Network Security

Project 1.2

Encrypted Socket Programming

Instructor: Shiuhpyng Shieh

TA：E-Lin Ho, Jui-Chien Jao

1. Project Description

In this project, you need to implement a socket program to communicate with the server at **140.113.194.88:30000** through a secure channel using RSA cryptosystem. The communication protocol, illustrated in Fig. 1, shall be implemented in your client socket program.

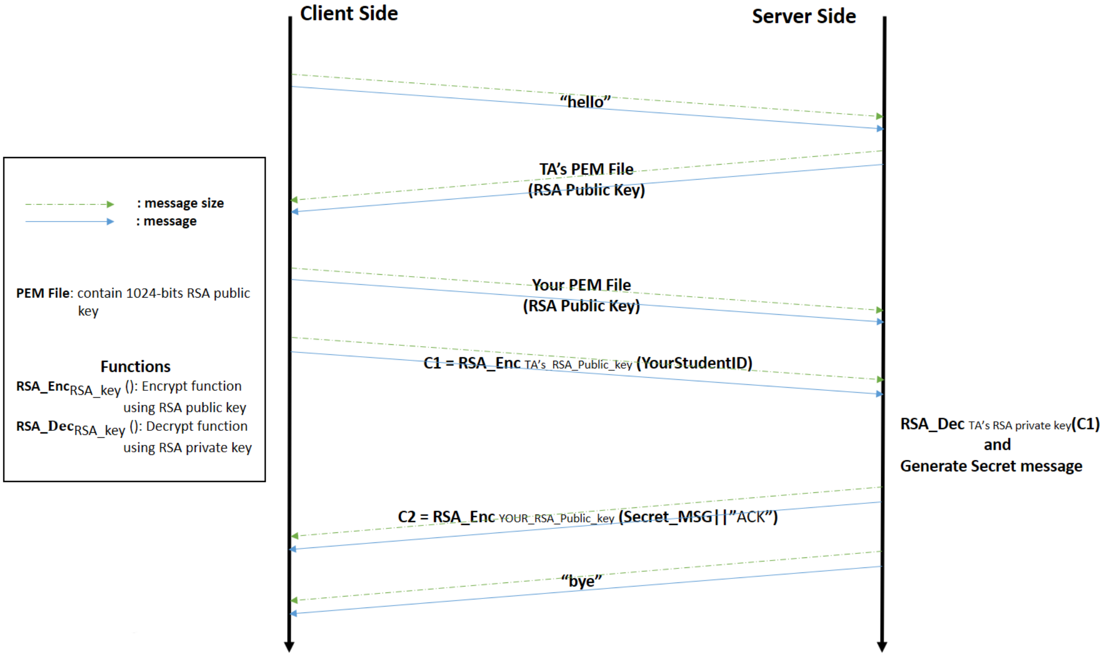
Client Side
Server Side
"hello"
TA's PEM File
(RSA Public Key)
Your PEM File
(RSA Public Key)
C1 = RSA_Enc TA's RSA_Public_key (YourStudentID)
RSA_Dec TA's RSA private key(C1)
and
Generate Secret message
C2 = RSA_Enc YOUR_RSA_Public_key (Secret_MSG||"ACK")
"bye"
: message size
: message
PEM File: contain 1024-bits RSA public key
Functions
RSA_Enc RSA_key (): Encrypt function using RSA public key
RSA_Dec RSA_key (): Decrypt function using RSA private key

Fig. 1 The protocol of Encrypted Socket Programming.

As shown in Fig. 1, a "hello" message sent by the client starts a new conversation. Firstly, the client and the server exchange their public keys. Thus a message encrypted by the RSA public key owned by the other side can be securely transmitted and decrypted. Secondly, the client sends a message that encrypts student ID by **server's public key**. The server will decrypt it and send a magic number back to client. Because the magic number is generated from the input data, every student shall receive his/her own unique value. The magic number is also encrypted, and it can be decrypted with **client's private key**.

The magic number will be in the form of ***XXXXX***, surrounded by three preceding and three succeeding ‘*’. The last “bye” message in plaintext terminates the conversation.

Reminder: For the implementation of RSA cryptosystem, the use of the OpenSSL library supporting C/C++ and Java is recommended.

2. Deliverables

Each student must work individually and submit a .zip file, named by “<YOUR_STUDENT_ID>.zip” containing:

a) The source code of your socket program.

b) A report,

  - ◆ The magic number you receive from the server.
  - ◆ The instructions for compiling your source code.

3. Reference

● pcap file:	Project1.2.pcap
● client proto code in python:	Project1.2.client.py

- Openssl: https://www.openssl.org/
  https://www.openssl.org/docs/manmaster/

Any anomaly connection such DDoS will be traced for punishment.

Server online time: 10/10(Mon)

Deadline: 2016/10/18(Tue) 23:59:59