NAME: Faraz Gurramkonda
UMICH ID:45489809

NAME: katerina Matraku
UMICH ID:74424324

INSTRUCTOR: PROF.GUO
COURSE : CIS 527

GROUP-4

READ ME FILE:

STEP1: HOW TO RUN THE PROGRAM

1. connect to the Umich VPN
2. connect to the umich server
3. Copy all the files from the tar/zip file and extract it on your local.
4. Create a folder and extract it in the folder.
5. Now copy everything in the folder. Basically copy the folder from your local to the remote umich sever.
6. For copying it ,you should before login into your umcih server.
7. If you are using linux or mac the scp them to the umcih server under destinated path. I did it under the Public Folder

STEP 2 : RUN THE SERVER

1. Run the server.py file using the command “python server.py”
2. Initially we saved all the messages in the Message_of_day file and we are reading everything from a the file from the server and after reading it we are closing it.
3. you don’t have to download any version of python in the umich server. Umich server has Python 2.7.5. so, We modified code according to the version which is on the server. we developed it in initially in the latest version but we modified it according the python 2.7.5. so, do not download anything to run. Just run the command which i mentioned earlier.

STEP 3: RUN THE CLIENT

1. Run the c1.py file using the command “python c1.py”
2. As per instruction use the loop address for the host address which the client ask you to insert. which is this 127.0.0.1
3. And choose the commands options that it shows and you can go on.

FUNCTIONS IMPLEMENTED:

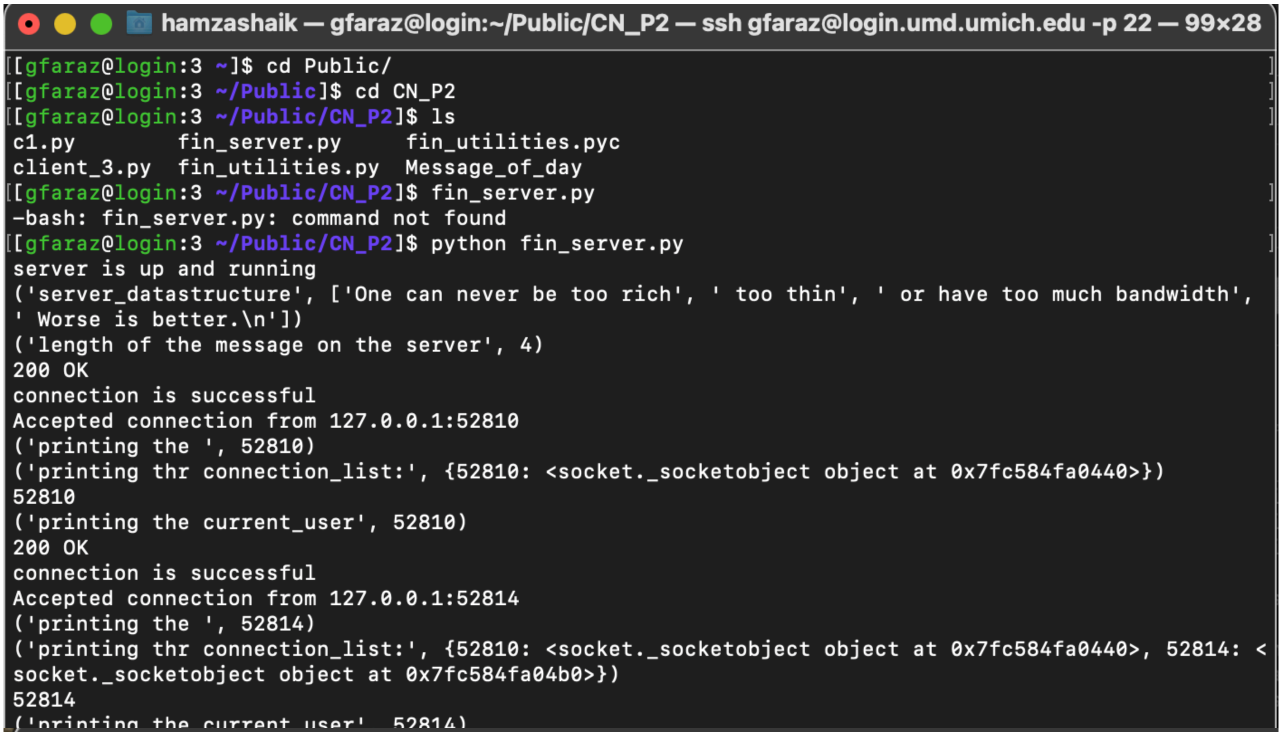
msgget: This function allows the message of the day from the sever to client.
msgstore: This function allows the client user to send the message of the day where the server will store the message in to the server data structure. if the user not logged in, Then the function return the error
shutdown: This function will allow only the root user to shutdown both server and the clients(all the clients in the multiple client server) from the sever side. If the user is not the root user then eventually you will get the error. If it is then it will broadcast the message “210 the server is about to shutdown ….” and shutdown all the clients connection which are active.
login: After choosing a login option,The server will ask the user for username and password if it does not match with users list on the server. Then an error will be generated . If everything is good then it shows 200 OK.
logout: This function will allow the user to logout only when the user is logged in or else an error will be generated.
quit: This function will quit both the client and server.
who : When use this command this command will display the current active clients which are connected to the server.
Send: When you use this command from the client side then the server ask you to which active client that client needs to send the private message and forwards the private message to that client. After getting the private message from that sender client. if there no active users logged in with the name of the active clients that inputed. Then an error will be generated”420 either the user does not exist or is not logged in”

converting_str_to_asc: This function takes an input of string and converts it into the ascii value. which helps to send the code from client to server.

Fin_utilities.py: I created a separate fin_utilities package to implement the functions.

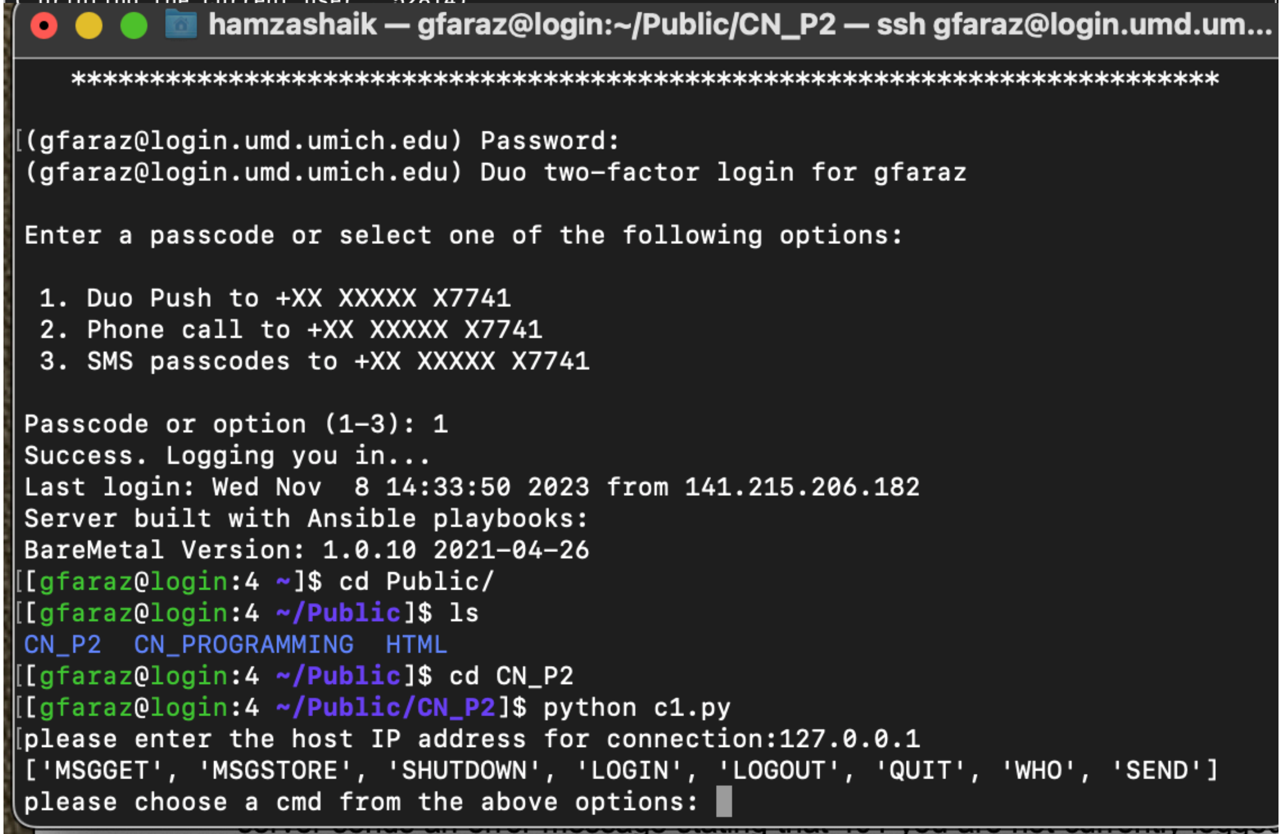
Sample output :

FIG (1)
This is the sever side output when send the commands from the client to server.



hamzashaik — gfaraz@login:~/Public/CN_P2 — ssh gfaraz@login.umd.umich.edu -p 22 — 99×28

