# INFA723 Cryptography and Network Security

# Lab8 A Combination of RSA and AES to Encrypt a File

# 1 Introduction

OpenSSL toolkit provides a command line tool and a crypto library used for a wide range of crypto operations. In this exercise, we are going to learn how to use the combination of RSA and AES to encrypt a file. Note that for all the functions implemented in the lab, you can find a way to implement them using the OpenSSL crypto library.

Both symmetric ciphers and public-key ciphers have their advantages and limitations. In practice, these two ciphers are usually used together. For example, in this example, the file is encrypted using a secret key and the secret key is generated from a password file. The password file is protected by public-key ciphers.

A lab package has been created for the lab. Go to the class website and download the lab package. Assume your Cygwin installation folder is c:\cygwin. Unzip the package to your Cygwin home folder, e.g., C:\cygwin\home\user name.

The process used to create the cipher text and encrypted password file is demonstrated below. This is just an example showing how to use a symmetric cipher and a public-key cipher together. To create a cipher text use a password file, enter the command:

**$ openssl enc -aes-256-cbc -salt -in plaintext1.txt -out cipher.txt -pass file:./password.txt**

To encrypt the password.txt using a 1024-bit public key, enter the command:

**$ openssl rsautl -encrypt -pubin -inkey public1024.key -in password.txt -out encryptedpwd.txt**

The cipher text, encrypted password file, and private key are given in the lab and you are required to restore the plaintext.

# 2 Objectives

* Benchmark DES and RSA algorithms
* Use a combination of RSA and AES to encrypt a file

# 3 Use a combination of RSA and AES to encrypt a file

1. DES benchmark

**$ openssl speed des**

1. Rsa benchmark

**$ openssl speed rsa**

RSA benchmark shows both the sign and the verify throughput. In general, verify throughput (using public key) can be used as encryption throughput and sign throughput (using private key) can be used as decryption throughput.

1. Use private key to decrypt secret password

**$ openssl rsautl -decrypt -inkey private1024.key -in encryptedpwd.txt -out password.txt**

**Enter pass phrase for private1024.key:**

The private key is password protected and you need to enter pass phase shown below (lower case)

dakota

1. Use password to decrypt the cipher text

**$ openssl enc -d -aes-256-cbc -in cipher.txt -out c.txt -pass file:./password.txt**

1. Check the plaintext

**$ cat c.txt**

# 4 Question

1. (6 points) What is the throughput when you sign and verify a message using a 1024-bit key in RSA? Which one is faster? Why?
2. (6 points) What is the throughput when you encrypt a 1024-bit message block using DES CBC mode? Compared with a 1024-bit RSA verify throughput, which one is faster? Compared with a 1024-bit RSA sign throughput, which one is faster?
3. (8 points) What is the plaintext you restored from the cipher text?