|  |  |
| --- | --- |
| **Practicum Case** |  |
| COMP6548  Programming for Penetration Testing |
| **Cyber Security** | **<Case Code>** |
| ***Valid on*** *[Odd/Even/Compact] Semester Year 9999/9999* | **Revision 00** |

## Learning Outcomes

* Socket Network Program for Penetration Testing
* Web Vulnerability with Programming
* Additional tools for Penetration Testing

## Topic

* Introduction to Socket Programming

## Subtopics

* Create Client Program using TCP Protocol
* Create Server Program using TCP Protocol
* Python Threading and Multi-processing

## Soal

*Case*

**Chatting**

You ask a Python Programmer are asked to code a program that can chat between one another using **socket library**. The chat room only consists of 2 people, here are the requirements:

Server (Chat room host)

1. The program contains argument parser with **getopt** **library** with the following specification:

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Short Args | Long Args | Description |
| 1. | -i | --ip | The ip address of the target (default ip 127.0.0.1) |
| 2. | -h | --help | Print the help menu |
| 3. | -p | --port | The port number (default port 1234) |

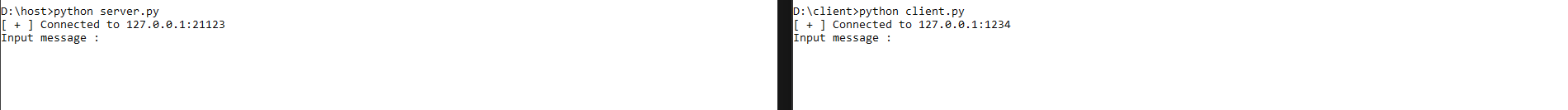
1. Make the program to wait for the client to connect to the chat room host. After there is a client connect to the host, initiate the **chat process**. Make the program can **send** message to the client and **receive** message from the client.
2. Make the program **exit** and **terminate** the **connection** if the user type “exit”.

Client

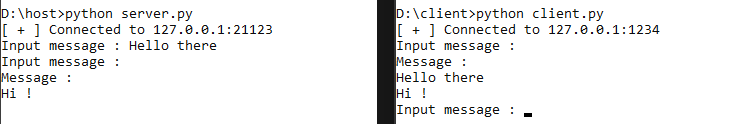
1. The program contains argument parser with **getopt** **library** with the following specification:

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Short Args | Long Args | Description |
| 1. | -i | --ip | The ip address of the target (default ip 127.0.0.1) |
| 2. | -h | --help | Print the help menu |
| 3. | -p | --port | The port number (default port 1234) |

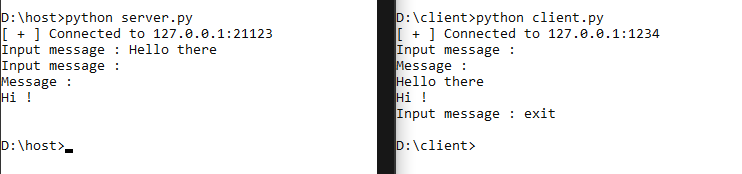
1. Make the program to connect to the chat room host. After client has connected to the host, initiate the **chat process**. Make the program can **send** message to the host and **receive** message from the host.
2. Make the program **exit** and **terminate** the **connection** if the user type “exit”.



**Figure 1. Client and Server Connected**



**Figure 2. Client and Server Sending Message**



**Figure 3. Client Exit from the Chat Room\***

\*Note: The host doesn’t automatically exit after client exit, you need to type enter first and then the host’s socket will terminate the connection. (The input() still hold the host program) and vice versa.