Lattice-Based Identification Schemes Secure Under Active Attacks*

Vadim Lyubashevsky

University of California, San Diego
9500 Gilman Drive, La Jolla, CA 92093-0404, USA
vlyubash@cs.ucsd.edu

Abstract. There is an inherent difficulty in building 3-move ID schemes based on combinatorial problems without much algebraic structure. A consequence of this, is that most standard ID schemes today are based on the hardness of number theory problems. Not having schemes based on alternate assumptions is a cause for concern since improved number theoretic algorithms or the realization of quantum computing would make the known schemes insecure. In this work, we examine the possibility of creating identification protocols based on the hardness of lattice problems. We construct a 3-move identification scheme whose security is based on the worst-case hardness of the shortest vector problem in all lattices, and also present a more efficient version based on the hardness of the same problem in *ideal* lattices.

1 Introduction

Public key identification (ID) protocols allow a party holding a secret key to prove its identity to any other entity holding the corresponding public key. The minimum security of such protocols should be that a passive observer who sees the interaction should not then be able to perform his own interaction and successfully impersonate the prover. In a more realistic model, the adversary should first be allowed to interact with the prover in a "dishonest" way in hopes of extracting some information, and then try to impersonate the prover. Identification schemes resistant to such impersonation attempts are said to be secure in the active attack model [7], and this is currently the *de facto* security notion.

Since Fiat and Shamir's seminal paper [9], there have been many proposals for constructing secure ID protocols. With a few notable exceptions, most of these protocols (e.g. [11,26,21,29,23,10]) are based on problems from number theory, and as such, they require fairly costly multiplication and exponentiation operations. Another potential problem is that the security of these protocols is based on problems that are easy if (when) practical quantum computers become reality [28]. Thus it is prudent to have viable alternative schemes based on different hardness assumptions.

The identification protocols not based on number theory problems (e.g. [27,30]) are generally combinatorial in nature. Because of this lack of algebraic structure,

* Supported by NSF grant CCF-0634909.

R. Cramer (Ed.): PKC 2008, LNCS 4939, pp. 162–179, 2008.

these combinatorial schemes all seem to have an inherent shortcoming in that they require a lot more rounds of communication than their algebraic counterparts. This problem arises because the proof of security is established by showing that the schemes are zero-knowledge proofs of knowledge. It is shown that the prover (or adversary) who successfully proves his identity, actually "knows" the secret (as defined in [7]), yet the protocol is zero-knowledge, and as such, the prover doesn't reveal anything about his secret key. The problem is that in order for the protocol to have negligible soundness error, it must be repeated a polynomial number of times. But zero-knowledge is not preserved under parallel-repetition, and so the protocol has to be run sequentially in order for it to maintain the claimed security.

In recent years, lattices have emerged as a possible alternative to number theory. Cryptography based on lattices was pioneered by Ajtai [1], who showed a fascinating connection between solving random instances of a certain problem and solving *all* instances of certain lattice problems. This opened up a way to base cryptographic functions on the hardness of worst-case problems. Since then, there has been a lot of work on improving the average case/worst-case reduction [19], building cryptographic primitives [3,24,25], and using similar techniques to build more efficient cryptographic primitives [17,22,15,16] based on similar worst-case assumptions. Additionally, there are currently no efficient quantum algorithms for solving lattice problems.

1.1 This Work

In this work, we present an ID scheme whose security is based on the worst-case hardness of lattice problems. In addition, we present a more efficient version of the scheme that is based on the hardness of problems on *ideal* lattices (see section 2.5). We prove security by showing that an adversary who successfully attacks our scheme can be used to solve random instances of problems defined in [19] and [17], which were proven to be as hard as lattice problems in the worst case. Thus, in this work, we do not deal with average-case/worst-case reductions directly.

We believe that the technical details of our ID protocol may also be of independent interest. While our scheme has the structure of a standard 3-move commit-challenge-response protocol, for security reasons, an honest prover sometimes "refuses" to respond to the verifier's challenge. It can be shown that if the prover always responds to the verifier, then his secret key is leaked to even a passive observer. On the other hand, by strategically refusing to reply, each round of the protocol can be shown to be *witness-indistinguishable*. And since witness-indistinguishability is preserved under parallel-composition, all the rounds can be performed in parallel.

1.2 Related Work

The one place in the literature that mentions constructions of lattice-based identification schemes is the work of Micciancio and Vadhan [20] on statistical zero

knowledge relating to lattice problems. In this work, the authors show an efficient-prover SZK proof system for certain lattice problems and mention that one can convert the proof system into an identification scheme. The conversion is non-trivial (due to the problem of zero-knowledge not being closed under parallel-composition), and many details remain to be filled in.

2 Preliminaries

2.1 Notation

We will represent vectors by bold letters. By $\boldsymbol{x} \xleftarrow{\$} X$, we mean that $\boldsymbol{x}$ is chosen uniformly at random from the set X. The notation $\tilde{O}(n^k)$ is equivalent to $O(n^k \log^c n)$ for some constant c.

2.2 Statistical Distance

Informally, statistical distance is a measure of how far apart two distributions are. Formally, if X and Y are random variables over a countable set A, then the statistical distance between X and Y, denoted $\Delta(X, Y)$, is defined as

$$\Delta(X, Y) = \frac{1}{2} \sum_{a \in A} |Pr[X = a] - Pr[Y = a]|$$

From the definition, it's easy to see that

$$\Delta(X, Z) \leq \Delta(X, Y) + \Delta(Y, Z)$$

2.3 Identification Schemes

An identification scheme consists of a key-generation algorithm and a description of an interactive protocol between a prover, possessing the secret key, and verifier possessing the corresponding public key. In general, it is required that the verifier accepts the interaction with a prover who behaves honestly with probability one. In this work, though, we need to relax this definition, and only require that the verifier accepts an honest prover with probability negligibly close to one (i.e $1 - 2^{-\omega(\log n)}$).

The standard active attack model against identification schemes proceeds in two phases [7]. In the first phase, the adversary interacts with the prover in an effort to obtain some information. In the second stage, the adversary plays the role of the prover and tries to make a verifier accept the interaction. We remark that in the second stage, the adversary no longer has access to the honest prover. We will say that the adversary has advantage *adv*, if the verifier accepts the interaction with the adversary with probability *adv* (where the probability is over the randomness of the prover, verifier, and the adversary).

2.4 Witness Indistinguishability

The concept of *witness indistinguishability* was introduced by Feige and Shamir in [8]. For a string x and relation R, a witness set $W_R(x)$ consists of all strings w such that $R(w, x) = 1$. For example, x could be a boolean formula and the relation R could be defined as $R(x, w) = 1$ iff w is an assignment that makes x evaluate to 1. Then the set $W_R(x)$ is the set of all assignments that make x evaluate to 1. In our case, the witness will correspond to the secret key and the string x is the public key.

