1. Delegation is an important concept in distributed systems security. Explain what delegation means, its use in a certain environment and the problems that must be solved to implement this correctly.

2. Assume that you have to design a cryptographic protocol for a (oversimplified) command and control center for battlefields. Many decisions have to be taken in real time and information is shared among data sources/consumers of different clearance levels. State your assumptions (simplifications) and design a cryptographic protocol to keep information flowing without compromising your security policy. As security policy, you can use the one described below or come up with yours (perhaps an extension of the one below). Sample policy:
   a) There are 3 classifications for sources of information (these classifications are their security levels). The source can be unclassified, confidential or secret. Any data produced by a specific source of information receives only one classification. This means that a secret source produce only data with secret level.
   b) There are 3 classifications for consumers of information (these classifications are their clearance levels). The consumers can be unclassified, confidential or secret.
   c) Secret consumers can receive any data. However, your protocol may enable some way for restricting that to prevent a consumer from receiving data that it is not interested. This also applies for the following two policy rules.
   d) Confidential consumers can receive only confidential and unclassified data.
   e) Unclassified consumers can only receive unclassified data.

   Well define the algorithms used and key management that allows you to have different security levels of data being distributed to consumers with different clearances. State what are the reasons for your decisions, considering the possible (hardware) limitations of data sources/consumers. State if your system uses a centralized or distributed approach.

3. Design a cryptographic protocol to be used by an Internet site doing electronic commerce (retail style, what means direct with a user and not in a B2B environment). An important requirement is that the site should be able to keep a user profile (including his/her credit card number) if a user so chooses. Design your protocol to provide as much protection mechanisms as you can, particularly confidentiality, authentication, authorization, and protection against denial of service. In addition, how much privacy does your protocol provide, and how user-friendly is it?

4. (a) Informally speaking, a one-way function is a function that is easy to evaluate but hard to invert. Give a formal definition of a one-way function, and two candidate one-way functions.
(b) Although it is hard to recover an entire pre-image of a one-way function, one-way functions may not hide all partial information of its input. Give an example of a candidate one-way function which leaks some partial information and explain what is the partial information leaked.

5. 
(a) Describe the (plain) RSA digital signature scheme.
(b) In a chosen message attack against a signature scheme, an adversary attacks in two phases: In the first phase, the adversary chooses messages and obtains their signatures. In the second phase, the adversary attempts to forge a signature of a message different from those chosen in the first phase. Give a formal definition of the security of a signature scheme against a chosen message attack.
(c) Is the plain RSA signature scheme secure against chosen message attacks? If so, give a proof of security. If not, describe an attack by which adversary can successfully forges, and suggest a fix to the plain RSA signature.

6. Define a collision-free hash function and give an example of how to use it.

7. If quantum computers can be built, describe the potential consequences for at least two cryptographic systems.

8. What are the key performance metrics of intrusion detection system? Given two real-time network intrusion detection systems, how do you go about evaluating/comparing them?

9. What are the approaches for building anomaly detection model of program execution? What are the pros and cons of these approaches?