_in` are printed.



```
Terminal Shell Edit View Window Help 57% Mon Nov 8 7:38 PM

whiskey — ssh skumar39@embankment.cse.buffalo.edu — 85x22
[msg]:128.205.36.36 HELLO
[RELAYED:END]
[RELAYED:SUCCESS]
msg from:128.205.36.36, to:255.255.255.255
[msg]:128.205.36.36 U_TE
[RELAYED:END]
STATISTICS
[STATISTICS:SUCCESS]
1  underground.cse.buffalo.edu  4  0  logged-in
2  highgate.cse.buffalo.edu     0  2  logged-in
[STATISTICS:END]
BLOCKED 28.205.36.33
[BLOCKED:ERROR]
[BLOCKED:END]
REFRESH
[REFRESH:ERROR]
[REFRESH:END]
BLOCKED 128.205.36.33
[BLOCKED:SUCCESS]
1  underground.cse.buffalo.edu  128.205.36.36  5001
[BLOCKED:END]

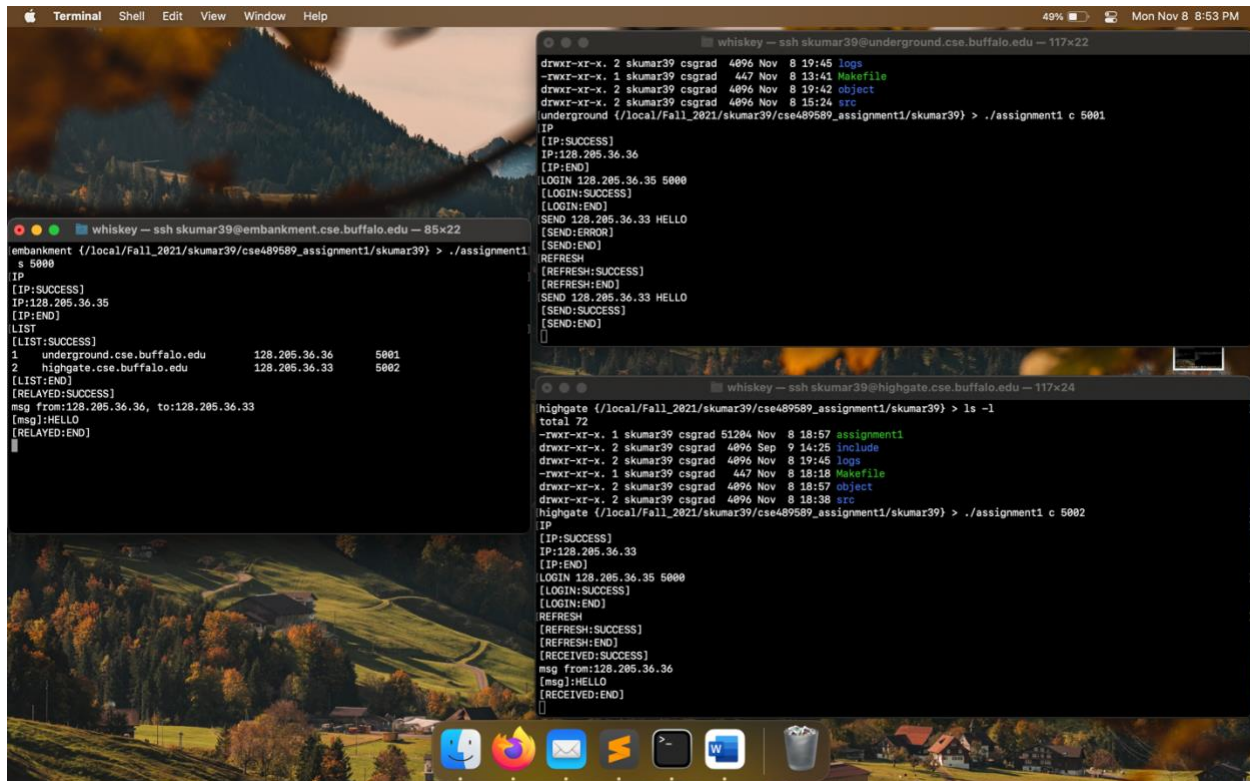
whiskey — ssh skumar39@underground.cse.buffalo.edu — 117x22
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
BROADCAST HELLO
[BROADCAST:SUCCESS]
[BROADCAST:END]
BROADCAST HELLO AGAIN!
[BROADCAST:SUCCESS]
[BROADCAST:END]
BROADCAST HELLOO
[BROADCAST:SUCCESS]
[BROADCAST:END]
BROADCAST U_TE*?
[BROADCAST:SUCCESS]
[BROADCAST:END]
SEND 128.205.36.33 Y_U_BLOCK_ME?
[SEND:ERROR]
[SEND:END]
SEND 128.205.36.33 HELLOOO
[SEND:ERROR]
[SEND:END]

whiskey — ssh skumar39@highgate.cse.buffalo.edu — 117x24
[RECEIVED:SUCCESS]
msg from:128.205.36.36
[msg]:HELLO
[RECEIVED:END]
BLOCK 128.205.36.36
[BLOCK:SUCCESS]
[BLOCK:END]
UNBLOCK 128.205.36.36
[UNBLOCK:SUCCESS]
[UNBLOCK:END]
[RECEIVED:SUCCESS]
msg from:128.205.36.36
[msg]:U_TE
[RECEIVED:END]
BLOCK 128.205.36.36
[BLOCK:SUCCESS]
[BLOCK:END]
BLOCKED 128.205.36.36
[BLOCKED:ERROR]
[BLOCKED:END]
REFRESH
[REFRESH:SUCCESS]
[REFRESH:END]
```

