Efficiency Limitations of Σ-Protocols for Group Homomorphisms Revisited

Björn Terelius and Douglas Wikström

KTH Royal Institute of Technology, Stockholm, Sweden
{terelius,dog}@csc.kth.se

Abstract. We study the problem of constructing efficient proofs of knowledge of preimages of general group homomorphisms. We simplify and extend the recent negative results of Bangerter et al. (TCC 2010) to *constant round* (from three-message) generic protocols over *concrete* (instead of generic) groups, i.e., we prove lower bounds on both the soundness error and the knowledge error of such protocols. We also give a precise characterization of what can be extracted from the prover in the direct (common) generalization of the Guillou-Quisquater and Schnorr protocols to the setting of general group homomorphisms.

Then we consider some settings in which these bounds can be circumvented. For groups with no subgroups of small order we present: (1) a three-move honest verifier zero-knowledge argument under some set-up assumptions and the *standard* discrete logarithm assumption, and (2) a Σ-proof of both the order of the group and the preimage. The former may be viewed as an offline/online protocol, where all slow cut-and-choose protocols can be moved to an offline phase.

1 Introduction

An honest-verifier zero-knowledge proof of knowledge is a two party protocol where a prover demonstrates knowledge of a secret and the verifier does not learn anything he can not compute himself. A protocol is complete if the honest verifier accepts when interacting with the honest prover. The prover is honest-verifier zero-knowledge if the view of the honest verifier interacting with the prover can be simulated efficiently.

The probability that a malicious prover convinces the verifier of a false statement is called the *soundness error*. The prover is said to *know* the secret when there exists an efficient *extractor* which after interacting with the prover outputs the secret. On the other hand, a malicious prover who does not know the secret still has some probability, called the *knowledge error*, of convincing the verifier. Making the knowledge error as small as possible at low computational and communication costs is an important goal in the construction of protocols.

Guillou's and Quisquater's [12] protocol for proving knowledge of an RSA root and Schnorr's well-known proof of knowledge of a discrete logarithm [14] in a group of prime order q are particularly nice proofs of knowledge. Recall that in Schnorr's proof of knowledge of w such that $y = g^w$, the prover first

I. Visconti and R. De Prisco (Eds.): SCN 2012, LNCS 7485, pp. 461–476, 2012.

commits to randomness $r \in \mathbb{Z}_q$ by sending $\alpha = g^r$ to the verifier. The verifier then sends a random challenge $c \in \mathbb{Z}_q$ and the prover responds with $d = cw + r \bmod q$. To extract the secret w, the extractor only needs to sample interactions until it finds two accepting transcripts (α, c, d) and (α, c', d'), where $c' \neq c$, and compute $w = (d - d')(c - c')^{-1} \bmod q$. Similar protocols with exponentially small knowledge errors can be constructed for statements involving several group elements. The protocols exhibiting this form, with three messages, extraction from two accepting transcripts sharing the first message, and a strong form of honest-verifier simulation, are called *Σ-proofs* [7].

There is a simple and well-known generalization of Schnorr's protocol that can be executed over a group of unknown order, or even prove knowledge of a preimage $w \in \mathcal{G}$ of an element $y \in \mathcal{H}$ under a group homomorphism $\phi : \mathcal{G} \to \mathcal{H}$. Unfortunately, the resulting protocol has soundness and knowledge error $1/2$. These errors can be reduced to 2^{-n} by n repetitions, but this approach is impractical because it increases both the computation and communication costs considerably. Thus, a natural question is whether there exist protocols with small knowledge error and a structure similar to the Guillou-Quisquater and Schnorr proofs, but which works for any groups $\mathcal{G}$ and $\mathcal{H}$ and group homomorphism ϕ.

1.1 Previous Work

Proofs of knowledge over groups of unknown order have been studied before and both positive and negative results are known. Shoup [15] gave a three-message protocol for proving knowledge of w such that $y = w^{2^m}$ in an RSA group and showed that a knowledge error of $1/2$ is optimal.

Bangerter, Camenisch and Krenn [1] considered generic Σ-protocols, i.e., three-message protocols where the prover computes integer linear combinations of the secret witness and random elements, and possibly applies the homomorphism. They proved a lower bound on the knowledge error of such protocols in the *generic group model*. They also showed that the lower bounds hold for some natural generalizations of Schnorr's protocol in concrete groups. Specifically, they generalized Shoup's result on powers in RSA groups to arbitrary exponents and proved a lower bound of the knowledge error of exponentiation homomorphisms $\phi(w) = g^w$ in groups of unknown order, under mild assumptions on the extractor.

There is a vast literature on constructing protocols for specific groups and homomorphisms, with and without computational assumptions, and we only mention a few. Fujisaki and Okamoto [9] created an integer commitment scheme (subsequently generalized and corrected by Damgård and Fujisaki [8]) along with an argument of knowledge of the opening of the commitment under the strong RSA assumption. The argument of knowledge is actually a Σ-protocol of $(w, r) \in \mathbb{Z}^2$ such that $y = g^w h^r$. Other protocols [3] have been proposed based on the same principles.

Bangerter, Camenisch and Maurer [2] proposed two protocols for proving knowledge of a preimage in a group with unknown order. The first protocol assumed that the players are given an auxiliary *pseudo-preimage* (e', u') such that $y^{e'} = \phi(u')$ where e' is a prime larger than any challenge. Thus, if the ordinary extractor has found a pseudo-preimage (e, u) such that $y^e = \phi(u)$, one knows that $\gcd(e, e') = 1$, which implies that there exist integers a, b such that $ae + be' = 1$. Hence $y = \phi(u)^a \phi(u')^b = \phi(au + bu')$. This method of finding a proper preimage is sometimes called "Shamir's trick". Their second protocol is based on running two Σ-protocols in parallel, one being a Damgård-Fujisaki commitment. This protocol was later criticized by Kunz-Jacques et al. [13], since a verifier can choose a bad RSA-modulus. The prefix protocol of Wikström [16] used to establish a safe modulus suffers from the same flaw, though his main protocol remains secure.

Groups of unknown order have been used in several identification schemes. Brickell and McCurley [5] constructed an identification scheme that is secure as long as either discrete logarithms or factoring is hard. In their scheme, the prover knows the (prime) order of the generator g but the verifier does not. Their protocol share some similarities with our Protocol 3 for proving knowledge of a multiple of the order and subsequent proof of knowledge of the discrete logarithm in Protocol 4. Girault et al. [10] suggested an identification scheme that uses Schnorr's protocol in a group of unknown order (cf. our Protocol 1) to prove knowledge of a secret w for the public identity g^w. However, the security model in [10] only requires the extractor to output a witness if the attacker can forge the proof for a *randomly chosen* public identity with non-negligible probability, and they note that this protocol is not a proof of knowledge.

