Signing with Multiple ID's and a Single Key

Mirosław Kutyłowski
Wrocław University of Technology, Wrocław, Poland
Email: Miroslaw.Kutylowski@pwr.wroc.pl

Jun Shao
Zhejiang Gongshang University, Hangzhou, P.R.C.
Email: chn.junshao@gmail.com

***Abstract*— We propose a solution for electronic identity cards that makes it possible to create multiple identities for a person creating digital signatures with just one private key stored on his personal electronic identity card. The public keys corresponding to this secret key belong to different *sectors*, devoted to different applications or activity areas (like private use and signing in behalf of a company or an authority).**

We provide strong privacy guarantees called *unlinkability*. It means that given two public keys from different sectors and some signatures corresponding to these keys, it is unfeasible to say if these keys (and signatures) come from the same person. The proof is performed in the random oracle model and reduces linkability to the Decisional Diffie-Hellman Problem.

Our proposal extends the idea of *Restricted Identification* introduced in German personal identity cards from unlinkable sector identification to unlinkable sector signatures.

I. Introduction

Today more and more data concerning relations between physical persons and service providers is exchanged in electronic way. This awakes strong concerns about privacy threats: electronic data can be processed cheaply and fast, so the scale of data misuse might be quite substantial. For this reason, quite many countries introduced sharp personal data protection legal rules. If a system designer implements these requirements on a purely procedural way, then the risk and the cost will be quite high. This creates a barrier for development of pervasive electronic systems, as violations of law become uneasy to diagnose. Even if in some countries the privacy protection is on a much lower level, this creates natural barrier for exporting goods and services. Last not least, according to cloud computing paradigm it could be quite hard to determine where the protected data is processed.

In this situation it seems to be necessary to provide tools that could guarantee personal data protection automatically. The (cryptographic) protocols designed according to this goal should provide those features and only those that are necessary for achieving desired functionality. A good example of this approach is a scheme, called *Restricted Identification*, introduced by German security authority BSI for the use on German personal identity cards [1]. According to this scheme, the identity card can create a separate identity for each application sector, so that the ID's of a given person from different sectors are not linkable. A similar solution (but based on symmetric cryptography) was introduced in Austria (*Bürgerkarte*). An alternative of [1] for a different application scenario was proposed in [3].

Our goal is to propose a similar mechanism for *signing* documents with a smart card. The signatures (and the public keys) for different application sectors of the same person, should be unlinkable. For instance, a person should use a different signature when operating business, when shopping online, when making personal communication. A trivial solution would be to generate a separate key pair for each sector. However, this approach is quite problematic from practical point of view. The first problem is the number of secret keys that we would have to hold on secure smart cards. As the memory for these devices is severely limited, there are substantial scalability problems. Second, independent creation of keys would help to perform Sybil attacks.

A. Our Contribution

We propose a framework in which a user holds a *single* private key of his smart card, but nevertheless can create signatures for practically unlimited number of sectors. Moreover, it is guaranteed that the public keys and the signatures from different sectors but created by the same person are unlinkable as long as some cryptographic assumptions are true.

II. Protocol Description

The protocol described below reuses the signature scheme proposed by Bao Feng in [2]. The difference is that we create multiple public keys for a single private key.

A. System Setup

We use a group G with a prime order q such that the Decisional Diffie-Hellman Problem (DDH) is hard in G.

We define the sectors and their corresponding public keys (the setup has a certain peculiarity: there are no private keys corresponding to these public keys). For each sector we use its legal name (which must be unique by definition). If A is such a legal name, then the corresponding public key is $g(A) := H_G(A)$ where H is a secure hash function $H_G : \{0,1\}^* \to G$.

Let g be a fixed generator of G. Due to pseudo-randomness of H_G we may also assume that the discrete logarithm of a public key $g(A)$ with respect to g is unknown.

Each user holds a personal identity card with a smart card capable of performing cryptographic operations. Among other possible functions, each ID card B holds its multi-sector signature key x_B. The key x_B is a random number from Z_q^*. The card should guarantee that the key x_B can be generated only once for the lifetime of the smart card. In this way we prevent Sybil attacks – a person with a bad record cannot erase the old key and generate a new one in order to start a "second life". This is easy to achieve, if the mechanism is implemented

978-1-4244-8790-5/11/$26.00 ©2011 IEEE

on personal identity cards (like in many EU countries), where there is a strict control over issuing the cards and a single person has at most one valid personal identity card.

For the key x_B, there is the corresponding *master* public key $P_B := g^{x_B}$. The key P_B is generated on the smart card and the confirmed by the card issuer in a certificate.

B. Registering to a Sector

In order to use sector signature a user B holding the smart card with his multi-sector secret x_B generates his public key $p(A)_B := g(A)^{x_B}$ for sector A.

In order to use the key $p(A)_B$ the user B must contact a Certificate Authority (CA). Then CA provides a certificate $C(A)_B$ stating that the owner of the public key $p(A)_B$ has registered for using sector A. Before issuing the certificate, CA checks that B is holding a smart card with the discrete logarithm of $p(A)_B$ with respect to $g(A)$ and that this discrete logarithm is equal to the discrete logarithm of P_B with respect to g. For this purpose a standard zero-knowledge proof [5] may be executed. In its database CA stores a record linking $p(A)_B$ to B for the case of a fraud committed by B in sector A.