The server displays the correct output for the command `BLOCKED` after a `REFRESH` is executed at one of the clients post blocking the other client.

[17.0] SEND <client-ip> <msg> + Exception Handling

Sends a message to the provided IP address. For the message to be successfully transmitted, the sender should not be in the list of IPs blocked by the receiver and the receiver should not be in the blocked list of the sender. Also, the user to which the message is being sent should be valid and currently logged in, otherwise the message will not be sent, and an error will be generated. This command also increments the count of messages sent by the sender the count of messages received by the receiver.



The screenshot shows a macOS desktop with three terminal windows. The top-left window is titled 'whiskey - ssh skumar39@embankment.cse.buffalo.edu - 85x22' and shows the output of the 'assignment1' command, including a list of users and a successful message relay. The top-right window is titled 'whiskey - ssh skumar39@underground.cse.buffalo.edu - 117x22' and shows a series of status messages like '[IP:SUCCESS]', '[LOGIN:SUCCESS]', and '[SEND:SUCCESS]'. The bottom-right window is titled 'whiskey - ssh skumar39@highgate.cse.buffalo.edu - 117x24' and shows the output of the 'ls -l' command, listing files in the current directory. The desktop background is a scenic view of a valley with green fields and trees. The dock at the bottom contains icons for various applications including Finder, Firefox, Mail, and VS Code.

```
embankment (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1
s 5000
IP
[IP:SUCCESS]
IP:128.205.36.35
[IP:END]
LIST