Cramer and Damgård [6] recently gave a method for *amortizing* the cost of cut-and-choose protocols over *several instances*, which reduces both the computational complexity and the size of the proof. This does not contradict our lower bounds since we only consider a single instance of the problem of proving knowledge of w such that $y = \phi(w)$.

1.2 Our Results

We begin by giving a precise characterization of the knowledge of the prover in the well-known generalization of the Guillou-Quisquater and Schnorr-protocols to the setting of group homomorphisms. We essentially prove that if a prover convinces the honest verifier with probability p, then we can extract $e \approx 1/p$ and u such that $y^e = \phi(u)$ in time $O(T(n)e)$ for some polynomial $T(n)$.

Then we consider a generalization of Bangerter et al.'s [1] class of generic Σ-protocols for proving knowledge of a preimage of a group homomorphism. We extend their model from three-message protocols to protocols with any *constant number of rounds* with challenges that could depend on previous messages and we prove lower bounds on both the knowledge error and the soundness error of protocols from this class.

- Under mild assumptions, we show that a malicious prover who knows a pseudo-preimage $u = 2w + \sigma$ of $y = \phi(w)$, where $\phi(\sigma) = 1$, can convince the verifier with some *constant* probability, where the constant depends on the protocol. Thus, an efficient extractor for w can, in general, not be constructed unless w can be computed from $(2, u)$. This generalizes the result for *three-message* protocols given in [1] to *constant-round* protocols. Furthermore, our analysis is simpler and does not rely on the generic group model.

- We show that if the group $\mathcal{H}$ has an element γ of small order and the verifier uses a natural type of verification test, then the proof does not even need to be sound. In particular, we construct a malicious prover who knows γ and w such that $y = \gamma\phi(w)$, yet manages to convince the verifier that $y = \phi(w')$ for some w'. The technique is similar to that of Kunz-Jacques et al. [13].

These results shed some new light on what is needed from a protocol for proving knowledge of a preimage.

Finally, we investigate two ways of circumventing the negative results. We present two honest-verifier zero-knowledge protocols that allow (a precisely characterized) partial knowledge extractor in general, and a proper knowledge extractor under assumptions on the order of the underlying group.

- Our first protocol, Protocol 2, only works for the exponentiation homomorphism under the *standard* discrete logarithm assumption in a different group of *known prime order*, and requires set-up assumptions. We show that if a prover convinces the verifier with probability p, then we can extract $e \approx 1/p$ and u such that $y^e = g^{eu}$. In contrast to the basic protocol, Protocol 1, this may, loosely, be viewed as an argument of knowledge of the preimage up to small subgroups, and an argument of knowledge of w when the order of the underlying group contains no small factors. The set-up assumptions require cut-and-choose protocols, but these can be executed in an offline phase.
- Our second protocol, Protocol 4, works for any group homomorphism, but requires that the prover knows the order of the underlying group (in fact it proves knowledge of both a multiple of the order and the preimage). We show that if a prover convinces the verifier with probability p, then we can extract $e \approx 1/p$ and u such that $y^e = \phi(u)$ and every factor of e divides the order of y. Again, if the order of the group contains no small factors, this gives a proof of knowledge.

Although neither protocol solves the problem of constructing an efficient proof of knowledge of a preimage in general, our protocols suffice in certain situations. The first protocol can, e.g., be used to prove knowledge of an integer w such that $y_0 = g_0^w$ and $y_1 = g_1^w$, where g_0 and g_1 are generators of two groups of *distinct* large prime orders.

1.3 Notation

Throughout the paper, we use the standard definitions of zero-knowledge protocols [11] and proofs of knowledge [4]. Let n and n_c to denote the security parameter and bit-size of challenges. When executing proofs of knowledge of exponents, we denote by n_w the bit-size of the exponent and n_r the additional bits in the randomizer. We let $\mathcal{G}$ and $\mathcal{H}$ denote abelian groups and let $\phi : \mathcal{G} \to \mathcal{H}$ be a group homomorphism. Since ϕ will often be an exponentiation homomorphism, we will write $\mathcal{G}$ additively and $\mathcal{H}$ multiplicatively.

We sometimes use overlining to indicate that something is a vector or a list, e.g., $\overline{z} \in \mathcal{G}^n$ is a list of n elements in $\mathcal{G}$. We write $\phi(\overline{z})$ as a short-hand for the list obtained by applying the homomorphism to each component of $\overline{z}$. Similarly, if $w \in \mathcal{G}$ and $\overline{\alpha} \in \mathbb{Z}^n$, we use the convention that $w^{\overline{\alpha}} = (w^{\alpha_1}, \ldots, w^{\alpha_n})$. For a list or vector $\overline{v}$, the number of components is denoted by $dim(\overline{v})$. We adopt the definition of a *pseudo-preimage* of Bangerter et al. [1].

Definition 1. *Let $\phi : \mathcal{G} \to \mathcal{H}$ be a group homomorphism and let y be an element of $\mathcal{H}$. A* pseudo-preimage *of y is a pair $(e, u) \in \mathbb{Z} \times \mathcal{G}$ such that $y^e = \phi(u)$.*

The following observation follows immediately from the definition.

Lemma 1. *Let $y = \phi(w)$. Any pseudo-preimage (e, u) of y must have the form $u = ew + \sigma$, where $\phi(\sigma) = 1$, i.e., σ is an element in the kernel of ϕ.*

2 Tight Analysis of the Basic Protocol

Below we recall the natural generalization of Schnorr's protocol for proving knowledge of a discrete logarithm to the setting where the prover instead needs to show that it knows a preimage $w \in \mathcal{G}$ of $y \in \mathcal{H}$ under a homomorphism $\phi : \mathcal{G} \to \mathcal{H}$.

Protocol 1 (Basic Protocol)
Common Input. An element $y \in \mathcal{H}$ and a homomorphism $\phi : \mathcal{G} \to \mathcal{H}$ of abelian groups $\mathcal{G}$ and $\mathcal{H}$.
Private Input. An element $w \in \mathcal{G}$ such that $y = \phi(w)$.

