Part3 操作說明文件

編譯Client 端程式：cd到client資料夾，執行 make指令

執行Client 端程式：在client資料夾中有client的執行檔，cd到client資料夾，執行 ./client {ip address} {port number}指令即可

編譯Server 端程式：cd到server資料夾，執行 make指令

執行Server 端程式：在server資料夾中有server的執行檔，cd到server資料夾，執行 ./server {port number}指令即可

程式執行環境：Ubuntu 20.04

安全傳輸實作的方法及流程說明：

Client 及 server 分別利用 rsa 演算法產生各自的公私鑰，大小為 2048 bits。在 A 要轉帳給 B 並通知 server 的過程中，那總共進行兩次加密，一次解密，以下分別敘述。

第一次加密（A 傳給 B 時）：

A 在轉出前把明文 *M* 利用 A 的私鑰加密成密文 *E*，再透過 ssl 傳輸傳給B。

第一次加密（B 傳給 server 時）：

B 收到 A 的密文 *E*，並且需要再加密一次。由於密文的大小為 256 bytes，大小超出 B 單次可加密的資料大小（我是利用 RSA_PKCS1_PADDING，最多加密 256 – 11 bytes），因此需在 B 受到密文後，將密文*E* 切成 *E1* 及 *E2*，並將 *E1* 及 *E2* 分別加密成 *E1'* 及 *E2'*，再合併傳給 server。

解密（server 收到訊息之後）：

Server 收到密文後，先將密文切回 *E1'* 及 *E2'*，再利用 B 的公鑰分別解密，變成 *E1* 及 *E2*。將 *E1* 及 *E2* 串成 *E* 後，再利用 A 的公鑰解密回明

文 M。有了明文即可用第二階段的方式進行交易。

參考資料：https://www.openssl.org/

http://neokentblog.blogspot.com/2012/10/openssl-ssl.html

https://aticleworld.com/ssl-server-client-using-openssl-in-c/