Daniel Garcia

ECE 548

Homework 2

2.1 Suppose that someone suggests the following way to confirm that the two of you are both in possession of the same secret key. You create a random bit string the length of the key, XOR it with the key, and send the result over the channel. Your partner XORs the incoming block with the key (which should be the same as your key) and sends it back. You check, and if what you receive is your original random string, you have verified that your partner has the same secret key, yet neither of you has ever transmitted the key. Is there a flaw in this scheme?

Yes, mainly due to the nature of XOR. Since XOR is a reversible system, with any two pieces of information out of three, you can construct the third. For example, if A XOR B = C, and A XOR C = B, then B XOR C = A. In our scenario, let’s call the secret key A, B the random bit string the length of the key, and the resulting ciphertext C. In this scenario, C is visible over the wire and the partner computes A XOR C = B and sends B over the wire. Even though A has never been sent, the attacker can simply XOR C and B which will give them A. So, this system is extremely vulnerable. Something as simple as hash functions would make this scenario much more secure.

2.5 In this problem we will compare the security services that are provided by digital signatures (DS) and message authentication codes (MAC). We assume that Oscar is able to observe all messages sent from Alice to Bob and vice versa. Oscar has no knowledge of any keys but the public one in case of DS. State whether and how (i) DS and (ii) MAC protect against each attack. The value auth(x) is computed with a DS or a MAC algorithm, respectively.

1. (Message integrity) Alice sends a message x = “Transfer $1000 to Mark” in the clear and also sends auth(x) to Bob. Oscar intercepts the message and replaces “Mark” with “Oscar.” Will Bob detect this?
2. (Replay) Alice sends a message x = “Transfer $1000 to Oscar” in clear and also sends auth(x) to Bob. Oscar observes the message and signature and sends them 100 times to Bob. Will bob detect this?
3. (Sender authentication with cheating third party) Oscar claims that he sent some message x with a valid auth(x) to Bob but Alice claims the same. Can Bob clear the question in either case?
4. (Authentication with Bob cheating) Bob claims that he received a message x with a valid signature auth(x) from Alice (e.g., “Transfer $1000 from Alice to Bob”) but Alice claims she has never sent it. Can Alice clear this question in either case?

a. Yes, Bob will detect this since the hash function at the end of the message will not match the result of the hash when Bob attempts to hash it therefore Bob will not trust the integrity of the message.

b. Yes, Bob will detect this because the MAC algorithm usually implements a sequence number or timestamp. This blocks any attempts to use the same message in a replay attack since Bob will know that the message is invalid as the sequence number or timestamp has not been changed.

c. No, Since the MAC only protects the integrity of the message itself and is not an authentication mechanism for the sender, there is no way for Bob to clarify who sent a message at any given time unless he has another method to authenticate senders. If instead of MAC the system incorporated digital signature, it would be impossible for a third party to claim such a message since it would have to have been signed by the users private key and there for would not be forgable.

d. No, Since the MAC cannot prove authentication of senders, there is no way for Alice to clear her name. If instead of MAC the system incorporated digital signature, it would be impossible for a third party to claim such a message since it would have to have been signed by the users private key and there for would not be forgable.