3 Attack On Public-Key Cryptography

3.1 Information that can be obtained by Eve

Considering RSA, I will generalise it to a public-key cryptography.The general idea of a public-key cryptography is that a message encrypted with the public key P can only be decrypted with the private key K.

In RSA, the public keys are n and e. If eve changes these 2 keys, she can decrypt all the messages sent between Alice and Bob.Here n is $p * q$ where p and q are two prime numbers and e is a coprime number to totient(n). totient(n) = (p-1)(q-1).

Consider Alice and Bob have their own seperate private keys and their own public keys. Their public keys are placed on a file in a server for which Eve has an access to that file. So potentially Eve has Alice and Bob's public keys.

The normal flow will be that if Alice has to send a message to Bob, Alice will encrypt the plain text using her public key and send it to Bob which Bob will decrypt it using his private key. Now Consider Eve is in the middle of this and has access to both their public keys.

1) Eve replaces Alice and Bob's public keys with two other public keys. So now she has altered the file to have her own public keys.

2) Alice now wants to send a message to Bob and she encrypts the plain text with the public key present in the file.(She thinks that this is the public key of Bob, but indeed it is the public key of Eve.)

3) Now Eve can decrypt the message using her private key and hence she can have access to the messages that are sent between Alice and Bob. This kind of a attack is called as the Man in the Middle Attack.

4)Furthermore, Eve can send forged/fake messages to Bob as Eve has Bob's public key as well.

3.2 Detecting Eve's subversion of the keys

The question here is can Alice and Bob find that Eve has changed their public keys. Alice sends a message using Eve's public key unknowingly as stated earlier. Eve can decrypt the message and she can send her own message to Bob with Bob's public key. In this case, Bob will not get a garbage value while decrypting the message and hence Alice and Bob cannot find that Eve is doing a Man in the Middle attack.

Also there is also another important factor called as the **Public Key Certificate.**A public key certificate is a digitally signed document that serves to validate the sender's authorization and name. The document consists of a specially formatted block of data that contains the name of the certificate holder (which may be either a user or a system name) and the holder's public key, as well as the digital signature of a certification authority for authentication. Using this factor, both Alice and Bob can find if Eve is in the middle.

But if Eve unknowingly sent the same message that was sent by Alice, to Bob, then while decrypting Bob will get a garbage value because the message sent by Alice was encrypted with Eve's public key and not Bob's.

For a clearer version, I would like to give an example.
Consider X and Y are the public keys of Alice and Bob respectively.
Eve replaces U and V instead of X and Y in the file.
Alice sends a message M using the public key of Bob(which is corrupted now, it should have been Y but it is corrupted as V).
Eve can easily decrypt the message M using her private key and can send a forged message F to Bob with Bob's public key Y.In this case,both Alice and Bob cannot find that Eve has changed their public Keys.
Instead of sending the forged message F, if Eve sends the message M that was encrypted with the public Key V, then Bob cannot decrypt with his private key or he will get some garbage value because of a different public-private key pair.In such a case, Bob and Alice can find that their public keys were altered by someone.

Example

- Alice Public Key = X and Bob Public Key = Y
- Eve replaces x by u and y by v
- Alice's Corrupted Public key = U and Bob's Corrupted Public key = V
- Alice encrypts a message M = "My Account Number 12345" using V to output ciphertext C.
- C is received by Eve and not Bob because M was encrypted with V and not with Y.