List of the Possible Questions

#1: What does “digital certificate” consist of?

#2: Who are CA’s (certificate authorities)?

#3: At a CA (certificate authority), how a digital certificate is constructed (describe it as a step-by-step procedure to construct one at a CA)?

#4: When a customer downloads a digital certificate of a merchant, how the web browser at the customer uses the digital certificate of the merchant (describe it as a step-by-step procedure at the customer’s web browser)?

#5: Which of the following information is (are) not included in a digital certificate?

(a) The identity of the owner of the certificate in text format

(b) The encrypted algorithm used to encrypt the hash digest of the certificate

(c) The public key of the owner of the certificate

(d) The public key of the CA who issued this certificate

(e) The hash algorithm used to calculate the hash digest for the certificate

(f) None of the above

#6: When a merchant requests a CA to create the merchant’s digital certificate (assume that this request is made through an electronic form over a network), how this merchant can securely send the merchant’s public key to the CA?

#7: If a merchant uses digital certificate of a CA to verify the identity of the CA (as well as the CA’s public key), the encrypted message digest in a digital certificate of a CA must be decrypted at the merchant’s local web browser. Whose public key should the merchant to use to decrypt the hash digest?

#8: Who are “root CA’s”? How are they special CA’s?

#9: Suppose that your browser downloaded a digital certificate from a web site which claims itself to be one of more than a couple hundreds of the CA’s in the world. How does your web browser confirm that the CA the browser is talking to is the CA the web browser intends to talk to (not a fake CA pretended by an attacker)?

#10: If a CA’s digital certificate is issued by another CA (i.e., a super CA), the chain of this relationship can continue. Where should the chain of this relationship be ended?
#11: Confirmations of the authenticity of digital certificate are automatic. Your web browsers try to reach a root CA for each digital certificate your browsers download. However, occasionally there are the situation when your web browser ask a question like shown below:

Regarding this message from your web browser:

(a) Why does your browser ask this question to you?

(b) What is (are) the possible security risk(s) if you accept this digital certificate (i.e., you press “OK” button)? Mention all possible security threat type(s).

#12: What are the six security services defined by ITUT-X800 standard?

#13: What does “authentication” mean?

#14: What does “authorization” mean?

#15: What does “access control” mean (from textbook)?

#16: What are the three aspects “access control” should manage (control)?

#17: What does “data integrity” mean?

#18: Which of the following four security threats does “data integrity for connection-oriented transmissions” prevent (select all that apply)?

   (a) No missing packet
   (b) All the packets are delivered in the correct order
   (c) No duplicated packet delivered
   (d) All packets are unmodified.

#19: Which of the following four security threats does “data integrity for connection-less
transmissions” prevent (select all that apply)?

(a) No missing packet  
(b) All the packets are delivered in the correct order  
(c) No duplicated packet delivered  
(d) All packets are unmodified.

#20: What does “nonrepudiation” guarantee (the three things “nonrepudiation” guarantees)?

#21: What are black-hole routers?

#22: What is “directory-traversals”? Explain how it happens.

#23: What is “SQL-injections”? Explain how it happens.

#24: What is the solution (the most primary solution) to prevent “SQL-injections”?

#25: What is “OS-command injections”? Explain how it happens.

#26: What is the solution (the most primary solution) to prevent “OS-command injections”?

#27: Explain how “Cross-Site Scripting attacks” are performed by attackers.