200 points. Individual Work Only. Due February 23, 2023 before 12:30 PM.

1. Objective

Design and implement a multi-threaded client-server program using secure sockets for communication between the client and server programs.

2. Description

1. Read the article "Using JSSE for secure socket communication" by Greg Travis (available in Blackboard as j-jsse.pdf) and then test the programs (available in Blackboard as *j-jsse.tar*). Download the file *j-jsse.tar* and extract in (*tar xvf j-jsse.tar*). It will create a directory called *j-jsse*. Change to that directory and follow the instructions provided in the README file to compile and test these programs. Note that this program uses mutual authentication and hence requires key exchanges between client and server.
2. Modify the socket-based client/server program developed in Homework-1 to include secure sockets communications.

3. Guidelines/Hints

First follow the article by Greg Travis on using secure sockets even if you are programming in Python to understand the concepts, key generation, and key exchange. Review the documentation provided in Section 10 to understand the APIs and example programs.

4. Graduate Students Only (Bonus for Undergraduates)

Write a short essay (at least 500 words) comparing virtual machines and containers. Make sure to discuss if there are any security implications in using with either one of these options. Make sure to cite any references that you use and feel free to use a diagram similar to Figure 3.8 or 3.9 in the textbook.

5. Working Environment

You can develop and test the client and server programs on any machine that you have access to. You may be required to demonstrate the working program to the instructor to get full credits for this assignment.

6. Short Answer Questions

Answer the following questions and submit your answers as a separate Word or PDF file.

1) Now that we have secure communication, if student A has the server program running on machine P, could student B use his client program running on machine Q to connect to A's server program if student B knows the port on which student A's server is running?
   a) If your answer to question (1) is yes, then explain how to prevent this from happening.
   b) If your answer to question (1) is no, then explain why this is not possible and what is necessary to support this option.
2) If we were to modify the server program such that we could execute the server program on multiple computers instead of one computer, what changes would be required to the server program and the client program? Write down the changes required, if any, for both server and client program (no need to implement any of these changes, just describe this in the report).

7. Feedback Questions (answer to these questions has no impact on your grade)

1) Was this homework too difficult, or too easy?
2) Was the assignment fun or challenging?
3) Was there something that was unclear?
4) Was the homework too long for the given amount of time?
5) What did you learn from this homework?

8. Submission Instructions

List ALL the references you used in this homework as well as test cases used to test your programs. This includes any classes that you used that you did not write and any help you received from any other sources. Use appropriate class name and include comments to indicate various operations performed by the program. Your program must have the following header information within comments:

```
/*
        Name:
        Section: CS 491 or CS 591
        Homework #:
*/
```

Make a directory CS591/homework2 and then two separate directories one for the server (server) and one for the client program (client). Include a README.md file in the homework2 directory that provides the instructions for executing the client and server programs.

Create a github repository (say, Spring2023_CS691_HW2) and upload your source files to the github repository. Make sure that you have created a **private repository/project** (click on visibility to be private). Add pvbangalore@ua.edu as a collaborator to this project.

Create a tar/zip file (only tar and zip are accepted, all other file formats will not be considered) of the homework1 directory using the following filename format: <crimsonId>-cs491-hw2.tar or <crimsonId>-cs591-hw2.zip. Upload the tar/zip file that includes the source code and instructions for executing the programs and a separate Word/PDF document for the typed solution to short answer questions and the feedback questions. In the comment section of the homework submission, include the link to your github repository for this homework.

9. Late Submissions

Submissions must be made on the due date before the beginning of the class. Any submissions received after the due date will receive a score of 0 for this homework.

10. Resources

1. For Java Secure Socket Extension (JSSE) Reference Guide: https://docs.oracle.com/en/java/javase/19/security/java-secure-socket-extension-jsse-reference-guide.html
2. For Java JSSE Examples: https://docs.oracle.com/javase/8/docs/technotes/guides/security/jsse/samples/index.html
3. For Python SSL Module Documentation: https://docs.python.org/3/library/ssl.html