[LIST:SUCCESS]
1  underground.cse.buffalo.edu  128.205.36.36  5001
2  highgate.cse.buffalo.edu     128.205.36.33  5002
[LIST:END]
[RELAYED:SUCCESS]
msg from:128.205.36.36, to:128.205.36.33
[msg]:HELLO
[RELAYED:END]

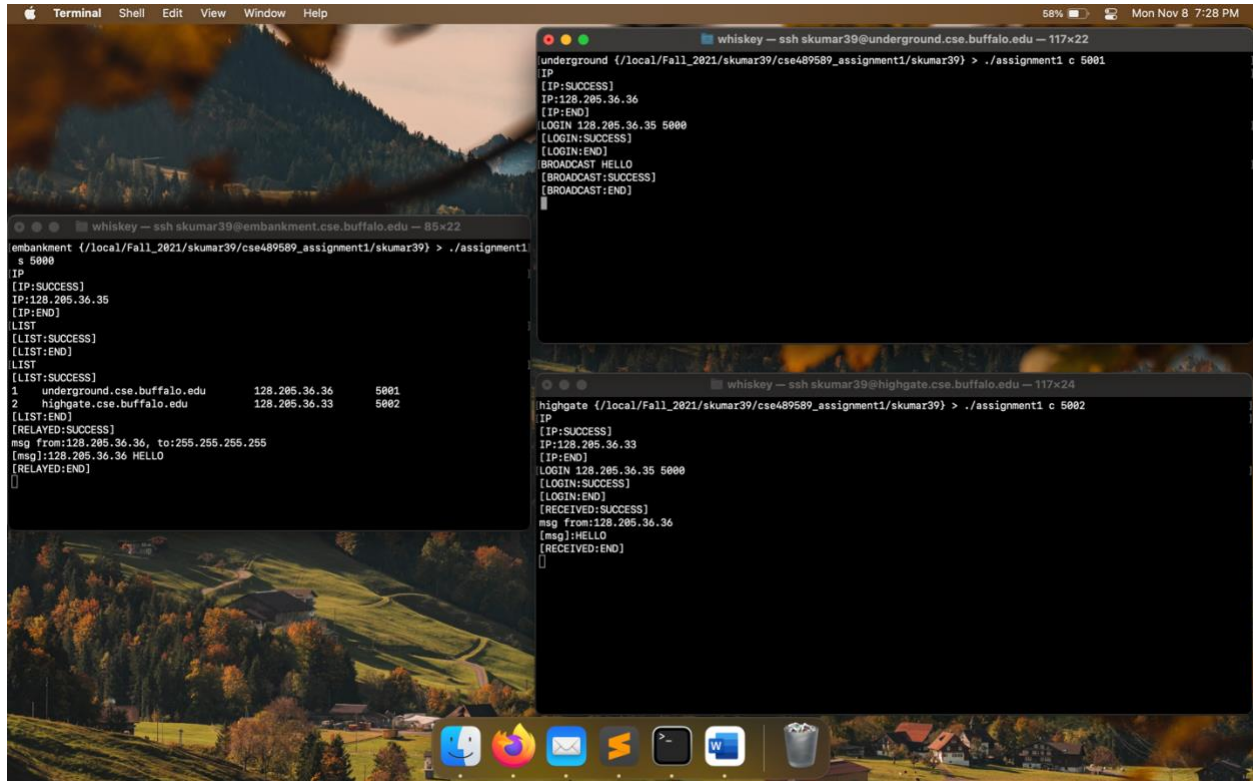
whiskey - ssh skumar39@underground.cse.buffalo.edu - 117x22
drwxr-xr-x. 2 skumar39 csgrad 4096 Nov 8 19:45 logs
-rwxr-xr-x. 1 skumar39 csgrad 447 Nov 8 13:41 Makefile
drwxr-xr-x. 2 skumar39 csgrad 4096 Nov 8 19:42 object
drwxr-xr-x. 2 skumar39 csgrad 4096 Nov 8 15:24 src
underground (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5001
IP
[IP:SUCCESS]
IP:128.205.36.36
[IP:END]
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
SEND 128.205.36.33 HELLO
[SEND:ERROR]
[SEND:END]
REFRESH
[REFRESH:SUCCESS]
[REFRESH:END]
SEND 128.205.36.33 HELLO
[SEND:SUCCESS]
[SEND:END]

whiskey - ssh skumar39@highgate.cse.buffalo.edu - 117x24
highgate (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ls -l
total 72
-rwxr-xr-x. 1 skumar39 csgrad 51204 Nov 8 18:57 assignment1
drwxr-xr-x. 2 skumar39 csgrad 4096 Sep 9 14:25 include
drwxr-xr-x. 2 skumar39 csgrad 4096 Nov 8 19:45 logs
-rwxr-xr-x. 1 skumar39 csgrad 447 Nov 8 18:18 Makefile
drwxr-xr-x. 2 skumar39 csgrad 4096 Nov 8 18:57 object
drwxr-xr-x. 2 skumar39 csgrad 4096 Nov 8 18:38 src
highgate (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5002
IP
[IP:SUCCESS]
IP:128.205.36.33
[IP:END]
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
REFRESH
[REFRESH:SUCCESS]
[REFRESH:END]
[RECEIVED:SUCCESS]
msg from:128.205.36.36
[msg]:HELLO
[RECEIVED:END]
```

Message received by the client on the bottom right when sent by the client on the top right.

