**IoT Security and Privacy**

**Assignment 1 - Introduction to Cryto**

**(10 points)**

### Instructions:

1. Note: Blue text points to a web link. Ctrl + Click to follow link.
2. This is an individual assignment.
3. Answers to all questions must be put into **ONE** document. That is, every time, each student can only submit one report document, answering all questions of this assignment.
4. Students must put answers following each question in this assignment. The instructor will not grade a report with only answers in it and the student gets zero for such an assignment. An assignment report must include original questions.
5. Students MUST submit the finished assignment in either Microsoft Word or pdf format to Blackboard. The doc must be submitted as ONE standalone file and cannot be tarred or zipped into a container.
6. Refer to [Print screen](http://en.wikipedia.org/wiki/Print_screen) on how to take a screenshot. Pressing the Alt key in combination with PrtSc will capture the currently selected window.

**Questions (Refer to Computer Networking a Top-Down Approach 6th Edition – Chapter 8 if necessary to answer the questions below):**

**1**. What are the differences between message confidentiality and message integrity? Can you have confidentiality without integrity? Can you have integrity without confidentiality? Justify your answer. (3 points)

**2**. Suppose n = 10,000, a = 10,023, and b = 10,004. Use an identity of modular arithmetic to calculate in your head (a • b) mod n. (1 point)

**3**. Consider RSA with p = 7 and q = 17.

a. What are n and z? (1 point)

b. Let e be 5. Why is this an acceptable choice for e? (1 point)

c. Find d such that de = 1 (mod z). (1 point)

d. Encrypt the message m = 8 using the key (n, e). Let c denote the corresponding ciphertext. Show all work. (1 points)

**4**. Reliably publishing public keys is a grand challenge. One popular way is the use of certificates created by a Certificate Authority (CA) such as [DigiCert](https://www.digicert.com/). To get a certificate from a CA, a client provides its public key, identity (such as IP and/or email) and other info to the CA, which rigidly verifies all the information. The CA then provides a certificate to the client. The certificate includes the information provided by the client (denoted as *M*), other information (denoted as *M’*) including the certificate expiration date, and digital signature of *M*|*M’* (where | means concatenation). Assume CA has a public/private key pair (*eca*, *dca*). The certificate looks like this: (*M*, *M’*), *dca*(H(*M*|*M’*)). Actually when people buy computers, public keys of CAs are shipped with the operating system (e.g., Windows/Linux/MacOS/iOS/Andorid/etc.) in the format of certificate too. That is, *eca* is already in the OS and trusted.

**Question**: When a client accesses a web server, the web server sends its certificate signed with a CA’s private key *dca* to the client. Please explain how the client may verify the certificate is valid and thus gets the web serve’s public key. (2 points)