1. $\mathcal{P}$ chooses $r \in \mathcal{G}$ randomly[1] and hands $\alpha = \phi(r)$ to $\mathcal{V}$.
2. $\mathcal{V}$ chooses $c \in [0, 2^{n_c} - 1]$ randomly and hands c to $\mathcal{P}$.
3. $\mathcal{P}$ computes $d = cw + r$ in $\mathcal{G}$ and hands d to $\mathcal{V}$.
4. $\mathcal{V}$ verifies that $y^c\alpha = \phi(d)$

We obtain Schnorr's proof of knowledge of a discrete logarithm in a group $\langle g \rangle$ of prime order q by setting $\mathcal{G} = \mathbb{Z}_q$, $\mathcal{H} = \langle g \rangle$ and $\phi(w) = g^w$. When the order of g is unknown, we treat $\phi(w) = g^w$ as a homomorphism from $\mathcal{G} = \mathbb{Z}$ to $\langle g \rangle$. More precisely, we assume that $w \in [0, 2^{n_w} - 1]$ and choose $r \in [0, 2^{n_w+n_c+n_r} - 1]$ which is statistically close to uniform modulo the order of g if n_r is large enough. As a slight generalization, we may let the verifier check that $d \in [0, 2^{n_w+n_c+n_r} - 1]$. This modification does not affect Theorem 1 and 2 below, except that the protocol is overwhelmingly complete rather than perfectly complete. We use this variant in Protocol 3.

[1] It is sufficient that the distribution of r is statistically close to uniform in $\mathcal{G}$ or even that the distribution of $cw + r$ is statistically close to $c'w' + r'$ for any c, c', w, w' allowed in the protocol.

It is well-known that Protocol 1 is honest-verifier zero-knowledge, so we state that theorem without proof. It is also well-known that the protocol, in general, is not a proof of knowledge of w such that $y = \phi(w)$. On the other hand, it is clear that the prover shows that it knows *something* related to w and as far as we know there is no *precise* characterization in the literature of what can, and can not, be extracted from a convincing prover. Such a characterization is useful in the rest of the paper.

Theorem 1 (Zero-Knowledge). *Protocol 1 is complete and honest-verifier statistical (perfect) zero-knowledge if for each w and each $c \in [0, 2^{n_c} - 1]$, the distributions of $cw + r$ and r are statistically close (identical).*

Informally, the following theorem says that if a prover convinces the verifier with probability p, then we can extract $e \approx 1/p$ and u such that $y^e = \phi(u)$. In other words, the more successful a prover is, the more it needs to know about the preimage of y. The extractor depends on a parameter ϵ that controls how close e is to $1/p$. The reader may think of $\epsilon = \frac{1}{4}$. The proof of Theorem 2 is given in the full version.

Theorem 2 (Extraction and Soundness). *There exists an extractor $\mathcal{E}_\epsilon$, parameterized by $\epsilon < 1/2$ with $\epsilon^{-1} \in \mathsf{Poly}(n)$, using any PPT prover $\mathcal{P}^*$ as an oracle such that if $\mathcal{P}^*$ convinces $\mathcal{V}$ with probability $\Delta > \kappa$ on common input (y, ϕ), then $\mathcal{E}_\epsilon(y, \phi)$ extracts an integer $0 < e \leq \frac{1}{(1-\epsilon)^2 \Delta}$ and $u \in \mathcal{G}$ such that $y^e = \phi(u)$. The extractor runs in expected time $O\big(\epsilon^{-2} T(n)/(\Delta - \kappa)\big)$, where $T(n)$ is a polynomial (independent of ϵ) and the knowledge error κ is defined as $\kappa = 2^{1-n_c}/\epsilon$.*

The following theorem shows that we can not hope to find an extractor which extracts significantly more. This theorem is very similar to a theorem in [1], but differs in that their formulation concerns a *restricted* class of extractors.

Theorem 3. *A malicious prover knowing only y, e and $ew + \sigma$ such that $y^e = \phi(ew + \sigma)$, where $\phi(\sigma) = 1$, can convince the verifier with probability $1/e - \mathsf{negl}(n)$ if the distributions of $r + \frac{c}{e}\sigma$ and r are statistically close for each $c \in [0, 2^{n_c} - 1]$ such that $e \mid c$.*

Proof. The prover chooses r and sends $\phi(r)$ as usual. After receiving the challenge c, the prover halts if $e \nmid c$, and responds with

$$d' = \frac{c}{e}(ew + \sigma) + r = cw + r + \frac{c}{e}\sigma$$

otherwise. We clearly have $\phi(d') = \phi(d)$, since $d = cw + r$. Furthermore, the verifier notices that the prover is cheating with negligible probability, since d' is statistically close in distribution to a correctly formed response. □

If c is not chosen uniformly from an interval as in Protocol 1, e.g., if e never divides c, we are not able to apply the previous theorem directly. However, it is always possible to find c^* such that $e \mid (c - c^*)$ with probability $1/e$. The malicious prover can then answer with $d' = r + \frac{(c-c^*)}{e}(ew + \sigma) = r - c^*w + \frac{(c-c^*)}{e}\sigma + cw$ whenever e divides $c - c^*$. This is indistinguishable from the real response provided that the distributions of $r - c^*w + \frac{(c-c^*)}{e}\sigma$ and r are statistically close, which happens for example if r is chosen from a sufficiently large interval. We generalize this approach in the next section.

3 Lower Bound on the Knowledge Error

Suppose that we wish to prove knowledge of w such that $y = \phi(w)$ in a group of unknown order. Bangerter et al. [1] defined generic Σ-protocols as the class of protocols where the verifier only sends random challenges and the prover only uses the homomorphism and linear combinations of group elements to generate his responses. They proved a lower bound on the knowledge error in the *generic group model* for such protocols and gave concrete examples of protocols where the bounds hold in the plain model.

In this section we generalize their result to any *constant round* protocol of the same type (Σ-protocols have three messages) and give the verifier more freedom in how it chooses its challenges. We also provide a novel analysis that does not rely on the generic group model.