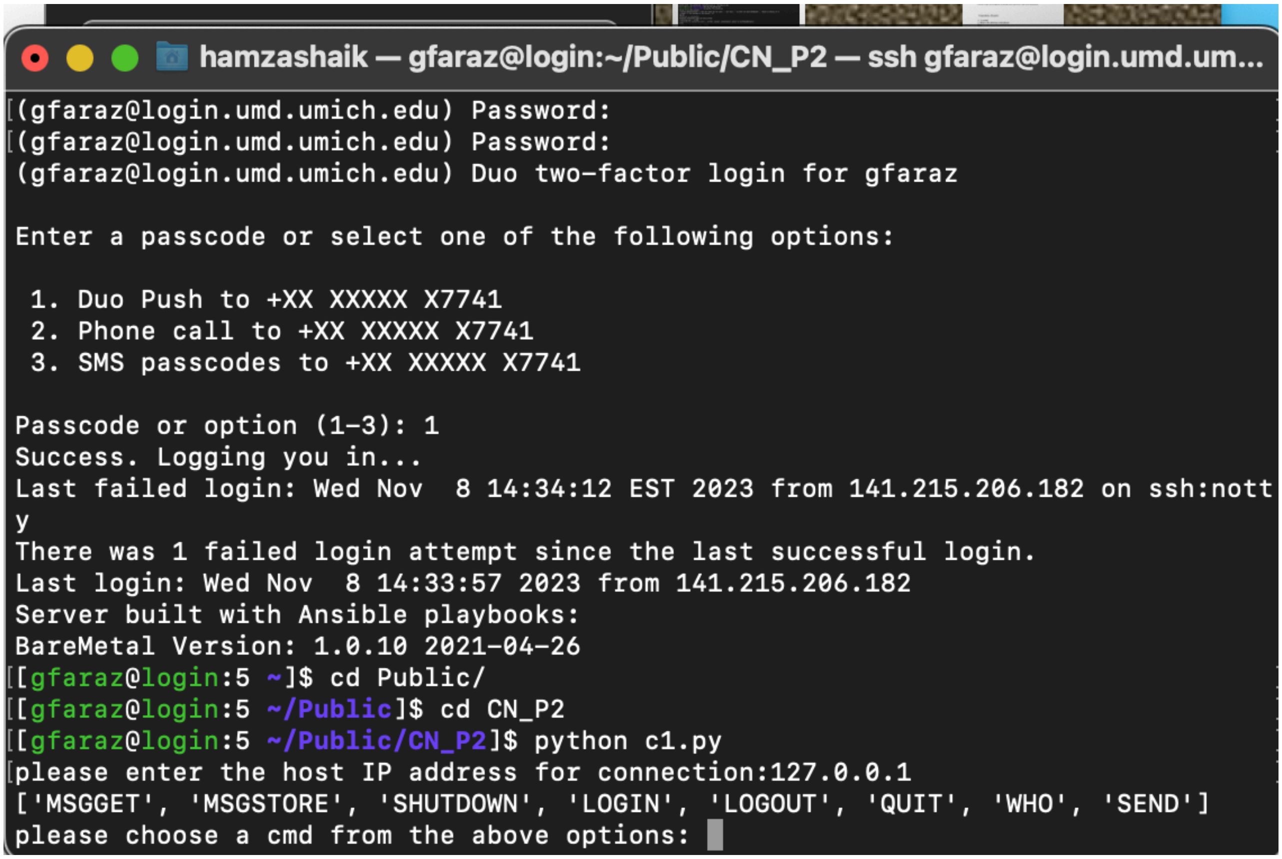
```
[[gfaraz@login:3 ~]$ cd Public/
[[gfaraz@login:3 ~/Public]$ cd CN_P2
[[gfaraz@login:3 ~/Public/CN_P2]$ ls
c1.py        fin_server.py     fin_utilities.pyc
client_3.py  fin_utilities.py  Message_of_day
[[gfaraz@login:3 ~/Public/CN_P2]$ fin_server.py
-bash: fin_server.py: command not found
[[gfaraz@login:3 ~/Public/CN_P2]$ python fin_server.py
server is up and running
('server_datastructure', ['One can never be too rich', ' too thin', ' or have too much bandwidth',
' Worse is better.\n'])
('length of the message on the server', 4)
200 OK
connection is successful
Accepted connection from 127.0.0.1:52810
('printing the ', 52810)
('printing thr connection_list:', {52810: <socket._socketobject object at 0x7fc584fa0440>})
52810
('printing the current_user', 52810)
200 OK
connection is successful
Accepted connection from 127.0.0.1:52814
('printing the ', 52814)
('printing thr connection_list:', {52810: <socket._socketobject object at 0x7fc584fa0440>, 52814: <
socket._socketobject object at 0x7fc584fa04b0>})
52814
('printing the current_user', 52814)
```



hamzashaik — gfaraz@login:~/Public/CN_P2 — ssh gfaraz@login.umd.um...

```
  **************************************************************************

[(gfaraz@login.umd.umich.edu) Password:
(gfaraz@login.umd.umich.edu) Duo two-factor login for gfaraz

Enter a passcode or select one of the following options:

 1. Duo Push to +XX XXXXX X7741
 2. Phone call to +XX XXXXX X7741
 3. SMS passcodes to +XX XXXXX X7741

Passcode or option (1-3): 1
Success. Logging you in...
Last login: Wed Nov  8 14:33:50 2023 from 141.215.206.182
Server built with Ansible playbooks:
BareMetal Version: 1.0.10 2021-04-26
[[gfaraz@login:4 ~]$ cd Public/
[[gfaraz@login:4 ~/Public]$ ls
CN_P2  CN_PROGRAMMING  HTML
[[gfaraz@login:4 ~/Public]$ cd CN_P2
[[gfaraz@login:4 ~/Public/CN_P2]$ python c1.py
[please enter the host IP address for connection:127.0.0.1
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options:
```

server sends an error message stating that 401 you are not currently logged
You need to login first to send the messages to store them.

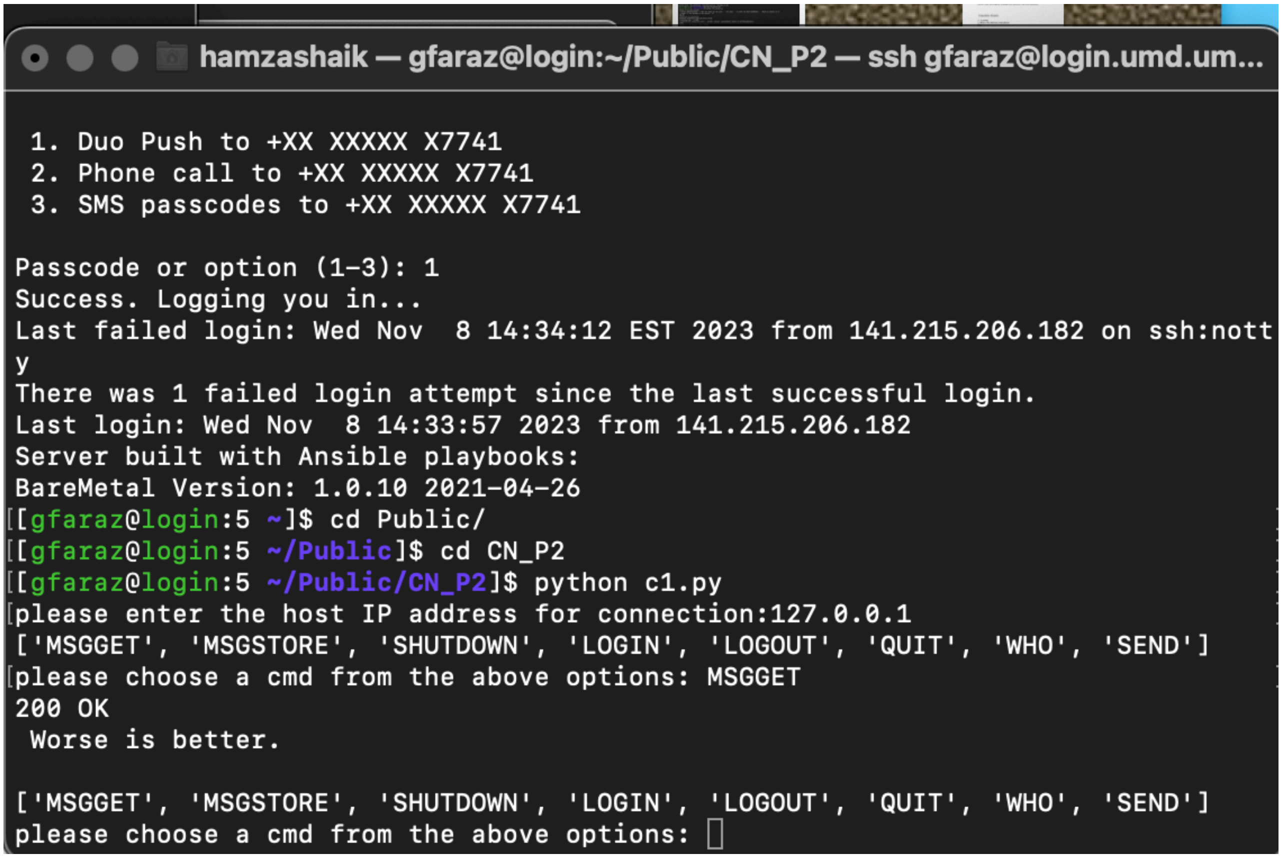

hamzashaik — gfaraz@login:~/Public/CN_P2 — ssh gfaraz@login.umd.um...

```
[(gfaraz@login.umd.umich.edu) Password:
[(gfaraz@login.umd.umich.edu) Password:
 (gfaraz@login.umd.umich.edu) Duo two-factor login for gfaraz

 Enter a passcode or select one of the following options:

  1. Duo Push to +XX XXXXX X7741
  2. Phone call to +XX XXXXX X7741
  3. SMS passcodes to +XX XXXXX X7741

 Passcode or option (1-3): 1
 Success. Logging you in...
 Last failed login: Wed Nov  8 14:34:12 EST 2023 from 141.215.206.182 on ssh:nott
 y
 There was 1 failed login attempt since the last successful login.
 Last login: Wed Nov  8 14:33:57 2023 from 141.215.206.182
 Server built with Ansible playbooks:
 BareMetal Version: 1.0.10 2021-04-26
[[gfaraz@login:5 ~]$ cd Public/
[[gfaraz@login:5 ~/Public]$ cd CN_P2
[[gfaraz@login:5 ~/Public/CN_P2]$ python c1.py
[please enter the host IP address for connection:127.0.0.1
 ['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
 please choose a cmd from the above options:
```

