

Python Project Report

Compsys302

Steven Yan | hyan506 | 07/06/2018

# Abstract

Our Client has requested for a way of online communication without worrying about being monitored. This report presents a solution to the request and the concept behind the application.

This application allows authorized user to log in. After users have logged in, they will be able to see other online users and send message or files to then. They can also edit their profiles and look at other’s profiles.

# 1. Introduction

This application is designed based on the Login server+P2P Network, which means the server is only used for user authorization and sending online user list. All the message or file transmission are using peer to peer network.

The application is implemented by Cherrypy, which is a python web framework. Webpages are implemented by HTML and CSS. Most of API has a output in JSON format, which makes the data light-weight and easy to deal with. This makes the network transmission and data processing (read and write) faster.

# 2. Top-Level View

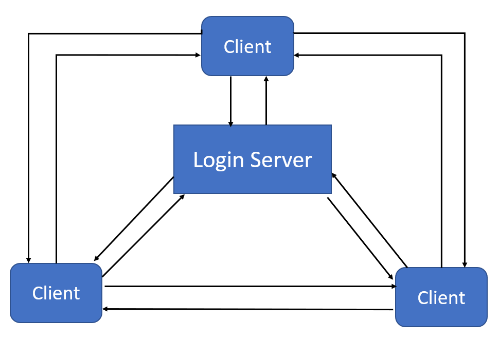
* 1. Login server & P2P Network

The network system that being used in this application is login server and peer-to-peer network. There are two main reasons of why we chose this network structure.

The first reason is security. The only information goes into the server is user’s username and password. All messages and files are sent from P2P network, which means the data will be protected even if someone attacks and takes over the control of server.

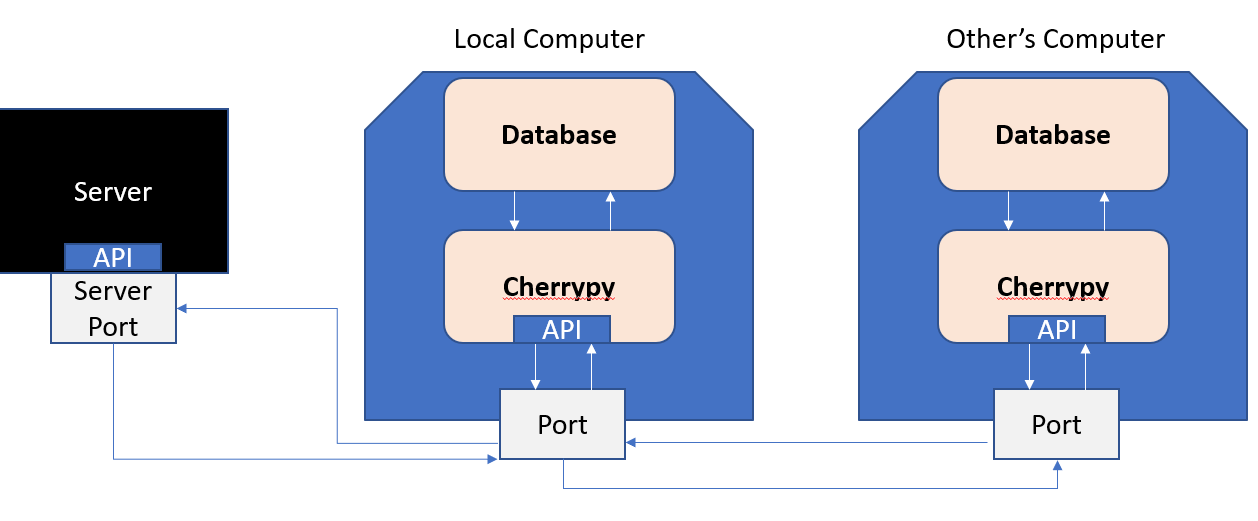
The second reason is of the server goes down, users already logged in will still be able to communicate with each other because communication between users are peer-to-peer. This is a very good feature to have in case of emergency.