Definition 2. *Consider a protocol for proving knowledge of a preimage, executed by a prover $\mathcal{P}$ and a verifier $\mathcal{V}$ with common input $\mathcal{G}$, $\mathcal{H}$, a group homomorphism $\phi : \mathcal{G} \to \mathcal{H}$ and $y \in \mathcal{H}$, and private input w such that $y = \phi(w)$. We call it a* constant-round generic protocol *if in the ith round:*

1. *$\mathcal{V}$ sends integer vectors $\overline{\alpha}^{(i)}$, $\overline{\beta}^{(i)}$ chosen according to some distributions, possibly depending on the messages in the earlier rounds, and*
2. *$\mathcal{P}$ responds with $\overline{t}^{(i)} = \phi\big(A^{(i)}\overline{r} + \overline{\alpha}^{(i)}w\big)$ and $\overline{s}^{(i)} = B^{(i)}\overline{r} + \overline{\beta}^{(i)}w$,*

where $A^{(i)}$ and $B^{(i)}$ are public integer matrices and $\overline{r}$ denotes the random tape, viewed as a vector of elements in $\mathcal{G}$, given to the prover. The verifier may use any polynomial time test to decide whether or not to accept the proof.

Remark 1. Readers familiar with the generic Σ-protocols in [1] may notice that their definition is a special case of Definition 2, obtained by restricting the protocol to two rounds. In the first round, $\overline{\alpha}^{(0)}$ and $\overline{\beta}^{(0)}$ are part of the protocol specification. In the second round, $\overline{\alpha}^{(1)}$ and $\overline{\beta}^{(1)}$ are defined (using the notation of [1]) as

$$\overline{\alpha}_j^{(1)} = f_j + \sum g_{ji}c_i \quad \text{and} \quad \overline{\beta}_j^{(1)} = d_j + \sum e_{ji}c_i \ ,$$

where f_j, d_j, g_{ji} and e_{ji} are public constants and $c_1, \ldots, c_p$ are the challenges chosen by the verifier.

In the following, we consider all of the rounds simultaneously. To simplify the exposition, we define $\overline{\alpha}$ and $\overline{t}$ as the column vectors formed by all elements of $\overline{\alpha}^{(i)}$ and $\overline{t}^{(i)}$ respectively, and let A be the matrix formed by the rows of all $A^{(i)}$. We define B, $\overline{\beta}$ and $\overline{s}$ analogously. This permits us to write all the equations concisely as

$$\overline{t} = \phi(A\overline{r} + \overline{\alpha}w) \quad \text{and} \quad \overline{s} = B\overline{r} + \overline{\beta}w \ .$$

Note that a malicious prover can generate $\overline{t}$ as $t_i = \phi\left((A\overline{r})_i\right)\phi(w)^{\alpha_i}$. This shows that the protocol could equivalently have been designed to just send $\phi\left((A\overline{r})_i\right)$ since the powers of $\phi(w)$ can be computed from the public information and added or removed from whatever the prover sends. It is, however, not as easy to generate $\overline{s}$, since the prover does not know w. Intuitively, we want to avoid this problem by constructing a prover that "almost" knows $2w$ and only answers when the challenge has a given parity.

Theorem 4. *Let $(\mathcal{P}, \mathcal{V})$ be a constant-round generic protocol as in Definition 2. There exists a prover $\mathcal{P}^*$ that takes as input the groups $\mathcal{G}$, $\mathcal{H}$, a homomorphism $\phi : \mathcal{G} \to \mathcal{H}$, an integer vector $\overline{v}^*$, a group element $y \in \mathcal{H}$, and a pseudo-preimage $(2, u)$ such that $u = 2w + \sigma$ where $y = \phi(w)$ and $\phi(\sigma) = 1$.*

Define $S = \{\overline{\beta} : \exists\overline{v}$ such that $\overline{\beta} = B\overline{v}\}$ as the set of challenges $\overline{\beta}$ in the protocol that have preimages under B. For each $\overline{\beta} \in S$, let $\overline{v}_{\overline{\beta}}$ denote a preimage[2] *of $\overline{\beta}$ under B, i.e., $\overline{\beta} = B\overline{v}_{\overline{\beta}}$. Let $T \subset \{\overline{v}_{\overline{\beta}} : \overline{\beta} \in S\}$ be a subset of the preimages $\overline{v}_{\overline{\beta}}$ such that for every $\overline{v}, \overline{v}' \in T$ for which $\overline{v} = \overline{v}' \bmod 2$, the statistical distance between the distributions of $\overline{r}$ and $\overline{r}^* = \overline{r} + \overline{v}'w - \frac{\overline{v}-\overline{v}'}{2}\sigma$ is at most ϵ.*

If the integer vector $\overline{v}^* \in T$ is chosen such that $\Pr[\overline{v}_{\overline{\beta}} = \overline{v}^* \bmod 2] \geq 2^{-\dim(\overline{v})}$, when the probability is taken over the choice of $\overline{\beta}$ conditioned on $\overline{\beta} \in S$ and $\overline{v}_{\overline{\beta}} \in T$, then $\mathcal{P}^*$ convinces $\mathcal{V}$ with probability at least

$$\Pr[\overline{\beta} \in S] \cdot \Pr[\overline{v}_{\overline{\beta}} \in T \mid \overline{\beta} \in S] \cdot 2^{-\dim(\overline{v})} - \epsilon \ ,$$

where $\dim(\overline{v})$ is the (constant) number of components in $\overline{v}$.

About the Assumptions. To ensure that $\overline{\beta}w$ is completely hidden in $B\overline{r} + \overline{\beta}w$, we expect that every integer vector $\overline{\beta}$ has a preimage under B, or in other words that the lattice spanned by B contains all points with integer coordinates. If this is the case, then $\Pr[\overline{\beta} \in S] = 1$ and moreover, the preimages $\overline{v}_{\overline{\beta}}$ can be chosen such that $\overline{v}_{\overline{\beta}} = \overline{v}_{\overline{\beta}'} \bmod 2$ whenever $\overline{\beta} = \overline{\beta}' \bmod 2$. Hence we can choose $\overline{v}^*$ such that $\Pr[\overline{v}_{\overline{\beta}} = \overline{v}^* \bmod 2] \geq 2^{-\dim(\overline{\beta})}$ rather that $\Pr[\overline{v}_{\overline{\beta}} = \overline{v}^* \bmod 2] \geq 2^{-\dim(\overline{v})}$.