[10.0] BROADCAST <msg>

Sends a message to all the currently logged in clients that are not a part of the list of IPs blocked by the sender. A receiver that has blocked the sender's IP address will also not receive the message. Like the SEND command, this also increments the count of messages sent by the client and the count of messages received by the currently logged in recipients.



```
Terminal Shell Edit View Window Help 88% Mon Nov 8 7:28 PM

whiskey — ssh skumar39@underground.cse.buffalo.edu — 117x22
underground (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5001
IP
[IP:SUCCESS]
IP:128.205.36.36
[IP:END]
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
BROADCAST HELLO
[BROADCAST:SUCCESS]
[BROADCAST:END]

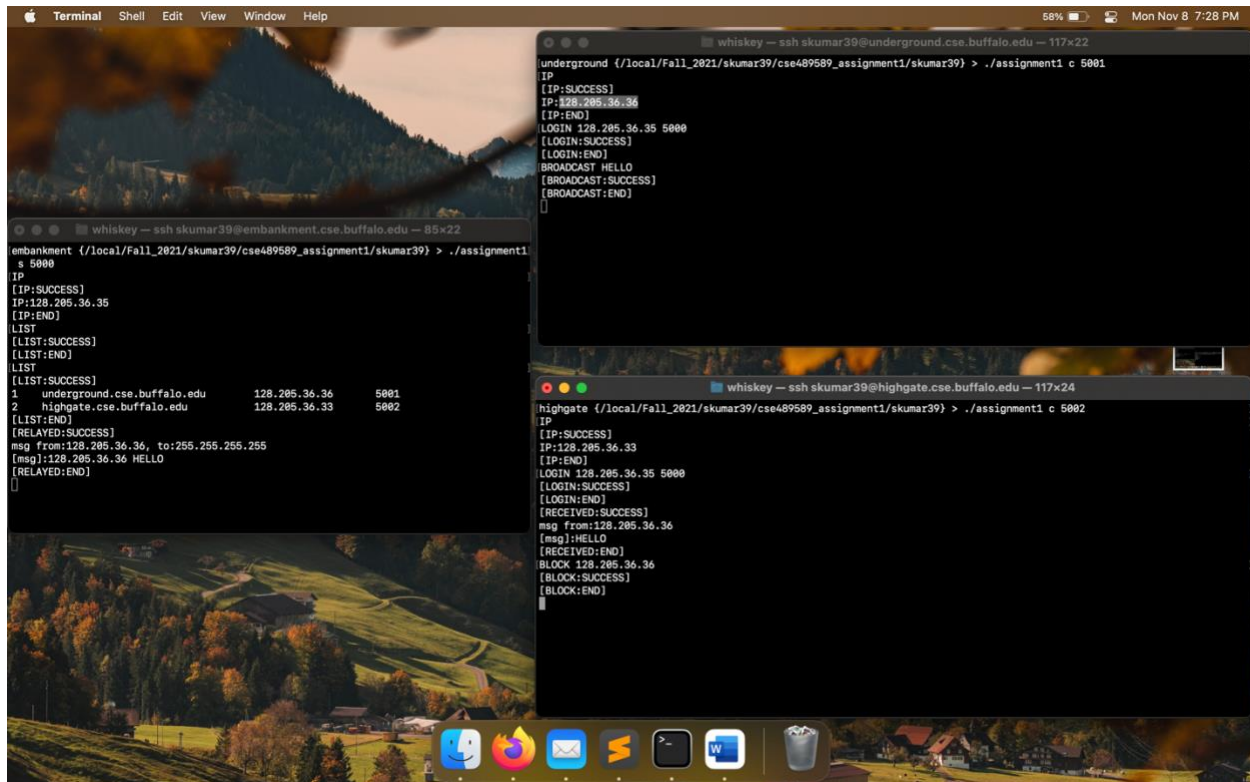
whiskey — ssh skumar39@embankment.cse.buffalo.edu — 85x22
embankment (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1
s 5000
IP
[IP:SUCCESS]
IP:128.205.36.35
[IP:END]
LIST
[LIST:SUCCESS]
[LIST:END]
LIST
[LIST:SUCCESS]
1 underground.cse.buffalo.edu 128.205.36.36 5001
2 highgate.cse.buffalo.edu 128.205.36.33 5002
[LIST:END]
[RELAYED:SUCCESS]
msg from:128.205.36.36, to:255.255.255.255
[msg]:128.205.36.36 HELLO
[RELAYED:END]

whiskey — ssh skumar39@highgate.cse.buffalo.edu — 117x24
highgate (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5002
IP
[IP:SUCCESS]
IP:128.205.36.33
[IP:END]
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
[RECEIVED:SUCCESS]
msg from:128.205.36.36
[msg]:HELLO
[RECEIVED:END]
```

Broadcasted message by the client (top right) received by the other client (bottom right).

[7.0] BLOCK <client-ip> + Exception Handling

Blocks a given IP for a client. Struct connection has an array blockedIPs that contains the list of all the IPs blocked for that client. The BLOCK command simply places the given IP in that array. A client is prohibited from blocking itself and trying to re-block an IP that is already blocked has no effect.