When B contacts A for the first time, then he presents the certificate for $p(A)_B$ – possibly without declaring (or proving) his identity. From this moment, in order to sign a document directed to sector A, user B creates signatures that can be verified with the key $p(A)_B$.

C. Signature Creation and Verification

Assume that a user B has to sign a document m for sector A. The following steps are executed:

1) B chooses a number $r \in [1, q-1]$ uniformly at random, and computes $R := (g(A))^r$,
2) B puts $S := H_q(g(A), p(A)_B, R, m) \cdot x_B + r \bmod q$,

(R, S) is the signature for document m. It is concerned to be valid as a a signature of m created by the holder of public key $p(A)_B$ for sector A if

$$g(A)^S = (p(A)_B)^{H_q(g(A),p(A)_B,R,m)} \cdot R$$

III. Security Features

A. Unforgeability

Theorem 1: If the multi-sector signature scheme can be forged in the random oracle model, then the discrete logarithm (DL) problem can be solved for G.

Proof: Due to the limited space, we just give the proof sketch here. We follow the method in [4] to simulate the hash oracle H_q and signature oracle $\mathcal{O}_s$. For the hash oracle H_G: on input a legal name I, $\mathcal{B}$ finds (I, r_G) in the table T_{H_G}. If it exists, then $\mathcal{B}$ returns $(g^a)^{r_G}$; otherwise, $\mathcal{B}$ chooses a random number r_G from Z_p^*, returns $(g^a)^{r_G}$, and records (I, r_G) in the table T_{H_G}.

According to the result in [4], we know that with a non-negligible probability, one can produce two valid signature transcripts (R, r_q, S) and (R, r'_q, S') on message m on behalf of $(g(I), p(I)_B)$. $\mathcal{B}$ solves the DL problem by $a = (S' - S)/(r_G \cdot r'_q - r_G \cdot r_q)$, where r_G is the value in the table T_{H_G} corresponding to I. ■

B. Unlinkability

When public keys $P(C)$, $P'(D)$ from sectors C and D are presented (together with some signatures verified with these keys, and the public keys g_C, g_D of the sectors), the question to answer is if $P(C)$, $P'(D)$ are assigned to the same person. If we cannot answer this question, we say that the scheme satisfies unlinkability in a weak sense.

Theorem 2: If the multi-sector signature scheme does not satisfy unlinkability in a weak sense in the random oracle model, then Decisional Diffie-Hellman Problem can be solved for G.

Proof: We set $P(C) = (g, \bar{g})$ and $P'(D) = (h, \bar{h})$, then use the same method in the proof of Theorem 1 to simulate the hash oracle H_G and the signature oracle. At the end, if $\mathcal{A}$ says $P(C) = (g, \bar{g})$ and $P'(D) = (h, \bar{h})$ are corresponding to the same user, then $\mathcal{B}$ outputs $\log_g \bar{g} = \log_g \bar{h}$; otherwise, $\mathcal{B}$ outputs $\log_g \bar{g} \neq \log_g \bar{h}$. So, the DDH problem is solved. ■

We say that the scheme is unlinkable in the strong sense if answering the following question is infeasible in all but a negligible number of cases:

- the input consists of: the public keys of the sectors, the set of public keys of users for each sector, the set of master public keys of the users in each sector, and some number of signatures that can be verified with these keys,
- the question is to link data of the same user in different sectors (even without disclosing the user concerned).

This is not directly equivalent to DDH Problem, since for DDH with probability $\frac{1}{2}$ the input is not a DH triple. In our problem the sets of users might be the same, so always there is some linking creating valid DDH triples (for details see [3]).

Since in the random oracle model one can simulate creating signatures (as described above), the argument from [3] can be extended to show the following theorem:

Theorem 3: If the multi-sector signature scheme is not unlinkable in the strong sense in the random oracle model, then Decisional Diffie-Hellman Problem can be solved for G.

Acknowledgment

This work has been supported by by EU within the 7th Framework Programme, contract 215270 (FRONTS), Foundation for Polish Science, “MISTRZ” Programme, and National Natural Science Foundation of China, grant 61003308.

References

[1] Bundesamt für Sicherheit in der Informationstechnik. Technical Report. Advanced Security Mechanisms for Machine Readable Travel Documents, TR-03110, 2009. `https://www.bsi.bund.de/cae/servlet/contentblob/532066/publicationFile/44792/TR-03110_v202_pdf.pdf`.

[2] F. Bao. Colluding Attacks to a Payment Protocol and Two Signature Exchange Schemes. In *ASIACRYPT 2004*, vol. 3329 of *LNCS*, pp. 137–144, 2004.

[3] M. Koza, P. Kubiak, Ł. Krzywiecki, and M. Kutyłowski. Restricted Identification Scheme and Diffie-Hellman Linking Problem. Technical report, Wroclaw University of Technology, 2010. Submitted for publication.

[4] D. Pointcheval and J. Stern. Security Proofs for Signature Schemes. In *EUROCRYPT 1996*, vol. 1070 of *LNCS*, pp. 387–398, 1996.

[5] C.P. Schnorr. Efficient identifications and signatures for smart cards. In *CRYPTO 1998*, vol. 435 of *LNCS*, pp. 239–251, 1998.