CS 3326: Networks Security

Spring 2016

Project Phase I

Assigned: February 12th , 2016 Due: March 1st, 2016

1 Background

In lecture, we have been learning about the basics of secret key/ symmetric key cryptosystems. In order to successfully complete later phases of the term project, you will need to not only understand how these tools work, but also how to use them when developing applications. In this short initial phase of the project, you will be responsible for studying the BouncyCastle Cryptography API, and learning to use the tools that it provides to encrypt and decrypt data within Java programs. To begin, visit http://www.bouncycastle.org/java.html and download the version of BouncyCastle best suited to your particular Java installation. This web site also contains a plethora of documentation regarding how to use these APIs, which provide a good starting point for your study. If you choose to use an IDE while developing your Java code, visit this website http://www.itcsolutions.eu/2011/08/22/how-to-use-bouncy-castle-cryptographic-api-in-netbeans-or-eclipse-for-java-jse-projects/ for a comprehensive explanation of how to install BouncyCastle APIs in your IDE (Netbeans or Eclipse).

2 What do you need to do?

After studying the BouncyCastle documentation, you are responsible for writing a Java program that makes use of the DES and AES cryptosystems to encrypt and decrypt text. In particular, you are to write a single Java class that does the following:

- Input a line of text from the console
- Crypto tests
  - Generate a 64-bit DES key and nother 128-bit AES key
  - Encrypt the string read from the console once using DES, and once using AES
  - Decrypt the resulting DES and AES ciphertexts and print out the equivalent plaintexts

- Extra Credit (*Up to 5 points*)
  - Generate an array of 100 different random strings (50 characters each)
  - Time how long it takes to encrypt all 100 strings using DES
  - Time how long it takes to encrypt all 100 strings using AES
  - Output the time difference between each crypto system

Note: You can use BouncyCastle's lightweight crypto API or you could instead use BouncyCastle as a provider for Java's cryptography extension (the JCE).

3 What to turn in ?

This project is to be implemented in groups (2 - 3). Each group must arrange a fixed date/times to meet and work on this together. Students who have less experience using Java should take advantage of this project to improve their programming skills. Each group must submit the following: A single Java class that carries out the above tasks when invoked from another test driver class. Both Java source files, as well as a text file named `instructions.txt` that explains how to compile and run your program should be bundled together in one zip file. Your zip file must be submitted to the Blackboard Project (Phase 1) Turin by 11:59 PM on Tuesday, March 1.