Group Signatures: Unconditional Security for Members

Lidong Chen Torben P. Pedersen[*]
Aarhus University
Denmark

Abstract

First a detailed definition of group signatures, originally suggested by Chaum and van Heijst, is given. Such signatures allow members of a group to sign messages anonymously on behalf of the group subject to the constraint that in case of disputes later on a designated authority can identify the signer. It is shown that if such schemes are to provide information theoretic anonymity, then the length of the secret information of the members and the authority increases with the number of members and the number of signatures each member is allowed to make. A dynamic scheme meeting these lower bounds is described. Unlike previous suggestions it protects each member unconditionally against framing, i.e. being hold responsible for a signature made by someone else.

1 Introduction

Group signatures as introduced in [CH91] allow members of a group (e.g. a company or family) to make signatures on behalf of the group in such a way that

- only members can make signatures,
- the actual member who made a given signature remains anonymous except that

[*] Supported by Carlsbergfondet

- in case of dispute a designated authority (who is given some extra information) can identify the signer.

Such a signature scheme can for example be used in invitations to submit tenders. All companies submitting a tender then form a group and each company signs its tender anonymously using the group signature. Later when the preferred tender has been selected the winner can be identified, whereas the signers of all other tenders will remain anonymous. All submitters are bound to their tender by the signature, as the signer can be identified without his cooperation.

1.1 Related Work

Group signatures should not be confused with the related notion of group oriented signatures first suggested in [Boy89b] and [CH89]. Here certain subsets of a group of people are allowed to sign on behalf of the group. Such schemes do not provide a method for identifying the (subset of) members who actually made the signature (see [D93] for an overview). Another related concept is that of multi-signatures which require a digital signature from many persons (see [O88] and [OO93]).

As mentioned above, group signatures were introduced by Chaum and van Heijst in [CH91] (see also [H92]). They present four schemes: one protects the anonymity of the signer unconditionally, whereas the other three only give computational protection. These schemes also differ with respect to the following two properties:

- Framing:
  A group member, P, is said to be *framed* if other persons (including group members) make a signature for which the trusted authority will identify P as the signer.

- Dynamic:
  A group signature scheme is called *dynamic* if the group members do not have to change their secret keys when the group is changed (members leaving or new members joining). Only the public key of the group and possibly the secret key of new members must be changed.

In particular, the scheme from [CH91] providing unconditional anonymity is not dynamic and it only protects against framing under a cryptographic assumption.

In [CP94] a dynamic scheme providing unconditional anonymity is presented, but security against framing relies on a cryptographic assumption.

1.2 Results and Contents

This paper contains three main results:

- Group signatures are defined in details in Section 2. Based on this definition, the method of double-signing introduced in [CP94] is formalised (Section 3).
- A dynamic group signature scheme providing unconditional anonymity and unconditional protection against framing is presented (see Section 4).
- Lower bounds on the sizes of secret keys and auxiliary information of the authority are given (see Section 5). These bounds say that the length of the secret key of each member grows as $T \log_2 n$, if each member can make T signatures and n is the number of members. Similarly, the length of the auxiliary information of the authority grows as $Tn \log_2 n$. The scheme presented in Section 4 actually meets these bounds except for constant factors.

2 Definitions

In this section secure group signatures are defined. Throughout this paper $\mathcal{M}$ denotes the message space.

Definition 1 A group signature for a group of n members $P_1, ..., P_n$ and an authority A is a tuple $(n, k, gen, sign, test, iden)$. Here k is the security parameter, and gen, $sign$, $test$, $iden$ are all polynomial time (in k) algorithms.

- gen is a ***probabilistic*** algorithm generating the keys. On input (k, n) it outputs
$$(pk, (s_1, s_2, ..., s_n), aux),$$
where pk is the public key of the group, s_i is the secret key of P_i, $i = 1, 2, ..., n$, and aux is the auxiliary information for A.

- $sign$ is a ***probabilistic*** algorithm which on input s_i and $m \in \mathcal{M}$ outputs $sign(s_i, m)$. A string σ is called a ***correct signature*** on $m \in \mathcal{M}$, if there exists $i \in \{1, 2, \ldots, n\}$ such that $\sigma = sign(s_i, m)$.

- $test$ is used to test signatures. On input pk, m, and a possible signature on m, it outputs $true$ or $false$. A string σ is called an ***acceptable signature*** on m with respect to pk if $test(pk, m, \sigma) = true$.

- $iden$ is used by A to identify the signer. On input aux, $m \in \mathcal{M}$ and an acceptable signature on m, it outputs $i \in \{1, 2, ..., n\} \cup \{?\}$ (the output ? indicates that $iden$ could not identify the signer).

For any $i \in \{1, 2, ..., n\}$, and any $m \in \mathcal{M}$, the scheme must satisfy

$$test(pk, m, sign(s_i, m)) = true,$$

and

$$iden(aux, m, sign(s_i, m)) = i.$$

Remark Different secret keys must produce different signatures:

$$\forall i, j \in \{1, 2, ..., n\} \, \forall m \in \mathcal{M} : \; i \neq j \Rightarrow sign(s_i, m) \neq sign(s_j, m).$$

Remark A correct signature is also acceptable, but an acceptable signature is not necessarily correct.

According to the informal description in the introduction group signatures must provide

- Security against forgeries.

- Anonymity of the signer.

- The authority must be able to identify the signer.

Each of these properties will be defined in the following.

2.1 Security Against Forgeries

It must be infeasible to forge signatures in adaptively chosen message attacks (see [GMR88]). Let $\mathcal{F}$ be a polynomial time algorithm, which on input pk and possibly aux, works as follows.

1. Repeat the following:
   (a) Generate a message $m \in \mathcal{M}$ and $i \in \{1, 2, ..., n\}$;
   (b) Get $sign(s_i, m)$.
2. Output a message $m_0 \in \mathcal{M}$ different from all m's generated above and $\tilde{\sigma}(m_0)$.

Definition 2 Let a group signature $(n, k, gen, sign, test, iden)$ be given. The scheme is ***secure against forgeries*** after signing T messages if the following holds: For any polynomial time $\mathcal{F}$ as above getting at most T signatures from each P_i, for all but a negligible fraction of the keys,

$$\begin{array}{l} \forall c > 0, \exists k_0, \boldsymbol{s.t.} \forall k > k_0 \\ Prob[test(pk, m_0, \tilde{\sigma}(m_0)) = true] \leq k^{-c}, \end{array}$$

where $(m_0, \tilde{\sigma}(m_0))$ is the output of $\mathcal{F}$. The probability is over the random coins of signatures and the random coins of $\mathcal{F}$.