The last reason is the system does not require a powerful server. All the information that the server needs to deal with is users status (username, password, ip address, etc.) This can save a lot of money as running a powerful server is very expensive.



* 1. Networking

API stands for application programming interface, which is a set of functions for building software. In this application, API is used for communicate and exchange information between local computer and the outside world (server + others’ computer). We can call APIs and deal with information in our database through cherrypy, which is pythonic.



# 3. Issues during Development

3.1 Making HTML page dynamic.

After getting information by calling API and store them into database, they are supposed to be displayed on the user interface, which are the page they open from browser. Inside the python, it is very unefficient to generate html code, so the page is a separated html file apart from the python code. Therefore, Jinja2 is used for rendering HTML templates. Jinja2 can replace the code written in jinja syntax by the parameter passed into the render function. This makes the page (HTML) changes every time it is rendered.

3.2 Using Database

When the information is received, they will be stored in the local computer. It is possible to store all data into TXT file, but it does not support any query method. In order to store the data properly, database is used. To edit a database file, we need to use SQL commands. However, we are using python to edit the database. Therefore, SQLite library is used for editing the database. In the application, data is stored inside different tables depending on with kind of data it is, so that the data can be easily extracted when it is needed.

# 4. Design Features

In the application, threading is used to keep reporting to the server in order to keep the current online. Also, threading is used to update the newest online user list every 20 second.

When the user clicks other’s username on the list, their profile pages will be displayed.

The message history will be displayed in the chat box if the user clicks the send message button beside other’s username

There is a notice board on the main page. When the user receives messages or files, it will show who has sent message/files to you. The user can click on the notice and their chat box will be displayed.

# 5. About Peer-to-Peer method

P2P (usually be known as Peer-to-Peer) Network is a network created between 2 or more computers without a separate hosting server. Therefore, every PC is playing both server and client. This is the reason why we did not choose to do a fully P2P application. It is very difficult to synchronize the authentication data. If the authentication does not take place, anyone can send and receive data which makes the app not secure. However, P2P method also has its advantage. It has ability to robust, which means if the system lost the connection of one node, the system will still be active as long as there are 2 computers connected. What is more, the P2P method is cheap to be implemented as no server needed.

# 6. About the Protocol

Protocols are rules or standards that APIs should follow. It functions like a language that API speaks. The communication will only success when both sides follows the protocol while building their applications. If developers build their apps following protocols, their APIs will know what parameters will be passed in and what parameters they need to return. In this application, APIs used to communicate with the login servers such as Report were built while following the login server protocol. APIs used to communicate with other users such as receiveMessage were built while following the application protocol. None of the communication will work if we do not follow the protocol. We can update more APIs in the protocol if we decided to further develop other functionalities of our applications. As long as the API is built following the protocol, the communication should work.

# 7. Tools Used

7.1 Python

Python is a high-level programming language which can be used on many purposes. People use python for media processing, website build and many other reasons. There are many plugins for python which makes python very flexible and powerful. In this project, python is used for building web application.

7.2 HTML and CSS

HTML is the standard language for building webpages. CSS is the style sheet which can edit elements in the html and make the webpage look better. They are very commonly used while people are trying to build a webpage. They are used to build a webpage which displays the UI in this application

7.3 SQL and Database

SQL is structured query language that used for accessing and editing data inside database. SQLite is the database engine that can be used in python. It is commonly used in the industry and easy to learn. In this application, SQLite is used to edit and access data in the database.

# 8. Future Development

8.1 User interface

The page is not really dynamic because it auto-refresh itself every 30 seconds, so that the jinja2 can re-render the page. Javascript and JQuery could be used to make the UI dynamic.

8.2 logout while the application exit

The user have to logout manually to get their status offline in the server. A thread could be used inside cherrypy and keep checking if the engine is stopped. Once it detect the engine is stopped, it should call the API to log off.

8.2 Encryption and decryption

Encryption and decryption could be added to make the application more secure, as other people needs keys to read your data, otherwise the data will be unreadable.