Let $\mathcal{P}$ and $\mathcal{V}$ be two randomized interactive Turing machines and $(\mathcal{P}, \mathcal{V})$ be a protocol between $\mathcal{P}$ and $\mathcal{V}$. We denote by $\mathcal{V}_{\mathcal{P}(x,w)}(x, y)$ the output of $\mathcal{V}$ after participating in the protocol $(\mathcal{P}, \mathcal{V})$. We say that $(\mathcal{P}, \mathcal{V})$ is statistically witness-indistinguishable if for all $\mathcal{V}'$, all large enough x, any y, and any two $w, w' \in W_R(x)$,

$$\Delta\left(\mathcal{V}'_{\mathcal{P}(x,w)}(x, y), \mathcal{V}'_{\mathcal{P}(x,w')}(x, y)\right) < 2^{-\omega(\log|x|)}.$$

In other words, every cheating verifier $\mathcal{V}'$ with any auxiliary input y, cannot distinguish whether the witness that $\mathcal{P}$ is using in the protocol is w or w'. An important feature of witness indistinguishability is that it is closed under parallel composition.

2.5 Lattices

General Lattices. An integer lattice $\mathcal{L}$ of dimension n is simply an additive subgroup of $\mathbb{Z}^n$. A fundamental set of parameters associated with a lattice $\mathcal{L}$ is the set of successive minima $\lambda_i(\mathcal{L})$ for $1 \leq i \leq n$. For every i, $\lambda_i(\mathcal{L})$ is defined as the minimal radius of a sphere centered at the origin that contains i linearly independent lattice vectors. For example, $\lambda_1(\mathcal{L})$ corresponds to the length of the shortest vector in $\mathcal{L}$, and finding a vector of length $\lambda_1(\mathcal{L})$ is known as the Shortest Vector Problem (SVP). Likewise, the problem of finding n independent vectors all of length at most $\lambda_n(\mathcal{L})$ is known as the Shortest Independent Vector Problem (SIVP). Approximation versions of SVP and SIVP are defined in the natural way. That is, an approximate solution to SVP within some factor γ is a vector in the lattice that is of length at most $\gamma\lambda_1(\mathcal{L})$. Similarly, an approximate solution to SIVP within a factor γ is a set of n linearly independent lattice vectors each having length at most $\gamma\lambda_n(\mathcal{L})$

The shortest vector problem was shown to be NP-hard by Ajtai [2] and NP-hard to approximate to within any constant factor by Khot [13]. The best known algorithm to find the exact shortest vector, or even some polynomial in n factor approximation of it, takes time $2^{O(n)}$ [4,14]. As far as SIVP is concerned, it is known that this problem is NP-hard to approximate for any constant factor [6], and finding the exact solution takes time approximately $n!$ [18] (although finding a $(1 + \epsilon)$ approximation takes time $2^{O(n)}$ for any constant ϵ [5]).

The aspect that makes lattices interesting in cryptography is that one can build collision-resistant hash function families that are as hard to break on the average, as solving approximate SIVP in the worst case. This work began with the seminal paper by Ajtai [1], and the currently tightest reduction is due to

Micciancio and Regev [19]. Below, we restate the main result of [19] in a way that will be convenient for our proof. [1]

Definition 1. *(The small integer solution* SIS($\boldsymbol{A}$) *problem) Given a matrix* $\boldsymbol{A} \in \mathbb{Z}_p^{n \times m}$*, find two distinct vectors* $\boldsymbol{z}, \boldsymbol{z}' \in \mathbb{Z}^m$ *such that* $\boldsymbol{A}\boldsymbol{z} \bmod p = \boldsymbol{A}\boldsymbol{z}' \bmod p$ *and* $\|\boldsymbol{z}\|, \|\boldsymbol{z}'\| \leq 10m^{1.5}$*.*

Theorem 2. *[19, Theorem 5.9] For integer* $m = \lceil 4n \log n \rceil$ *and some integer* $p = \tilde{\Theta}(n^3)$*, if there exists a polynomial-time algorithm that solves* SIS($\boldsymbol{A}$) *for uniformly random* $\boldsymbol{A} \in \mathbb{Z}_p^{n \times m}$*, then the SIVP problem can be approximated in polynomial time to within a factor of* $\tilde{O}(n^2)$ *in every* n*-dimensional lattice.*

Ideal Lattices. Ideal lattices were first studied in the context of cryptography by Lyubashevsky and Micciancio in [15]. Such lattices are a special class of general lattices and a generalization of cyclic lattices [17]. Their usefulness is attributed to the fact that very efficient and practical collision-resistant hash functions can be built based on the hardness of finding an approximate shortest vector in such lattices. Roughly speaking, ideal lattices are lattices corresponding to ideals in rings of the form $\mathbb{Z}[x]/\langle f \rangle$ for some irreducible polynomial f of degree n. For simplicity we will only concentrate on rings of the form $\mathbb{Z}[x]/\langle x^n + 1 \rangle$, as they have proved to be the most useful for practical applications [16]. An n-dimensional ideal lattice in the ring $\mathbb{Z}[x]/\langle x^n + 1 \rangle$ is a lattice with the additional restriction that for every vector $(a_1, \ldots, a_{n-1}, a_n)$ in the lattice, the rotated vector with the first coordinate negated $(-a_n, a_1, \ldots, a_{n-1})$ must also be in the lattice. It was shown in [15] that efficient collision resistant hash functions could be built based on the hardness of finding the shortest vector in ideal lattices. The average-case hard problem in [15] is essentially the SIS problem in Definition 1, with the one difference being (and this is what gives the hash function its efficiency) that the matrix $\boldsymbol{A} \in \mathbb{Z}_p^{n \times m}$ is no longer chosen from the entire domain $\mathbb{Z}_p^{n \times m}$. Instead, it is chosen as follows: first pick any vector $\boldsymbol{a}_1 \in \mathbb{Z}_p^n$ and make it the first column of $\boldsymbol{A}$. The next $n-1$ columns of $\boldsymbol{A}$ consist of consecutive rotations (while always negating the coordinate that gets rotated to the beginning of the vector) of $\boldsymbol{a}_1$. For column $n+1$, we choose another random vector $\boldsymbol{a}_2$ and then fill the next $n-1$ columns with its rotations. We continue repeating this process until all m columns are filled (we assume that m is a multiple of n). We will call this domain of all such matrices $\mathrm{ROT}(n, m, p)$, and selecting a random $\boldsymbol{A} \in \mathrm{ROT}(n, m, p)$ corresponds to performing the above procedure while choosing $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots \boldsymbol{a}_{m/n}$ randomly from $\mathbb{Z}_p^n$.

Notice that because of the repetition, it is not necessary to store all m columns of matrices chosen from $\mathrm{ROT}(n, m, p)$. Another extremely important feature is that multiplying such matrices by any vector in $\mathbb{Z}_p^m$ requires only $\tilde{O}(m \log n)$ time rather than $\tilde{O}(mn)$. This is because the multiplication can be done using the Fast Fourier Transform (see [17,15] for details).

[1] We point out that the below result is weaker than what was proved in [19]. Unfortunately, in this paper we cannot construct an identification scheme with security based on the strongest results from [19].

We will now state a convenient form of the main result of [15] [2].