2.2 Anonymity

Every group member should be able to make signatures on behalf of the group without leaking any (Shannon-) information about his identity. To define this the distribution of the secret keys is needed.

A public key pk, produced by gen, corresponds to a set of possible secret keys defined as

$$\begin{aligned} SK(pk) &= \{(sk_1, sk_2, ..., sk_n) \mid \exists aux : \\ &\qquad gen(n,k) = (pk, (sk_1, sk_2, \ldots, sk_n), aux)\}. \end{aligned}$$

We will omit pk in the following. The set $SK^{(i)}$ is defined as all the possible secret keys of P_i, $i = 1, 2, \ldots, n$, i.e. $SK^{(i)}$ is the projection of SK on the i'th coordinate. If $s_i \in SK^{(i)}$ denotes the actual secret key of P_i, then

$$(s_1, s_2, \ldots, s_n) \in SK.$$

For any subset J of $\{1, 2, ..., n\}$ and for all positive integers t and L, $0 < L \leq |J|t$, define a subset of J^L by

$$\mathcal{I}_J(t, L) = \{(i_1, \ldots, i_L) \in J^L \mid \forall j \in J : |\{l \in \{1, \ldots, L\} \mid i_l = j\}| \leq t\},$$

Thus each $j \in J$ appears at most t times in $\underline{i} = (i_1, \ldots, i_L) \in \mathcal{I}_J(t, L)$. For $J = \{1, 2, \ldots, n\}$, $\mathcal{I}_J(t, L)$ will be denoted $\mathcal{I}(t, L)$.

If $\sigma(m_i)$ is a correct signature on $m_i \in \mathcal{M}$ for $i = 1, \ldots, L$, then $\sigma(\underline{m})$ denotes $(\sigma(m_1), \sigma(m_2), \ldots, \sigma(m_L))$. For every $\underline{i} \in \mathcal{I}_J(t, L)$, "$\sigma(\underline{m}) \Leftarrow \underline{i}$" denotes the event that there exists $(sk_1, sk_2, \ldots, sk_n) \in SK$ such that for all $j \in \{1, 2, \ldots, L\}$:

$$sign(sk_{i_j}, m_j) = \sigma(m_j).$$

Definition 3 Let a group signature $(n, k, gen, sign, test, iden)$ be given. The scheme provides *anonymity* for signing T messages if for any $J \subseteq \{1, 2, ..., n\}$, and for any $L \leq |J|T$ different messages

$$\underline{m} = (m_1, m_2, \ldots, m_L),$$

and correct signatures on these made by $(P_j)_{j \in J}$

$$\sigma(\underline{m}) = (\sigma(m_1), \sigma(m_2), \ldots, \sigma(m_L))$$

the following holds. If each P_j has made at most T signatures, then for any $\underline{i} \in \mathcal{I}_J(T, L)$,

$$Prob[\sigma(\underline{m}) \Leftarrow \underline{i}] = \frac{1}{|\mathcal{I}_J(T, L)|}.$$

The probability is over the choice of $(sk_1, sk_2, \ldots, sk_n) \in SK$ and the random coins used in the signatures.

2.3 Signer Identification

For any subset J of $\{1, 2, ..., n\}$, let $\mathcal{F}_J$ be a polynomial time algorithm, which on input pk and $\{s_j\}_{j \in J}$, works as follows:

1. Repeat the following:
   (a) Generate a message $m \in \mathcal{M}$, and a number $i \in J^c$;
   (b) Get $sign(s_i, m)$.
2. Output a message $m_0 \in \mathcal{M}$ different from all m's in 1 and an acceptable signature $\sigma(m_0)$ on m_0.

Definition 4 Let $(n, k, gen, sign, test, iden)$ be a group signature. The scheme provides ***signer identification*** for signing T messages if the following holds: For any subset J of $\{1, 2, ..., n\}$, and for any polynomial time algorithm $\mathcal{F}_J$ as above getting at most T signatures from each P_i $(i \in J^c)$,

$$\begin{array}{l} \forall d > 0, \exists k_0, \text{ s.t. } \forall k > k_0 \\ Prob[iden(aux, m_0, \sigma(m_0)) \in J] \geq 1 - k^{-d}, \end{array}$$

where $(m_0, \sigma(m_0))$ is the output of $\mathcal{F}_J$. The probability is over the random coins of $\mathcal{F}_J$ and the choices of the received signatures.

There are two aspects of this definition. Firstly, for $|J| = 1$ it says that the signer must be identified by the authority with overwhelming probability. Secondly, it says that no subset of (polynomially bounded) group members can frame a member outside this subset.

2.4 Secure Group Signatures

The preceding three definitions give

Definition 5 A group signature scheme is ***secure*** for signing T messages, if it is secure against forgery, provides anonymity and signer identification after each member has made at most T signatures.

Remark The definition easily generalises to let P_i sign T_i messages, $i = 1, 2, \ldots, n$.

3 Identifying the signer

[CP94] sketched a general method by which the authority can identify the signer. In the following this method is described in terms of the previous definitions.

Let $(n, k, gen, sign, test, iden)$ be a group signature scheme which satisfies Definition 2 and 3 for signing T messages. This scheme can be used to construct a new one which under certain conditions satisfies Definition 5 for signing T messages. The new scheme will be denoted by $(n, k, gen', sign', test', iden')$ and is defined as

- $gen'(k, n)$: execute $gen(k, n)$ twice with independent random bits. This gives $(pk_i, (s_{1i}, s_{2i}, \ldots, s_{n,i}), aux_i)$ for $i = 1, 2$. The output of $gen'(k, n)$ is now defined as

  $$((pk_1, pk_2), ((s_{11}, s_{12}), \ldots, (s_{n1}, s_{n2})), (s_{11}, s_{21}, \ldots, s_{n,1})).$$

- $sign'((s_{i1}, s_{i2}), m) = (sign(s_{i1}, m), sign(s_{i2}, m)) = (\sigma_1, \sigma_2)$.
- $test'((pk_1, pk_2), m, (\sigma_1, \sigma_2)) = test(pk_1, m, \sigma_1) \wedge test(pk_2, m, \sigma_2)$
- $iden'((s_{11}, \ldots, s_{n,1}), m, (\sigma_1, \sigma_2))$ outputs id where

  $$id = \begin{cases} i & \text{if } \sigma_1 = sign(s_{i1}, m) \\ ? & \text{if no such } i \text{ exists.} \end{cases}$$

Since different group members make different signatures $iden'$ is well defined. Thus, the new scheme consists of two independent versions of the original scheme. Each member has two secret keys, and the authority knows one of these.