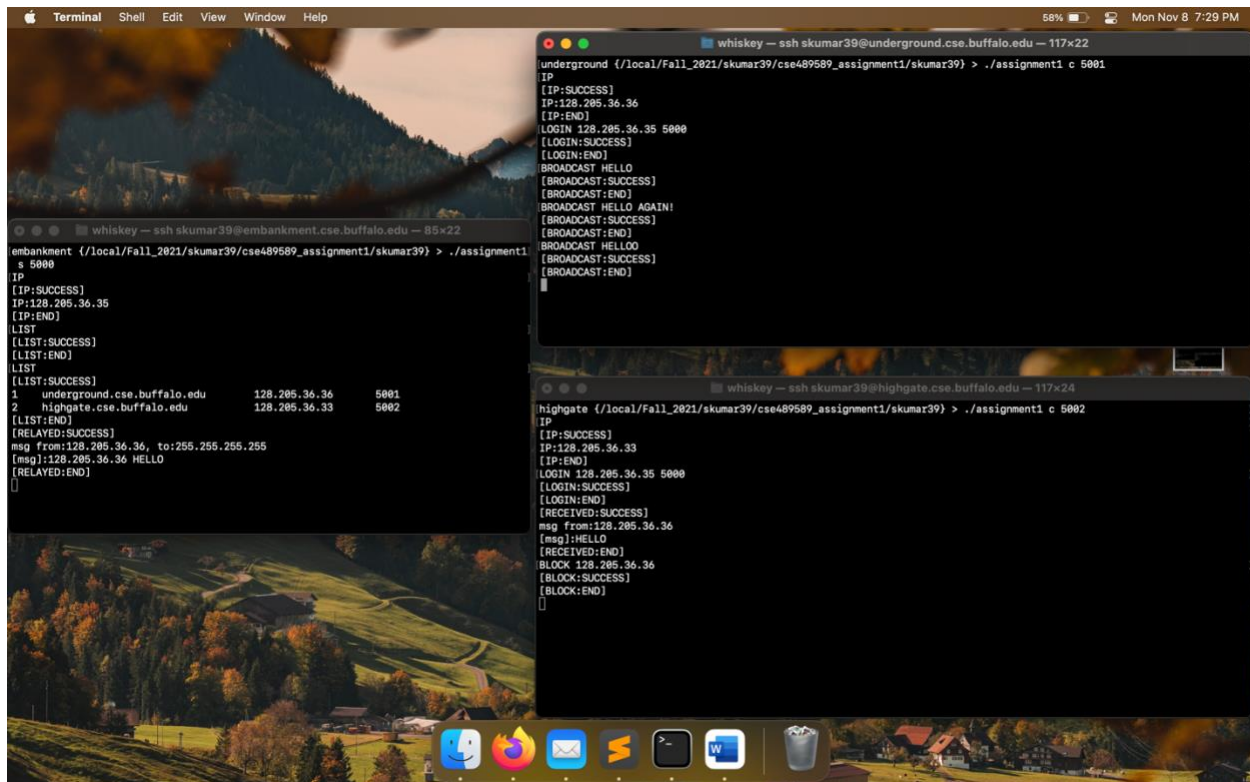
```
Terminal Shell Edit View Window Help 68% Mon Nov 8 7:28 PM

whiskey — ssh skumar39@embankment.cse.buffalo.edu — 85x22
embankment (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1
s 5000
IP
[IP:SUCCESS]
IP:128.205.36.35
[IP:END]
LIST
[LIST:SUCCESS]
[LIST:END]
LIST
[LIST:SUCCESS]
1 underground.cse.buffalo.edu 128.205.36.36 5001
2 highgate.cse.buffalo.edu 128.205.36.33 5002
[LIST:END]
[RELAYED:SUCCESS]
msg from:128.205.36.36, to:255.255.255.255
[msg]:128.205.36.36 HELLO
[RELAYED:END]

whiskey — ssh skumar39@underground.cse.buffalo.edu — 117x22
underground (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5001
IP
[IP:SUCCESS]
IP:128.205.36.36
[IP:END]
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
BROADCAST HELLO
[BROADCAST:SUCCESS]
[BROADCAST:END]

whiskey — ssh skumar39@highgate.cse.buffalo.edu — 117x24
highgate (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5002
IP
[IP:SUCCESS]
IP:128.205.36.33
[IP:END]
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
[RECEIVED:SUCCESS]
msg from:128.205.36.36
[msg]:HELLO
[RECEIVED:END]
BLOCK 128.205.36.36
[BLOCK:SUCCESS]
[BLOCK:END]
```