Theorem 3. *[15, Theorem 2] For integer $m = \lceil 4n \log n \rceil$ and some integer $p = \tilde{\Theta}(n^3)$, if there exists a polynomial-time algorithm that solves* SIS($\boldsymbol{A}$) *for uniformly random $\boldsymbol{A} \in ROT(n, m, p)$, then SIVP (and also SVP[3]) can be approximated in polynomial time to within a factor of $\tilde{O}(n^2)$ in every n-dimensional lattice corresponding to an ideal in $\mathbb{Z}[x]/\langle x^n + 1 \rangle$.*

2.6 Leftover Hash Lemma

In this section, we review the leftover hash lemma [12]. This lemma will be crucial in proving the witness-indistinguishability property of our protocol.

Lemma 4. ***(Leftover Hash Lemma)*** *Let X and Y be two finite sets and U be the uniform distribution over Y. If $\mathcal{H}$ is a universal family of hash functions[4] from X to Y, then for all but a $2^{\frac{\log|Y|-\log|X|}{4}}$ fraction of the possible $h_i \in \mathcal{H}$, $\Delta(h_i(x), U) \leq 2^{\frac{\log|Y|-\log|X|}{4}}$ where x is chosen uniformly at random from X.*

The following lemma is a straightforward consequence of the leftover hash lemma.

Lemma 5. *Let X be some subset of $\mathbb{Z}_p^m$. Then for all but a $2^{\frac{n\log p-\log|X|}{4}}$ fraction of all $\boldsymbol{A} \in \mathbb{Z}_p^{n\times m}$, we have*

$$\Delta(\boldsymbol{A}\boldsymbol{x} \bmod p, \boldsymbol{u}) \leq 2^{\frac{n\log p-\log|X|}{4}},$$

where $\boldsymbol{x}$ is a random variable distributed uniformly in X and $\boldsymbol{u}$ is a random variable distributed uniformly in $\mathbb{Z}_p^n$.

Proof. We consider a family of hash functions $\mathcal{H}$ consisting of functions $h_{\boldsymbol{A}}$ indexed by $\boldsymbol{A} \in \mathbb{Z}_p^{n\times m}$, where $h_{\boldsymbol{A}}(\boldsymbol{x})$ is defined as $\boldsymbol{A}\boldsymbol{x} \bmod p$. The domain of these functions is any subset of $\mathbb{Z}_p^m$ and the range is $\mathbb{Z}_p^n$. To apply the Leftover Hash Lemma, we need to show that $\mathcal{H}$ is a universal family of hash functions. In other words, for any distinct $\boldsymbol{x}, \boldsymbol{x}' \in X$, we need to show that for a randomly chosen $\boldsymbol{A} \in \mathbb{Z}_p^{n\times m}$,

$$Pr[h_{\boldsymbol{A}}(\boldsymbol{x}) = h_{\boldsymbol{A}}(\boldsymbol{x}')] = \frac{1}{2^{n\log p}}.$$

In other words, we need to show that for a randomly chosen $\boldsymbol{A} \in \mathbb{Z}_p^{n\times m}$,

$$\begin{aligned}\frac{1}{2^{n\log p}} &= Pr[\boldsymbol{A}\boldsymbol{x} \bmod p = \boldsymbol{A}\boldsymbol{x}' \bmod p] \\ &= Pr[\boldsymbol{A}(\boldsymbol{x}-\boldsymbol{x}') \bmod p = \boldsymbol{0}] = Pr[\boldsymbol{A}\boldsymbol{y} \bmod p = \boldsymbol{0}]\end{aligned}$$

where $\boldsymbol{y}$ is some non-zero vector. Without loss of generality, assume that the last coefficient of $\boldsymbol{y}$ is non-zero, and let $\boldsymbol{y}'$ be the first $m-1$ coefficients of $\boldsymbol{y}$.

[2] As for general lattices, the below result is weaker than what was proved in [15].

[3] This is because lattices of this form have the property that $\lambda_1(\mathcal{L}) = \ldots = \lambda_n(\mathcal{L})$.

[4] Recall that a hash function family $\mathcal{H} : X \to Y$ is called universal if for every two distinct elements $x, x' \in X$, we have $Pr_{h \xleftarrow{\$} \mathcal{H}}[h(x) = h(x')] = 1/|Y|$.

Similarly, let $\boldsymbol{a}$ be the last column of $\boldsymbol{A}$ and let $\boldsymbol{A'}$ be the first $m-1$ columns of $\boldsymbol{A}$. Then,

$$\begin{aligned} Pr[\boldsymbol{Ay} \bmod p = \boldsymbol{0}] &= Pr[\boldsymbol{A'y'} + \boldsymbol{a}y_m \bmod p = \boldsymbol{0}] \\ &= Pr[\boldsymbol{a} \equiv y_m^{-1}(-\boldsymbol{A'y'})(\bmod\ p)] = \frac{1}{2^{n \log p}} \end{aligned}$$

Since p is prime and y_m is non-zero, the multiplicative inverse of y_m modulo p exists. And since $\boldsymbol{a}$ is chosen uniformly at random from $\mathbb{Z}_p^n$, the probability that it is equal to any specific value is $\frac{1}{2^{n \log p}}$. And now that we have shown that $\mathcal{H}$ is a family of universal hash functions, the claim of the lemma follows from the Leftover Hash Lemma. □

The below corollary is obtained by applying Lemma 5 twice, and using the triangular inequality property of statistical distance.

Corollary 6. *Let X and Y be any two subsets of $\mathbb{Z}_p^m$. Then for all but a $2^{\frac{n \log p - \log |X|}{4}} + 2^{\frac{n \log p - \log |Y|}{4}}$ fraction of all $\boldsymbol{A} \in \mathbb{Z}_p^{n \times m}$, we have*

$$\Delta(\boldsymbol{Ax} \bmod p, \boldsymbol{Ay} \bmod p) \leq 2^{\frac{n \log p - \log |X|}{4}} + 2^{\frac{n \log p - \log |Y|}{4}},$$

where $\boldsymbol{x}$ is a random variable distributed uniformly in X and $\boldsymbol{y}$ is a random variable distributed uniformly in Y.

3 The Identification Scheme

We will first describe one round of our identification scheme (Figure 1). The prover picks a secret key $\tilde{\boldsymbol{w}} \in \{0,1\}^m$, and publishes the public keys $\boldsymbol{A} \xleftarrow{\$} \mathbb{Z}_p^{n \times m}$ and $\boldsymbol{w} \leftarrow \boldsymbol{A}\tilde{\boldsymbol{w}} \bmod p$, where $m = \lceil 4n \log n \rceil$ and p is some integer of order $\tilde{\Theta}(n^3)$.[5] We note that the matrix $\boldsymbol{A}$ may either be created by the prover or be created by a trusted third party. In fact, all users may share the same matrix $\boldsymbol{A}$. In the first step of the protocol, the prover picks a uniformly random vector $\tilde{\boldsymbol{y}}$ from the set of vectors $\{0, 1, \ldots, 5m-1\}^m$, and sends $\boldsymbol{y} \leftarrow \boldsymbol{A}\tilde{\boldsymbol{y}} \bmod p$ to the verifier. The verifier then sends a challenge $c \leftarrow \{0,1\}$. If $c = 0$, the prover simply sends $\boldsymbol{z} \leftarrow \tilde{\boldsymbol{y}}$ as the response. If, on the other hand, $c = 1$, the prover first checks whether the quantity $\tilde{\boldsymbol{w}} + \tilde{\boldsymbol{y}}$ is in the set SAFE$=\{1, 2, \ldots, 5m-1\}^m$. If it is, then the prover sends $\boldsymbol{z} \leftarrow \tilde{\boldsymbol{w}} + \tilde{\boldsymbol{y}}$, and if it is not, then the prover sends $\boldsymbol{z} \leftarrow \perp$ which signifies that he refuses to answer. If the prover sends $\perp$, then the verifier obviously rejects the interaction. Otherwise, the verifier checks whether $\|\boldsymbol{z}\| \leq 5m^{1.5}$ and $\boldsymbol{Az} \bmod p = c\boldsymbol{w} + \boldsymbol{y}$. The verifier accepts if and only if those two conditions are satisfied.

