

Computer Network Final Project Report

B06902006 王俊翔 B06902058 吳崇維

0) 所有程式皆在工作站上執行，測試

1) 架server, client 資訊

對於server，只需要編譯並跑出執行檔即可。不過對於在不同IP上架設的server，client需要特別的指定server的所在，因此需要在 client.cpp 程式中的19行處，需隨著server架設的IP來改變其值！

2) 執行server, client

兩者分處不同的資料夾，執行前先make，之後便可 ./server 以架設server，用戶端則同樣先 make 編譯檔案，輸入./client 便可連上server。

以下皆針對client跑的程序結果。

3) 登入介面

- 任何時候，輸入 "/b" 皆為『返回上一頁』
- 登入時，若有帳號則直接輸入帳號密碼，否則輸入 "/signup" 來創立新的帳號
- 任何時候，以 ^C 指令都可強制中斷程式

4) 主畫面

主畫面列出了可以執行的操作，鍵入對應的英文字母就可以執行該操作

- (C) chat !

進入後，server會詢問你想要跟誰聊天，並列出建議選項 (你的朋友們)，決定聊天對象後，輸入他的ID便可開始聊天。(若對方在線並且也打開對你的聊天視窗，即可即時通訊。亦有離線保存訊息)

- (V) view friend list

此選項可以看你目前有哪些好友。

- (A) add friend

此選項可以讓你加入好友 (單方向的加入)

- (W) who is online

可以看到目前所有人的上線情況，以便進行聊天

- (L) log out

登出client端

- (F) file

進入此選項後，將有兩個項目可以選擇，分別是(R)receive file 及(S)send file，進入(R)後可以查看是否有新檔案傳來，而(S)可以傳檔案給特定的人(可以一次傳多個檔案)。

運作原理

檔案的部份每個user在註冊的時候server會在其User資料夾中新建一個名為file的資料夾，往後若要傳檔案給某人，則在要收取的人的file中新增檔案，對方要收取時server就查看file中是否有檔案以進行辨別。

5) 運作原理

整個server倚賴一個名為 User 的資料夾，裡面保存了所有client的資訊，server在運行的過程中若接到client，則fork出一個process專門處理該client，並且注意client結束時需要用 SIGCHLD來避免 zombie process。

對於每個client的資料夾，裡面保存了他的密碼，好友們，通訊紀錄，未讀訊息...等資料。並在client的command下更改互動，達到通訊的效果。

通訊範例：

User wayne , 假設目前為server1服務

User jim, 假設為server2服務

若wayne傳訊息給jim，server1收到該訊息後便會到jim的資料夾內，寫入某個檔案中，代表存放 wayne → jim 的訊息，與此同時，若是jim打開了wayne聊天視窗，則server2會負責讀取server1寫入的訊息，並呈現在螢幕上。

反之亦然，藉由server的不斷讀取以及輸出，變有了即時通訊的效果。

其餘的功能皆只是單純的建檔讀檔，並未有太多的技巧。

