CS380 — Project 8

March 3, 2015

Due: Friday, March 13, 2014 (80 points)

Description

In this project, we'll use the Java Cryptography Extension to send data securely across a network. You will first obtain a public key from me that will be posted on Blackboard as a file containing the key's serialized form. You'll generate a symmetric session key to use for communication and send it to me encrypted with the given public key. We'll use this symmetric session key for the rest of the communication.

The public key that I provide will be an RSA key. It is given in serialized form, so use an `ObjectInputStream` to read from the file in to an instance of `RSAPublicKey`. Use an instance of `Cipher` with the public key. Generate an AES key using `KeyGenerator` as shown in the example on Blackboard. See https://docs.oracle.com/javase/1.5.0/docs/guide/security/CryptoSpec.html for more information about JCE. `ByteArrayInputStream` and `ByteArrayOutputStream` can be useful for writing data (like when serializing the session key) directly to an array of bytes.

You must perform the following sequence of operations:

1. Deserialize the given RSA public key from the file.
2. Generate an AES session key using `KeyGenerator` as shown in the JCE example on Blackboard.
3. Serialize the session key, encrypt its serialized form with the given public key, and send this ciphertext as the data in a UDP packet.
4. I will send back `0xCAFEBABE` if I successfully received your key, otherwise you will get an error code indicating the problem.
5. Now, you must follow the same process as project 5 by sending 12 UDP packets (destination port must be set to 38008 this time) starting with length 2 and doubling each packet, but encrypt the *entire* packet using the session key. I will send back `0xCAFEBABE` after each packet if I can decrypt it correctly.
6. Have your program output the server's response and round trip time as in project 5 for each of the 12 packets, then output the average round trip time for all 12 packets.

The server should be running by the end of Thursday, March 5 and the public key will be posted by then on Blackboard. The public key file name should be `public.bin`.

Code	Reason
`0xBAADF00D`	Problem with IPv4 portion of packet
`0xCAFED00D`	Incorrect destination port in UDP packet
`0xDEADC0DE`	Invalid UDP checksum
`0xBBADBEEF`	Incorrect UDP data length
`0xDEADF00D`	Unable to decrypt session key

Submission

Submit a single Java file, `CryptoClient.java` to Blackboard. Don't package your classes. If you include multiple classes, place them all within the `CryptoClient.java` file.