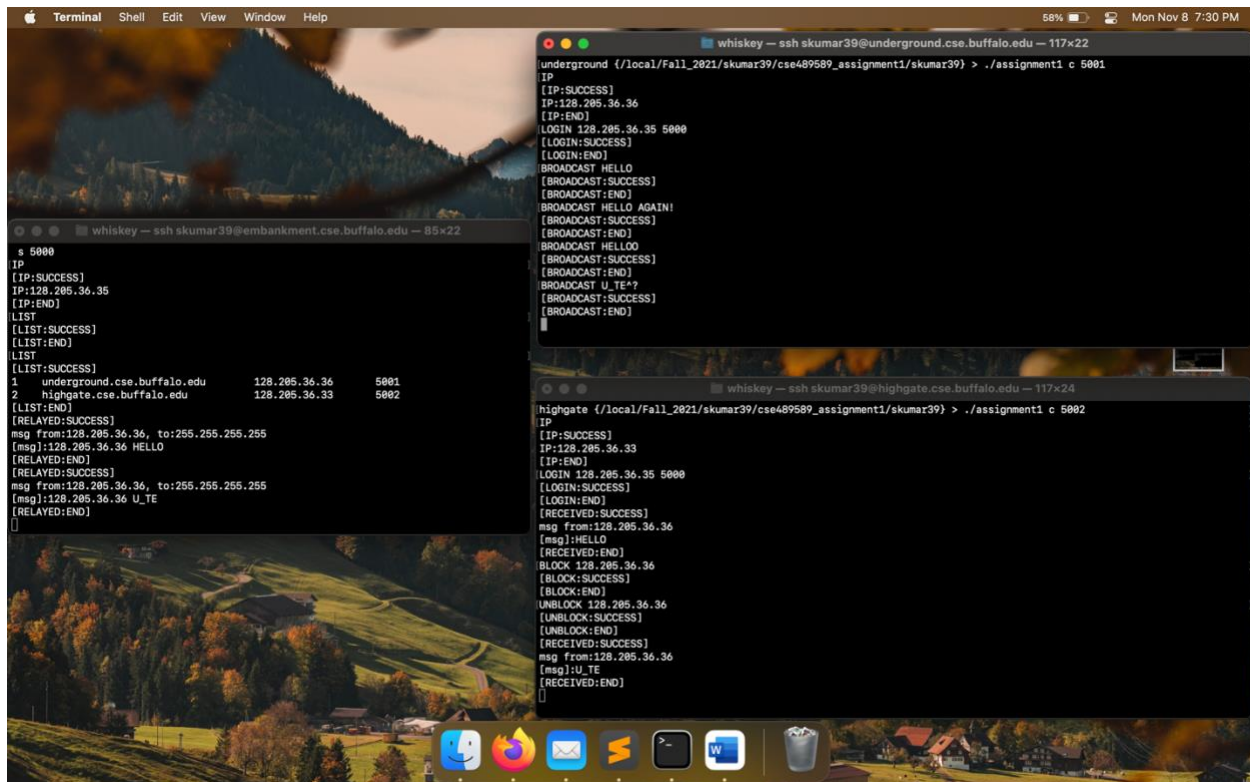
Here the client on the bottom right blocks the client on the top right.



The client on the top right then attempts to message the client on the bottom right but fails since it is currently blocked.

[4.5] UNBLOCK <client-ip> + Exception Handling

Unblocks an already blocked IP. Removes that IP from the blockedIP array by iterating through all its elements. Only an IP that has been previously blocked can be unblocked using this command. Trying to unblock an IP that has not been blocked will result in an error.