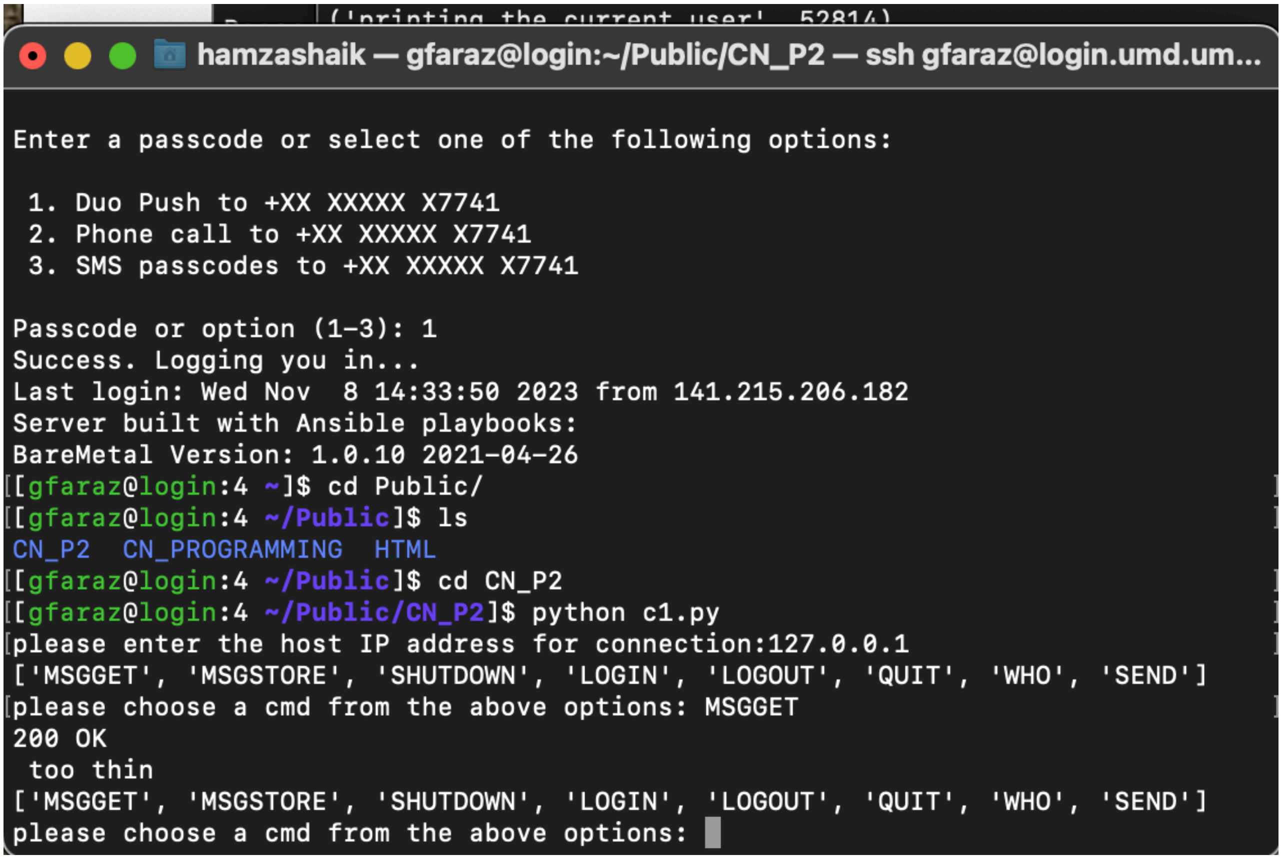
we have multiple clients here.

Above show in the FIG(2):
Now first you will start the sever and then you will start the client program. For starting the client connection you need to give the loop back IP address as i mentioned earlier.

After That when you choose the MSGGET command. if the connection is good and everything is ready then the server sends 200OK message. After the 200OK message the server will then send the message of the day message.


hamzashaik — gfaraz@login:~/Public/CN_P2 — ssh gfaraz@login.umd.um...

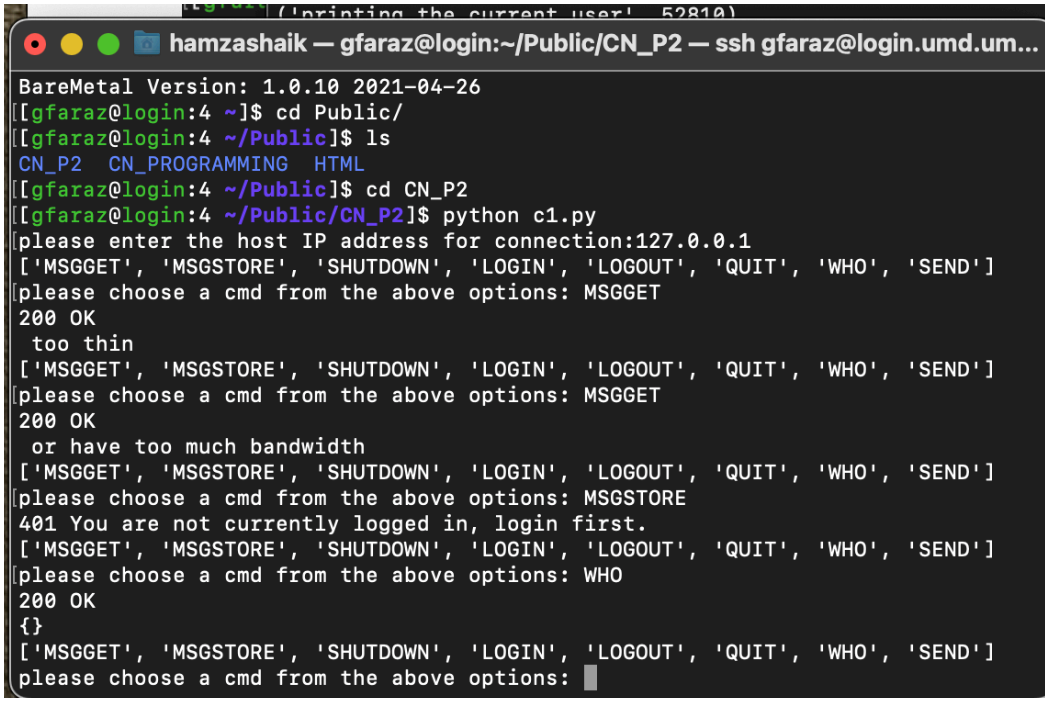
```
 1. Duo Push to +XX XXXXX X7741
 2. Phone call to +XX XXXXX X7741
 3. SMS passcodes to +XX XXXXX X7741

Passcode or option (1-3): 1
Success. Logging you in...
Last failed login: Wed Nov  8 14:34:12 EST 2023 from 141.215.206.182 on ssh:nott
y
There was 1 failed login attempt since the last successful login.
Last login: Wed Nov  8 14:33:57 2023 from 141.215.206.182
Server built with Ansible playbooks:
BareMetal Version: 1.0.10 2021-04-26
[[gfaraz@login:5 ~]$ cd Public/
[[gfaraz@login:5 ~/Public]$ cd CN_P2
[[gfaraz@login:5 ~/Public/CN_P2]$ python c1.py
[please enter the host IP address for connection:127.0.0.1
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: MSGGET
200 OK
 Worse is better.

['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: 
```


('printing the current user', 52814)
hamzashaik — gfaraz@login:~/Public/CN_P2 — ssh gfaraz@login.umd.um...

```
Enter a passcode or select one of the following options:

 1. Duo Push to +XX XXXXX X7741
 2. Phone call to +XX XXXXX X7741
 3. SMS passcodes to +XX XXXXX X7741

Passcode or option (1-3): 1
Success. Logging you in...
Last login: Wed Nov  8 14:33:50 2023 from 141.215.206.182
Server built with Ansible playbooks:
BareMetal Version: 1.0.10 2021-04-26
[[gfaraz@login:4 ~]$ cd Public/
[[gfaraz@login:4 ~/Public]$ ls
CN_P2  CN_PROGRAMMING  HTML
[[gfaraz@login:4 ~/Public]$ cd CN_P2
[[gfaraz@login:4 ~/Public/CN_P2]$ python c1.py
[please enter the host IP address for connection:127.0.0.1
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: MSGGET
200 OK
 too thin
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: 
```

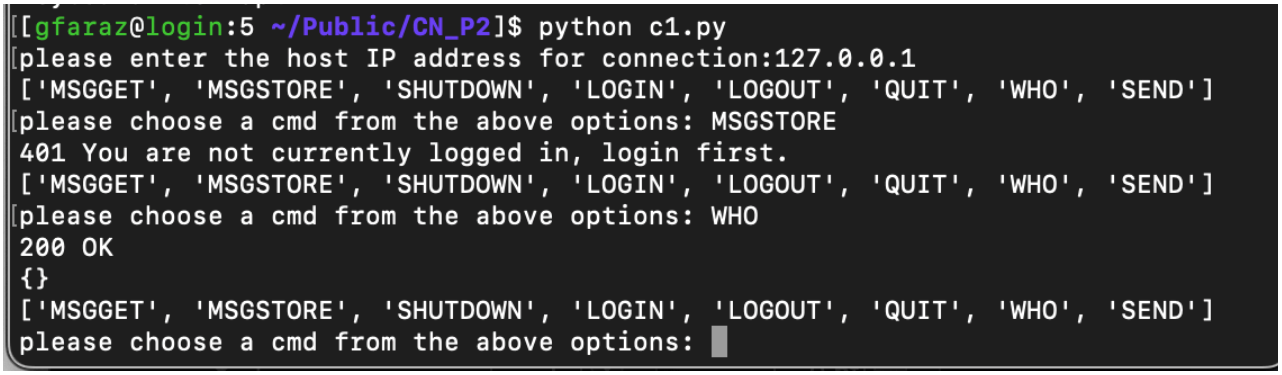
The two client connection are shown above

If you choose the MSGSTORE command and if your not logged into the system then the server sends an error message stating that 401 you are not currently logged in, login first.so, You need to login first to send the messages to store them.


