CSI 4139 / CEG 4399
Design of Secure Computer Systems

Laboratory #1
Due: September 20, 2018 (written report due September 24, 2018))

Goals: (1) Become familiar with some of the cryptographic libraries and functions available on the system. (2) View vulnerable states of a network server.

Details:
(1) Generate 2 public-key / private-key pairs, one for encryption/decryption and one for signing/verifying. Take a file as input and call the appropriate routines to hash and sign it, and also to encrypt it with a symmetric key and then encrypt the symmetric key with the public key of a recipient. In addition, your program must be able to input a protected file, decrypt it, hash it, and verify the digital signature using the originator's public verification key. During the lab demonstration of your program, the TA will supply a file to be protected. The TA will observe the protection of an unprotected file, and the un-protection of a protected file.

(2) Decide on the number of tests and scans your group is going to do. (The tests and scans chosen must serve a purpose in assessing the security of the server.) Use several assessment tools (downloaded from the Internet, but open source only) to detect as many vulnerabilities as possible in the web server provided by the TA in the lab.

Deliverables: You are expected to write 2 documents and demonstrate your program.

Document: (1) Write a brief description (no more than two pages) of your file protection / unprotection program and the relevant choices you have made. In particular, you should describe your environment, programming language, and crypto library, and you should explain and defend the algorithms and key lengths you have chosen. Any underlying assumptions about the input, output, and communications environment should also be discussed. (2) Describe (no more than two more pages) all the server weaknesses you were able to detect. The report must be detailed enough to explain each weakness and give suggested solutions. The purpose for each test/scan you performed must be clearly stated in the report, along with how the tool helped you to achieve your goal.

Software: You must implement the software to do protection and un-protection. This will consist primarily of calls to an underlying set of library routines (such as the Java routines available on the lab machine), as well as a user interface sufficient to allow the TA to see what your program is doing (e.g., the input and output of each function call).

The laboratory software may be done in groups of up to 3 students, but each student should ensure that s/he gets sufficient practice with the cryptographic and scanning tools. Report contents may be discussed as a group, but are to be written individually.