Wang Zhe 29184622

Question1:

What is a key distribution center(KDC)?

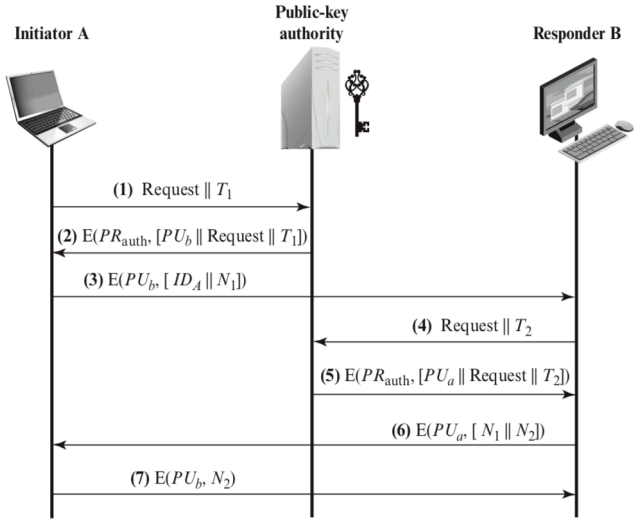
Answer：

A session key is a temporary encryption key used between two principals. A master key is a long-lasting key that is used between a key distribution center and a principal for the purpose of encoding the transmission of session keys. Typically, the master keys are distributed by noncryptographic means.

Question2:

In the public key distribution approach public key authority, how it prevent hacker altering the message before arriving the authority.

Answer:

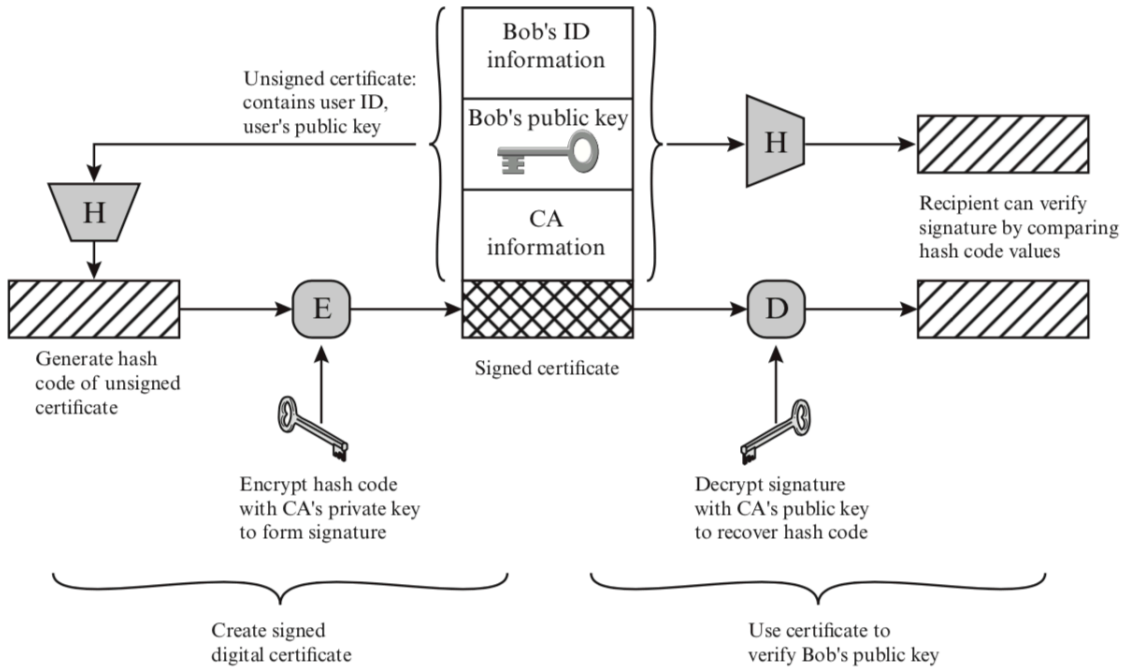


The original request used to enable A to match this response with the corresponding earlier request and to verify that the original request was not altered before reception by the authority. The original timestamp given so A can determine that this is not an old message from the authority containing a key.