hamzashaik — gfaraz@login:~/Public/CN_P2 — ssh gfaraz@login.umd.um...

```
BareMetal Version: 1.0.10 2021-04-26
[gfaraz@login:4 ~]$ cd Public/
[gfaraz@login:4 ~/Public]$ ls
CN_P2  CN_PROGRAMMING  HTML
[gfaraz@login:4 ~/Public]$ cd CN_P2
[gfaraz@login:4 ~/Public/CN_P2]$ python c1.py
please enter the host IP address for connection:127.0.0.1
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: MSGGET
200 OK
 too thin
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: MSGGET
200 OK
 or have too much bandwidth
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: MSGSTORE
401 You are not currently logged in, login first.
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: WHO
200 OK
{}
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: 
```

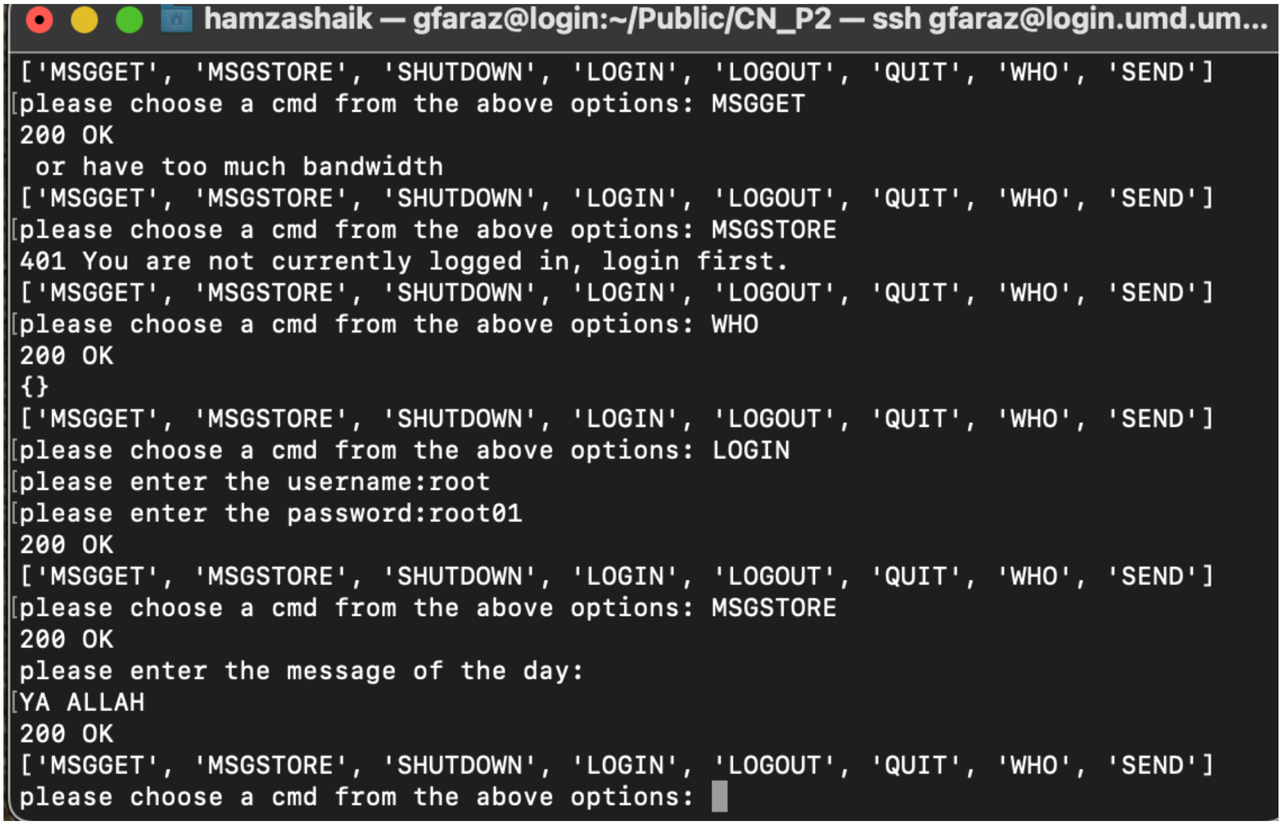
The above image shows client 1 terminal



```
[gfaraz@login:5 ~/Public/CN_P2]$ python c1.py
please enter the host IP address for connection:127.0.0.1
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: MSGSTORE
401 You are not currently logged in, login first.
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: WHO
200 OK
{}
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: 
```

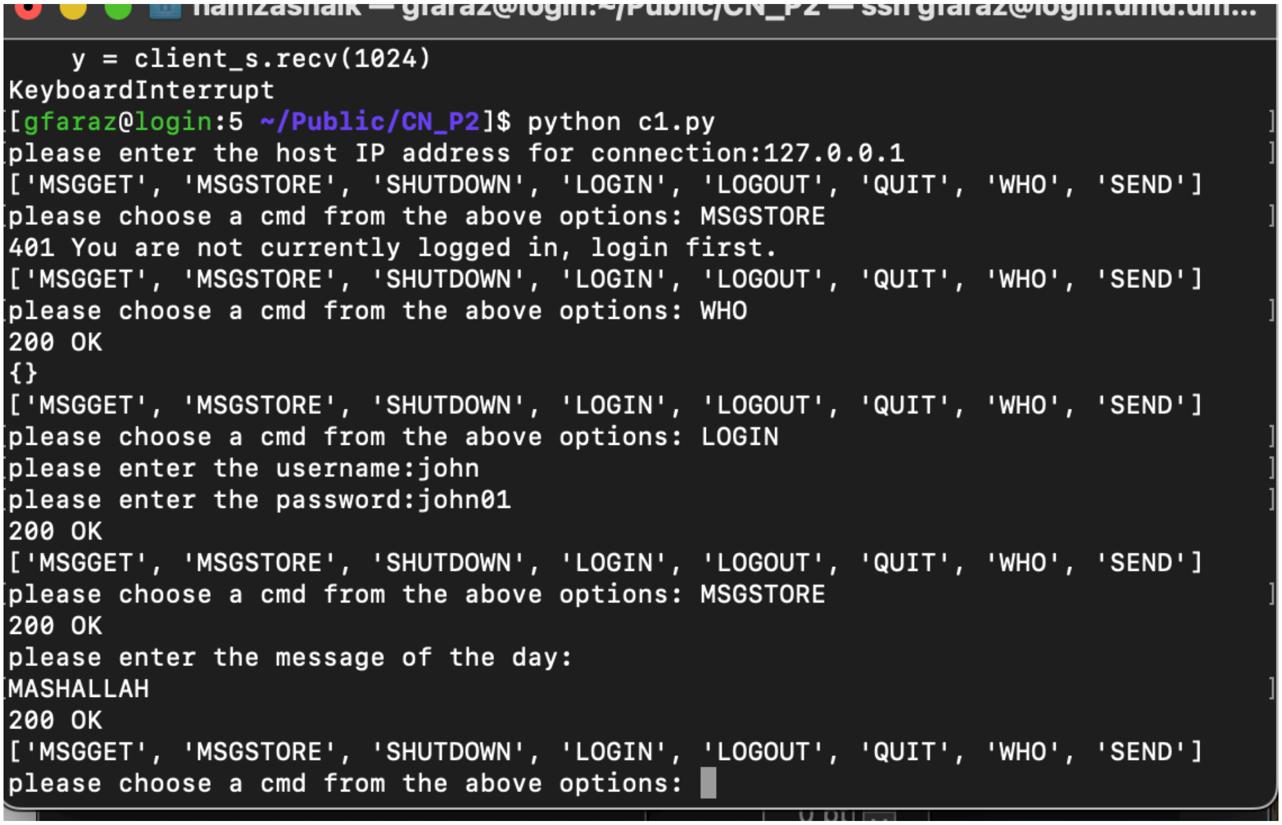
The above image shows client 2 terminal

if logged in and sends the request to send the messages then the output would like this in the below picture.



hamzashaik — gfaraz@login:~/Public/CN_P2 — ssh gfaraz@login.umd.um...

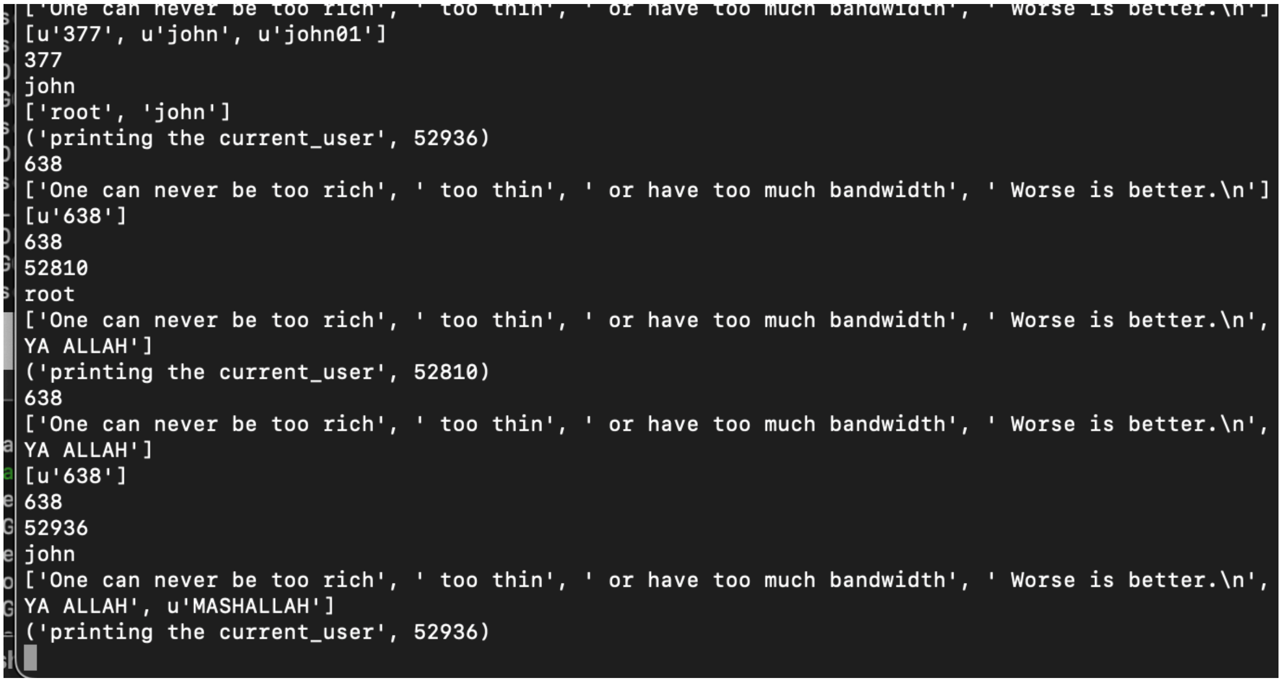
```
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: MSGGET
200 OK
 or have too much bandwidth
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: MSGSTORE
401 You are not currently logged in, login first.
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: WHO
200 OK
{}
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: LOGIN
please enter the username:root
please enter the password:root01
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: MSGSTORE
200 OK
please enter the message of the day:
YA ALLAH
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: 
```