Proposition 6 *The scheme $(n, k, gen', sign', test', iden')$ defined above is secure against forgeries and provides anonymity for signing T messages.*

Proof Forging a signature require forging a signature with respect to pk_2. This is infeasible by the properties of the original scheme.

The scheme provides anonymity because the original scheme provides anonymity. □

By the definition of $iden'$ the extended scheme can be used to identify members making correct signatures. Furthermore, under certain circumstances it can be shown that the extended scheme satisfies the requirements to signer identification. This proof often depends on the actual schemes (see Section 4.2.3 for an example).

In three of the schemes in [CH91] double-signing will make it easier to identify the signer than using the interactive protocols proposed there (at the cost of twice as long signatures).

4 Obtaining Unconditional Anonymity

This section presents a group signature scheme giving unconditional anonymity. First, the basic ingredients are presented, and then it is shown how these can be used to construct a group signature scheme.

Throughout this section let G_q denote a (multiplicative) group of prime order, q.

4.1 Basic Signature

The basic signature can very briefly be described as a combination of the identification protocol of [O93] and the fail-stop signature scheme of [HP93].

Let two generators g_1 and g_2 of G_q be given. Let $g_2 = g_1^e$ and let the message space be $\mathcal{M} = \mathbb{Z}_q \setminus \{e\}$. It is easy to test membership in $\mathcal{M}$ as $m \in \mathbb{Z}_q$ is in $\mathcal{M}$ if and only if $g_2 \neq g_1^m$.

A person having secret key (s_1, s_2) and a corresponding public key $h = g_1^{s_1} g_2^{s_2}$ signs a message $m \in \mathcal{M}$ by publishing $\sigma = s_1 + ms_2 \bmod q$ and proving that this is indeed correct. This proof is obtained from the interactive protocol in Figure 1 by computing the challenge c as $\mathcal{H}(m, \sigma, a, \tau)$ where $\mathcal{H}$ is a hash function with "pseudo-random properties" (see [FS87]). More precisely, $\Sigma = (\sigma, c, r_1, r_2)$ is a correct signature on m with respect to h if $\tau = r_1 + mr_2 - c\sigma$ and $a = g_1^{r_1} g_2^{r_2} h^{-c}$ satisfy $c = \mathcal{H}(m, \sigma, a, \tau)$.

$$\begin{array}{lcl}
\textsf{P} & & \textsf{V}\\
t_1, t_2 \in_{\mathcal{R}} \mathbb{Z}_q^* & & \\
a \leftarrow g_1^{t_1} g_2^{t_2} & & \\
\tau \leftarrow t_1 + mt_2 & & \\
 & \xrightarrow{(a,\tau)} & \\
 & & c \in_{\mathcal{R}} \mathbb{Z}_q^* \\
 & \xleftarrow{c} & \\
r_1 \leftarrow t_1 + cs_1 \bmod q & & \\
r_2 \leftarrow t_2 + cs_2 \bmod q & & \\
 & \xrightarrow{(r_1,r_2)} & \\
 & & r_1 + mr_2 \overset{?}{=} \tau + c\sigma \\
 & & g_1^{r_1} g_2^{r_2} \overset{?}{=} ah^c
\end{array}$$

Figure 1: Interactive proof that $\sigma = s_1 + ms_2$.

The scheme is only intended for signing one message, because given signatures on two different messages, the secret key can be derived by solving two linear equations.

This also means that in order to forge a signature (given a signature) the forger must be able to compute the secret key. Thus, it is sufficient to argue that the secret key cannot be computed from a single signature.

Firstly, σ does not help computing the secret key, because given the public key all values of σ are equally likely (there are q possible secret keys and they will all give a different value of σ because $m \neq e \bmod q$). Secondly, if c is chosen uniformly at random, an execution of the protocol in Figure 1 does not help computing the secret key. Thus under the assumption that computing c as $\mathcal{H}(m, \sigma, a, \tau)$ corresponds to choosing it at random, the signature scheme is secure.

4.2 Group Signatures

We only consider the case with two persons (P_1 and P_2) in the group (the general case is obtained by a straightforward extension). Let T be a parameter, and let $T + 1$ generators $g_0, g_1, \ldots, g_T$ of G_q be given.

These are chosen initially by a key authentication centre (or the group authority) such that for some $e \in \mathbb{Z}_q$

$$g_i = g_{i-1}^e \qquad \text{for } i = 1, 2, \ldots, T.$$

The message space is $\mathcal{M} = \mathbb{Z}_q \setminus \{e\}$ as before. It is important that no group member knows e. We therefore need the following extended discrete logarithm assumption:

Assumption 1 Let A be any polynomially bounded algorithm which takes q and $(g_0, \ldots, g_T)$ chosen at random as described above as input and outputs a number $d \in \mathbb{Z}_q$. Then the probability that

$$g_i = g_{i-1}^d \qquad \text{for } i = 1, 2, \ldots, T$$

is smaller than the inverse of any polynomial for q sufficiently large.

The secret key of P_i is $(s_{i0}, \ldots, s_{iT}) \in \mathbb{Z}_q^{T+1}$ for $i = 1, 2$. The public key of the group is

$$(g_0, g_1, \ldots, g_{T+1}, h_1, h_2)$$

where

$$h_i = \prod_{j=0}^{T} g_j^{s_{ij}} \qquad \text{for } i = 1, 2$$

(assume $h_1 \neq h_2$). The secret key of P_i will be denoted by s_i, where s_i is the polynomial

$$s_i(x) = \sum_{j=0}^{T} s_{ij} x^j \bmod q.$$

P_i's signature on a message $m \in \mathcal{M}$ is

$$\sigma = s_i(m)$$

plus a proof that σ is correct with respect to either h_1 or h_2. A witness indistinguishable proof of this can be constructed from the protocol in Figure 1 using the techniques of Schoenmakers (see [S93] — briefly sketched in Appendix A). The resulting protocol is shown in Figure 2. The digital signature is then obtained as before using a pseudo-random hash function. Thus the signature on m is (pretty long):