Question3:

Please briefly describe how recipient verify signature in X.509 schema

Answer:



The certificate for Bob’s public key includes unique identifying information for Bob, Bob’s public key, and identifying information about the CA, plus other information as explained subsequently. This information is then signed by computing a hash value of the information and generating a digital signature using the hash value and the CA’s private key. X.509 indicates that the signature is formed by encrypting the hash value. Then recipient decrypt signature with CA's public key to recover hash code, and verify signature by comparing hash code values.

Shen Li 29184452

Question 1:

Reference the suppress-replay attack described in Section 15.2 to answer the following.

a. Give an example of an attack when a party's clock is ahead of that of the KDC.

b. Give an example of an attack when a party's clock is ahead of that of another party.

Answer:

a. An unintentionally postdated message (message with a clock time that is in the future with respect to the recipient's clock) that requests a key is sent by a client. An adversary blocks this request message from reaching the KDC. The client gets no response and thinks that an omission or performance failure has occurred. Later, when the client is off-line, the adversary replays the suppressed message from the same workstation (with the same network address) and establishes a secure connection in the client's name.

b. An unintentionally postdated message that requests a stock purchase could be suppressed and replayed later, resulting in a stock purchase when the stock price had already changed significantly.

Question 2:

There are three typical ways to use nonces as challenges. Suppose Na is a nonce generated by A, A and B share key K, and f() is a function (such as an increment). The three usages are:

Usage 1	Usage 2	Usage 3
(1) A → B: N_a	(1) A → B: E(K, N_a)	(1) A → B: E(K, N_a)
(2) B → A: E(K, N_a)	(2) B → A: N_a	(2) B → A: E(K, f(N_a))

Describe situations for which each usage is appropriate.

Answer:

All three really serve the same purpose. The difference is in the vulnerability. In Usage 1, an attacker could breach security by inflating Na and withholding an answer from B for future replay attack, a form of suppress-replay attack. The attacker could attempt to predict a plausible reply in Usage 2, but this will not succeed if the nonces are random. In both Usage 1 and 2, the messages work in either direction. That is, if N is sent in either direction, the response is E[K, N]. In Usage 3, the message is encrypted in both directions; the purpose of function f is to assure that messages 1 and 2 are not identical. Thus, Usage 3 is more secure.

Question 3:

In Kerberos, what does the Ticket contain that allows Alice and Bob to talk securely?

Answer:

It contains the session key encrypted by the KDC-Bob secret key. The client needs to

send secret information to TGS in a secure way to prove its identity.

(5)	$\mathbf{C} \rightarrow \mathbf{V}$	$Ticket_v \| Authenticator_c$
(6)	$\mathbf{V} \rightarrow \mathbf{C}$	$E(K_{c,v}, [TS_5 + 1])$ (for mutual authentication)
		$Ticket_v = E(K_v, [K_{c,v} \| ID_C \| AD_C \| ID_v \| TS_4 \| Lifetime_4])$
		$Authenticator_c = E(K_{c,v}, [ID_C \| AD_C \| TS_5])$

(c) Client/Server Authentication Exchange to obtain service

Q1 According to the following procedure, explain what each step does.

1. $A \rightarrow KDC: \quad ID_A \| ID_B$
2. $KDC \rightarrow A: \quad E(PR_{auth}, [ID_B \| PU_b])$
3. $A \rightarrow B: \quad E(PU_b, [N_a \| ID_A])$
4. $B \rightarrow KDC: \quad ID_A \| ID_B \| E(PU_{auth}, N_a)$
5. $KDC \rightarrow B: \quad E(PR_{auth}, [ID_A \| PU_a]) \| E(PU_b, E(PR_{auth}, [N_a \| K_s \| ID_B]))$
6. $B \rightarrow A: \quad E(PU_a, [E(PR_{auth}, [(N_a \| K_s \| ID_B)]) \| N_b])$
7. $A \rightarrow B: \quad E(K_s, N_b)$

1_A:
step 1, A informs the KDC of its intention to establish a secure connection with B