hamzashaik — gfaraz@login:~/Public/CN_P2 — ssh gfaraz@login.umd.um...

```
    y = client_s.recv(1024)
KeyboardInterrupt
[gfaraz@login:5 ~/Public/CN_P2]$ python c1.py
please enter the host IP address for connection:127.0.0.1
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: MSGSTORE
401 You are not currently logged in, login first.
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: WHO
200 OK
{}
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: LOGIN
please enter the username:john
please enter the password:john01
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: MSGSTORE
200 OK
please enter the message of the day:
MASHALLAH
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: 
```

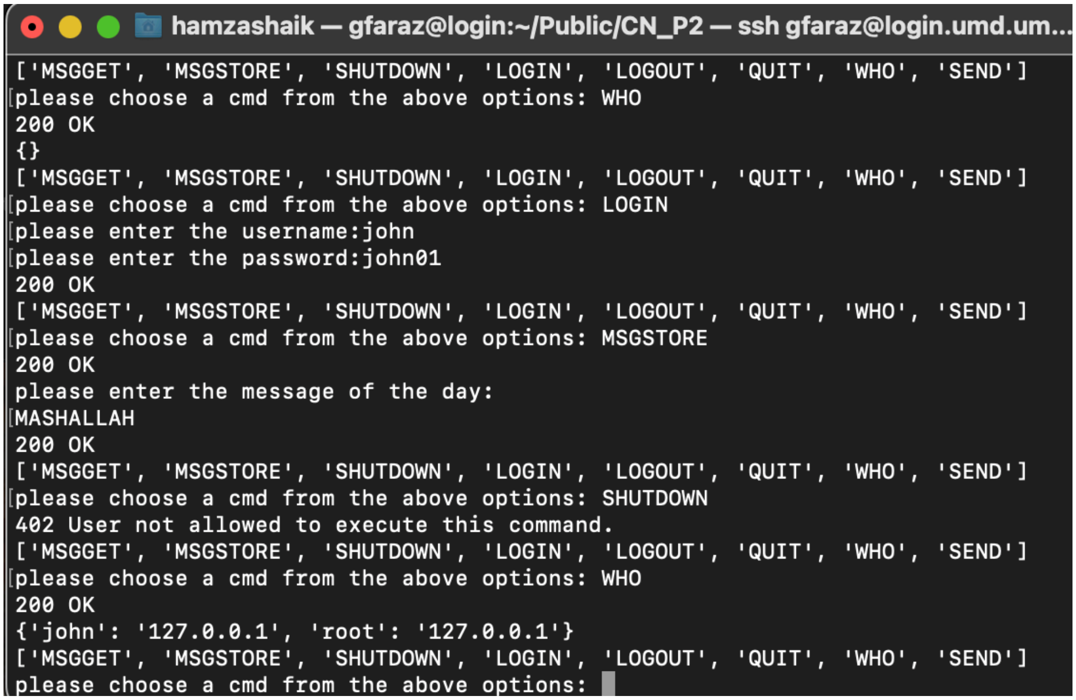
The above two terminals are the client 1 and client 2 terminals



```
['One can never be too rich', ' too thin', ' or have too much bandwidth', ' Worse is better.\n']
[u'377', u'john', u'john01']
377
john
['root', 'john']
('printing the current_user', 52936)
638
['One can never be too rich', ' too thin', ' or have too much bandwidth', ' Worse is better.\n']
[u'638']
638
52810
root
['One can never be too rich', ' too thin', ' or have too much bandwidth', ' Worse is better.\n',
YA ALLAH']
('printing the current_user', 52810)
638
['One can never be too rich', ' too thin', ' or have too much bandwidth', ' Worse is better.\n',
YA ALLAH']
[u'638']
638
52936
john
['One can never be too rich', ' too thin', ' or have too much bandwidth', ' Worse is better.\n',
YA ALLAH', u'MASHALLAH']
('printing the current_user', 52936)
```

This is the sever window. In the bottom of the image you can see a list of messages that tired to store.

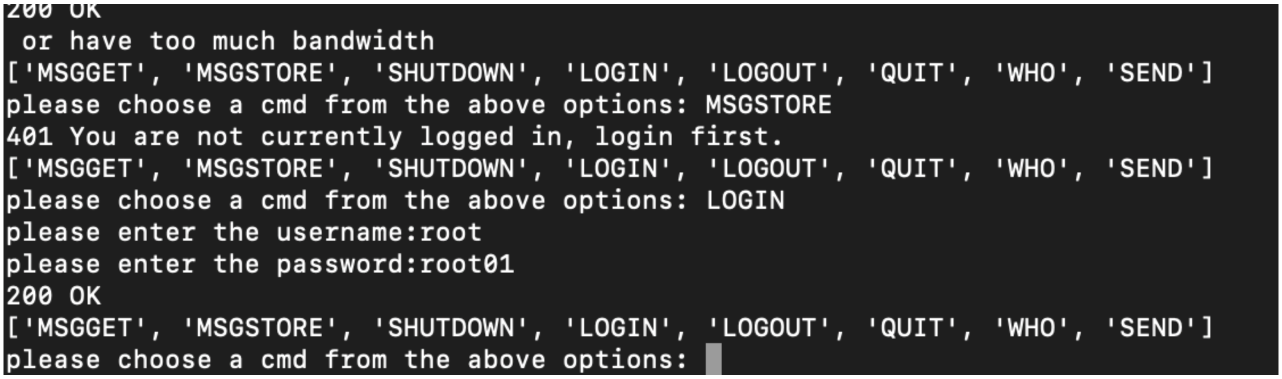
If you choose SHUTDOWN option, Then the server checks whether your are a root user if not it will generate an error 402 user not allowed to execute the command . if you are root user then it will shutdown all the client by sending message that the “210 the server is about to shutdown”.


hamzashaik — gfaraz@login:~/Public/CN_P2 — ssh gfaraz@login.umd.um...

```
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: WHO
200 OK
{}
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: LOGIN
please enter the username:john
please enter the password:john01
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: MSGSTORE
200 OK
please enter the message of the day:
MASHALLAH
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: SHUTDOWN
402 User not allowed to execute this command.
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: WHO
200 OK
{'john': '127.0.0.1', 'root': '127.0.0.1'}
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options:
```

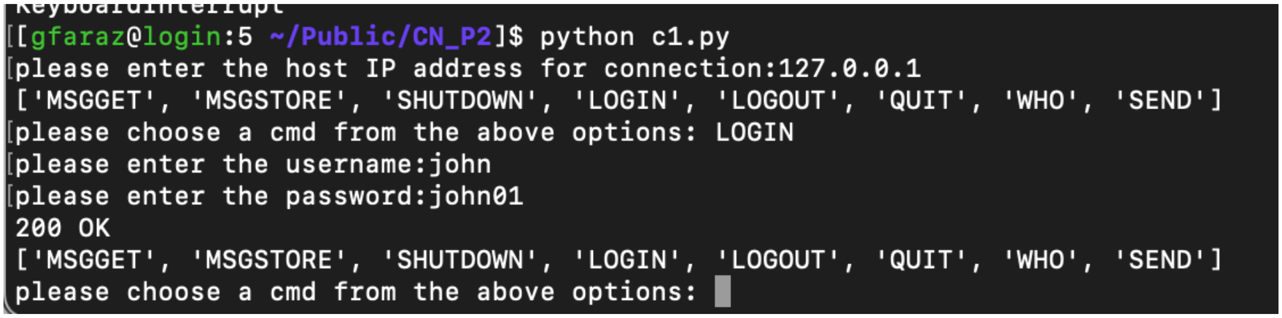
Here, in the below image:
I gave wrong credentials to it. so it generated an error 410 wrong UserID or Password .
I am making this for better user experiences. But when you check on the server side the command goes into it as the single line as it is mentioned by instructor Guo. I made easy for user to understand what to do. i chose this type. But the functionality of this is same as mentioned in the instructions.

If the userID and password is correct then it will respond with 200 OK message.