Some comments are in order about the somewhat unusual way in which the prover picks his response $\boldsymbol{z}$ when the challenge is $c = 1$. Notice that if the prover

[5] For the reader's convenience, we will make the convention of putting tildes over the variables which are kept "secret" by the prover (e.g. $\tilde{\boldsymbol{w}}, \tilde{\boldsymbol{y}}$).

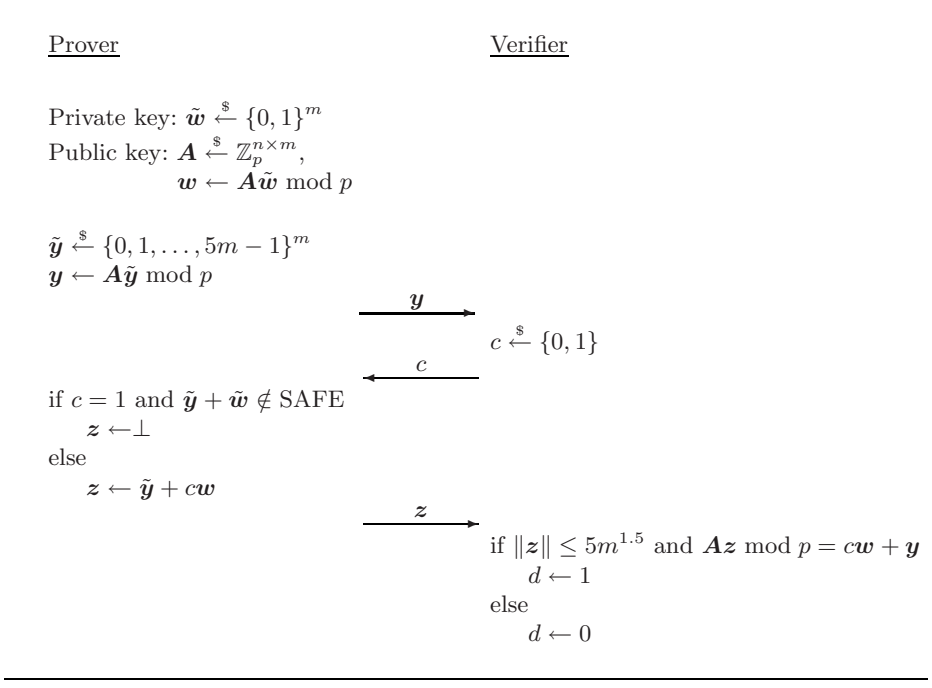

Fig. 1. One round of our identification scheme. The parameters are $p = \tilde{O}(n^3)$, $m = \lceil 4n \log n \rceil$, and the set SAFE is defined as $\{1,\ldots,5m-1\}^m$.

always sends $\boldsymbol{z} \leftarrow \tilde{\boldsymbol{w}} + \tilde{\boldsymbol{y}}$ for $c = 1$, then even a passive observer can deduce the secret $\tilde{\boldsymbol{w}}$ after he sees enough rounds. This is because if any coordinate of $\boldsymbol{z}$ is ever 0, the observer knows that the corresponding bit of $\tilde{\boldsymbol{w}}$ must also be 0. Similarly, if any coordinate of $\boldsymbol{z}$ is $5m$, then the corresponding bit of $\tilde{\boldsymbol{w}}$ must be 1. One might think that a way to resolve this problem would be to choose $\tilde{\boldsymbol{y}}$ in a way such that seeing $\tilde{\boldsymbol{w}} + \tilde{\boldsymbol{y}}$ will not give anything away about $\tilde{\boldsymbol{w}}$. The problem with this approach is that when the verifier sends $c = 0$, the prover will have to reveal $\tilde{\boldsymbol{y}}$, and the distribution of the $\tilde{\boldsymbol{y}}$'s may actually end up revealing the secret $\tilde{\boldsymbol{w}}$. (Consider the naïve idea of never setting any coordinates of $\tilde{\boldsymbol{y}}$ to 0 if the corresponding bits of $\tilde{\boldsymbol{w}}$ are 0. Then the fact that some coordinates of $\tilde{\boldsymbol{y}}$ are never 0 will give away the fact that those bits of $\tilde{\boldsymbol{w}}$ were themselves 0's.) At the present, the only way that we know of to "fix" this, is to make the integers m of order $n^{\omega(1)}$. This way, with high probability, the coefficients of $\tilde{\boldsymbol{y}}$ will never be 0 or $5m-1$, and so $\tilde{\boldsymbol{w}}$ will potentially be safe. Unfortunately, setting m to such a large number significantly weakens the result of the security proof.

A consequence of the prover sometimes refusing to answer is that the verifier may end up rejecting an honest prover. So it is important that the honest prover is not rejected too often in each round. This way, if the protocol is repeated enough times, the prover will answer correctly enough times so that the verifier will be able to distinguish between an honest prover and an impersonator.

We will now outline the rest of this section. We first show that an honest prover will able to get the verifier to accept with a "high enough" probability (Lemma 7). We then show that every round of the protocol is statistically *witness-indistinguishable* (Theorem 9). Since witness indistinguishability is preserved under parallel composition, we can repeat the protocol in Figure 1 many times in parallel. The result of this is the identification protocol in Figure 2. In Theorem 13, we show that this protocol is secure in the active attack model by showing that an adversary who successfully attacks the protocol can be used to solve the SIS problem from Definition 1, which by Theorem 2 implies being able to solve the approximate Shortest Independent Vector Problem in every lattice.