The set T encodes a subset of preimages $\overline{v}_{\overline{\beta}}$ for which the distributions $\overline{r}$ and $\overline{r}^*$ are statistically close. If the components of $\overline{r}$ are chosen from a sufficiently large subset of $\mathcal{G}$, then T contains all preimages, so $\Pr[\overline{v}_{\overline{\beta}} \in T \mid \overline{\beta} \in S] = 1$. This

[2] There may be several integer vectors $\overline{v}$ such that $\overline{\beta} = B\overline{v}$. Let $\overline{v}_{\overline{\beta}}$ be some choice among those preimages.

happens for example if $\overline{r}$ is chosen uniformly from a finite group, or if $\mathcal{G} = \mathbb{Z}$ and $\overline{r}$ is chosen from a large interval. We remark that this assumption was also made implicitly for $\mathcal{G} = \mathbb{Z}$ by Bangerter et al. [1]. Using these two stronger assumptions gives us the following corollary.

Corollary 1. *Let $(\mathcal{P}, \mathcal{V})$ be a constant-round generic protocol as in Theorem 4 and suppose that every integer vector has a preimage under B and that the randomness $\overline{r}$ is chosen from a sufficiently large subset of $\mathcal{G}$ so that $T = \{\overline{v}_{\overline{\beta}} : \overline{\beta} \in S\}$. Then the malicious prover $\mathcal{P}^*$, who knows $\overline{v}^*$, $y = \phi(w)$, and a pseudo-preimage $(2, u)$, convinces $\mathcal{V}$ with probability at least $2^{-\dim(\overline{\beta})} - \epsilon$.*

Interpretation of Theorem 4. Recall that Protocol 1 and Theorem 2 showed that we can extract a pseudo-preimage in any group. This is sufficient if a preimage can be computed from the pseudo-preimage, which is the case in groups of known prime order, for example. On the other hand, computing w from $2w + \sigma$ in $\mathbb{Z}_N^*$ where N is a safe RSA modulus, would imply computing a multiple of the group order $\phi(N)$. This is believed to be infeasible, even for machines running in expected polynomial time. Theorem 4 shows that (under some plausible assumptions) we can not extract more than the pseudo-preimage, since that is all the malicious prover is given. In particular, it gives a lower bound on the knowledge error *assuming* that it is infeasible to compute a preimage from the pseudo-preimage. (To see this, suppose that there is an extractor which after interacting with any prover outputs a true preimage in expected time $T(n)/(\Delta - \kappa)$ where Δ is the prover's success probability and κ is the knowledge error. Running this extractor with e.g. the malicious prover of Corollary 1 gives an algorithm which takes a pseudo-preimage and outputs a preimage in expected time $T(n)/(2^{-\dim(\overline{\beta})} - \epsilon - \kappa)$. Since it was assumed hard to compute a preimage, the distance between κ and $2^{-\dim(\overline{\beta})}$ must be negligible.) We note, however, that it may be possible to construct an efficient protocol by violating the hypothesis of the theorem. Thus, like many other negative results in cryptography, the result should be viewed as a guide for future research, and not as the final answer.

Proof (of Theorem 4). We consider only the case where $\overline{\beta} \in S$ and $\overline{v}_{\overline{\beta}} \in T$, which explains the factor

$$\Pr[\overline{\beta} \in S] \Pr[\overline{v}_{\overline{\beta}} \in T \mid \overline{\beta} \in S]$$

in the success probability of our adversary. There exists a $\overline{v}^* \in T$ such that

$$\Pr[\overline{v}_{\overline{\beta}} = \overline{v}^* \bmod 2 \mid \overline{\beta} \in S \wedge \overline{v}_{\overline{\beta}} \in T] \geq 2^{-\dim(\overline{v})} \enspace ,$$

where the probability is taken over the choice of $\overline{\beta}$. This follows, since there are at most $2^{\dim(\overline{v})}$ possibilities for the parities. If each had probability less than $2^{-\dim(\overline{v})}$, the probabilities would not sum to 1.

Define $\overline{v}$ as $\overline{v} = \overline{v}_{\overline{\beta}}$. The malicious prover $\mathcal{P}^*$ samples $\overline{r}'$ with the distribution of $\overline{r}$ in the protocol and then generates $\overline{s}'$ and $\overline{t}'$ as follows

$$\overline{t}' = \phi(A\overline{r}')y^{\overline{\alpha} - A\overline{v}^*} \quad \text{and} \quad \overline{s}' = B\overline{r}' + \frac{\overline{\beta} - \overline{\beta}^*}{2}u \enspace ,$$

where $\overline{\beta}^* = B\overline{v}^*$. Consider now an $\overline{s}$ formed in an execution with the honest prover, conditioned on $\overline{\beta} \in S$, $\overline{v} \in T$, and $\overline{v} = \overline{v}^* \bmod 2$. It can be expressed as

$$\begin{aligned}\overline{s} = B\overline{r} + \overline{\beta}w &= B(\overline{r} + \overline{v}^*w) + \frac{\overline{\beta} - \overline{\beta}^*}{2}2w \\ &= B\left(\overline{r} + \overline{v}^*w - \frac{\overline{v} - \overline{v}^*}{2}\sigma\right) + \frac{\overline{\beta} - \overline{\beta}^*}{2}u \\ &= B\overline{r}^* + \frac{\overline{\beta} - \overline{\beta}^*}{2}u \enspace ,\end{aligned}$$

where $\overline{r}^* = \overline{r} + \overline{v}^*w - \frac{\overline{v}-\overline{v}^*}{2}\sigma$. We may similarly express $\overline{t}$ as

$$\begin{aligned}\overline{t} = \phi(A\overline{r} + \overline{\alpha}w) &= \phi\left(A\overline{r}^* - A\overline{v}^*w + A\frac{\overline{v} - \overline{v}^*}{2}\sigma + \overline{\alpha}w\right) \\ &= \phi(A\overline{r}^*)\phi(w)^{\overline{\alpha} - A\overline{v}^*} \enspace .\end{aligned}$$

To conclude the proof, we note that the statistical distance between $(\overline{s}', \overline{t}')$ and $(\overline{s}, \overline{t})$ is at most ϵ since the statistical distance between the distributions of $\overline{r}^*$ and $\overline{r}$ is at most ϵ. □

4 Lower Bound on the Soundness Error

Next, we show that if the group has a subgroup of small order and the verifier uses a constant round generic protocol with a natural type of acceptance test, then the proof does not even need to be sound. In particular, we show that a malicious prover knowing an element γ of small order and w such that $\tilde{y} = \gamma\phi(w)$ can convince the verifier that $\tilde{y} = \phi(w')$ for some w'. Note that γ does not have to be in the image of ϕ.

