Suitable function

Daniel Nager
daniel.nager@gmail.com

June 27, 2020

This file and referenced files are on the address https://github.com/danielnager/xifrat/

We will use a substitution table, for example:

5	3	0	4	1	10	9	2	8	6	11	7
3	7	5	0	9	6	2	8	10	1	4	11
0	5	4	11	6	8	1	9	2	10	7	3
4	0	11	7	10	2	6	1	9	8	3	5
1	9	6	10	7	5	11	4	0	3	8	2
10	6	8	2	5	4	3	7	11	0	9	1
9	2	1	6	11	3	4	0	5	7	10	8
2	8	9	1	4	7	0	5	3	11	6	10
8	10	2	9	0	11	5	3	7	4	1	6
6	1	10	8	3	0	7	11	4	5	2	9
11	4	7	3	8	9	10	6	1	2	5	0
7	11	3	5	2	1	8	10	6	9	0	4

to define a function $c = f(a, b)$, where c is the element in the a-th row and b-th column.

The following two properties hold:

$f(f(a, b), c) \neq f(a, f(b, c))$ – non-associativity in general

$f(f(a, b), f(c, d)) = f(f(a, c), f(b, d))$ – restricted commutativity

Next we define a list of N integers in the range $[0, 11]$ to meet the size required. For 256 bits we need $256/log_2(12) = 71$ approximately. So let's set $N = 71$. This list can be interpreted as a 71 digit base-12 number.

Next we define a mixing procedure of elements of this kind, t and k, N-element lists of numbers in the integer range $[0, 11]$.

The mixing procedure is:

```
function m(t,k) returns t

    for M number of rounds -- 64 for example
        //one-to-one mixing of k and t
        for i in 0..N-1
            t[i]<-f(t[i],k[i])
        end for
        // accumulative mixing of t with itself, t[-1]=t[N-1]
        for i in 0..N-1
            t[i]<-f(t[i],t[i-1])
        end for
    end for

return t
```

The function m is neither associative nor commutative, and meets the restricted commutativity property:

$m(m(a,b),m(c,d)) = m(m(a,c),m(b,d))$

With this a Secret agreement and a Digital signature can be done as explained in the document:

https://github.com/danielnager/xifrat/blob/master/cryptosystem.pdf

The computationally hard problem proposed is:

in $c = m(t,k)$, knowing c and t, find k.

Now lets define the secret agreement and the digital signature using the mixing function m. To put it more clear we will use the following notation:

$m(a,b)$ is written as (a b)

$m(m(a,b),m(c,d))$ is written as (a b)(c d)

For the secret agreement the procedure is the following:

Both Alice and Bob agree on some constant C. Alice chooses a random key K, and Bob does the same choosing a random key Q. Alice sends to Bob (C K), Bob sends to Alice (C Q). Alice computes using bob sent value (C Q)(K C), and Bob does the same and computes (C K)(Q C).

By the property of restricted commutatitvity (C Q)(K C)=(C K)(Q C)

For the signature the procedure is the following:

Alice, the signer, chooses a public value C and a random key K. Its credentials are C and (C K). To sing a value, H, Alice simply computes S=(H K). H must be different from C.

Bob needs to verify if Alice has signed H. Computes (C C)(H K) and
(C H)(C K). Both values must be equal due to restricted commutativity if (H K) is a valid signature from Alice.

In order to do smaller signatures, of 128 bits in this case, we apply the following equality:

$$(CCK_2K_1)(H_1H_2K_1K_2) = (CH_1)(CH_2)(K_2K_1)(K_1K_2)$$

In this formula, C is a public constant provided by Alice, the signer, K_1 and K_2 are two 128 bit keys, and H_1 and H_2 is a 256 bits value to be signed split in two halves.

The credentials of Alice are (K_2K_1), (K_1K_2) and (CCK_2K_1).

In order to sing a value represented by H_1 and H_2, the Alice computes $S = (H_1H_2K_1K_2)$.

To verify the signature Bob computes (CH_1) and (CH_2) and checks for the initial equality to hold, as Bob has all the elements needed.

Daniel Nager - daniel.nager@gmail.com