$$\Sigma = (\sigma, (r_{ik})_{i=1,2,k=0,\ldots,T}, (d_1, d_2)).$$

$$\begin{array}{lcl}
\mathsf{P} & & \mathsf{V} \\
t_{ik} \in_{\mathcal{R}} \mathbb{Z}_q^* & & \\
d_2 \in_{\mathcal{R}} \mathbb{Z}_q^* & & \\
a_1 \leftarrow \prod_{k=0}^{T} g_k^{t_{1k}} & & \\
a_2 \leftarrow \prod_{k=0}^{T} g_k^{t_{2k}} h_2^{-d_2} & & \\
\tau_1 \leftarrow \sum_{k=0}^{T} t_{1k} m^k & & \\
\tau_2 \leftarrow \sum_{k=0}^{T} t_{2k} m^k - d_2\sigma & & \\
 & \xrightarrow{(a_1, a_2; \tau_1, \tau_2)} & \\
 & & c \in_{\mathcal{R}} \mathbb{Z}_q^* \\
 & \xleftarrow{\quad c \quad} & \\
d_1 = c - d_2 \bmod q & & \\
r_{1k} \leftarrow t_{1k} + d_1 s_{1k} \bmod q & & \\
r_{2k} \leftarrow t_{2k} & & \\
 & \xrightarrow{(r_{ik})_{i,k}} & \\
 & & c \overset{?}{=} d_1 + d_2 \bmod q \\
 & & \sum_{k=0}^{T} r_{ik} m^k \overset{?}{=} \tau_i + d_i\sigma \\
 & & \prod_{k=0}^{T} g_k^{r_{ik}} \overset{?}{=} a_i h_i^{d_i}
\end{array}$$

Figure 2: Interactive proof that σ is correct with respect to h_1 or h_2 — P knows the secret key corresponding to h_1. Here $i = 1, 2$ and $k = 0, 1, \ldots, T$.

The last two tuples of this signature will also be called the ***proof-part*** of Σ. The signature can be verified by computing

$$\tau_i = \sum_{k=0}^{T} r_{ik} m^k - d_i \sigma \bmod q \qquad \text{for } i = 1, 2$$

and

$$a_i = \prod_{k=0}^{T} g_k^{r_{ik}} h_i^{-d_i} \qquad \text{for } i = 1, 2$$

and verifying that $d_1 + d_2$ equals $\mathcal{H}(m, \sigma, a_1, a_2, \tau_1, \tau_2)$.

In the analysis of this scheme it is sometimes necessary to consider the general scheme with n members. This scheme is easily derived from the case $n = 2$, and is shown in Appendix B.

Proposition 7 *The interactive protocol in Figure 2 is a witness indistinguishable proof of knowledge (see [FS90]) of* $(w_0, \ldots, w_T)$ *such that*

$$\sigma = \sum_{j=0}^{T} w_j m^j \bmod q \quad \wedge \quad \left(h_1 = \prod_{j=0}^{T} g_j^{w_j} \quad \vee \quad h_2 = \prod_{j=0}^{T} g_j^{w_j} \right).$$

Proof Using the same arguments as in [S93] it can be shown that the protocol is a proof of knowledge as claimed.

Witness indistinguishability is proved by considering the distribution of the messages, which the prover sends. First it is shown that two provers knowing different witnesses to the same h_i, say h_1, produce messages with the same distribution. Then it shown that a prover knowing a witness to h_1 cannot be distinguished from a prover knowing a witness to h_2 (the protocol for a prover knowing a witness to h_2 is symmetric to that in Figure 2).

An execution with witness $(s_{10}, \ldots, s_{1T})$ using $(t_{10}, \ldots, t_{1T})$ will give exactly the same messages as an execution with secret key $(s'_{10}, \ldots, s'_{1T})$ and random choices $(t'_{10}, \ldots, t'_{1T})$, where

$$t'_{1k} = t_{1k} + d_1(s_{1k} - s'_{1k}) \qquad \text{for } k = 0, 1, \ldots, T.$$

In particular,

$$\prod_0^T g_k^{t'_{1k}} = \prod_0^T g_k^{t_{1k} + d_1(s_{1k} - s'_{1k})} = \prod_0^T g_k^{t_{1k}} h^{d_1} h^{-d_1} = \prod_0^T g_k^{t_{1k}}$$

Similarly is

$$\sum_0^T t'_{1k} m^k = \sum_0^T t'_{1k} m^k.$$

Next, an execution with witness $(s_{10}, \ldots, s_{1T})$ (to h_1) and random choices $(t_{10}, \ldots, t_{1T}, t_{20}, \ldots, t_{2k})$ will result in exactly the same messages as an execution with secret key $(s_{20}, \ldots, s_{2T})$ (to h_2) and random choices $(t'_{10}, \ldots, t'_{1T}, t'_{20}, \ldots, t'_{2k})$ where

$$\begin{aligned} t'_{1k} &= t_{1k} + d_1 s_{1k} \\ t'_{2k} &= t_{2k} - d_2 s_{2k} \end{aligned}$$

for $k = 0, 1, \ldots, T$. It is not hard to see that for the same values of d_1 and d_2 in the two conversations all messages are equal. □

The following lemma is essential for showing that the group signature scheme is secure.

Lemma 8 *Let* $0 \leq l \leq T$, *and let correct signatures on different messages* $m_1, m_2, \ldots, m_l$ *be given. For any* h_i, *there are* q^{T-l} *possible* $(T+1)$*-tuples in* $\mathbb{Z}_q$, *which could be the secret key corresponding to* h_i *and these* l *signatures.*

Proof Given these l signatures any secret key corresponding to h_i must satisfy the following equations:

$$\begin{aligned} h_i &= g_0^{s_{i0}} g_1^{s_{i1}} \cdots g_T^{s_{iT}} \\ \sigma_j &= s_i(m_j) \qquad (1 \leq j \leq l). \end{aligned}$$

For $h_i = g_0^{e_i}$ these are equivalent to

$$\begin{pmatrix} 1 & e & e^2 & \cdots & e^T \\ 1 & m_1 & m_1^2 & \cdots & m_1^T \\ 1 & m_2 & m_2^2 & \cdots & m_2^T \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ 1 & m_l & m_l^2 & \cdots & m_l^T \end{pmatrix} \begin{pmatrix} s_{i0} \\ s_{i1} \\ s_{i2} \\ \cdot \\ \cdot \\ \cdot \\ s_{iT} \end{pmatrix} = \begin{pmatrix} e_i \\ \sigma_1 \\ \sigma_2 \\ \cdot \\ \cdot \\ \cdot \\ \sigma_l \end{pmatrix}$$

By the definition of $\mathcal{M}$ this matrix has maximal rank and therefore there are exactly q^{T-l} solutions.