Recall that Cauchy's theorem states that if $\mathcal{H}$ is a finite group and q is a prime dividing the order of $\mathcal{H}$, then there is an element of order q in $\mathcal{H}$. Thus, when the order of $\mathcal{H}$ is unknown, we can not exclude the possibility of elements of small order.

Theorem 5. *Let $\mathcal{H}$ be an abelian group, let $\phi : \mathbb{Z} \to \mathcal{H}$ be a group homomorphism, and let $\gamma \in \mathcal{H}$ be an element of prime order q. Define $\overline{s}$, $\overline{t}$, $\overline{\alpha}$, and $\overline{\beta}$ as in Definition 2 except that we only allow $\overline{\alpha}$ and $\overline{\beta}$ to depend on $\overline{s}$, not on $\overline{t}$. Let $f_i(\cdot,\cdot,\cdot)$ and $g_{ij}(\cdot,\cdot,\cdot)$ be polynomials and let $h_i(\cdot,\cdot,\cdot)$ be a polynomial time computable function. If the verifier's acceptance test is of the form*

$$y^{f_i(\overline{s},\overline{\alpha},\overline{\beta})} \prod\nolimits_j t_j^{g_{ij}(\overline{s},\overline{\alpha},\overline{\beta})} = \phi(h_i(\overline{s},\overline{\alpha},\overline{\beta})) \quad \forall i \in I \enspace ,$$

where the product is taken over all components of $\overline{t}$, then there exists a PPT prover $\mathcal{P}^*$, a PPT algorithm M_σ, and a PPT algorithm $M_{\mathcal{V}^*}$ such that at least one of the following holds for each γ, w and $y = \phi(w)$, where $\Delta = q^{-(\dim\overline{s} + \dim\overline{\beta} + \dim\overline{\alpha})}/3$:

1. *On input γ and w, $\mathcal{P}^*$ convinces the honest verifier on common input $\tilde{y} = \gamma\phi(w)$ with probability at least Δ over the random tapes of the prover and verifier.*
2. *On input w, M_σ outputs a non-zero element in the kernel of ϕ with probability at least Δ over the random tape of M_σ.*
3. *On input q and a transcript of an execution between an honest prover and an honest verifier on private input w and common input $y = \phi(w)$, $M_{\mathcal{V}^*}$ outputs either $w \bmod q$ or $\perp$, where $w \bmod q$ is output with probability at least Δ over the random tapes of the prover and verifier.*

The intuition is that the malicious prover $\mathcal{P}^*$ can guess the residue modulo q of $f_i(\overline{s}, \overline{\alpha}, \overline{\beta})$ and $g_{ij}(\overline{s}, \overline{\alpha}, \overline{\beta})$ and be correct with probability 3Δ. $\mathcal{P}^*$ then generates $(\overline{s}, \overline{t})$ as if producing a correct proof for $y = \phi(w)$, but modifies $\overline{t}$ to cancel any factors γ that appear in the verifier's acceptance test when run with $\tilde{y} = \gamma y$. This modification can be done as long as a certain linear system is solvable. Case 2 and 3 in the theorem give the possibilities when this system is not solvable. The full proof of Theorem 5 is given in the full version of this paper.

To see why it may be hard to compute an element in the kernel, note that for, e.g., the exponentiation homomorphism, finding an element in the kernel corresponds to finding a multiple of the order of the underlying group.

We require that $\overline{\alpha}$ and $\overline{\beta}$ do not depend on $\overline{t}$ in order to be able to take a valid transcript and modify $\overline{t}$ without changing anything else. On the other hand, the requirement that f_i and g_{ij} are polynomials is only used to express the probability that we correctly guess the residue modulo q in terms of the number of messages. This requirement can be relaxed to allow any polynomial time computable functions if there are not too many functions f_i and g_{ij}.

5 On Circumventing the Limitations

The previous work mentioned in the introduction, and the results in previous sections, give considerable evidence that an efficient zero-knowledge proof (of knowledge) of a preimage of a group homomorphism can not be constructed. In this section we consider two settings where we nevertheless are able to construct efficient protocols for proving knowledge of something more than a pseudo-preimage.

5.1 When We Know That a Committed Integer Is Small

In this section we restrict our attention to the exponentiation homomorphism $\phi(w) = g^w$, where $g \in \mathcal{H}$. Our protocol can be used to prove knowledge of e and u such that $y^e = g^{eu}$ and $e = \mathsf{Poly}(n)$. The small remaining exponent e is needed to circumvent Theorem 5.

Note that if the order of (y, g), considered as an element in $\mathcal{H} \times \mathcal{H}$, contains no factors of polynomial size, then this shows that the prover knows u such that $y = g^u$. An example of an application where this is the case is a prover that needs to show knowledge of an integer w of bounded size, such that $y_0 = g_0^w$ and $y_1 = g_1^w$, where g_i generates a group of prime order q_i and $q_0 \neq q_1$.

Our protocol takes a commitment $\mathsf{C}(w, s)$ of the witness w as additional input and the prover is given the randomness s used to form the commitment. Before the protocol is executed, the verifier must be convinced that the committed value is an integer with bounded absolute value. We postpone the discussion of how this can be enforced to Section 5.1 below.

We use a statistically hiding homomorphic commitment scheme with an efficient Σ-proof of the committed value, e.g., Pedersen's commitment scheme, and write $\mathsf{C}_{ck}(w, s)$ for a commitment of $w \in \mathbb{Z}_m$ using randomness $s \in \mathcal{R}$, where $\mathbb{Z}_m$ and $\mathcal{R}$ are the message space and randomizer spaces of the commitment scheme. We stress that the message space $\mathbb{Z}_m$ of the commitment scheme can depend on the commitment parameter ck and that there are no restrictions on m except that $2^{n_w+n_c+n_r} < m/2$. In particular, *we do not need an integer commitment scheme* and we can rely on the standard discrete logarithm assumption.

Similarly to standard Σ-proofs over groups of prime order, our protocol can easily be generalized to prove various more complicated statements involving multiple exponents.

Protocol 2 (Proof of Knowledge of Logarithm)
Common Input. Elements y and g of an abelian group $\mathcal{H}$, a joint commitment parameter ck, and a commitment W.
Private Input. An exponent $w \in [0, 2^{n_w} - 1]$ such that $y = g^w$, and $s \in \mathcal{R}$ such that $W = \mathsf{C}_{ck}(w, s)$.

