CS338 Spring 2022
Antonia Ritter
Conferred with McKenna Wirth

CRYPTOGRAPHIC SCENARIOS

1.

Alice wants to send Bob a long message, and she doesn't want Eve to be able to read it. Assume for this scenario that PITM is impossible.

This is what the Diffie-Hellman key exchange algorithm was designed for. Alice and Bob can use it to agree on a secret key K which Eve isn't able to determine. Then Alice can send Bob the long message M by sending $C = AES(K, M)$. Eve can't easily decrypt this without K, but Bob can read it using $AES_D(K, C)$.

2.

Alice wants to send Bob a long message. She doesn't want Mal to be able to modify the message without Bob detecting the change.

Assuming Alice doesn't care if Mal can read the message M, she can send the message without encrypting it. If she wants to encrypt it they can use Diffie-Hellman and the following can proceed using the encrypted message.

To ensure that Bob can detect if Mal modifies it, Alice also sends

$$check = E(S_A, H(M)).$$

Bob can verify the message by decrypting using Alice's public key, and making sure that matches the hash of the version of the message he received:

$$E(P_A, check) \stackrel{?}{=} H(M).$$

To change the message M to M' and send Bob a believable *check* value, Mal would have to re-encrypt $H(M')$ using Alice's private key, which they do not have (and they could not pass their own off as Alice's, since Bob already has Alice's public key.)

3.

Alice wants to send Bob a long message, she doesn't want Eve to be able to read it, and she wants Bob to have confidence that it was Alice who sent the message. Assume for this scenario that PITM is impossible.

Since PITM is impossible Alice and Bob can use Diffie-Hellman to agree on a common key K. Alice encrypts the message M using K to obtain $C = AES(K, M)$.

Let Alice's signature be

$$Sig = E(S_A, H(M)).$$

Then Alice sends $(C||Sig)$ to Bob. He can hash the decrypted message and decrypt the signature using Alice's public key (which he already has), which should be the same if Alice sent the message:

$$E(P_A, Sig) \stackrel{?}{=} H(M).$$

No one else can sign this message without Alice's private key.

4.

Consider scenario #3 above. Suppose Bob sues Alice for breach of contract and presents as evidence: the digitally signed contract $(C||Sig)$ and Alice's public key P_A. Suppose Alice says in court "C is not the contract I sent to Bob". Alice will now need to explain to the court what she believes happened that enabled Bob to end up with an erroneous contract. List at least three things Alice could claim happened. For each of Alice's claims, state briefly how plausible you would find the claim if you were the judge.

(Assuming that is actually Alice's key and the signature shows that particular C was signed using Alice's key.)

Claim #1: Someone hacked my computer and got my private key.

Plausibility: This is fairly plausible, especially if Alice can show evidence of a security breach. However, this still might be Alice's fault.

Claim #2: That's not the contract I sent, that's a different contract that has the same hash.

Plausibility: Very unlikely. Finding something that generates the same hash is hard enough, never mind finding something that generates the same hash and is the text of a contract.

Claim #3: Someone (maybe Bob) cracked RSA and figured out my private key, and rewrote and re-signed the contract.

Plausibility: Not plausible, unless this is something really important and the people involved have a lot of computing power.

5.

For this scenario, suppose the assumption that everybody has everybody else's correct public keys is no longer true. Instead, suppose we now have a certificate authority CA, and that everybody has the correct P_{CA} (i.e. the certificate authority's key). Suppose further that Bob sent his public key P_B to CA, and that CA then delivered to Bob this certificate:

$$Cert_B = \text{“}bob.com\text{”}||P_B||Sig_{CA}$$

In terms of P_{CA}, S_{CA}, H, E, etc., of what would Sig_{CA} consist? That is, show the formula CA would use to compute Sig_{CA}.

The signature would be

$$Sig_{CA} = E(S_{CA}, H(\text{“}bob.com\text{”}||P_B)).$$

Then anyone can verify Bob's certificate by decrypting Sig_{CA} with P_{CA} and comparing that to the hash of the TBS portion of the certificate.

6.

Bob now has the certificate $Cert_B$ from the previous question. During a communication, Bob sends Alice $Cert_B$. Is that enough for Alice to believe she's talking to Bob? (Hint: no.) What could Alice and Bob do to convince Alice that Bob has the S_B that goes with the P_B in $Cert_B$?

First Alice and Bob agree on a common key K using Diffie Hellman. Alice also sends Bob a random number R. Bob sends Alice $Cert_B$ and an encrypted hash of K and R:

$$check = E(S_B, H(K||R)).$$

Alice can check this by hashing $K||R$ herself and comparing the result to $E(P_B, check)$. If they are the same, she knows only Bob could have encrypted *check* because of the $Cert_B$. Now Alice knows she is talking to Bob and they can proceed with AES using that K.

7.

Finally, list at least two ways this certificate-based trust system could be subverted, allowing Mal to convince Alice that Mal is Bob.

- The certificate authority messed up and gave Mal a certificate saying they're Bob.
- Mal got a certificate for "therealbob.com" and Alice fell for it.
- Mal got Bob's secret key by hacking his computer (or threatening him with a wrench).
- Mal got Bob's secret key by using a lot of computing power to factor the modulus.
- Mal hacked Alice and modified the software used to do the check in #6.