The screenshot shows a macOS desktop with three terminal windows. The top-left window is titled 'whiskey - ssh skumar39@embankment.cse.buffalo.edu - 85x22' and shows a list of active connections. The top-right window is titled 'whiskey - ssh skumar39@underground.cse.buffalo.edu - 117x22' and shows a series of network events including IP success, login, broadcast, and block/unblock actions. The bottom-right window is titled 'whiskey - ssh skumar39@highgate.cse.buffalo.edu - 117x24' and shows similar network events, including a successful unblock of IP 128.205.36.36. The desktop background is a scenic image of a valley with autumn foliage. The dock at the bottom contains icons for Finder, Firefox, Mail, Slack, a terminal icon, and a trash can.

```
whiskey - ssh skumar39@embankment.cse.buffalo.edu - 85x22
s 5000
IP
[IP:SUCCESS]
IP:128.205.36.35
[IP:END]
LIST
[LIST:SUCCESS]
[LIST:END]
LIST
[LIST:SUCCESS]
1  underground.cse.buffalo.edu  128.205.36.36  5001
2  highgate.cse.buffalo.edu    128.205.36.33  5002
[LIST:END]
[RELAYED:SUCCESS]
msg from:128.205.36.36, to:255.255.255.255
[msg]:128.205.36.36 HELLO
[RELAYED:END]
[RELAYED:SUCCESS]
msg from:128.205.36.36, to:255.255.255.255
[msg]:128.205.36.36 U_TE
[RELAYED:END]

whiskey - ssh skumar39@underground.cse.buffalo.edu - 117x22
underground (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5001
IP
[IP:SUCCESS]
IP:128.205.36.36
[IP:END]
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
BROADCAST HELLO
[BROADCAST:SUCCESS]
[BROADCAST:END]
BROADCAST HELLO AGAIN!
[BROADCAST:SUCCESS]
[BROADCAST:END]
BROADCAST HELLO
[BROADCAST:SUCCESS]
[BROADCAST:END]
BROADCAST U_TE?
[BROADCAST:SUCCESS]
[BROADCAST:END]

whiskey - ssh skumar39@highgate.cse.buffalo.edu - 117x24
highgate (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5002
IP
[IP:SUCCESS]
IP:128.205.36.33
[IP:END]
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
[RECEIVED:SUCCESS]
msg from:128.205.36.36
[msg]:HELLO
[RECEIVED:END]
BLOCK 128.205.36.36
[BLOCK:SUCCESS]
[BLOCK:END]
UNBLOCK 128.205.36.36
[UNBLOCK:SUCCESS]
[UNBLOCK:END]
[RECEIVED:SUCCESS]
msg from:128.205.36.36
[msg]:U_TE
[RECEIVED:END]
```

The client on the bottom right starts receiving messages from client on the top right after it gets unblocked.

[2.5] LOGOUT

This command is used to logout from a server. This command closes the socket associated with the connection and clears its file descriptor using the FD_CLR command. However, this command does not destroy any states associated with the client, those remain intact in the connection structure and can be accessed in the future.

```
Terminal Shell Edit View Window Help 96% Mon Nov 8 7:47 PM

whiskey — ssh skumar39@embankment.cse.buffalo.edu — 85x22
embankment (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1
s 5000
embankment (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1
s 5005
ip
[IP:SUCCESS]
IP:128.205.36.35
[IP:END]
LIST
[LIST:SUCCESS]
1 underground.cse.buffalo.edu 128.205.36.36 5006
2 highgate.cse.buffalo.edu 128.205.36.33 5007
[LIST:END]
LIST
[LIST:SUCCESS]
1 underground.cse.buffalo.edu 128.205.36.36 5006
[LIST:END]
LIST
[LIST:SUCCESS]
[LIST:END]

whiskey — ssh skumar39@underground.cse.buffalo.edu — 117x22
underground (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5003
1
LOGIN 128.205.36.35 5000
[LOGIN:SUCCESS]
[LOGIN:END]
LOGOUT
[LOGOUT:SUCCESS]
[LOGOUT:END]
REFRESH
[REFRESH:ERROR]
[REFRESH:END]
EXIT
[EXIT:SUCCESS]
[EXIT:END]
underground (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5006
LOGIN 128.205.36.35 5005
[LOGIN:SUCCESS]
[LOGIN:END]
LOGOUT
[LOGOUT:SUCCESS]
[LOGOUT:END]

whiskey — ssh skumar39@highgate.cse.buffalo.edu — 117x24
BLOCK 128.205.36.36
[BLOCK:SUCCESS]
[BLOCK:END]
BLOCKED 128.205.36.36
[BLOCKED:ERROR]
[BLOCKED:END]
REFRESH
[REFRESH:SUCCESS]
[REFRESH:END]
IP
[IP:SUCCESS]
IP:128.205.36.33
[IP:END]
EXIT
[EXIT:SUCCESS]
[EXIT:END]
highgate (/local/Fall_2021/skumar39/cse489589_assignment1/skumar39) > ./assignment1 c 5007
LOGIN 128.205.36.35 5005
[LOGIN:SUCCESS]
[LOGIN:END]
LOGOUT
[LOGOUT:SUCCESS]
[LOGOUT:END]
```