Lemma 7. *For $m \geq 10$, the probability that the verifier will accept (i.e. set $d = 1$) an interaction with an honest prover during a round is at least* .81*.*

Proof. Notice that if $c = 0$, then the verifier will always accept because the prover will always send $\boldsymbol{z} = \tilde{\boldsymbol{y}}$ and thus $\boldsymbol{A}\boldsymbol{z} \equiv \boldsymbol{A}\tilde{\boldsymbol{y}} \equiv \boldsymbol{y} (\text{mod } p)$. Similarly, if $c = 1$ and $\tilde{\boldsymbol{w}} + \tilde{\boldsymbol{y}} \in \text{SAFE}$, then the verifier will always accept because the prover sends $\boldsymbol{z} = \tilde{\boldsymbol{w}} + \tilde{\boldsymbol{y}}$ and so $\boldsymbol{A}\boldsymbol{z} \equiv \boldsymbol{A}(\tilde{\boldsymbol{w}} + \tilde{\boldsymbol{y}}) \equiv \boldsymbol{w} + \boldsymbol{y} (\text{mod } p)$. Thus the probability that the verifier accepts is at least the probability that $\tilde{\boldsymbol{w}} + \tilde{\boldsymbol{y}} \in \text{SAFE}$.

$$Pr[d = 1] \geq Pr[\tilde{\boldsymbol{w}} + \tilde{\boldsymbol{y}} \in \text{SAFE}] = \left(1 - \frac{1}{5m}\right)^m \geq .81 \text{ for } m \geq 10 \qquad (1)$$

The equality is true because for every i, only one of $5m$ possibilities for the coefficient $\tilde{y}_i$ of $\tilde{\boldsymbol{y}}$ will lead to $\tilde{\boldsymbol{w}} + \tilde{\boldsymbol{y}}$ to be not in the set SAFE. That is, if $\tilde{w}_i = 0$, then $\tilde{y}_i$ can be anything except 0, and if $\tilde{w}_i = 1$, then $\tilde{y}_i$ can be anything except $5m - 1$. □

Before showing that every round of the protocol is witness-indistinguishable, we need to show that with extremely high probability over the choices of the public key, there does indeed exist more than one possible secret key.

Lemma 8. *For any matrix $\boldsymbol{A} \in \mathbb{Z}_p^{n \times m}$ and a randomly chosen $\tilde{\boldsymbol{w}} \xleftarrow{\$} \{0,1\}^m$, the probability that there exists another $\tilde{\boldsymbol{w}}' \in \{0,1\}^m \setminus \tilde{\boldsymbol{w}}$ such that $\boldsymbol{A}\tilde{\boldsymbol{w}} \bmod p = \boldsymbol{A}\tilde{\boldsymbol{w}}' \bmod p$ is at least $1 - 2^{n \log p - m}$.*

Proof. The result of $\boldsymbol{A}\tilde{\boldsymbol{w}} \bmod p$ falls into $\mathbb{Z}_p^n$, and thus there can be at most $|\mathbb{Z}_p^n| = 2^{n \log p}$ elements $\tilde{\boldsymbol{w}} \in \{0,1\}^m$ such that $\boldsymbol{A}\tilde{\boldsymbol{w}} \bmod p$ leads to a unique element in $\mathbb{Z}_p^n$. Thus the probability that a randomly chosen $\tilde{\boldsymbol{w}} \in \{0,1\}^m$ collides with some other $\tilde{\boldsymbol{w}}' \in \{0,1\}^m$ is at least $1 - 2^{n \log p - m}$. □

We now move to showing witness indistinguishability. The proof will roughly proceed as follows. First, we observe that when the challenge is $c = 0$, the protocol is trivially witness indistinguishable because the secret key is completely uninvolved in the response. So we concentrate on the case where $c = 1$. In that case, two things can happen. In one case, $\tilde{\boldsymbol{w}} + \tilde{\boldsymbol{y}}$ will be in the set SAFE and the prover sends $\boldsymbol{z} \leftarrow \tilde{\boldsymbol{w}} + \tilde{\boldsymbol{y}}$. In this case, we will show that the protocol is *perfectly* witness-indistinguishable. In the case that $\tilde{\boldsymbol{w}} + \tilde{\boldsymbol{y}}$ is not in SAFE and

the prover sends $\boldsymbol{z} \leftarrow \perp$, we will show that the protocol is *statistically* witness indistinguishable.

The below theorem actually proves witness indistinguishability of the protocol for all but a $2^{-\Omega(n \log^2 n)}$ fraction of $\boldsymbol{A} \in \mathbb{Z}_p^{n \times m}$. Since the matrix $\boldsymbol{A}$ is chosen at random, there is only a $2^{-\Omega(n \log^2 n)}$ chance that it is one of the "bad" $\boldsymbol{A}$'s that doesn't result in the protocol being witness indistinguishable.

Theorem 9. *For all but a $2^{-\Omega(n \log^2 n)}$ fraction of $\boldsymbol{A} \in \mathbb{Z}_p^{n \times m}$, the following holds true. For any two vectors $\tilde{\boldsymbol{w}}, \tilde{\boldsymbol{w}}' \in \{0,1\}^m$ where $\boldsymbol{A}\tilde{\boldsymbol{w}} \bmod p = \boldsymbol{A}\tilde{\boldsymbol{w}}' \bmod p = \boldsymbol{w}$, any cheating verifier $\mathcal{V}$, and auxiliary input string r,*

$$\Delta\left(\mathcal{V}_{\mathcal{P}(\boldsymbol{A},\tilde{\boldsymbol{w}})}(\boldsymbol{A}, \boldsymbol{w}, r), \mathcal{V}_{\mathcal{P}(\boldsymbol{A},\tilde{\boldsymbol{w}}')}(\boldsymbol{A}, \boldsymbol{w}, r)\right) \leq 2^{-\Omega(n \log^2 n)}.$$

Since the protocol is clearly witness indistinguishable when the verifier sends $c = 0$, we will assume that $c = 1$. We will show that

$$\Delta\left(\mathcal{V}_{\mathcal{P}(\boldsymbol{A},\tilde{\boldsymbol{w}})}(\boldsymbol{A}, \boldsymbol{w}, r), \mathcal{V}_{\mathcal{P}(\boldsymbol{A},\tilde{\boldsymbol{w}}')}(\boldsymbol{A}, \boldsymbol{w}, r)\right) \leq 2^{-n \log^2 n}$$

by showing that the distribution of the messages that the prover sends to the verifier is almost independent of whether the witness is $\tilde{\boldsymbol{w}}$ or $\tilde{\boldsymbol{w}}'$.

The messages that the prover sends to the verifier consist of the elements $\boldsymbol{y}$ and $\boldsymbol{z}$. For convenience, in the case that the witness is $\tilde{\boldsymbol{w}}$, we will use the variables $\boldsymbol{y}, \boldsymbol{z}$ and when the witness is $\tilde{\boldsymbol{w}}'$, we will use the variables $\boldsymbol{y}', \boldsymbol{z}'$.

$$\begin{aligned}
&\Delta\left(\mathcal{V}_{\mathcal{P}(\boldsymbol{A},\tilde{\boldsymbol{w}})}(\boldsymbol{A}, \boldsymbol{w}, r), \mathcal{V}_{\mathcal{P}(\boldsymbol{A},\tilde{\boldsymbol{w}}')}(\boldsymbol{A}, \boldsymbol{w}, r)\right) && (2)\\
&\leq \frac{1}{2} \sum_{(\boldsymbol{\alpha},\boldsymbol{\beta})} |Pr[(\boldsymbol{y}, \boldsymbol{z}) = (\boldsymbol{\alpha}, \boldsymbol{\beta})] - Pr[(\boldsymbol{y}', \boldsymbol{z}') = (\boldsymbol{\alpha}, \boldsymbol{\beta})]| && (3)\\
&= \frac{1}{2} \sum_{(\boldsymbol{\alpha},\boldsymbol{\beta} \neq \perp)} |Pr[(\boldsymbol{y}, \boldsymbol{z}) = (\boldsymbol{\alpha}, \boldsymbol{\beta})] - Pr[(\boldsymbol{y}', \boldsymbol{z}') = (\boldsymbol{\alpha}, \boldsymbol{\beta})]| && (4)\\
&+ \frac{1}{2} \sum_{(\boldsymbol{\alpha},\boldsymbol{\beta} = \perp)} |Pr[(\boldsymbol{y}, \boldsymbol{z}) = (\boldsymbol{\alpha}, \perp)] - Pr[(\boldsymbol{y}', \boldsymbol{z}') = (\boldsymbol{\alpha}, \perp)]| && (5)
\end{aligned}$$

In the above equations, the sums are over all $\boldsymbol{\alpha} \in \{0, 1, \ldots, 5m-1\}^m$ and $\boldsymbol{\beta} \in \{1, 2, \ldots, 5m-1\}^m \cup \{\perp\}$.