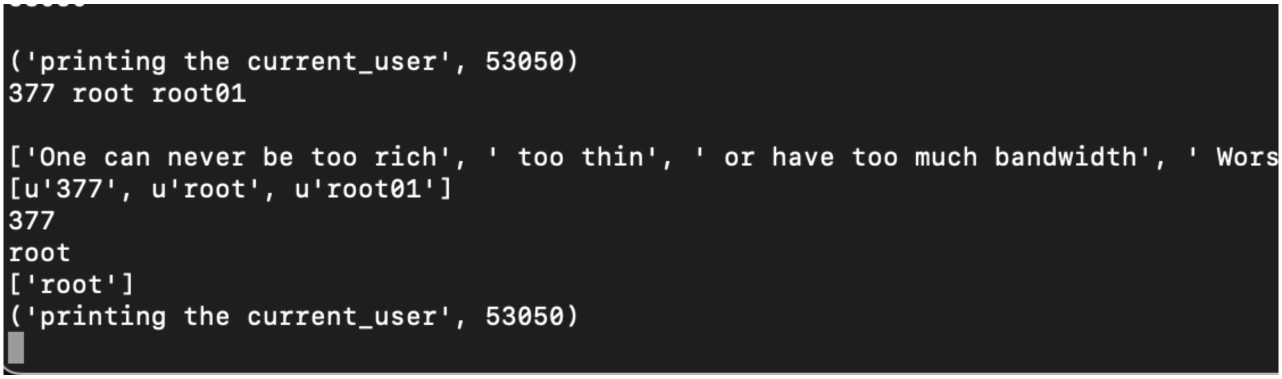
```
200 OK
 or have too much bandwidth
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: MSGSTORE
401 You are not currently logged in, login first.
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: LOGIN
please enter the username:root
please enter the password:root01
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options:
```

This is the root user



```
 KeyboardInterrupt
[[gfaraz@login:5 ~/Public/CN_P2]$ python c1.py
[please enter the host IP address for connection:127.0.0.1
 ['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: LOGIN
[please enter the username:john
[please enter the password:john01
 200 OK
 ['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
 please choose a cmd from the above options:
```

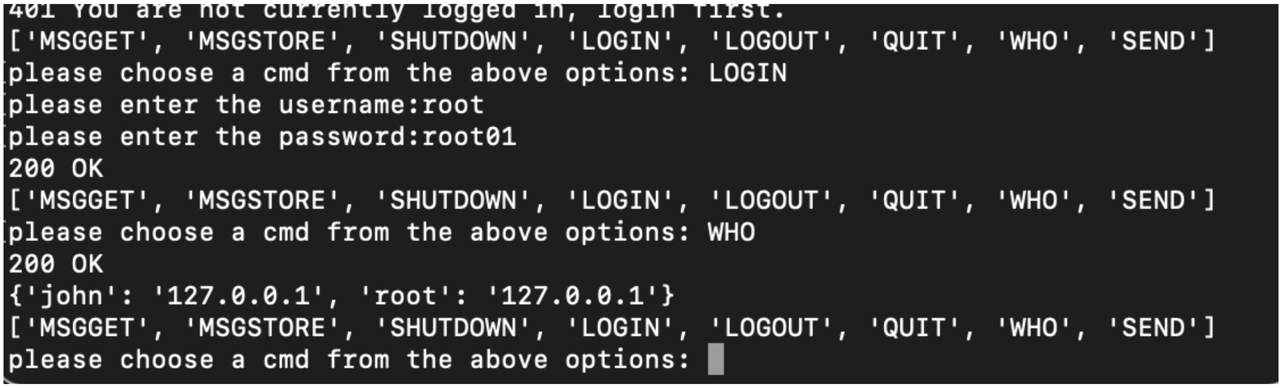
This is the user John



```

('printing the current_user', 53050)
377 root root01

['One can never be too rich', ' too thin', ' or have too much bandwidth', ' Wors
[u'377', u'root', u'root01']
377
root
['root']
('printing the current_user', 53050)
```

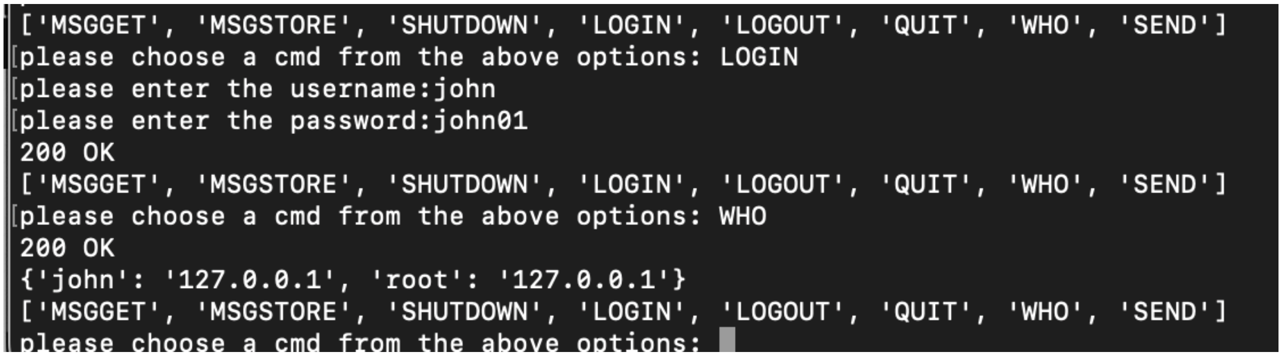
Now here in the above LOGIN command Images . we are getting the same login ascii code and userid and password in the same command . Everything is same including functionality . we made some change in users input for better user experience.

WHO:



```
401 You are not currently logged in, login first.
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: LOGIN
please enter the username:root
please enter the password:root01
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: WHO
200 OK
{'john': '127.0.0.1', 'root': '127.0.0.1'}
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options:
```

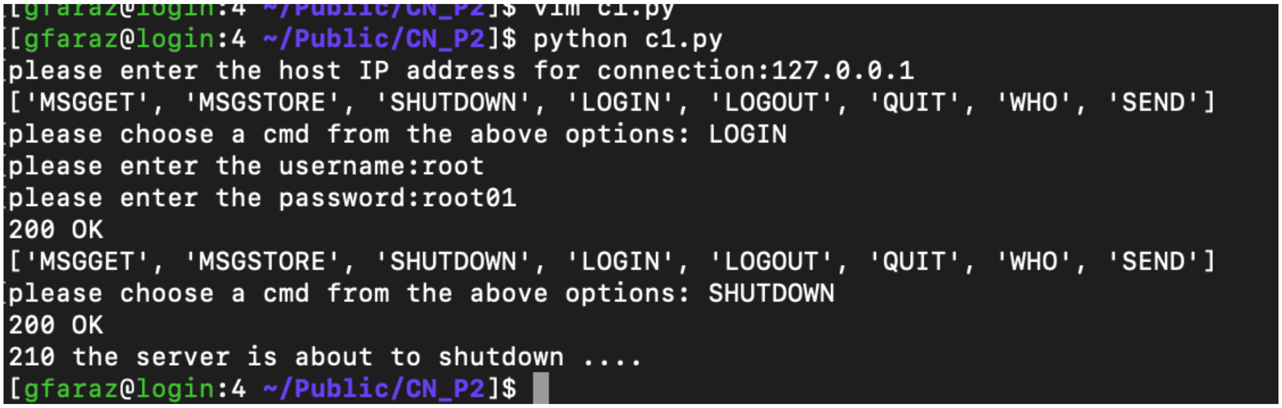
This the output when we use the command “WHO ” to check as the active users in the root users screen.



```
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: LOGIN
please enter the username:john
please enter the password:john01
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: WHO
200 OK
{'john': '127.0.0.1', 'root': '127.0.0.1'}
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options:
```

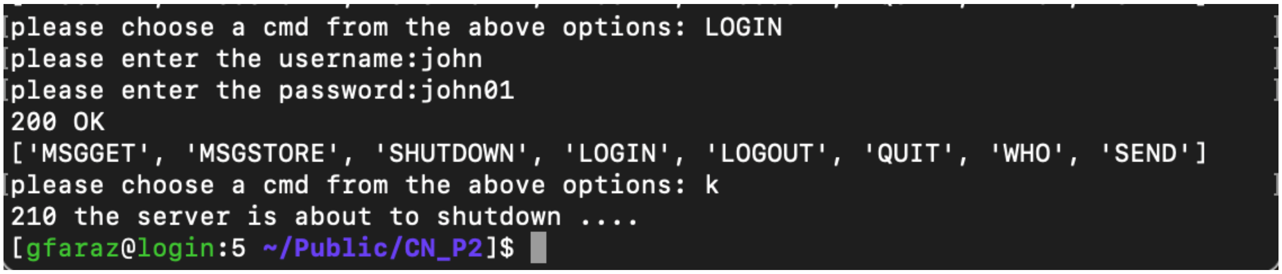
This is the output when we use the command “WHO” to check as the active users in the john users screen.

SHUTDOWN : As you can see if the current user is a root user and logged in then the shutdown will occur and you will be out from the program



```
[gfaraz@login:4 ~/Public/CN_P2]$ vim c1.py
[gfaraz@login:4 ~/Public/CN_P2]$ python c1.py
please enter the host IP address for connection:127.0.0.1
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: LOGIN
please enter the username:root
please enter the password:root01
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: SHUTDOWN
200 OK
210 the server is about to shutdown ....
[gfaraz@login:4 ~/Public/CN_P2]$
```