The lemma now follows from the fact that the proof in Figure 2 is witness indistinguishable (i.e., the proof-part of the signature reveals no additional information about the actual secret key). □

Remark If a member makes less than T signatures, his secret key is information-theoretically protected. If he makes T signatures, the key can be computed if e and e_i are known. However, this is assumed to be hard (see Assumption 1).

4.2.1 Security Against Forgeries

If the challenge c is chosen uniformly at random, Proposition 7 shows that σ must equal $s_i(m)$. Thus, if the hash function has the property that it is just as hard to convince a verifier who chooses c using $\mathcal{H}$ as a verifier who chooses c at random, the following assumption is reasonable:

Assumption 2 If σ is the first component of an acceptable signature on the message, m, then for some $i \in \{1, 2, \ldots, n\}$ there exists $a_0, a_1, \ldots, a_T \in \mathbb{Z}_q$ such that

$$\sigma = \sum_0^T a_j m^j \quad \text{and} \quad h_i = \prod_0^T g_i^{a_i}.$$

Furthermore, in order to produce such a signature it is necessary to know $(a_0, a_1, \ldots, a_T)$.

By knowing $(a_0, a_1, \ldots, a_T)$ we simply mean that the ability to forge a signature requires the ability to convince a verifier in the interactive protocol. As this is a proof of knowledge we can use the corresponding knowledge extractor to obtain $(a_0, a_1, \ldots, a_T)$.

Lemma 9 *Given P_i's signatures on $T-1$ different messages even with unlimited computing power it is infeasible to find P_i's signature on a new message with probability better than $1/q$.*

Proof Given l signatures from P_i there are q^{T-l} possible secret keys. A forger, who can construct a signature on a new message is able, to bound the number of possibilities to q^{T-l+1}. A contradiction. □

This lemma shows that it is hard to find a correct signature. In order to rule out the possibility of making an acceptable signature the following consequence of Assumption 1 is needed.

Lemma 10 *By Assumption 1 it is hard to find $a_0, \ldots, a_T, b_0, \ldots, b_T \in \mathbb{Z}_q$ such that*

$$(a_0, \ldots, a_T) \neq (b_0, \ldots, b_T) \quad \textit{and} \quad \prod_{i=0}^T g_i^{a_i} = \prod_{i=0}^T g_i^{b_i}.$$

Proof Given $a_0, \ldots, a_T, b_0, \ldots, b_T \in \mathbb{Z}_q$ as described. Then

$$\prod_{i=0}^{T} g_i^{a_i - b_i} = 0$$

and hence e is a root of the polynomial

$$\sum_{i=0}^{T} (a_i - b_i) x^i$$

over $\mathbb{Z}_q$. Thus a probabilistic, polynomial time algorithm which computes $a_0, \ldots, a_T, b_0, \ldots, b_T \in \mathbb{Z}_q$ can be used to find e in expected polynomial time by finding the roots of this polynomial (see [K81]).

□

This lemma says that it is hard to find two different secret keys corresponding to the same public key.

Proposition 11 *Under Assumptions 1 and 2 the scheme is secure against forgeries after signing $T - 1$ messages.*

Proof By Assumption 2 it is not feasible to construct a signature unless the forger knows a secret key corresponding to the public key of one of the members. Thus if the forged signature is acceptable and not correct, the forger must know a possible secret key which is different from those held by the members. By Lemma 10 it is infeasible to find such a key.

Next, Lemma 9 shows that even an unlimited powerful forger cannot construct a correct signature. □

The above proposition only proves security after signing $T - 1$ messages (for each member). However, it is conceivable that the scheme also provides security against forgery after each member has signed T messages. In particular, from an additional correct signature it is easy to find the secret key of one of the members. Furthermore, no matter how many signature a member makes it remains hard to find an acceptable signature, which is not correct (under Assumptions 1 and 2).

Thus it is sufficient to show that it is infeasible to find the secret key of P_i given P_i's signatures on T messages $m_1, \ldots, m_T$. However, we know of no formal proof of this.

4.2.2 Anonymity

If a member signs $T+1$ messages, then his secret key can be calculated. However, the following shows that the scheme provides anonymity for signing T messages.

Proposition 12 *The scheme provides anonymity for signing T messages.*

Proof Let a subset J of $\{1, 2, \ldots, n\}$ and $L \leq |J|T$ different messages be given. Given signatures on these messages made by the members of J such that each P_i has made at most T of these. We have to show that for each $\underline{i} \in \mathcal{I}_J(T, L)$ the event $\sigma(\underline{m}) \Leftarrow \underline{i}$ occurs with the same probability.

Let $\underline{i} \in \mathcal{I}_J(T, L)$ be given. If r occurs l_r times in $\underline{i}$ then there are exactly q^{T-l_r} possible secret keys of P_r. The probability that P_r has a secret key in this set is q^{-l_r}. Since the secret key of each member is chosen independently of each other, the probability that all P_r's have a secret key corresponding to $\underline{i}$ is

$$\prod_{r=1}^{|J|} q^{-l_r} = q^{-L},$$

which is independent of $\underline{i}$. □

4.2.3 Identifying the Signer

In order to obtain a group signature scheme, we use the method of double-signing described in Section 3. By Proposition 6 it is sufficient to show that Definition 4 is satisfied.

Proposition 13 *Under Assumptions 1 and 2 the scheme provides signer identification for signing $T-1$ messages if double-signing is used.*

Proof Suppose there exists a subset J of $\{1, 2, \ldots, n\}$, and an algorithm $\mathcal{F}_J$, which after getting at most $T-1$ signatures from each P_i, $i \in J^c$, can output a message m_0 and an acceptable signature $\sigma(m_0)$ such that for some $d > 0$

$$Prob[iden(aux, m_0, \sigma(m_0)) \notin J] > k^{-d}$$

for infinite many values of k. This probability is over the random coins of $\mathcal{F}_J$ and the randomness of the received signatures. Consider such a k and let $id = iden(aux, m_0, \sigma(m_0))$. Then

$$Prob[id \notin J] \leq Prob[id = ?] + Prob[id \in J^c].$$

By Lemma 9 (even if $\mathcal{F}_J$ has unlimited computing power)

$$Prob[id \in J^c] < q^{-1}$$

which is exponentially small in k. Next consider the event that $id = ?$. This means that the signature is acceptable, but not correct. By the same arguments as in the proof of Lemma 10 it can be shown to be hard to make such a signature in polynomial time. Thus $Prob[id = ?]$ is smaller than the inverse of any polynomial for q sufficiently large. □

Remark Even with unlimited computing power it is infeasible to *frame* another group member (this corresponds to the event $id \in J^c$).