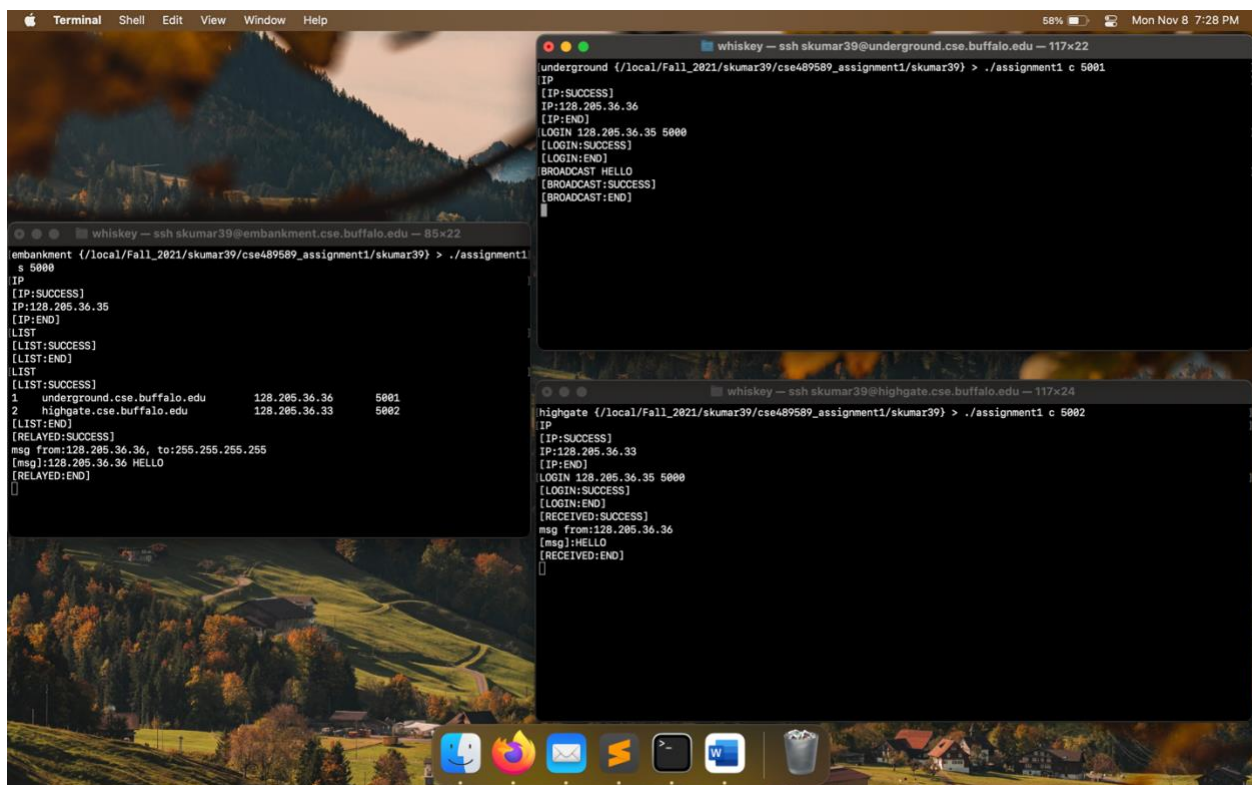
Initially, both the clients are connected to the server because of which we see two entries when the LIST command is executed. The client on the top right then logs out after which the LIST command results in one entry only. The client on the bottom right also logs out, resulting in no entries when the LIST command is executed by the server on the left.

[2.5] EXIT

```
highgate {/local/Fall_2021/skumar39/cse489589_assignment1/skumar39} > ./assignment1 c 5001
[IP
[IP:SUCCESS]
IP:128.205.36.33
[IP:END]
[EXIT
[EXIT:SUCCESS]
[EXIT:END]
highgate {/local/Fall_2021/skumar39/cse489589_assignment1/skumar39} > █
```

This command closes the connection and unlike the LOGOUT command, destroys all states associated with the client. Destroying the states implies removing all data associated with that client from the connection structure.

[EVENT]: Message Received



Message relayed from one client (top right) to the other (bottom right) by the server (left).