We will finish the proof of the theorem by showing that (4) is 0 for all matrices $\boldsymbol{A} \in \mathbb{Z}_p^{n \times m}$ (Lemma 10), and (5) is negligibly small for all but a $2^{-\Omega(n \log^2 n)}$ fraction of $\boldsymbol{A} \in \mathbb{Z}_p^{n \times m}$ (Lemma 11).

Lemma 10

$$\frac{1}{2} \sum_{(\boldsymbol{\alpha},\boldsymbol{\beta} \neq \perp)} |Pr[(\boldsymbol{y}, \boldsymbol{z}) = (\boldsymbol{\alpha}, \boldsymbol{\beta})] - Pr[(\boldsymbol{y}', \boldsymbol{z}') = (\boldsymbol{\alpha}, \boldsymbol{\beta})]| = 0$$

Proof. We will show that for every $\boldsymbol{\alpha}$ and $\boldsymbol{\beta} \neq \perp$,

$$Pr[(\boldsymbol{y}, \boldsymbol{z}) = (\boldsymbol{\alpha}, \boldsymbol{\beta})] = Pr[(\boldsymbol{y}', \boldsymbol{z}') = (\boldsymbol{\alpha}, \boldsymbol{\beta})]. \quad (6)$$

We rewrite $Pr[(\boldsymbol{y}, \boldsymbol{z}) = (\boldsymbol{\alpha}, \boldsymbol{\beta})]$ as

$$\begin{aligned} Pr[(\boldsymbol{y}, \boldsymbol{z}) = (\boldsymbol{\alpha}, \boldsymbol{\beta})] &= Pr[\boldsymbol{A}\tilde{\boldsymbol{y}} \bmod p = \boldsymbol{\alpha} \wedge \tilde{\boldsymbol{y}} + \tilde{\boldsymbol{w}} = \boldsymbol{\beta}] \\ &= Pr[\boldsymbol{A}\tilde{\boldsymbol{y}} \bmod p = \boldsymbol{\alpha} | \tilde{\boldsymbol{y}} + \tilde{\boldsymbol{w}} = \boldsymbol{\beta}] Pr[\tilde{\boldsymbol{y}} + \tilde{\boldsymbol{w}} = \boldsymbol{\beta}] \end{aligned}$$

And similarly,

$$Pr[(\boldsymbol{y}', \boldsymbol{z}') = (\boldsymbol{\alpha}, \boldsymbol{\beta})] = Pr[\boldsymbol{A}\tilde{\boldsymbol{y}}' \bmod p = \boldsymbol{\alpha} | \tilde{\boldsymbol{y}}' + \tilde{\boldsymbol{w}}' = \boldsymbol{\beta}] Pr[\tilde{\boldsymbol{y}}' + \tilde{\boldsymbol{w}}' = \boldsymbol{\beta}].$$

Notice that the probability $Pr[\boldsymbol{A}\tilde{\boldsymbol{y}} \bmod p = \boldsymbol{\alpha} | \tilde{\boldsymbol{y}} + \tilde{\boldsymbol{w}} = \boldsymbol{\beta}]$ is being conditioned on $\tilde{\boldsymbol{y}}$, which is the only random variable in the expression, and thus the probability evaluates to either 1 or 0. It is 1 whenever $\boldsymbol{A}(\boldsymbol{\beta} - \tilde{\boldsymbol{w}}) \bmod p = \boldsymbol{\alpha}$ and it is 0 otherwise. Similarly, $Pr[\boldsymbol{A}\tilde{\boldsymbol{y}}' \bmod p = \boldsymbol{\alpha} | \tilde{\boldsymbol{y}}' + \tilde{\boldsymbol{w}}' = \boldsymbol{\beta}] = 1$ whenever $\boldsymbol{A}(\boldsymbol{\beta} - \tilde{\boldsymbol{w}}') \bmod p = \boldsymbol{\alpha}$ and 0 otherwise. The important thing is that $\boldsymbol{A}(\boldsymbol{\beta} - \tilde{\boldsymbol{w}}) \bmod p = \boldsymbol{A}(\boldsymbol{\beta} - \tilde{\boldsymbol{w}}') \bmod p$ (because $\boldsymbol{A}\tilde{\boldsymbol{w}} \bmod p = \boldsymbol{A}\tilde{\boldsymbol{w}}' \bmod p$) and thus

$$Pr[\boldsymbol{A}\tilde{\boldsymbol{y}} \bmod p = \boldsymbol{\alpha} | \tilde{\boldsymbol{y}} + \tilde{\boldsymbol{w}} = \boldsymbol{\beta}] = Pr[\boldsymbol{A}\tilde{\boldsymbol{y}}' \bmod p = \boldsymbol{\alpha} | \tilde{\boldsymbol{y}}' + \tilde{\boldsymbol{w}}' = \boldsymbol{\beta}].$$

So all that remains to show to prove the equality in equation (6) is to show that

$$Pr[\tilde{\boldsymbol{y}} + \tilde{\boldsymbol{w}} = \boldsymbol{\beta}] = Pr[\tilde{\boldsymbol{y}}' + \tilde{\boldsymbol{w}}' = \boldsymbol{\beta}].$$

This is done by observing that since $\boldsymbol{\beta} \neq \perp$, it must be in the set SAFE, which means that all coefficients of $\boldsymbol{\beta}$ are between 1 and $5m - 1$. And since the coefficients of $\tilde{\boldsymbol{w}}$ are all 0 or 1, the coefficients of $\boldsymbol{\beta} - \tilde{\boldsymbol{w}}$ are between 0 and $5m - 1$, which is exactly the range that $\tilde{\boldsymbol{y}}$ is chosen uniformly from. Thus,

$$Pr[\tilde{\boldsymbol{y}} + \tilde{\boldsymbol{w}} = \boldsymbol{\beta}] = Pr[\tilde{\boldsymbol{y}} = \boldsymbol{\beta} - \tilde{\boldsymbol{w}}] = 1/(5m)^m$$

for all values of $\boldsymbol{\beta}$ and any secret key $\tilde{\boldsymbol{w}}$. And by the same reasoning, we have $Pr[\tilde{\boldsymbol{y}}' = \boldsymbol{\beta} - \tilde{\boldsymbol{w}}'] = 1/(5m)^m$. □

