6.915 Computer and Network Security	Fall Term, 1995
Lecture 20 : November 7, 1995	
Lecturer: Ron Rivest	*Scribe: Arthur Housinger*

1 Overview

Last lecture, we were given a system for anonymous digital cash. However, it had a lot of overhead and was generally heavy-duty. Today, a more compact and simple method will be presented. The same material can be found in Handout 31.

2 Background Math

Define q to be prime, such that we can also define p to be prime and

$$p = 2q + 1^1$$

For example, $q = 5$ and $p = 11$. Continuing with this example, we also have

$$Z_{11}^* = \{2, 4, 8, 5, 10, 9, 7, 3, 6, 1\}$$

¿From here, we can easily find a sub-group of size q by choosing any element from Z_{11}^* (with the exception of 1) and using it as a generator. So, now we have

$$G_q = \{4, 5, 9, 3, 1\} = \{g^x : 0 \leq x \leq q - 1\}$$

where g is and element of G_q, such as 4.

Finally, note that solving $y = g^x$ for x given g and y is a Discrete Log Problem (DLP), which is hard.

3 The Setup

Let g_1 and g_2 be distinct generators of G_q. y is an arbitrary number.

[1] All math from here on out will be mod p.

A User know a representation of y if he knows x_1 and x_2 such that

$$y = g_1^{x_1} g_2^{x_2}$$

There are multiple such representations.

Since DLP is hard, finding 2 representations for a given y is also hard. (g_1 and g_2 must be randomly chosen. That is, there is no known relationship between them.)

Before we can start actually issuing anonymous electronic cash, we must first make sure that the user and the bank do the following.

- Bank makes a secret x.
- Bank publicizes g, $h = g^{x2}$, g_1, $h_1 = g_1^x$, g_2, and $h_2 = g_2^x$.
- User makes a secret u_1. Not even the Bank knows this.
- User gives $I = g_1^{u_1}$ to the Bank. This is the User's account number.
- User computes and remembers

$$Z = (I_{g_2})^x = g_1^{u_1 x} g_2^x$$

4 The Coin

We need to have a coin that is anonymous, and yet if the user double spends we want to be able to compute I and catch him. The protocol first has the user pick s, x_1, and x_2 at random. He then computes

$$A = g_1^{u_1 s} g_2^s$$
$$B = g_1^{x_1} g_2^{x_2}$$

In other words, the user knows a representation for A and B. Finally, we contruct our coin as the triplet (A, B, sig_{bank}). Just assume that the blind signature is possible; we've had enough of this already.

[2]This is never defined in the paper, so you might actually want to remember this if you read the paper.

5 Payment protocol

So, now that we have the coin, what do we do with it?

$$\begin{array}{ccc} User & \overset{A,B,sig}{\longrightarrow} & Vendor \\ User & \overset{d}{\longleftarrow} & Vendor \\ User & \overset{\substack{r_1=d(u_1 s)+x_1 \\ r_2=ds+x_2}}{\longrightarrow} & Vendor \end{array} \tag{1}$$

where

$$d = Hash(A, B, Shop - ID^3, date/time)$$

At this point, the shop checks to make sure that

$$g_1^{r_1} g_2^{r_2} = A^d B$$

If it is okay, then

$$Vendor \overset{(A,B,sig),d,date/time,(r_1,r_2)}{\longrightarrow} Bank$$

6 Catching Double Spending

This section is just pure math. The goal is to take two information packets coming into the bank and find out what I is. Here is how.

The bank has

$$\begin{array}{c} (A, B, sig_{bank}), d, (r_1, r_2), date, shop \\ (A, B, sig_{bank}), d', (r'_1, r'_2), date', shop' \end{array}$$

The bank can then compute the two subtractions

$$- \begin{array}{rcl} r_1 & = & d(u_1 s) + x_1 \\ r'_1 & = & d'(u_1 s) + x_1 \\ \hline r_1 - r'_1 & = & u_1 s(d - d') \end{array} \qquad - \begin{array}{rcl} r_2 & = & ds + x_2 \\ r'_r & = & d's + x_2 \\ \hline r_2 - r'_2 & = & s(d - d') \end{array}$$

[3] Shop-ID is the shop's account number. It is a unique identifying number.

Dividing the result on the left by the one on the right, it gets

$$u_1 = \frac{r_1 - r_1'}{r_2 - r_2'} = u_1$$

Finally, the bank needs to compute

$$I = g^{u_1} = account\ number$$