# CSCE 4013 Applied Cryptography – Spring 2016

# Lab 1: Secure Two-Party Communications

**Due: 11:59 PM, March 31, 2016**

**NOTE: Up to two students can form a team to work on this lab.**

## 1. Introduction

In this assignment, you will implement secure communications between two parties, Alice and Bob. This assignment is designed to practice key exchange, encryption/decryption, and integrity protection with secret key cryptography and public key cryptography.

## 2. Task Description

Communication scenario: Alice (as a Client) needs to send a message of 2000 bytes to Bob (as a Server), where each byte is 'a'; Bob needs to return a message of 1000 bytes back to Alice, where each byte is 'b'. Either TCP or UDP is fine for the transport protocol. Both messages must be encrypted and integrity-protected.

Step 1: Set up shared secret keys for encryption: For the communication from Alice to Bob, they agree on a shared secret key using RSA-based encryption. You can assume that they know each other's public key in advance. For the communication from Bob to Alice, they agree on a shared secret key using the Diffie-Hellman protocol. You can assume that Bob selects the public parameters of Diffie-Hellman protocol, and send them to Alice.

Step 2: Set up shared secret key for integrity protection: Generate two shared secret keys (one for each communication direction) using the same method as in the previous step.

Step 3: Send secure messages: Alice sends the 2000-byte message to Bob. This message is protected with an RSA digital signature signed over the hash (SHA-256) of the message, and encrypted using AES. Bob verifies the integrity and decrypts the message. Bob then returns the 1000-byte message to Bob. This message is protected with HMAC where the underlying hash algorithm is SHA-256, and encrypted using AES. Alice verifies the integrity and decrypts the message.

Programming language: C, C++, Java, or Python. Other languages need the instructor's approval.

You should follow the best practices of using the above cryptography algorithms. That is, use the algorithms in the correct ways.

**4. Submission Instructions**

Email your submission as a .zip file to the instructor AND the grader. The submission should include your well-commented source code and a report. The report should cover:

- A description of the structure of your program.
- A description of which part works and which part doesn't.

The email subject should include your name(s) and the assignment number like "Lab1_FirstnameLastname" or "Lab1_FirstnameLastname_FirstnameLastname" if you are a team. Your email attachment should also be named in the same way.