Lemma 11. *For all but a $2^{-\Omega(n \log^2 n)}$ fraction of possible $\boldsymbol{A} \in \mathbb{Z}_p^{n \times m}$,*

$$\frac{1}{2} \sum_{(\boldsymbol{\alpha}, \boldsymbol{\beta} = \perp)} |Pr[(\boldsymbol{y}, \boldsymbol{z}) = (\boldsymbol{\alpha}, \perp)] - Pr[(\boldsymbol{y}', \boldsymbol{z}') = (\boldsymbol{\alpha}, \perp)]| \leq 2^{-\Omega(n \log^2 n)}$$

Proof. Define the set $S_{\tilde{\boldsymbol{w}}} = \{\tilde{\boldsymbol{y}} \in \{0, \ldots, 5m - 1\}^m$ such that $\tilde{\boldsymbol{y}} + \tilde{\boldsymbol{w}} \notin \text{SAFE}\}$. The two important characteristics of the sets $S_{\tilde{\boldsymbol{w}}}$ and $S_{\tilde{\boldsymbol{w}}'}$, for any two secret keys $\tilde{\boldsymbol{w}}$ and $\tilde{\boldsymbol{w}}'$, is that their sizes are equivalent and "large enough". Both of these are implicit from equation (1) in Lemma 7. More precisely,

$$|S_{\tilde{\boldsymbol{w}}}| = |S_{\tilde{\boldsymbol{w}}'}| = (5m)^m - (5m)^m \left(1 - \frac{1}{5m}\right)^m \tag{7}$$

$$\geq (5m)^m - (5m)^m \left(\frac{1}{e}\right)^{1/5} \geq \frac{(5m)^m}{6} \tag{8}$$

We now proceed with the proof of the lemma.

$$\frac{1}{2}\sum_{(\boldsymbol{\alpha},\boldsymbol{\beta}=\perp)} |Pr[(\boldsymbol{y},\boldsymbol{z}) = (\boldsymbol{\alpha},\perp)] - Pr[(\boldsymbol{y}',\boldsymbol{z}') = (\boldsymbol{\alpha},\perp)]| \tag{9}$$

$$= \frac{1}{2}\sum_{\boldsymbol{\alpha}} |Pr[\boldsymbol{A}\tilde{\boldsymbol{y}} \bmod p = \boldsymbol{\alpha} \ \wedge \ \tilde{\boldsymbol{y}} \in S_{\tilde{\boldsymbol{w}}}] \tag{10}$$

$$- Pr[\boldsymbol{A}\tilde{\boldsymbol{y}}' \bmod p = \boldsymbol{\alpha} \ \wedge \ \tilde{\boldsymbol{y}}' \in S_{\tilde{\boldsymbol{w}}'}]| \tag{11}$$

$$= \frac{1}{2}\sum_{\boldsymbol{\alpha}} |Pr[\boldsymbol{A}\tilde{\boldsymbol{y}} \bmod p = \boldsymbol{\alpha} | \tilde{\boldsymbol{y}} \in S_{\tilde{\boldsymbol{w}}}] Pr[\tilde{\boldsymbol{y}} \in S_{\tilde{\boldsymbol{w}}}] \tag{12}$$

$$- Pr[\boldsymbol{A}\tilde{\boldsymbol{y}}' \bmod p = \boldsymbol{\alpha} | \tilde{\boldsymbol{y}}' \in S_{\tilde{\boldsymbol{w}}'}] Pr[\tilde{\boldsymbol{y}}' \in S_{\tilde{\boldsymbol{w}}'}]| \tag{13}$$

$$\leq \frac{1}{2}\sum_{\boldsymbol{\alpha}} |Pr[\boldsymbol{A}\tilde{\boldsymbol{y}} \bmod p = \boldsymbol{\alpha} | \tilde{\boldsymbol{y}} \in S_{\tilde{\boldsymbol{w}}}] - Pr[\boldsymbol{A}\tilde{\boldsymbol{y}}' \bmod p = \boldsymbol{\alpha} | \tilde{\boldsymbol{y}}' \in S_{\tilde{\boldsymbol{w}}'}]| \tag{14}$$

$$= \frac{1}{2}\sum_{\boldsymbol{\alpha}} \left| Pr_{\tilde{\boldsymbol{y}} \xleftarrow{\$} S_{\tilde{\boldsymbol{w}}}}[\boldsymbol{A}\tilde{\boldsymbol{y}} \bmod p = \boldsymbol{\alpha}] - Pr_{\tilde{\boldsymbol{y}}' \xleftarrow{\$} S_{\tilde{\boldsymbol{w}}'}}[\boldsymbol{A}\tilde{\boldsymbol{y}}' \bmod p = \boldsymbol{\alpha}] \right| \tag{15}$$

The inequality in equation (14) is true because $|S_{\tilde{\boldsymbol{w}}}| = |S_{\tilde{\boldsymbol{w}}'}|$, and so $Pr[\tilde{\boldsymbol{y}} \in S_{\tilde{\boldsymbol{w}}}] = Pr[\tilde{\boldsymbol{y}}' \in S_{\tilde{\boldsymbol{w}}'}] < 1$. We now notice that equation (15) is the statistical distance between the distributions $\boldsymbol{A}\tilde{\boldsymbol{y}} \bmod p$ and $\boldsymbol{A}\tilde{\boldsymbol{y}}' \bmod p$ where $\tilde{\boldsymbol{y}}$ and $\tilde{\boldsymbol{y}}'$ are chosen uniformly from the sets $S_{\tilde{\boldsymbol{w}}}$ and $S_{\tilde{\boldsymbol{w}}'}$ respectively. Using the fact that $|S_{\tilde{\boldsymbol{w}}}| = |S_{\tilde{\boldsymbol{w}}'}| = \Omega(m \log m) = \Omega(n \log^2 n)$ and $p = \tilde{O}(n^3)$, we apply Corollary 6 to obtain the claim of the lemma. □

Having shown that one round of the protocol is witness indistinguishable, we move on to building the full identification scheme (see Figure 2). As we alluded to earlier, the scheme will not have perfect completeness since an honest prover will sometimes have to refuse to answer and thus get rejected by the verifier. Nevertheless, by having enough rounds, an adversary will reject an honest adversary with negligible probability.

Lemma 12. *The identification protocol in Figure 2 has completeness error less than* $2^{-t/14}$.

Proof. By Lemma 7, we know that the honest prover will respond correctly to challenge c_i with probability at least .81. Since the prover is honest, the probabilities of success are independent for all the challenges, and so using the Chernoff bound, we obtain:

$$Pr[\text{REJECT}] = Pr[sum < .65t] = Pr[sum < (.81 - .16)t] \leq e^{-2t(.16^2)} < 2^{-t/14}$$

□

Thus setting $t = \omega(\log n)$ results in the protocol having negligible completeness error.