4.3 The Scheme is Dynamic

The scheme is dynamic in the sense that new members can always join the group using the following procedure:

1. Select two secret keys $\tilde{s}_i = (\tilde{s}_{i0}, \ldots, \tilde{s}_{iT})$ and compute the corresponding public key,
$$\tilde{h}_i = \prod_{j=0}^{T} g_i^{\tilde{s}_{ij}}$$
for $i = 1, 2$.

2. The public key of the group is extended by adding $(\tilde{h}_1, \tilde{h}_1)$.

3. The secret key $\tilde{s}_1$ is added to the auxiliary information of the authority.

Dynamic schemes have not been defined formally, but it should be intuitively clear that the new scheme satisfies the same properties as the original one.

5 Lower Bounds

The scheme in Section 4.2 has the unfortunate property that the length of the secret keys as well as the auxiliary information grows as the number of of signatures grows. In this section it is shown that this cannot be avoided in schemes providing unconditional anonymity (see [CH91] and [CP94] for schemes with only computational anonymity in which the length of the secret keys and auxiliary information is independent of the number of signatures).

5.1 Secret Key

The main idea for proving the lower bound of the secret keys is to divide the set of possible secret keys of each member into nonempty, disjoint subsets. Then the number of possible secret keys is bounded by the number of subsets.

For a t-tuple $\underline{i} = (i_1, i_2, ..., i_t) \in \{1, 2, ..., n\}^t$, and t different messages $\underline{m} = (m_1, m_2, ..., m_t)$, for every $r, 1 \leq r \leq n$ define

$$SK_{\underline{i}}^{(r)}(\underline{m}) = \{sk \in SK^{(r)} | sign(sk, m_j) = sign(s_{i_j}, m_j), j = 1, 2, ..., t\},$$

where s_i is the secret key of P_i $(i = 1, 2, \ldots, n)$. $SK_{\underline{i}}^{(r)}(\underline{m})$ is the set of possible keys of P_r which will give P_{i_j}'s signature on m_j for $j = 1, 2, \ldots, t$.

Lemma 14 *If a group signature* $(n, k, gen, sign, test, iden)$ *provides anonymity for signing* T *messages, then for any* $t \leq T$*, the following holds: For all* $\underline{i} = (i_1, i_2, ..., i_t)$*, and any* t *different messages* $\underline{m} =$

$(m_1, m_2, ..., m_t)$,

$$SK_{\underline{i}}^{(r)}(\underline{m}) \neq \emptyset,$$

$r = 1, 2, ..., n.$

Proof Assume there exist $t \leq T$ different messages $\underline{m} = (m_1, ..., m_t)$, and $\underline{i} = (i_1, i_2, ..., i_t)$, such that

$$SK_{\underline{i}}^{(r_0)}(\underline{m}) = \emptyset,$$

for some r_0.

Let $\sigma(m_j) = sign(s_{i_j}, m_j)$, $j = 1, 2, ..., t$ and $\underline{i}_0 = (r_0, r_0, \ldots, r_0)$. Then

$$Prob[\sigma(\underline{m}) \Leftarrow \underline{i}_0] = 0,$$

which contradicts the definition of anonymity. □

Theorem 15 *Let a group signature* $(n, k, gen, sign, test, iden)$ *be given. If it provides anonymity for signing* T *messages, then for any* $r \in \{1, 2, ..., n\}$,

$$|SK^{(r)}| \geq n^T.$$

Proof First, for any $t \leq T$ different messages $\underline{m} = (m_1, m_2, ..., m_t)$, if

$$\underline{i} = (i_1, i_2, ..., i_t) \neq (i'_1, i'_2, ..., i'_t) = \underline{i}',$$

then

$$SK_{\underline{i}}^{(r)}(\underline{m}) \cap SK_{\underline{i}'}^{(r)}(\underline{m}) = \emptyset.$$

Otherwise there exists

$$sk \in SK_{\underline{i}}^{(r)}(\underline{m}) \cap SK_{\underline{i}'}^{(r)}(\underline{m}),$$

such that for some $j \in \{1, 2, \ldots, n\}$, $i_j \neq i'_j$,

$$sign(sk, m_j) = sign(s_{i_j}, m_j) \quad \text{and} \quad sign(sk, m_j) = sign(s_{i'_j}, m_j),$$

which contradicts Definition 1, since different members must make different signatures.

Second, by Lemma 14, for any t different messages $\underline{m} = (m_1, ..., m_t)$, and any t-tuple $\underline{i} = (i_1, i_2, ..., i_t) \in \{1, 2, ..., n\}^t$,

$$|SK_{\underline{i}}^{(r)}(\underline{m})| \geq 1.$$

Finally, for any t different messages $\underline{m} = (m_1, m_2, ..., m_t)$

$$|SK^{(r)}| \geq \sum_{\underline{i} \in \{1,2,...,n\}^t} |SK_{\underline{i}}^{(r)}(\underline{m})| \geq n^t,$$

for any $t \leq T$. □

Thus each member must have a secret key chosen from a set of at least n^T possible secret keys. In other words, at least $T \log n$ bits are needed to represent some of the secret keys of each group member. Thus, its length grows linearly in the number of signatures.

5.2 Auxiliary Information

In this section, we consider the length of the auxiliary information held by the authority. To this end some random variables are needed.

Definition 16 For any L, $0 < L \leq nT$, a tuple

$$hist_L(\underline{m}) = ((m_1, \sigma(m_1)), (m_2, \sigma(m_2)), \ldots, (m_L, \sigma(m_L)))$$

is called an (L, T)-history, if

$$\underline{m} = (m_1, m_2, \ldots, m_L)$$

consists of L different messages and there exists a tuple

$$\underline{i} = (i_1, i_2, \ldots, i_L) \in \mathcal{I}(T, L)$$

such that

$$\sigma(m_l) = sign(s_{i_l}, m_l), \quad l = 1, 2, \ldots, L.$$

Let $(n, k, gen, sign, test, iden)$, T and an integer L, $0 < L \leq nT$ be given. Consider the following experiment given L different messages $m_1, m_2, \ldots, m_L$:

1. Generate $(pk, (s_1, \ldots, s_n), aux)$ using gen.
2. Choose $i_1, i_2, \ldots, i_L \in \mathcal{I}(T, L)$ uniformly at random.
3. Let $hist_l(\underline{m})$ be defined by
$$\sigma(m_j) = sign(s_{i_j}, m_j) \qquad \text{for } j = 1, \ldots, L.$$

Let AUX be the random variable of the authority's auxiliary information (defined on the probability space induced by gen). Let ID be the uniformly distributed random variable taking the value $(i_1, i_2, \ldots, i_L)$.

