Sean Lee and Peter McCrea
Cs 231

Scenarios

1. Alice wants to send Bob a long message, and she doesn't want Eve to be able to read it. (I say "Eve" here because I want you to assume for this scenario that person-in-the-middle is impossible, and give an answer that is as simple as possible under that assumption.)

Alice and Bob use Diffie-Hellman to agree on a shared secret key K. To send the long message, they then use the shared key K in a symmetric encryption algorithm (e.g. AES). This would look like: Alice sends Bob $S_K(M)$. Bob can then decrypt this with the shared key K doing: $S_K^{-1}(S_K(M)) = M$, ending up with the original message.

This scenario would work with the Diffie-Hellman since Eve, not being the person-in-the-middle, would not be able to listen in and brute force the shared key agreed upon using Diffie-Hellman, and therefore could not decrypt the message that was encrypted using that same sharek key in a a symmetric encryption algorithm.

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

For each element needed to be transferred for Diffie Helman, Alice can send ciphertext $C = E(P_B, M)$ and Bob can send ciphertext $C = E(P_A, M)$. Alice can compute $E(S_A, C) = E(S_A, E(P_A, M)) = M$. Bob can compute $E(S_B, C) = E(S_B, E(P_B, M)) = M$. This will allow them to negotiate a shared key, K that can then be used for in a symmetric encryption algorithm. This would look like: Alice sends Bob $S_K(M)$. Bob can then decrypt this with the shared key K doing: $S_K^{-1}(S_K(M)) = M$, ending up with the original message.

This plan would work because assuming that Mal is the person-in-the-middle, she wouldn't be able to decrypt the elements shared between Alice and Bob for each element in the Diffie Hellman exchange because Mal does not know the secret keys of Alice or Bob. Then, similar to scenario 1, Mal would also be unable to decrypt the symmetric encryption used to encrypt the actual message, and Mal also wouldn't be able to change the message without having the 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. (Again, don't worry about Mal and person-in-the-middle here.)

Alice and Bob use Diffie-Hellman to agree on a shared secret key K. To send the long message, they then use the shared key K in a symmetric encryption algorithm (e.g. AES). At the end of this message, Alice will concatenate a digest D which is signed by her private key. Bob can decrypt this digest using Alice's public key. This would look like D = H(M) where H is some hash function. The signature will be in the form Sig = $E(S_A,D)$. Alice will then send the message M, encrypted using AES and the shared key K obtained from Diffie Helman concatenated with this signature (M||Sig). Bob will decrypt the message using the shared key K and hash the message. Finally, Bob can decrypt Alice's digital signature using $E(P_A, Sig) = D$. If Bob is able to decrypt Alice's signature with her public key, this is proof that Alice shared the message.

This scenario would work with the Diffie-Hellman since Eve, not being the person-in-the-middle, would not be able to listen in and brute force their way through due to the one way functions and thus would not be able to find the shared key. And using digital signatures, Bob can be assured that the message is from Alice since after decrypting Alice's message and Alice's digest using her public key, Bob knows that only Alice's private key could have generated that digest which matches the message.

4. Alice wants to send Bob a long message (in this case, it's a contract between AliceCom and BobCom). She doesn't want Eve to be able to read it. She wants Bob to have confidence that it was Alice who sent the message. She doesn't want Bob to be able to change the document and claim successfully in court that the changed version was the real version. And finally, Bob doesn't want Alice to be able to say in court that she never sent the contract in the first place.

If we care about stopping a person in the middle attack, For each element needed to be transferred for Diffie Helman, Alice can send ciphertext $C = E(P_B, M)$ and Bob can send ciphertext $C = E(P_A, M)$. Alice can compute $E(S_A, C) = E(S_A, E(P_A, M)) = M$. Bob can compute $E(S_B, C) = E(S_B, E(P_B, M)) = M$. This will allow them to negotiate a shared key, K that can then be used for a Message Authentication Code. Alice will end up also using this key to encrypt the message sent in the MAC. This would look like: First Alice computes $S_K(M)$, which we will call M'. C will be computed by Alice using C = MAC(K, M'). Finally, Alice can encrypt C using her private key, resulting in C'. $C' = E(P_A, M')$. Bob can decrypt this using Alice's public key.

This plan would work because assuming that Mal is the person-in-the-middle, she wouldn't be able to decrypt the elements shared between Alice and Bob for each element in the Diffie Hellman exchange because Mal does not know the secret keys of Alice or Bob. Then, by using a symmetric encryption algorithm to encrypt the message, she can use MAC to compute C which ensures authenticity such that Bob won't be able to change the document. Then by encrypting C using her own private key to obtain C', she can send this over to Bob in which he can decrypt using Alice's public key ensuring that Alice sent the contract in the first place. If Alice claims that she never sent the contract, Bob can point out that only Alice's secret key could have generated something that can be decrypted with Alice's public key, proving Alice sent the contract in the first place.