Massachusetts Institute of Technology
6.915: Network and Computer Security
Professor Ronald L. Rivest

Handout 32
November 21, 1995

Problem Set 9

This problem set is due on *Thursday, November 30* at the end of class.

Each solution is to be handed in *separately*, i.e. on separate sheets of three-hole punched paper. If the solution to a single problem takes more than one page, staple together only the pages relative to the same problem. Mark the top of each sheet with your name, 6.915, the problem number, and the date.

If you collaborate with other students, you MUST write up solutions on your own and acknowledge the people you work with.

Problem 9-1. Atomicity of withdrawal protocol

Recall the protocol that allows a User to withdraw a coin of \$1 from the Bank . Let $(n, 3)$ be the RSA public key of the Bank .

1. The User prepares 100 messages $m_1, \ldots, m_{100}$ which are all \$1 coins. The User *blinds* them, that is she chooses at random $r_1, \ldots, r_{100}$ and computes $w_i = r_i^3 m_i$. The User sends $w_1, \ldots, w_{100}$ to the Bank .
2. The Bank chooses at random 99 of the blindings and asks the User to open them. That the Bank chooses $i_1, \ldots, i_{99}$ and sends it to the User .
3. The User opens the required blindings by revealing $r_{i_1}, \ldots, r_{i_{99}}$.
4. The Bank checks that the blindings are constructed correctly and then finally signs the unopened blinding. W.l.o.g. assume this to be the first one. So the Bank signs w_1 by sending to the User $w_1^{\frac{1}{3}} = r_1 m_1^{\frac{1}{3}}$
5. The User divides this signature by r_1 and gets a signature on m_1 which is a valid coin.

Notice that the User has a probability of 1/100 to succesfully cheat.

Suppose now that the protocol is not atomic. That is the communication line may go down at the end of each step between the Bank and the User . What protocol should be followed for each step if the line goes down at the end of that step in order to prevent abuse or fraud by either party?

Problem 9-2. Blinding with ElGamal

In class we saw a way to blind messages for signatures using RSA. In this problem we ask you to construct blind signatures for a variation of the ElGamal signature scheme.

The ElGamal signature we will consider is as follows. Let p ve a large prime, g a generator, x the secret key of the Bank and $y = g^x$ the corresponding public key. Let H be a collision-free hash function (as MD5 for example).

When the Bank wants to sign a message m she computes

$$a = g^k \bmod p$$

for a random k and

$$c = H(m, a)$$

and finally

$$b = kc + xa \bmod (p-1)$$

The signature of the meesgae m is $sig(m) = (a, b)$. Given the triple (m, a, b) the verification is performed by computing $c = H(m, a)$ and checking that

$$g^b = a^c y^a$$

So the withdrawal protocol could be as following:

1. The User tells the bank she wants a \$1 coin.
2. The Bank replies with 100 values $a_i = g^{k_i}$ for random k_i.
3. The User sends back $c_i = H(m_i, a_i)$ where m_i are all \$1 coins.
4. The Bank asks the user to open 99 of those.
5. The User reveals 99 of the m_i's.
6. The Bank replies with $b_i = k_i c_i + x a_i \bmod (p-1)$ for the unopened index i

However this is not anonymous since the Bank can recognize the User when they come back. In order to make them really anonymous, the User has to change the value of "challenge" c_i computed at step 3. This modification will allow him to compute a different signature on m_i on her own which will not be recognizable to the Bank when the coin comes back. A possible way to do this is for the User to "behave" as if the a_i's are equal to $g^{k_i+\alpha_i}$ where the α_i's are values that she knows.

During the protocol the Bank will check as usual that this modification has been performed correctly by asking the User to open 99 random blindings.