1. $\mathcal{P}$ chooses $r \in [0, 2^{n_w+n_c+n_r} - 1]$ and $t \in \mathcal{R}$ randomly, computes $\alpha = g^r$ and $R = \mathsf{C}(r, t)$, and hands (α, R) to $\mathcal{V}$.
2. $\mathcal{P}$ proves knowledge of u, s, r, and t such that $W = \mathsf{C}(u, s)$ and $R = \mathsf{C}(r, t)$. This is done in parallel with the remaining three rounds (the honest prover sets $u = w$).
3. $\mathcal{V}$ hands a randomly chosen challenge $c \in [0, 2^{n_c} - 1]$ to $\mathcal{P}$.
4. $\mathcal{P}$ computes $d_0 = cw + r$ over $\mathbb{Z}$ and $d_1 = cs + t$ over $\mathcal{R}$, and hands (d_0, d_1) to $\mathcal{V}$.
5. $\mathcal{V}$ verifies that $d_0 \in [0, 2^{n_w+n_c+n_r} - 1]$, $W^c R = \mathsf{C}(d_0, d_1)$, and $y^c \alpha = g^{d_0}$.

Theorem 6 (Zero-Knowledge). *Protocol 2 is overwhelmingly complete and honest-verifier statistical zero-knowledge.*

The proof of Theorem 6 is given in the full version.

Informally, if a prover convinces the verifier with probability p, then we can extract integers e and u such that $y^e = g^{eu}$ and $e \approx 1/p$. Formally, we need to take care of the commitment parameter ck and commitment W given as additional inputs. In our theorem, the adversary may in a first phase choose the instance (y, g, W) based on the commitment parameter. This is formalized as a PPT instance chooser $\mathcal{I}$. In an application the instance chooser represents the events occurring before the protocol is executed.

Theorem 7 (Soundness and Knowledge Extraction). *Let ck be a random commitment parameter. Given a PPT instance chooser $\mathcal{I}$, we define $(y, g, u, s, z) = \mathcal{I}(ck)$ and $W = \mathsf{C}(u, s)$, where u is an integer satisfying $|u| \in [0, 2^{n_w} - 1]$.*

There exists an extractor $\mathcal{E}_\epsilon$, parametrized by $\epsilon < 1/2$, with $\epsilon^{-1} \in \mathsf{Poly}(n)$, using any prover $\mathcal{P}^*(z)$ as an oracle such that if $\mathcal{P}^*(z)$ convinces $\mathcal{V}(y, g, ck, W)$ with non-negligible probability Δ, then the extractor $\mathcal{E}_\epsilon(y, g, ck, W)$ outputs (e, u) such that $0 < e \leq \frac{1}{(1-\epsilon)^2 \Delta}$ and $y^e = g^{eu}$ with overwhelming probability under the assumption that the commitment scheme is binding. The extractor runs in expected time $O\big(\epsilon^{-2} T(n)/(\Delta - \kappa)\big)$ for some polynomial $T(n)$ and negligible κ.

The proof of Theorem 7 is given in the full version.

Enforcing Small Exponents. We could of course construct a cut-and-choose protocol for proving that a committed value is small when viewed as an integer, but then we could just as well prove knowledge of the exponent *directly* using this approach. Similarly, it makes little sense to let a trusted party certify a commitment of the secret exponent w, when it can just as well certify (y, g) directly. For Protocol 2 to be of interest we need to *decouple the size-guarantee* of the committed value *from the choice of a particular exponent w*.

A trusted party certifies several commitments $Z_1, \ldots, Z_k$ with $Z_i = \mathsf{C}_{ck}(z_i, t_i)$, where both $z_i \in [0, 2^{n_w + n_r} - 1]$ and $t_i \in \mathcal{R}$ are randomly chosen and handed to the prover. Then it is easy to prove that another commitment $W = \mathsf{C}(w, s)$ contains an integer with absolute value less than $2^{n_w + n_r}$ by simply revealing $(z, t) = (w + z_i, s + t_i)$ such that $Z_i W = \mathsf{C}_{ck}(z, t)$. The receiver verifies the certificate of Z_i and that $0 < z < 2^{n_w + n_r}$. Note that using this method, we must effectively reduce the maximum size of w by n_r bits, i.e., the security parameters in Protocol 2 must be modified slightly. The trusted party can go offline after publishing the commitments, but $z = w + z_i$ reveals w, so the trusted party learns w. The latter can be avoided by instead letting the prover register its commitments with the trusted party (or directly with the receiver) and prove that they are correctly formed using a (slow) cut-and-choose protocol in a preliminary phase.

5.2 When the Prover Knows the Order of the Group

We consider the problem of constructing a protocol for proving knowledge of a preimage of a homomorphism, when the prover knows (a multiple of) the order of y. From here on, we use σ to denote the group order rather than any element in the kernel.

Recall that in the protocol for proving knowledge of a discrete logarithm in groups of unknown order, Bangerter et al. [2] assume that the prover has been given a pseudo-preimage (e, u) such that $y^e = g^u$ and $e > 2^{n_c}$ is a large prime such that $e \nmid u$. The reason that the prime is large is to ensure that when a pair of accepting transcripts are extracted in the basic protocol, we have a relation $y^{c-c'} = \phi(u - u')$ and $\gcd(c - c', e)$ is already one, effectively terminating the extraction procedure of Theorem 2 in a single step.

We observe that if the prover knows such a pseudo-preimage and a proper witness w such that $y = g^w$ as well, then it can actually compute a multiple $ew - u \neq 0$ of the order of g. Thus, it seems that the prover essentially knows the order of g in their setting.

The idea of the main protocol of this section is that the prover first proves knowledge of the group order and then proves knowledge of a preimage using the basic protocol (Protocol 1). Combining knowledge of a pseudo-preimage (e, u) with small e with knowledge of the order σ of the group allows the extractor to simplify both quantities.

Proof of Knowledge of a Multiple of the Order of an Element. We do not know how to construct an efficient proof of knowledge of the order of a group element, but it turns out that a proof of knowledge of a *multiple* of the order suffices for our purposes. The proofs of the following theorems appear in the full version of this paper.

Protocol 3 (Knowledge of Multiple of the Order of an Element)
Common Input. An element g of an abelian group $\mathcal{H}$ and an upper bound 2^{n_w} of $|\langle g \rangle|$.
Private Input. The order $|\langle g \rangle|$ of g.