From the definition of unconditional anonymity, the following lemma is obtained.

Lemma 17 *If the group signature scheme $(n, k, gen, sign, test, iden)$ provides anonymity for signing T messages, then for any (L, T)-history $hist_L(\underline{m})$, ID is uniformly distributed on $\mathcal{I}(T, L)$. Especially, the conditional entropy of ID given $hist_L(\underline{m})$ is*

$$H(ID \mid hist_L(\underline{m}) = \log_2 |\mathcal{I}(T, L)| = \log_2 \left(\frac{(Tn)!}{(T!)^n} \right).$$

Theorem 18 *If the group signature scheme $(n, k, gen, sign, test, iden)$ provides anonymity for signing T messages and signer identification, then*
$$H(AUX) \geq Tn(\log n - 1).$$

Proof Let $L = Tn$, and consider an (L, T)-history, $h = hist_L(\underline{m})$. The entropy of AUX can be written

$$H(AUX \mid h) = H(AUX|ID, h) + H(ID \mid h) - H(ID|AUX, h).$$

Since the scheme provides signer identification $H(ID|AUX, h) = 0$ and thus

$$H(AUX) \geq H(AUX \mid h) = H(ID \mid h) + H(AUX|ID, h) \geq H(ID \mid h).$$

From the lemma above,

$$H(ID \mid h) = \log \frac{(Tn)!}{(T!)^n}.$$

Stirlings Formula

$$n! \approx e^{-n} n^n \sqrt{2\pi n}$$

gives

$$\log \frac{(Tn)!}{(T!)^n} \approx Tn \log n + \log \sqrt{2\pi Tn} - n \log \sqrt{2\pi T} \geq Tn(\log n - 1).$$

This completes the proof. □

This bound can be interpreted as follows. The authority needs some information corresponding to each signature that each member is allowed to make — in total nT pieces. Each of these must be be linked to the actual member — this requires $\log n$ bits.

5.3 Comparison with Suggested Scheme

In the scheme presented in Section 4.2 the length of the secret key is $2(T+1)\log q$ bits. Taking into account that this scheme allows up to q members this scheme meets the lower bound except for a factor of 2 originating from double signing.

The length of the auxiliary information is $n(T+1)\log q$ bits. Again this meets the lower bound.

Finally, it should be mentioned that the length of the signatures in the scheme of Section 4.2 grows linearly in the number of group members and signatures. However, this need not always be the case (e.g. see [CH91] for a scheme with constant length signatures).

6 Conclusion

We have given a detailed definition of group signature schemes providing unconditional anonymity, and presented a scheme which satisfies this definition (the security against forgery relied on some assumptions). This scheme has the disadvantage that the length of the secret keys and the auxiliary information grows linearly in the number of signatures, but as shown in Section 5 this cannot be avoided. Thus group signatures with unconditional anonymity have some limits which might make them less attractive in some applications.

A One out of n Witnesses

This appendix sketches Schoenmakers method for proving knowledge of one out of many witnesses given in [S93] and further elaborated on in [CDS94].

Let G_q denote a group of prime order q and let g be a generator of G_q. The common input to the prover and verifier is $(g, h_1, \ldots, h_n)$ for some $n \in \mathbb{N}$, where each $h_i \in G_q$. Let $h_i = g^{x_i}$, $i = 1, 2, \ldots, n$. The protocol in Figure 3 is a proof of knowledge of $x_i, i = 1, 2, \ldots, n$.

Now suppose that the prover only knows one of the n witnesses. Given one of x_i's as secret input, the prover shows that he knows w such that for some $i \in \{1, 2, \ldots, n\}$: $h_i = g^w$. The protocol is sketched in Figure 4 for the case $w = x_1$.

Intuitively, the challenge $c = \sum_1^n d_i$, gives the prover freedom to choose $(n-1)$ of the d_j's. Therefore, the prover must know at least one of the n witnesses. However the prover's messages do not reveal any information about which d_j's the prover chooses initially.

Proposition 19 ([S93]) *The protocol in Figure 4 is a witness indistinguishable proof of knowledge (see [FS90]) of w satisfying*

$$h_i = g^w \qquad \textit{for some } i \in \{1, 2, \ldots, n\}.$$

Remark An extension of this protocol allows the prover to show that he knows at least k out of n secret keys (see [S93]).

$$\begin{array}{lcl}
\textsf{P} & & \textsf{V}\\
s_1, s_2, \ldots, s_n \in_{\mathcal{R}} \mathbb{Z}_q^* & & \\
(a_i \leftarrow g^{s_i})_{i=1,2,\ldots,n} & & \\
& \xrightarrow{(a_1, a_2, \ldots, a_n)} & \\
& & d_1, d_2, \ldots, d_n \in_{\mathcal{R}} \mathbb{Z}_q^* \\
& \xleftarrow{(d_1, d_2, \ldots, d_n)} & \\
(r_i \leftarrow s_i + d_i x_i)_{i=1,2,\ldots,n} & & \\
& \xrightarrow{(r_1, r_2, \ldots, r_n)} & \\
& & \left(g^{r_i} \overset{?}{=} a_i h_i^{d_i}\right)_{i=1,2,\ldots,n}
\end{array}$$

Figure 3: Proving knowledge of n witnesses

$$\begin{array}{lcl}
\textsf{P} & & \textsf{V}\\
s_1, s_2, \ldots, s_n \in_{\mathcal{R}} \mathbb{Z}_q^* & & \\
d_2, \ldots, d_n \in_{\mathcal{R}} \mathbb{Z}_q^* & & \\
a_1 \leftarrow g^{s_1} & & \\
\left(a_i \leftarrow g^{s_i} h_i^{-d_i}\right)_{i=2,\ldots,n} & & \\
& \xrightarrow{(a_1, a_2, \ldots, a_n)} & \\
& & c \in_{\mathcal{R}} \mathbb{Z}_q^* \\
& \xleftarrow{c} & \\
d_1 \leftarrow c - \sum_2^n d_i & & \\
r_1 \leftarrow s_1 + d_1 x_1 & & \\
(r_i \leftarrow s_i)_{i=2,\ldots,n} & & \\
& \xrightarrow{(d_1, \ldots, d_n; r_1, \ldots, r_n)} & \\
& & c \overset{?}{=} \sum_1^n d_i \\
& & \left(g^{r_i} \overset{?}{=} a_i h_i^{d_i}\right)_{i=1,2,\ldots,n}
\end{array}$$

Figure 4: Proving knowledge of one of n witnesses.

