Assignment - 2

Dipika Tanwar (MCS202456)
Topic : Secret Key Sharing Protocol

Assumptions :

1. The Public key of all parties is published and known to other parties.
2. Charlie public key is trusted.
3. Charlie has way to verify public keys.
4. Alice and Bob public key's need authentication.
5. Use AES (with block size = 128 bits) for encryption using a secret key.
6. Use RSA (with n = 1024) for encryption using a public/private key.

Condition-1: Assume that there is a trusted third party, Charlie. In other words, Charlie is trusted by both Alice and Bob. Under these conditions, design a key agreement protocol to enable Alice and Bob to share a secret key Ka,b.

Solution for Condition 1 :
Alice: User who wants to initiate communication.
Bob: User to whom Alice wants to communicate.
Charlie: Trusted Third Party acts like a certificate Authority.

Protocol :

1. Alice requests Charlie **encrypting** the request message with Charlie's public key and asks to verify his and Bob's public keys and provide him the digital certificates for Bob and share his digital certificate with Bob for requested period of time.
2. Charlie **decrypts** request of Alice and verifies Alice and Bob.
    a. Generates digital certificates for both
    b. **Encrypts** the certificates with his private key.
    c. Send Alice certificate to Bob and Bob certificate to Alice.
3. Alice verifies Bob's certificate sent by Charlie by **decrypting** with Charlie's public key and request to share a secret key (Ka,b).
4. Bob verifies Alice's certificate sent by Charlie by **decrypting** with Charlie's public key and request to share a secret key (Ka,b).
5. Charlie generates the secret key Ka,b.
    a. **Encrypts** it with Alice's public key and send to Alice.
    b. **Encrypts** it with Bob's public key and send it to Bob.
6. Alice **decrypts** the key shared by Charlie with his private key..
7. Bob **decrypts** the shared key with its private key.

Security :

1. **Man in middle** :The protocol is safe as Bob and Alice are authenticated by Charlie. Hence, it is known that they are the genuine owners of their public keys.
2. **Eavesdropping :** This protocol is safe from as Messages transmitted are always transmitted by encrypting with the verified public key of the receiver.
    a. Security is ensure by encrypting the request message.
    b. Digital certificate is encrypted to hide the validation period from eavesdropper.
3. **Replay attacks** : The protocol is safe from as if messages are received later than valid period of communication then, Bob can ask for digital certificate again as earlier digital certificate doesnot hold valid and Intruder cannot show Alice digital certificate.

Computational efficiency:

1. Encryption / Decryption : The complete protocol requires to encrypt and decrypt with asymmetric keys to be performed **three times (RSA).**
    a. There exists a tradeoff for hiding the request packets as it has information about with whom communication needs to be done and for how much time. For greater security reason more encryption steps are added.

Total Number of operations : 3 imes encrypt + 3 times decrypt = 6 operations.

Communicational efficiency :

Bits transmitted :

1. Request message : Has communicating entities information and Time period.
    a. Assuming the 2 userIds and time period bits as : as 32 bits each.
    b. Total : 3*32
2. Digital certificate : Can be made with single bit field indicating the other user public certificate validity. With limiting two two users. User Id can be excluded. And validation period can be defined by 32 bits :
    a. Total - 32+1
3. Sharing secret to both Alice and Bob and encrypted with RSA so,
    a. Total bits : 2*1024

Total Acutal Transmitted bits : 2177 bits.

Condition-2: Assume that there is an honest-but-curious third party, Charlie. In other words, Alice and Bob do not want to reveal their secret key to Charlie. However, Alice and Charlie share a secret key Ka,c, and Bob and Charlie share a secret key Kb,c. Under these conditions, design a key agreement protocol to enable Alice and Bob to share a secret key Ka,b.

Solution for Condition 2:

Alice: User who wants to initiate communication.
Bob: User to whom Alice wants to communicate.
Charlie: Trusted Third Party acts like a key distribution system.

Protocol :

1. Alice generates a secret key Ka,b and append time stamp ts.
    a. **Encrypts** Ka,b with public key of Bob say, Kb.
        i. E(Kb , Ka,b || ts)
    b. Creates Request packet with receiver information and encrypted key.
        i. R[“Bob”, E(Kb , Ka,b || ts)] .
    c. **Encrypts** request packet with secret key of Alice and Charlie’s.
        i. E(Ka,c, R[“Bob”, E(Kb , Ka,b || ts)])
    d. Sends encrypted request packet to Charlie.

2. Charlie receives E(Ka,c, R[“Bob”, E(Kb , Ka,b || ts)])
    a. **Decripts** with its secreat key Ka,c .
        i. R[“Bob”, E(Kb , Ka,b || ts)] .
    b. Sees receiver as Bob.
    c. **Encrypts** the message packet with secret key shared with Bob , Kb,c
        i. E(Kb,c, R[“Bob”, E(Kb , Ka,b || ts)])

3. Bob receives packet from Charlie E(Kb,c, R[“Bob”, E(Kb , Ka,b || ts)]).
    a. **Decrypts** with its secret key Kb,c to R[“Bob”, E(Kb , Ka,b || ts)]
    b. **Decrypts** message from Request packet by its private key to (Ka,b || ts)
    c. It verifies the times tamp to validate for the possibility of replay attack.

Security :

1. **Man in middle attack** not possible as secret keys are used beween Alice and Charlie Ka,c and Charlie and Bob, Kb,c.
2. **Eavesdropping** : Messages are always transmitted in encrypted formats.
3. **Replay Attacks**: Time stamp is incorporated by Alice to prevent the replay attack.

Computational efficiency:

1. Encryption/Decryption :
    a. One time Asymmetric encryption (RSA).
    Two time symmetric encryption (AES).

    Optimization arguments:

    1. Encryption with Alice public key can be skipped as authentication is provided by Ka,c.
    2. Bob public key encryption needs to be incorporated so, Charlie cannot know the secret key Ka,b.

Total operations : 3 operation of encrypt + 3 operations of decrypt + 1 operation time stamp append.
Total operations : 7.

Communicational efficiency :

1. Request packet : Assume secret key length as 256 bits., time stamp 32 bits and encrypted with 1024 bits.

a. Alice sends :Encrypted message as encryption done by RSA - total : 1024.Than again encryption with key length 256.;
    i. Total length remains : 1024.
b. Charlie send : by encrypting RSA so, total bits : 1024.
c. B doesnot send after receiving packet .

Total bits send : 2048 bits