IS2150/TEL2810 Introduction to Security

Programming Project

This report includes project guidance and answers.

1. Message Digest

This program demonstrates the use of hashing using MD5 and SHA scheme and the MessageDigest class. Here are the steps: 1) Compile and run “Hash.java” 2) Input the message 3) Choose the method

2. Various Crypto Techniques

A. Authentication

In the authentication part, program could implement double-strength password login using message digest. Here are the steps: 1) Compile “*.java” in Authentication file 2) Run “ProtectedServer.java” 3) Run “ProtectedClient.java”

Finally, after the verifying the password, the console of “ProtectedServer.java” will print “Client logged in.”

B. Signature

In the signature part, program could implement ElGamal key generation and signature creation algorithms (for Alice), and signature verification algorithm (for Bob). Here are the steps: 1) Compile “*.java” in Encryption file 2) Run “ElGamalBob.java” 3) Run “ElGamalAlice.java”

Finally, the program the console of “ElGamalBob.java” will print the original message and “Signature verified.”.

C. Encryption

In the encryption part, program could generate DES key and encrypt and decrypt message. Here are the steps: 1) Compile “*.java” in Signature file 2) Run “CipherServer.java” 3) Run “CipherClient.java”

Finally, after decryption the encrypted message, the program will create a file to save password and the console of “CipherServer.java” will print the original message.

D. Public-Key System

In the Public-Key System part, program could generate a pair of keys to verify the information. Here are the steps: 1) Compile “*.java” in Public-Key System file 2) Run “PublicKeySystem_Bob.java” 3) Run “PublicKeySystem_Alice.java”

Finally, the program will generate two files to store the public key and verify the transmitted information.

E. X.509 Certification

In the X.509 Certification part, program could use X.509 certificate to realize the message of switch confidentiality. Here are the steps: 1) Compile “*.java” in X.509 Certification file 2) Run “X509_Server.java” 3) Run “X509_Client.java”

Finally, after verifying the public key and print the original message, the program will print the content of certificate and the expiration date.

3. **Question**

1) What are the limitations of using self-signed certificates?

a. Compared with CA, self-signed certificate is generated by the server; therefore, it is not easy to be revoked and vulnerable to attack, which is a extremely difficult way to control.

b. Due to the reasons mentioned in a and other insecurity, the application scope of self-signed certificate is relatively narrow.

2) What are they useful for?

The application scope of self-signed certificate is relatively narrow, so it will be applied to internal network. Because of its high flexibility and high speed, it will also be applied in the software development stage.