1. $\mathcal{P}$ and $\mathcal{V}$ compute $u = 2^{n_w + 2n_c + n_r + 2}$ and $y = g^u$.
2. $\mathcal{P}$ computes $w = u \bmod |\langle g \rangle|$.
3. Using Protocol 1, $\mathcal{P}$ proves knowledge of integers e and u' (e.g., 1 and w) such that $|e| < 2^{n_c+1}$, $|u'| < 2^{n_w + n_c + n_r + 1}$, and $y^e = g^{u'}$.

Theorem 8. *Protocol 3 is complete and an honest-verifier statistical zero-knowledge proof of knowledge of a multiple of the order of g.*

Proof of Knowledge of a Preimage. Using the above protocol for proving knowledge of a multiple of the order of a group element, we now construct a proof of knowledge of a preimage for provers that know the order of y.

Protocol 4 (Proof of Knowledge of a Preimage)
Common Input. An element $y \in \mathcal{H}$ and a homomorphism $\phi : \mathcal{G} \to \mathcal{H}$ of abelian groups $\mathcal{G}$ and $\mathcal{H}$.
Private Input. A preimage $w \in \mathcal{G}$ such that $y = \phi(w)$ and the order σ of y.

1. $\mathcal{P}$ and $\mathcal{V}$ execute Protocol 3 on common input y and an upper bound 2^{n_w} of the order $\sigma = |\langle y \rangle|$, and private input σ, i.e., $\mathcal{P}$ proves knowledge of a multiple of σ.
2. $\mathcal{P}$ and $\mathcal{V}$ execute Protocol 1 on common input y and private input w, i.e., $\mathcal{P}$ proves knowledge of a pseudo-preimage of y under ϕ.

Theorem 9. *Protocol 4 is complete and honest-verifier statistical zero-knowledge.*

The only difference between the results below and Theorem 2 is that here we can not only bound the size of e, we can also reduce it further until all its factors appear in σ, the order of y. When σ has no small factors, the result is a proper proof of knowledge.

Theorem 10 (Extraction and Soundness). *There exists an extractor $\mathcal{E}_\epsilon$, parameterized by $\epsilon < 1/2$ with $\epsilon^{-1} \in \mathsf{Poly}(n)$, such that for every (y, ϕ), and every PPT prover $\mathcal{P}^*$ which convinces $\mathcal{V}(y, \phi)$ with probability $\Delta > \kappa$, $\mathcal{E}_\epsilon(y, \phi)$, using $\mathcal{P}^*$ as an oracle, extracts an integer $0 < e \leq \frac{1}{(1-\epsilon)^2 \Delta}$ and $u \in \mathcal{G}$ such that $y^e = \phi(u)$ and each factor of e divides σ. The extractor runs in expected time $O\big(\epsilon^{-2} T(n)/(\Delta - \kappa)\big)$, where $T(n)$ is a polynomial and κ is defined as $\kappa = 2^{1-n_c}/\epsilon$.*

Corollary 2. *If every factor of σ is greater than 2^{n_c}, then Protocol 4 is a proof of knowledge with negligible soundness/knowledge error of a preimage w such that $y = \phi(w)$.*

References

1. Bangerter, E., Camenisch, J., Krenn, S.: Efficiency Limitations for Σ-Protocols for Group Homomorphisms. In: Micciancio, D. (ed.) TCC 2010. LNCS, vol. 5978, pp. 553–571. Springer, Heidelberg (2010)
2. Bangerter, E., Camenisch, J., Maurer, U.M.: Efficient Proofs of Knowledge of Discrete Logarithms and Representations in Groups with Hidden Order. In: Vaudenay, S. (ed.) PKC 2005. LNCS, vol. 3386, pp. 154–171. Springer, Heidelberg (2005)
3. Bangerter, E., Krenn, S., Sadeghi, A.-R., Schneider, T., Tsay, J.-K.: On the design and implementation of efficient zero-knowledge proofs of knowledge. In: ECRYPT Workshop on Software Performance Enhancements for Encryption and Decryption and Cryptographic Compilers, SPEED-CC 2009 (2009)
4. Bellare, M., Goldreich, O.: On Defining Proofs of Knowledge. In: Brickell, E.F. (ed.) CRYPTO 1992. LNCS, vol. 740, pp. 390–420. Springer, Heidelberg (1993)
5. Brickell, E.F., McCurley, K.S.: An interactive identification scheme based on discrete logarithms and factoring. J. Cryptology 5(1), 29–39 (1992)
6. Cramer, R., Damgård, I.: On the Amortized Complexity of Zero-Knowledge Protocols. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 177–191. Springer, Heidelberg (2009)
7. Cramer, R., Damgård, I., Schoenmakers, B.: Proof of Partial Knowledge and Simplified Design of Witness Hiding Protocols. In: Desmedt, Y.G. (ed.) CRYPTO 1994. LNCS, vol. 839, pp. 174–187. Springer, Heidelberg (1994)
8. Damgård, I., Fujisaki, E.: A Statistically-Hiding Integer Commitment Scheme Based on Groups with Hidden Order. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 125–142. Springer, Heidelberg (2002)
9. Fujisaki, E., Okamoto, T.: Statistical Zero Knowledge Protocols to Prove Modular Polynomial Relations. In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 16–30. Springer, Heidelberg (1997)
10. Girault, M., Poupard, G., Stern, J.: On the fly authentication and signature schemes based on groups of unknown order. J. Cryptology 19(4), 463–487 (2006)
11. Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof systems. SIAM J. Comput. 18(1), 186–208 (1989)

12. Guillou, L.C., Quisquater, J.-J.: A Practical Zero-Knowledge Protocol Fitted to Security Microprocessor Minimizing Both Transmission and Memory. In: Günther, C.G. (ed.) EUROCRYPT 1988. LNCS, vol. 330, pp. 123–128. Springer, Heidelberg (1988)
13. Kunz-Jacques, S., Martinet, G., Poupard, G., Stern, J.: Cryptanalysis of an Efficient Proof of Knowledge of Discrete Logarithm. In: Yung, M., Dodis, Y., Kiayias, A., Malkin, T. (eds.) PKC 2006. LNCS, vol. 3958, pp. 27–43. Springer, Heidelberg (2006)
14. Schnorr, C.-P.: Efficient signature generation by smart cards. J. Cryptology 4(3), 161–174 (1991)
15. Shoup, V.: On the security of a practical identification scheme. J. Cryptology 12(4), 247–260 (1999)
16. Wikström, D.: Designated Confirmer Signatures Revisited. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 342–361. Springer, Heidelberg (2007)