School of Information Systems

IS302: Information Security & Trust

Week 3 Lab: Symmetric Key Cipher

16 Jan 2017

SINGAPORE MANAGEMENT UNIVERSITY
SCHOOL OF INFORMATION SYSTEMS
IS302 - INFORMATION SECURITY AND TRUST

LABORATORY EXERCISE

1 OBJECTIVE AND LEARNING OUTCOMES

1.1 OBJECTIVE

The objective of this exercise is to provide a general overview of the Symmetric Key Cipher. OpenSSL and JCE will be used to demonstrate symmetric encryption.

1.2 LEARNING OUTCOMES

At the end of the laboratory session, students should be able to:

1. Understand the concept of symmetric encryption
2. Explore OpenSSL and crypto features.
3. Able to use OpenSSL and JCE to perform symmetric encryption/decryption.

2 LAB EXERCISE

2.1 Preparation

- Install OpenSSL (\\10.0.104.26\is302)
  - Windows users: download installation package
  - Setup PATH environment variables
- Install JDK
- Install a hex editor (e.g., HHD HEX Editor)

2.2 OpenSSL AES Encryption/Decryption

1. Go to your working directory (e.g., C:\is302) and run the following command line to encrypt your plaintext file (review)

   openssl aes-192-cbc -e -in *input-file-name* -out *output-file-name* -pass pass:<*your-password*>

   Example: **`openssl aes-192-cbc -e -in`** `msg.txt` **`-out`** `cipher` **`-pass`** `pass:asdfgh`

   Remember your password for decryption.

2. Run the following command line to decrypt the file using the same password.

openssl aes-192-cbc -d -in *input-file-name* -out *output-file-name* -pass pass:<*your-password*>

Example: **`openssl aes-192-cbc -d -in cipher -out msg1.txt -pass pass:asdfgh`**

Exercise: Compare the output file with your original file. Are they the same?

Ans: _____

3. Open the ciphertext with a HEX editor such as HHD HexEditor
4. Do not edit the first and last few bytes in the file. Change 1 byte of the cipher text in the file and save it as "*mod-cipher*".
5. Use the same method to decrypt *mod-cipher*.

Exercise:

a. Was "*mod-cipher*" decrypted correctly?

Ans: _____

b. Does encryption provide message integrity?

Ans: _____

2.3 JCE AES Encryption/Decryption (optional homework)

2.3.1 Key Generation (review)

1. Download the java program "AesGenKey.java" from course website to your working directory (e.g., c:\is302).

   This program generates an AES key and stores the key under an alias provided by the user. The key will be store in a JCE KeyStore file, "keystorefile.jce". This file will be created in the current directory when the program is run for the first time.

   Usage: java AesGenKey <key alias>

2. In the windows command shell, change to the directory where the file is downloaded and compile the program.

   Example:

   C:\is302>javac AesGenKey.java

Note: Ensure that the PATH variable has been set to the Java "bin" directory.
(E.g., SET PATH=%PATH%; C:\Program Files\Java\jdk1.5.0_08\bin)

3. Run the following command to create an AES key with alias "myaeskey".

   java AesGenKey myaeskey

4. Verify that the file "keystorefile.jce" is created in the current directory.

Exercise: Refer to the source code in AESGenKey.java and answer the questions below.

a) What is the length of the key generated by AESGenKey?

   Ans: _____bits

   What is the password used in AESGenKey to unlock the KeyStore file, "keystorefile.jce"?

   Ans: _____

2.2.2 Encryption/Decryption

1. Download the java program "AesEncrypt.java" from course website to your working directory (e.g., c:\is302).

   This program encrypts a plaintext file using AES with the AES key stored in "keystorefile.jce". The ciphertext will be saved as a file in the current directory.

   Usage: java AesEncrypt <key alias> <message file> <ciphertext file>

2. In the windows command shell, change to the directory where the file is downloaded and compile the program.

   Example:

   C:\is302>javac AesEncrypt.java

3. Download the clear text file "largefile.txt" from course website to your working directory.

4. Run the following command to encrypt the plaintext file "largefile.txt" with the key alias "**myaeskey**".

   java AesEncrypt myaeskey largefile.txt aesencrypt.txt

5. Verify that the file "aesencrypt" is created.

6. Download the java program "AesDecrypt.java" from course website to your working directory.

   This program decrypts a ciphertext file using AES with the AES key stored in "keystorefile.jce". The decrypted text will be saved as a file in the current directory.

   Usage: java AesDecrypt <key alias> <message file> <ciphertext file>

7. In the windows command shell, change to the directory where the file is downloaded and compile the program.

   Example:

   C:\is302>javac AesDecrypt.java

8. Run the following command to decrypt the ciphertext file "aesencrypt.txt" with the key alias "myaeskey".

   java AesDecrypt myaeskey aesdecrypt.txt aesencrypt.txt

9. Verify that the file "aesdecrypt.txt" is created and has been decrypted.