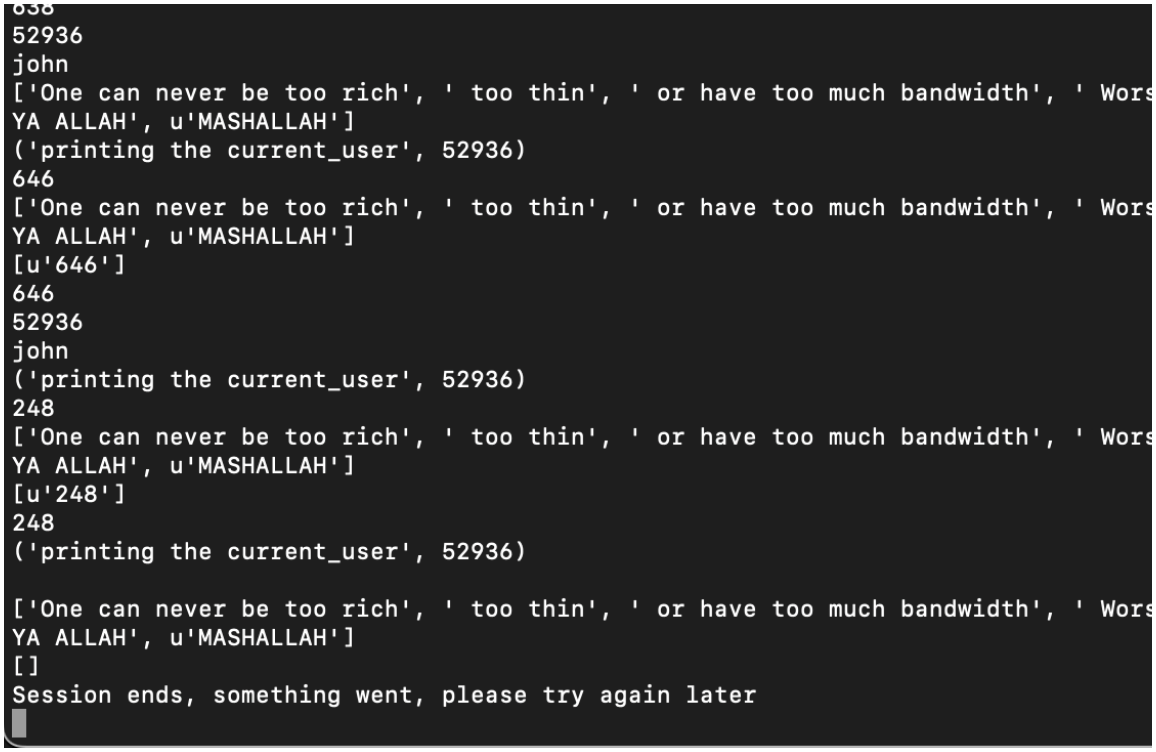
This is client 1. The root user made the server to SHUTDOWN.



```
please choose a cmd from the above options: LOGIN
please enter the username:john
please enter the password:john01
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: k
210 the server is about to shutdown ....
[gfaraz@login:5 ~/Public/CN_P2]$
```

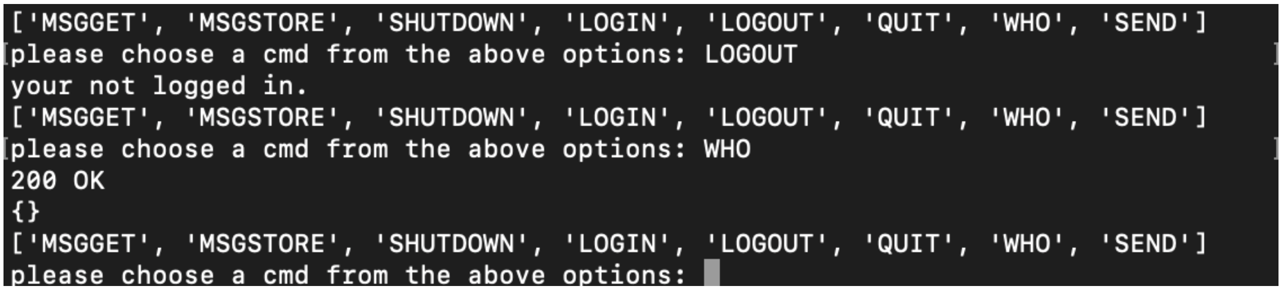
This is client 2. In here Client was trying to the give some command in the above options. But as the server getting shutdown the server broadcasted the message its getting down. And the if you use any command. After that , then you will not get the reply from past command due to server the shutdown . But as i mentioned only root can do it.

If you did any kind of keyboard interruptions during the program execution. You will get this error message session ends, something went, please try later.



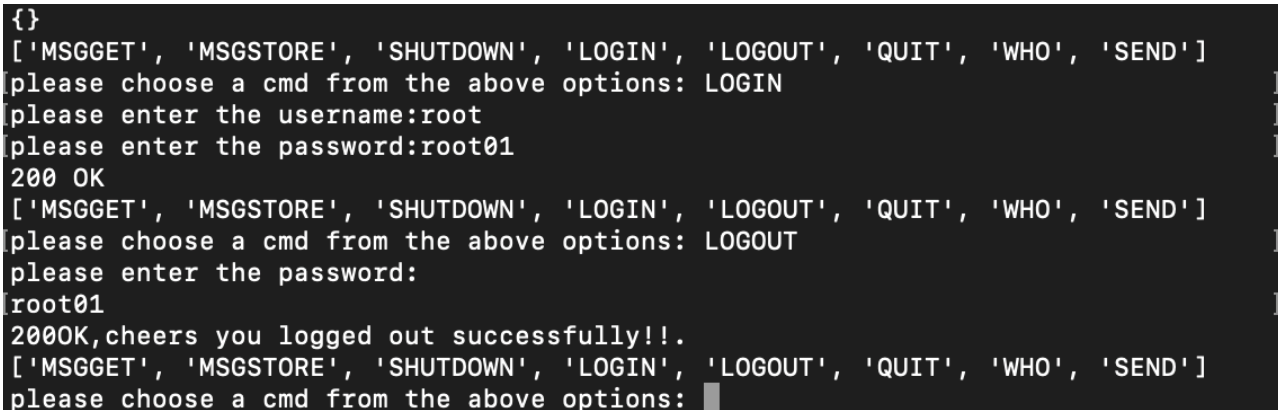
```
638
52936
john
['One can never be too rich', ' too thin', ' or have too much bandwidth', ' Wors
YA ALLAH', u'MASHALLAH']
('printing the current_user', 52936)
646
['One can never be too rich', ' too thin', ' or have too much bandwidth', ' Wors
YA ALLAH', u'MASHALLAH']
[u'646']
646
52936
john
('printing the current_user', 52936)
248
['One can never be too rich', ' too thin', ' or have too much bandwidth', ' Wors
YA ALLAH', u'MASHALLAH']
[u'248']
248
('printing the current_user', 52936)

['One can never be too rich', ' too thin', ' or have too much bandwidth', ' Wors
YA ALLAH', u'MASHALLAH']
[]
Session ends, something went, please try again later
```

LOGOUT:



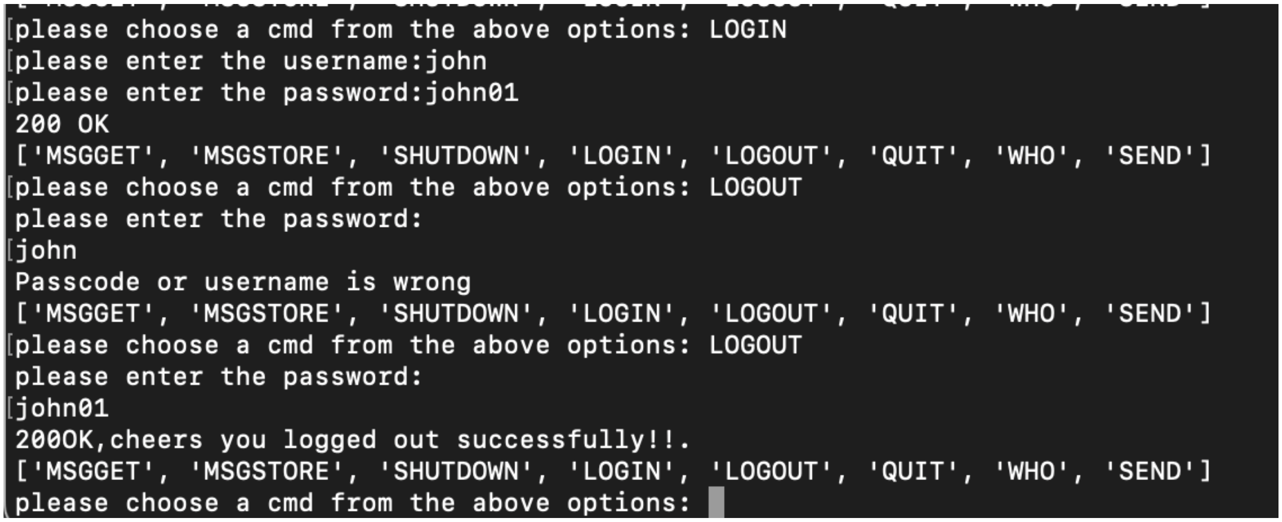
```
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: LOGOUT
your not logged in.
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: WHO
200 OK
{}
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options:
```

In the above image i did not logged any user from the terminal. so it showed an error that i am not logged in.



```
{}
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: LOGIN
please enter the username:root
please enter the password:root01
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: LOGOUT
please enter the password:
root01
200OK,cheers you logged out successfully!!.
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: 
```

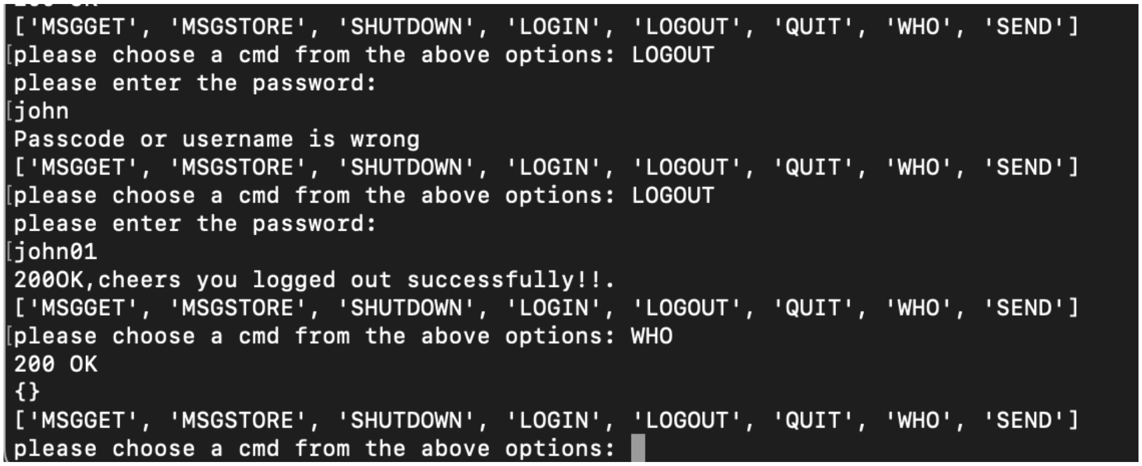
This the root users screen. where the root user logged in and logged out from the server. I made it more secured in such way that if you want to logout just provide the password and then the server logs the user out.



```
please choose a cmd from the above options: LOGIN
please enter the username:john
please enter the password:john01
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: LOGOUT
please enter the password:
john
Passcode or username is wrong
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: LOGOUT
please enter the password:
john01
200OK,cheers you logged out successfully!!.
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: 
```

Here, in the above screen in the john’s window. The users wants to logout but he provided a wrong password. so, the users couldn’t able to logout. But after providing the users password the user get logged out.

