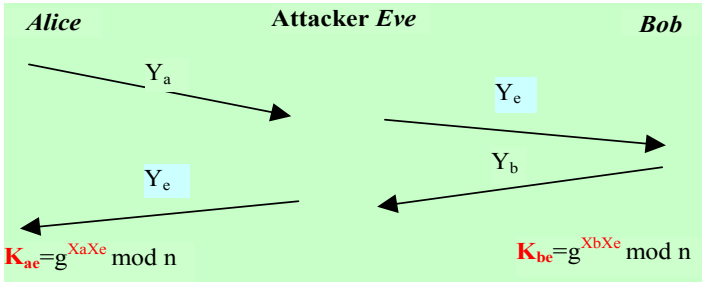
Answers to Exercise 8

(8a-i)

- Alice knows Kae, and Alice thought she shares this key with Bob. In other words, Alice thought only Bob knows Kae.
- Bob knows Kbe, and Bob thought that, apart from himself, only Alice knows this key.
- Eve knows Kae and Kbe. Eve can intercept the communication between Alice and Bob.
- Assuming that Alice is to send a confidential message, M, to Bob, so Alice sends E(Kae, M) to Bob, but this message is intercepted by Eve. Eve can decrypt this ciphertext using Kae, and then re-encrypt M using Kbe before forwarding the ciphertext to Bob.
- Bob decrypt the ciphertext thinking this is from Alice. However, what Alice and Bob do not know is that Eve has already read M.



(8a-**ii) There are three possible solutions with varying levels of security.**

The key point to thwart this attack is that we need a mechanism to ensure Alice is indeed talking to Bob (not to Eve), and Bob is indeed talking to Alice. The Ephemeral DH method can provide this assurance. The following gives two DH variants.

(1) Fixed DH

Server's and client's public key certificates contain their respective DH public key parameters. In other words, the DH public keys are certified by a CA. But there has to be a way that Alice and Bob could prove to each other that they are the rightful owners of the corresponding certificates (unless the identity of one entity is already known to the other). With this method, the established

symmetric key is not really a session-valid key. So the security level is not as high as the second method.

(2) Ephemeral DH

DH public keys are exchanged, signed using the sender's private DSA key; the receiver uses the corresponding public keys (certified in the certificates) to verify the signatures. The certificates are used to certify the DSA keys, and DH shares are signed using DSA by the respective signers. This method is more secure than the other two DH variants (anonymous DH and Fixed DH); it is used to create a temporary, one-time session key. This is the so-called DH-DSA method.

Anonymous DH – what is shown in the picture in the slide, i.e. the base DH algorithm is used, with no authentication.

(3) This is mentioned here for completeness; it is not in the domain of this question. Session key transportation using RSA encryption

With this method, one of the entities (say Alice) generate a random secret session key, and use the other entity's public key to transport the session key. The public key should be certified by a CA. If Bob is not already known to Alice, Bob should demonstrate that he is the rightful owner of the certificate by proving the knowledge of the secret key corresponding to the public key certified.

(8c-**i) Benefits:**

- Reduced involvement of KDC, thus less overhead not just for KDC also for B;
- This protocol (designed in this way) can also be used for authentication purpose, in addition to confidential communication between A and B.

(8c-ii). Two application areas: one is establishing a secure communication channel; and the other is for authentication service.

Benefits:

1. Party A (i.e. user) only need to remember a single key while still being able to use a different key for a different correspondent.
2. When used for authentication, the protocol supports single sign-on, i.e. a user only needs to remember a single password, but be able to use different session secrets for different servers/services and also the

users’ master secrets (i.e. long-term passwords) are only managed by one entity, i.e. KDC.