B The Complete Scheme

Figure 5 shows the interactive protocol in the general situation of n members. The corresponding signature on $m \in \mathcal{M}$ is

$$\Sigma = (\sigma, (r_{kj})_{j=1,\ldots,n,k=0,\ldots,T}, (d_j)_{j=1,\ldots,n}).$$

$$\begin{array}{lcl}
\mathsf{P} & & \mathsf{V} \\
t_{jk} \in_{\mathcal{R}} \mathbb{Z}_q^* & & \\
d_j \in_{\mathcal{R}} \mathbb{Z}_q^* \text{ for } j \neq i & & \\
a_i \leftarrow \prod_{k=0}^{T} g_k^{t_{ik}} & & \\
a_j \leftarrow \prod_{k=0}^{T} g_k^{t_{jk}} h_j^{-d_j} & & \\
\tau_i \leftarrow \sum_{k=0}^{T} t_{ik} m^k & & \\
\tau_j \leftarrow \sum_{k=0}^{T} t_{jk} m^k - d_j \sigma & & \\
 & \begin{array}{c}(a_1, \ldots, a_n) \\ (\tau_1, \ldots, \tau_n) \\ \xrightarrow{\qquad\qquad} \end{array} & \\
 & & c \in_{\mathcal{R}} \mathbb{Z}_q^* \\
 & \xleftarrow{\qquad c \qquad} & \\
d_i = c - \sum_{j \neq i} d_j \bmod q & & \\
r_{ik} \leftarrow t_{ik} + d_i s_{ik} \bmod q & & \\
r_{jk} \leftarrow t_{jk} & & \\
 & \xrightarrow{\quad (r_{jk})_{j,k} \quad} & \\
 & & c \overset{?}{=} \sum_{k=0}^{T} d_k \bmod q \\
 & & \sum_{k=0}^{T} r_{jk} m^k \overset{?}{=} \tau_j + d_j \sigma \\
 & & \prod_{k=0}^{T} g_k^{r_{jk}} \overset{?}{=} a_j h_j^{d_j}
\end{array}$$

Figure 5: Interactive proof that σ is correct with respect to one of $h_1, h_2, \ldots, h_n$ — here P knows the secret key corresponding to h_i. The subscripts j and k are over $\{1, 2, \ldots, n\}$ and $\{0, 1, \ldots, T\}$, respectively.

References

[Boy89b] C. Boyd. Digital Multisignatures. In *Cryptography and Coding*, pages 241 – 246, 1989.

[CDS94] R. Cramer, I. Damgård and B. Schoenmakers. Proofs of Partial Knowledge and Simplified Design of Witness Hiding Protocols. In *Advances in Cryptology - proceedings of CRYPTO 94*.

[CP94] L. Chen and T. P. Pedersen New Group Signature Schemes. In *Advances in Cryptology - Proceedings of Eurocrypt '94*.

[CH91] D. Chaum and E. van Heijst. Group Signatures. In *Advances in Cryptology - proceedings of EUROCRYPT 91*, Lecture Notes in Computer Science #547, pages 257-265. Springer-Verlag, 1991.

[CH89] R. A. Croft and S. P. Harris. Public-Key Cryptography and Reusable Shared Secrets. In *Cryptography and Coding*, pages 189 – 201, 1989.

[D93] Y. Desmedt. Threshold Cryptosystems. In *Advances in Cryptology - proceedings of AUSCRYPT 92*, Lecture Notes in Computer Science #718, pages 3–14, 1993.

[FFS88] U. Feige, A. Fiat, and A. Shamir. Zero-knowledge Proofs of Identity. *Journal of Cryptology*, 1(2):77–94, 1988.

[FS87] A. Fiat and A. Shamir. How to Prove Yourself: Practical Solutions to Identification and Signature Problems. In *Preproceedings of EUROCRYPT 86*, pages 186 – 194. 1987.

[FS90] U. Feige and A. Shamir. Witness Indistinguishable and Witness Hiding Protocols. In *Proceedings of the 22nd Annual ACM Symposium on the Theory of Computing*, pages 416 – 426, 1990.

[GMR88] S. Goldwasser, S. Micali, and R. L. Rivest. A Digital Signature Scheme Secure Against Adaptive Chosen Message Attack. *SIAM Journal on Computing*, 17(2):281 – 308, April 1988.

[H92] E. van Heijst. *Special Signature Schemes.* PhD thesis, CWI, 1992.

[HP93] E. van Heyst and T. P. Pedersen. How to make efficient fail-stop signatures. In *Advances in Cryptology—EUROCRYPT '92*, volume 658 of *Lecture Notes in Computer Science*, pages 366–378. Springer-Verlag, 1993.

[K81] D. E. Knuth. *The Art of Computer Programming: Seminumerical Algorithms*, volume II. Addison-Wesley, 2 edition, 1981.

[O88] T. Okamoto. A Digital Multisignature Scheme Using Bijective Public-Key Cryptosystems. *ACM Trans. on Comp. Sys.*, 6(8):432 – 441, 1988.

[O93] T. Okamoto. Provably secure and practical identification schemes and corresponding signature schemes. In *Advances in Cryptology—CRYPTO '92*, volume 740 of *Lecture Notes in Computer Science*, pages 31–53, Berlin, 1993. Springer-Verlag.

[OO93] K. Ohta and T. Okamoto. A Digital Multisignature Scheme Based on the Fiat-Shamir Scheme. In *Advances in Cryptology - proceedings of ASIACRYPT 91*, Lecture Notes in Computer Science #739, pages 139 – 148. Springer-Verlag, 1993.

[S93] B. Schoenmakers. Efficient Proofs of Or. Manuscript, 1993.

[TW87] M. Tompa and H. Woll. Random Self-reducibility and Zero Knowledge Interactive Proofs of Possession of Information. In *Proceedings of the 28th IEEE Symposium on the Foundations of Computer Science*, pages 472–482, 1987.