Johns Hopkins ■ CS600.424: Network Security ■ Spring 2012

Homework 4 – Programming Assignment
Due Date – April 10th

Instructions: Please submit your work (code) via Blackboard or email attachment to Marcus Sanchez.
Email address: msanch15@jhu.edu

To apply the lessons you have learned in lecture you will implement a secure chat program that will perform public key encryption, symmetric key encryption, and integrity validation. Your program will implement a simple secure echo chat program in which one client connects to a server and anything the client writes to the server is echoed back to the client. All of the code should be written in Java with the help of the Java.net, Java.crypto and Java.security libraries and any other standard Java library that can help you implement the assignment. The Java security and crypto libraries give access to an assortment of security schemes some nice ones to work with for this assignment are RSA, AES, and HMacMD5 but you can use any scheme you would like. The flow of the program should go as follows:

1. Client connects to the server
2. Server or client initiates a handshake so that the server has the client's public key
3. Server exchanges the symmetric key and any additional keys needed for integrity checks, with the symmetric key encrypted by the client's public key and any further keys encrypted by the symmetric key
4. Now any message passed between the client and server is hashed by the integrity function and encrypted by a symmetric key
5. For example, client sends a chat message to server -> server decrypts messages and checks that it passes the integrity validation then encrypts and hashes the message and sends it back to the client

The server and the client should handle any exceptions such messages that fail integrity validation by reporting the error to the console and dropping the message.

Questions: The above scheme seems secure but a man in the middle attack could easily gain access to the chat messages between the server and client.

1. After writing the program, write out an explanation of how a man in the middle attack would exploit this system.

2. What technology would be needed to make sure that the client and the server know for sure that they are talking to each other and not a man in the middle?