CS3235 Tutorial for week 10
(Oct 13th-Oct 17th, 2003)

10th October 2003

1. **Question 7, Chapter 10 in textbook:** Show that the Needham-Schroeder protocol solves the problem of replay attacks in the case of stolen keys.

2. In the coin-tossing example described in class, Alice chooses two large prime numbers p and q and then sends $n = pq$ to Bob. Bob chooses an x and then sends $y = x^2 \bmod n$ back to Alice who then... can easily calculate the four square roots of y. Why is this easy for Alice? How does she do it?

3. The voting protocol described in the notes has a serious drawback that precludes it from being used for (say) the Singapore election. What is this drawback?

4. Design, or discover a contract signing protocol, which uses a third party to oversee the contract.

5. In the proof that the final step of RSA ($c^D \bmod N$) calculates the original message m, we had

$$\begin{aligned} c^D \bmod N &= m^{ED} \bmod N \\ &= m^{k(P-1)(Q-1)+1} \bmod PQ \\ &= m * m^{k(P-1)(Q-1)} \bmod PQ \end{aligned}$$

   and then we noted that:

   $m^{P-1} \bmod P = 1$, so $(m^{(P-1)})^{k(Q-1)} \bmod P = 1$

   $m^{Q-1} \bmod Q = 1$, and so (tutorial) $(m^{(P-1)})^{k(Q-1)} \bmod PQ = 1$.

   Why is it that given $m^{P-1} \bmod P = 1$ and $m^{Q-1} \bmod Q = 1$ then $(m^{(P-1)})^{k(Q-1)} \bmod PQ = 1$?

6. Describe how a digital signature can be appended to a (plaintext) message, and how such a digital signature will be trusted. (For example - couldn't someone just change the message and then re-create a new changed signature?)

7. Alice and Bob are using Diffie-Hellman key exchange, and agree on the intital values $p = 1637$, and $g = 331$. Alice chooses a secret key $a = 1433$, and Bob chooses a secret key $b = 977$.

   (a) What value does Alice send to Bob?

   (b) What value does Bob send to Alice?

   (c) What is the shared key?

   (d) Harry-the-hacker has a difficult problem (to calculate $g^{ab} \bmod p$), in normal Diffie-Hellman key exchange, but not here. Why not?