step 2, The KDC returns to A a copy of B's public-key certificate

step 3, A informs B of its desire to communicate and sends a nonce *Na*

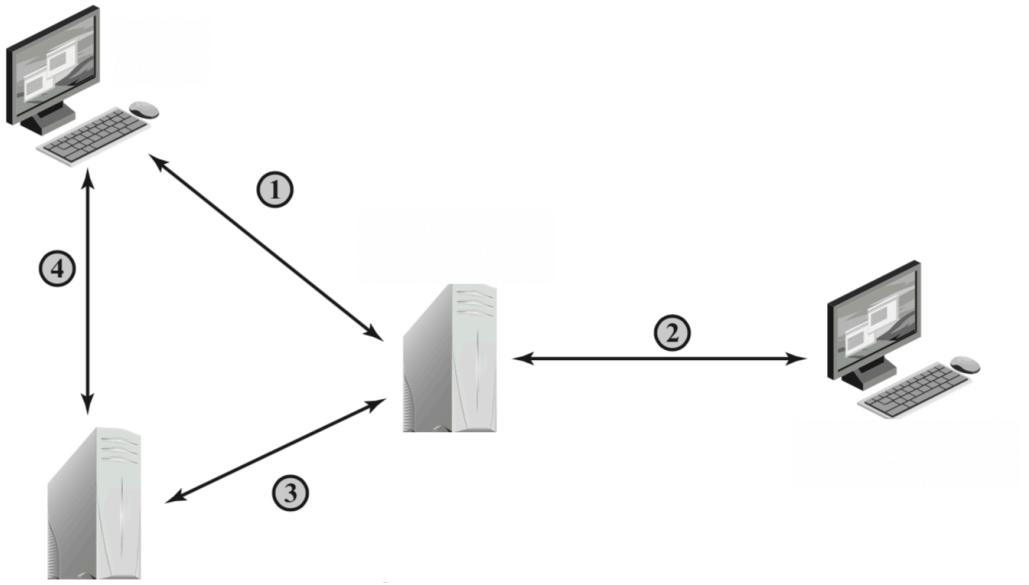
step 4, B asks the KDC for A's public-key certificate and requests a session key

step 5, The KDC returns to B a copy of A's public-key certificate, plus the information {*Na*, *Ks*, *IDB*}

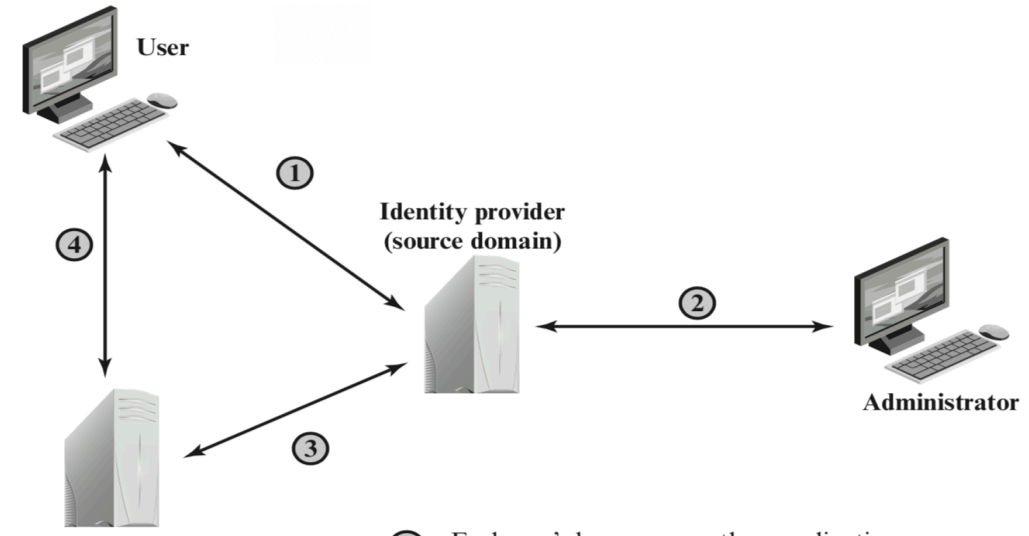
step 6, The triple {*Na*, *Ks*, *IDB*}, still encrypted with the KDC's private key, is relayed to A, together with a nonce *Nb* generated by B. All the foregoing sent to A are encrypted using A's public key

step 7, A retrieves the session key *Ks*, uses it to encrypt *Nb*, and returns it to B. This last message assures B of A's knowledge of the session key

Q2 Write the name of each device in the generic federated identity management architecture and explain the data flow.



2_A:



1. End user's browser or other application engages in an authentication dialogue with identity provider in the same domain. End user also provides attribute values associated with user's identity.
2. Some attributes associated with an identity, such as allowable roles, may be provided by an administrator in the same domain.
3. A service provider in a remote domain, which the user wishes to access, obtains identity information, authentication information, and associated attributes from the identity provider in the source domain.
4. Service provider opens session with remote user and enforces access control restrictions based on user's identity and attributes.