Now , lets check after with “WHO” command. what is the results. we logged in two users and logged out two users.so there should be none clients who are actives.

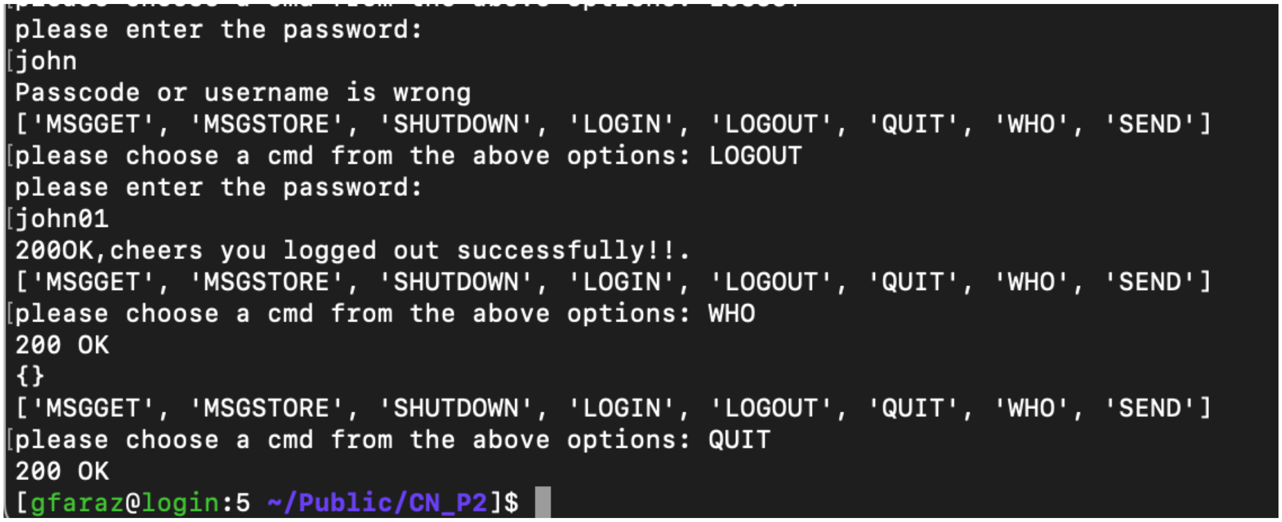


```
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: LOGOUT
please enter the password:
[john
Passcode or username is wrong
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: LOGOUT
please enter the password:
[john01
200OK,cheers you logged out successfully!!.
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: WHO
200 OK
{}
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options:
```

So if you check in the above image. Their is no client is in active as expected.

QUIT:

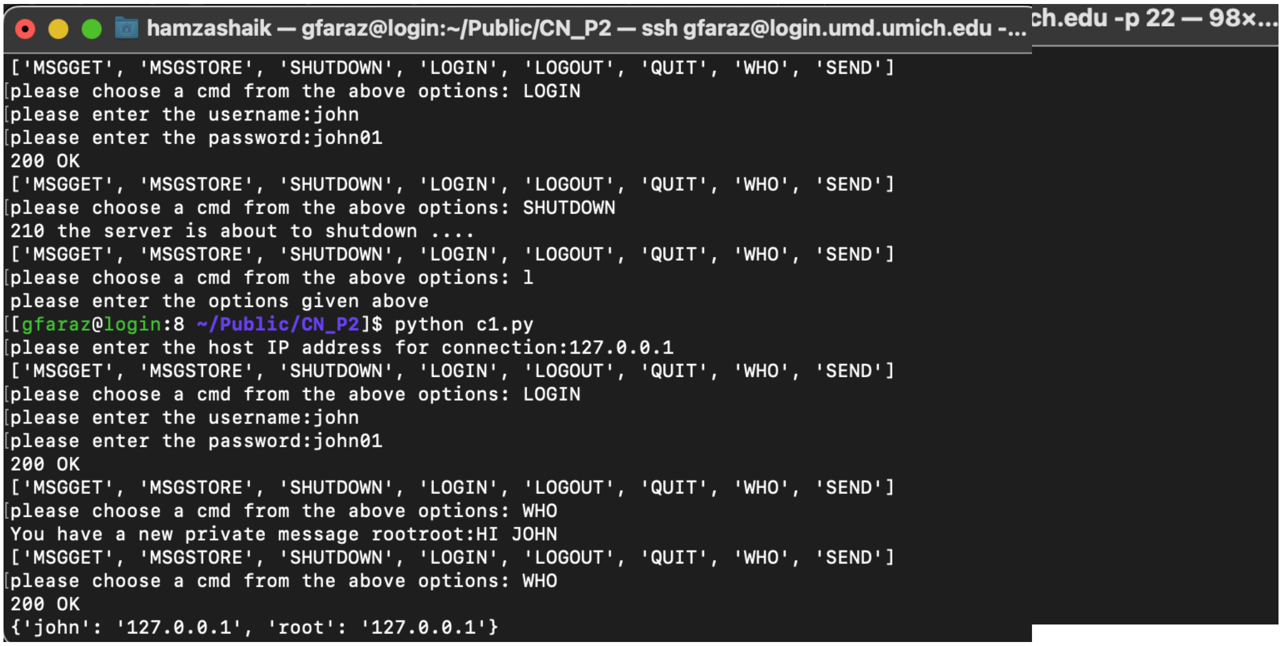
After choosing the QUIT option in the command option. You will be out from program and the 200OK message will be displayed on the client screen.



```
please enter the password:
[john
Passcode or username is wrong
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: LOGOUT
please enter the password:
[john01
200OK,cheers you logged out successfully!!.
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: WHO
200 OK
{}
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: QUIT
200 OK
[gfaraz@login:5 ~/Public/CN_P2]$
```

SEND:

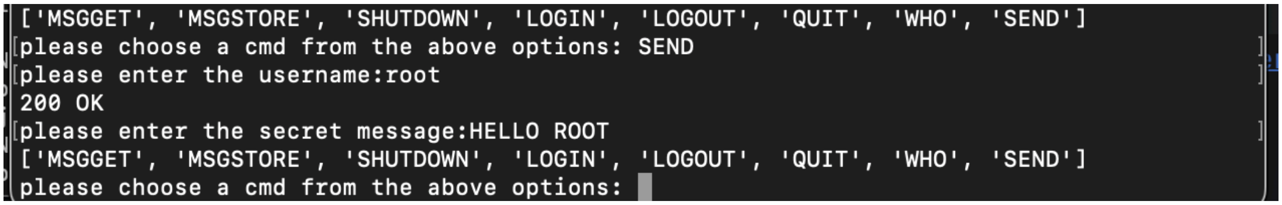
If you try to send to a user who is not logged in. you will get an error like above show in the image “420 either the user does not exists or is not logged in”. If user logged in then the server sends the secret message to the users that the user intended to. As in the above picture.


hamzashaik — gfaraz@login:~/Public/CN_P2 — ssh gfaraz@login.umd.umich.edu -...
ch.edu -p 22 — 98×...

```
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: LOGIN
[please enter the username:john
[please enter the password:john01
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: SHUTDOWN
210 the server is about to shutdown ....
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: l
please enter the options given above
[[gfaraz@login:8 ~/Public/CN_P2]$ python c1.py
[please enter the host IP address for connection:127.0.0.1
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: LOGIN
[please enter the username:john
[please enter the password:john01
200 OK
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: WHO
You have a new private message rootroot:HI JOHN
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: WHO
200 OK
{'john': '127.0.0.1', 'root': '127.0.0.1'}
```

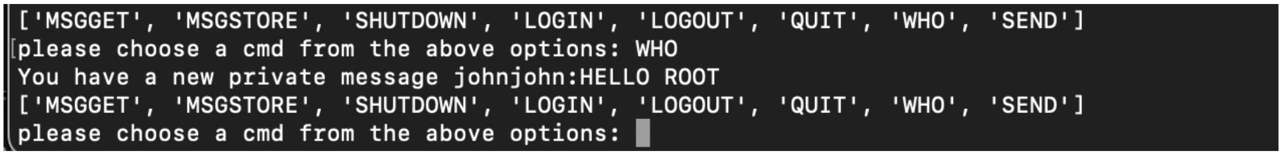
Now, on the john’s screen the private message is displayed.

Now let us try to do it in reverse order. sending message from the john to root.



```
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: SEND
[please enter the username:root
200 OK
[please enter the secret message:HELLO ROOT
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: 
```

Now john sent a private message to the user root.



```
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
[please choose a cmd from the above options: WHO
You have a new private message johnjohn:HELLO ROOT
['MSGGET', 'MSGSTORE', 'SHUTDOWN', 'LOGIN', 'LOGOUT', 'QUIT', 'WHO', 'SEND']
please choose a cmd from the above options: 
```

Now here the root got the message from the John that what he sent. When users was executing the WHO command.

KNOWN BUGS:

1. I don’t think there are any known Bugs in the code.

CONTRIBUTION OF THE WORK

My name is Faraz Gurramkonda and my part is in developing and testing of this work, I developed both server.py and utilities.py.

My name is katerina Matraku and my part of development is on the client.py and testing of this Work.

NOTE: WE ARE NOT SUBMITTING THE MAKEFILE AS IT IS NOT GOOD PRACTICE FOR PYTHON IMPLEMENTATION OF THE PROJECT AS PROFESSOR INSTRUCTED.

ALL THE PROJECT IS DONE BY USING MULTITHREADING CONCEPT AS PROFESSOR MENTIONED.