Group UDP, IRC documentation:

This documents explains the basic functionality and coding used in the group ALL project.

**Requirements:**

* Computer capable of running python 2.7
  + The following imports should be available to use:
  + sys
  + socket
  + select
  + time
  + base64
  + Os
  + paramiko
* Python module ‘crypto’ installed. This can be done using pip.
* Active internet connection

The recommended and tested OS for running this program is Kali Linux, however any debian based operating system should also work.

Some users may wish to generate new private keys instead of using the ones already included with the program

**Set up:**

This will require some basic linux bash commands. If you do not feel comfortable doing this, use google and get some help first.

To use the program and chat using the server, the server must first be started. In this instance, the chat server is running on a VPS located at ‘chat.teamudp.co.uk’ and runs on port 80.

Starting the server:

The server should be up on chat.teamudp.co.uk, however, if you are running the server on a different network you can use the following steps to start an instance of the server:

*(note: if you are starting the server on a different network, the client will need reconfiguring; see steps below)*

1. Use the *cd* command to change to the directory where the downloaded program is stored.
2. Using the *ls* command, you should see a folder named *server*
3. *cd* into the server folder and run *python server.py*
4. If you see a message displaying “Chat server started on port [port]” the server has successfully started.

Creating users:

This part of the document references files in /server

The server includes an authentication system where users will need to enter a correct username and password to login. A dictionary is created with the username as they key and the data being a hashed version of the password (using SHA\_512). The dictionary is then pickled into a file called users.db … if usernames/password are forgotten this file can be deleted, but this will delete ALL users

To add a new user, this folder also contains a script called dbfill.py

This script accepts two system arguments in the form of: *python dbfill.py [username] [password]*

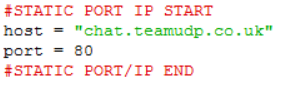
(It is recommended that user names should not contain special characters and should be different)

Starrting the client:

1. Use the *cd* command to change to the directory where the downloaded program is stored.
2. Using the *ls* command, you should see a folder named *client*
3. *cd* into the client folder and run *python client.py*
4. If you see a message displaying “Connected to chat.teamudp.co.uk” then you have successfully connected and may proceed to enter your designated username and password

Reconfiguring the client for a different network:

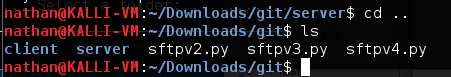
The program can be made to run across any type of network by reconfiguring a few lines of code in the client. If this is the case, follow these instruction:

* *Look for the line of code that reads “#STATIC PORT IP START#”*
* 
* *Change the host and port to the location of the server.*

**Using the program:**

**(some of the interfaces you see may differ slightly, however the steps for using the client remain the same)**

In a terminal window, navigate to the downloaded files. You should see a folder called client:



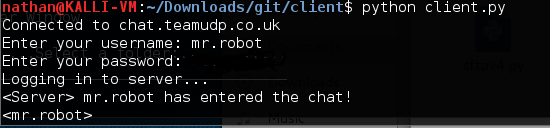
Navigate to the client folder by running cd client. You can verify you are in the right folder as this folder should contain a file named *client.py* and *pub.pem*



Initiate the client by running *client.py.* You can do this by using the command *python client.py* … If this was successful, your screen should look something like below:

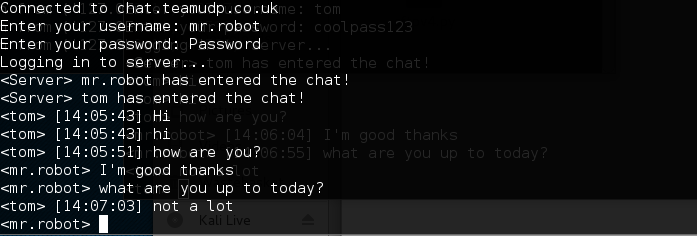


If you do not connect and receive an error, please refer to the troubleshooting part of this document.

You can now proceed to enter your username and password, designated to you by the administrator. If the login succeeded your screen should look something like below: 

If you do not connect and receive an error, please refer to the troubleshooting part of this document.

Wohooo! You are now connected. Type your message and hit return to send it. You can see all the messages that have been typed on the server by looking at the thread of messages:



You can exit at any time by using cntrlC (a keyboard interrupt)

**Troubleshooting:**

In an event where the program does not work, this part of the document should help you diagnose issues you may be having.

|  |  |  |
| --- | --- | --- |
| Error | Issue | Troubleshoot |
| Unable to connect to chat.teamudp.co.uk. Is the server running? | No connectivity to server | * Check for internet connectivity * Check that chat.teamudp.co.uk resolves to 46.101.23.38 * Check firewalls/ip tables that traffic passing on port 80 is allowed * Try running the program as an administrator/sudo |
| Login failed | The username of password has not been entered correctly. | * Check for caps lock/num lock * Try again * Try again and cry * Delete the db file which stores all the users and password and run dbfill.py to create a new database of users and password |
| Socket.error [errno13] | Can sometimes occur when starting the sever; the program does not have permission to create a new socket | * Run the server as root: *sudo python server.py* |
| Keyboard interrupt | Keyboard interrupt has occurred by the user hitting cntrlC accidently | * Learn how to type, idiot. |