Exercise 9.8 #5

Needham and Schroeder suggest the following variant of their protocol:

1. Alice → Bob : Alice

2. Bob → Alice : { Alice, $rand_3$ } k_{Bob}

3. Alice → Cathy : { Alice, Bob, $rand_1$, { Alice, $rand_3$ } k_{Bob} }

4. Cathy → Alice : { Alice, Bob, $rand_1$, $k_{session}$, {Alice, $rand_3$, $k_{session}$} k_{Bob} } k_{Alice}

5. Alice → Bob : { Alice, $rand_3$, $k_{session}$ } k_{Bob}

6. Bob → Alice : { $rand_2$ } $k_{session}$

7. Alice → Bob : { $rand_2 - 1$ }$k_{session}$

Show that this protocol solves the problem of replay as a result of stolen session keys.

The original Needham-Scheroeder protocol can be subverted with a stolen session key as follows. If in step 3 of the original protocol, Eve replays an old message with a compromised session key, she can intercept the next message, decrypt rand2 using the compromised key and send back rand2-1 to Bob. Therefore, she can deceive Bob thinking he is talking to Alice, while he is really talking to Eve.

In the proposed variant, Eve will replay message 5 to Bob. If Bob does not have an ongoing session with Alice he will discard the message. If he has received message 1 from Alice before, he simply compares rand3 in message 5 with rand3 that he has sent in message 2. Since rand3 is a nonce, if they are different the message is definitely a replay (nonce can be only used once).

Exercise 9.8 #6

Consider an RSA digital signature scheme (see Section 9.5.2). Alice tricks Bob into signing messages m_1 and m_2 such that $m = m_1m_2$ mod n_{Bob}. Prove that Alice can forge Bob's signature on m.

Given, $m = m_1 \times m_2 \bmod n_{\mathrm{Bob}}$

Bob's Digital Signature on m_1 and m_2

$$c_1 = m_1^{\mathrm{dBob}} \bmod n_{\mathrm{Bob}}$$
$$c_2 = m_2^{\mathrm{dBob}} \bmod n_{\mathrm{Bob}}$$

Bob's Digital Signature on m

$$c = m^{\mathrm{dBob}} \bmod n_{\mathrm{Bob}}$$

Since Alice has c_1 and c_2, she can construct c from them as follows. (note n_{Bob} is publicly known)

$$= [c_1 \times c_2] \bmod n_{\mathrm{Bob}}$$
$$= [(m_1^{\mathrm{dBob}} \bmod n_{\mathrm{Bob}}) \times (m_2^{\mathrm{dBob}} \bmod n_{\mathrm{Bob}})] \bmod n_{\mathrm{Bob}}$$
$$= (m_1^{\mathrm{dBob}} \times m_2^{\mathrm{dBob}}) \bmod n_{\mathrm{Bob}}$$
$$= (m_1 \times m_2)^{\mathrm{dBob}} \bmod n_{\mathrm{Bob}}$$
$$= m^{\mathrm{dBob}} \bmod n_{\mathrm{Bob}}$$

Thus, the forgery is possible.

Exercise 9.8 #7

Return to the example on page 140. Bob and Alice agree to sign the contract G (06). This time, Alice signs the message first and then enciphers the result. Show that the attack Bob used when Alice enciphered the message and then signed it will now fail.

In the example of page 140, Alice first enciphers the message:
$c_1 = m^{eBob} \bmod n_{\text{Bob}}$
Then she signs it and sends it to Bob:
$c_2 = c_1^{dAlice} \bmod n_{\text{Alice}}$
Since c_1 is known to Bob and he has control over his keys, he can modify his key pair to (d'_{Bob}, e'_{Bob}) such that for a different message f: $c_1 = m^{eBob} \bmod n_{\text{Bob}} = f^{e'Bob} \bmod n_{\text{Bob}}$. Therefore, he can claim Alice has signed *f*.

However, this time, Alice first signs and then enciphers it:
$c_1 = m^{dAlice} \bmod n_{\text{Alice}}$
$c_2 = c_1^{eBob} \bmod n_{\text{Bob}}$
Since Bob does not know Alice's private key, and also cannot change it, he cannot make a claim that c_1 is a signature of any other contract than the original.

Alternatively, you could do the same computations as before, and show that this time Bob's attack fails.

Exercise 11.9 #2
[sample solution, Lyndsi Hughes]

P >= TG/N
T <= PN/G
P = .1
G = 10,000 guess per second

a. $N = 127^8$
   $T \le .1\,(127^8) / 10000$
   $T \le 6.767 \times 10^{11}$ seconds
   T <= 21,459.676 years

b. $N = (26+26+10)^8 = 62^8$
   $T \le .1\,(62^8) / 10000$
   T <= 2,183,401,056 seconds
   T <= 69.235 years

c. $N = 10^8$
   $T \le .1\,(10^8) / 10000$
   T <= 1000 seconds
   T <= 16.667 minutes