This HW/Lab assignment covers Lectures 8. Please review these thoroughly before starting to work on the assignment.

It is a combination of a hands-on lab exercise and conceptual problems. Problem 2 is a lab exercise while Problem 1 is a conceptual problem. You are strongly encouraged to utilize your lab session to accomplish the lab exercise.

All soft copy submissions (with answers to the problems) must be turned in via Blackboard Learn. Name your files as “Lastname_Firstname_HW3”. Please make sure that you have correctly submitted/uploaded the files.

You can submit hand-written hard copies, if you wish. However, we would prefer that you submit soft copies. If you choose to submit hard copies in any case, please turn them into my mailbox in the CS office or hand them in before the lectures. Please also make sure that your hand-written solutions and handwriting is legible and easy to understand.

You must submit by the deadline – 11 am on 11/16/2015. This applies to both soft and hard copy submissions. Late submissions will not be graded.

You will be graded based on the correctness of your answer and also on the steps that you took to come to that answer, whenever possible and applicable. Please try to show all your work, when feasible.

The assignment needs to be solved individually by every student. No collaboration of any sort is allowed, unless stated otherwise. No plagiarism is allowed. Please check the course policies against misconduct (discussed in Lecture 1). When in doubt, please consult the instructor.

Please submit early to avoid any last minute issues.

Please do start working on the homework early and do not wait until the deadline.
A. Lab Exercise and Conceptual Problems

[Please review Section B, which will be useful in answering these questions]

1. [35pts] Analyze the certificate for https://www.regions.com/
   a. [5pts] Who has signed the certificate, i.e., who is the CA? Is the CA trusted by your browser?
   b. [5pts] What is the certification authority hierarchy for the certificate?
   c. [5pts] Has the certificate expired?
   d. [5pts] What is the CRL Distribution Point (the URL from which the CRL can be downloaded) corresponding to this certificate?
   e. [5pts] What digital signature scheme is used to sign the certificate?
   f. [5pts] What cryptosystem is used to generate the public (private) keys for the subject, regions?
   g. [5pts] What is the public key that’s being certified (just copy-paste it)?

2. [65pts] Do a Wireshark trace for establishing a connection to https://google.com and answer the following questions. Provide a snapshot from your trace to support your answer (highlight specific parts in the snapshot). Also, specify the web browser used and its version, and the TLS version used by the browser in the trace.
   a. [5pts] What number identifies the SSL Handshake content type?
   b. [5pts] What number identifies the SSL Application Data content type?
   c. [5pts] What number identifies the SSL Change Cipher Spec content type?
   d. [5pts] In your Wireshark trace, what all sets of messages are bundled together into single frames?
   e. [5pts] What is the cipher suite selected for this session?
   f. [5pts] What are the first 5 cipher suites suggested by the client machine?
   g. [5pts] What is ClientRandomValue included in the Client Hello message?
   h. [5pts] Is it possible for you to learn the pre-master secret from this trace? If yes, provide the pre-master secret. If not, explain your answer.
   i. [25pts] Establish another (fresh) session with https://google.com, (close the browser and start a new session from the same browser), and identify whether the answers to each of the above questions (a-h) change or not?
B. Lab Description
In this lab, we are going to cover two main parts: Certificate Analysis and examining the SSL/TLS handshaking procedure.

Certificate analysis can be done using your browser. We will discuss methodologies for doing this in both Firefox and Chrome.

In this lab, we will also introduce Wireshark, which is a network activity analysis (and sniffing) tool. In this lab, we are going to use Wireshark to examine the SSL/TLS handshake between a client and server.

Certificate Analysis

There’s a few ways we can analyze a web certificate.

Firefox Method
Launch Firefox, go to Preferences->Advanced->View Certificates->Add Exception.

To get the certificate of the remote server, type in the URL and select Get Certificate. Select View to get a more detailed overview of the certificate’s contents.
Certificate General View for https://stackexchange.com

Alternatively, you can navigate to the desired website, click the lock icon on the navigation bar, and then View Certificate.

**Chrome Method**

To view a certificate on Chrome, you can navigate to the website, select the padlock icon on the navigation bar, and then select Certificate Information. Select Details to see all the certificate information.
Internet Tool Method
There are several websites that will allow you to lookup SSL Certs on websites (generally these tools are made available so users can check if their SSL Certificates are properly installed). One such tool can be found here: http://www.digicert.com/help/

SSL/TLS Analysis
Now that we have analyzed x509 Certificates, let’s take a look at the actual handshake protocol that allows the client and server to negotiate a session key and initiate encrypted/authenticated communication. We will accomplish this by using Wireshark to analyze the messages sent to and received from the server.
Starting Wireshark
We can initiate wireshark with the following command:

```
sudo wireshark
```

We need to run it using the sudo command because we will be actively accessing and monitoring network interfaces.

To start capturing packets, go to Capture->Interfaces (or Ctrl+I), select eth0, and click start. You should then see the capture interface, and a lot of traffic appearing in the top third of the application.

Navigate to a website that uses an SSL connection (e.g., https://google.com), and after the page loads, stop the capture.
As you can tell, there are a huge number of packets (even in such a short time period). Let's reduce them by typing the following into the Filter section ‘(ssl)’, which should filter it down to only SSL/TLS packets.

Below is a snapshot for a sample Wireshark capture to Amazon. Note, this is the Wireshark Interface for OS X, it will vary somewhat from that of Linux or Windows.

Once filtered, you can easily break down the individual packets in the SSL handshake protocol used when connecting to `https://google.com` (or any other website).

Also note that portions of the handshake will be bundled together in the same frame for efficiency.

Note: Wireshark’s breaks each packet down according to the OSI Network Layer Model, and presents it in a human readable format, the hexadecimal data of the frame, and an ASCII representation of the data. Wireshark’s usage will be covered as part of the lab.