We now move to proving the security of the ID scheme. We will show that an adversary who successfully attacks the protocol can be used to successfully solve

Prover		Verifier
Private key: $\tilde{\boldsymbol{w}} \xleftarrow{\$} \{0,1\}^m$ Public key: $\boldsymbol{A} \xleftarrow{\$} \mathbb{Z}_p^{n \times m}$ $\boldsymbol{w} \leftarrow \boldsymbol{A}\tilde{\boldsymbol{w}} \bmod p$		
for $i = 1$ to t $\tilde{\boldsymbol{y}}_i \xleftarrow{\$} \{0, 1, \ldots, 5m-1\}^m$ $\boldsymbol{y}_i \leftarrow \boldsymbol{A}\tilde{\boldsymbol{y}}_i \bmod p$		
	$\boldsymbol{y}_1, \ldots, \boldsymbol{y}_t \rightarrow$	
		for $i = 1$ to t $c_i \xleftarrow{\$} \{0,1\}$
	$\leftarrow c_1, \ldots, c_t$	
for $i = 1$ to t if $c_i = 1$ and $\tilde{\boldsymbol{y}}_i + \tilde{\boldsymbol{w}}_i \notin$ SAFE $\boldsymbol{z}_i \leftarrow \perp$ else $\boldsymbol{z}_i \leftarrow \tilde{\boldsymbol{y}}_i + c\boldsymbol{w}_i$		
	$\boldsymbol{z}_1, \ldots, \boldsymbol{z}_t \rightarrow$	

		for $i = 1$ to t
		if ($\|\boldsymbol{z}_i\| \leq 5m^{1.5}$ and
		$\boldsymbol{A}\boldsymbol{z}_i \bmod p = c_i\boldsymbol{w} + \boldsymbol{y}_i$)
		$d_i \leftarrow 1$
		else
		$d_i \leftarrow 0$
		$sum = d_1 + \ldots + d_t$
		if $sum \geq 0.65t$ then ACCEPT
		else REJECT

Fig. 2. The identification scheme. The parameters are $p = \tilde{O}(n^3)$, $m = \lceil 4n \log n \rceil$, $t = \omega(\log n)$, and the set SAFE is defined as $\{1, \ldots, 5m-1\}^m$.

the SIS problem for random $\boldsymbol{A}$. By Theorem 2, this implies that this adversary can be used to approximate the length of the Shortest Vector to within a factor of $\tilde{O}(n^2)$ in every lattice.

Theorem 13. *If there exists a polynomial-time adversary who can break the ID protocol in Figure 2 with probability adv in the active attack model, then there exists a polynomial-time algorithm that solves the* SIS($\boldsymbol{A}$) *problem with success probability* $\Omega\left((adv)^2 - 2 \cdot 2^{-t/18}\right)$ *when* $\boldsymbol{A}$ *is chosen uniformly at random from* $\mathbb{Z}_p^{n \times m}$.

Proof. We explain how to build an algorithm that solves the SIS($\boldsymbol{A}$) problem using an adversary attacking the identification scheme. Given a random matrix $\boldsymbol{A} \xleftarrow{\$} \mathbb{Z}_p^{n \times m}$, we create a random secret key $\tilde{\boldsymbol{w}} \xleftarrow{\$} \{0,1\}^m$, and output $\boldsymbol{A}$ and $\boldsymbol{w} \leftarrow \boldsymbol{A}\tilde{\boldsymbol{w}} \bmod p$ as the public key of the identification scheme. Since we know the

secret key, we can perfectly simulate the identification scheme with an adversary who is acting as the verifier. If the adversary wishes to interact with more than one prover, we can easily accommodate him by creating more secret keys $\tilde{\boldsymbol{w}}_i$ and public keys $\boldsymbol{w}_i \leftarrow \boldsymbol{A}\tilde{\boldsymbol{w}}_i \bmod p$ and perfectly simulate those interactions as well.

After the adversary finishes his interaction with the prover(s), it's now his turn to perform an impersonation of the prover whose public key is $(\boldsymbol{A}, \boldsymbol{w})$. We will use this impersonation to extract a solution to the SIS($\boldsymbol{A}$) problem. In the first step of the protocol, the adversary sends us t vectors $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_t$. We reply by sending t random challenges $c_1, \ldots, c_t$. The adversary replies with vectors $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_t$. We then rewind the adversary, and send another set of independently random challenges $c'_1, \ldots, c'_t$ and receive responses $\boldsymbol{z}'_1, \ldots, \boldsymbol{z}'_t$. We then find an i such that $c_i \neq c'_i$, $\boldsymbol{A}\boldsymbol{z}_i \bmod p = c_i\boldsymbol{w} + \boldsymbol{y}_i$, and $\boldsymbol{A}\boldsymbol{z}'_i \bmod p = c'_i\boldsymbol{w} + \boldsymbol{y}_i$ (the fact that such an i exists will be shown later). Without loss of generality, suppose that $c_i = 1$ and $c'_i = 0$. We thus obtain that

$$\boldsymbol{A}(\boldsymbol{z}_i - \boldsymbol{z}'_i) \bmod p = \boldsymbol{w} = \boldsymbol{A}\tilde{\boldsymbol{w}} \bmod p. \tag{16}$$

Since our identification scheme is witness-indistinguishable, and there is at least one other $\tilde{\boldsymbol{w}}' \in \{0,1\}^m$ such that $\boldsymbol{A}\tilde{\boldsymbol{w}} \bmod p = \boldsymbol{A}\tilde{\boldsymbol{w}}' \bmod p$ (Lemma 8), the probability that $\boldsymbol{z}_i - \boldsymbol{z}'_i = \tilde{\boldsymbol{w}}$ is at most 1/2. Also, $\|\boldsymbol{z}_i - \boldsymbol{z}'_i\| \leq \|\boldsymbol{z}_i\| + \|\boldsymbol{z}'_i\| \leq 10m^{1.5}$. Thus, with probability at least 1/2, the values $\boldsymbol{z}_i - \boldsymbol{z}'_i$ and $\tilde{\boldsymbol{w}}$ are a solution to the SIS($\boldsymbol{A}$) problem.

What we now need to show that with high probability, there indeed will exist an i such that $c_i \neq c'_i$, $\boldsymbol{A}\boldsymbol{z}_i \bmod p = c_i\boldsymbol{w} + \boldsymbol{y}_i$, and $\boldsymbol{A}\boldsymbol{z}'_i \bmod p = c'_i\boldsymbol{w} + \boldsymbol{y}_i$. We will call this condition ($\star$). We will say that a pair of challenge sequences $c_1, \ldots, c_t$ and $c'_1, \ldots, c'_t$ is *good* if $\sum_i |c_i - c'_i| > .35t$ (they differ on more than $.35t$ coordinates). Notice that if the adversary succeeds in impersonating on both sequences of a *good* pair, then by the pigeonhole principle, ($\star$) will be satisfied[6]. By the Chernoff bound, the probability that a random pair of sequences is not *good* is

$$Pr\left[\sum_{i=1}^{t} |c_i - c'_i| < .36t\right] \leq e^{-2t(.14^2)} < 2^{-t/18}$$

The adversary succeeds on a random challenge sequence with probability adv, and thus succeeds on a pair of independently random sequences with probability $(adv)^2$. Since we just showed that at most a $2^{-t/18}$ fraction of all pairs is not *good*, we know that the adversary must be able to answer correctly on a randomly chosen *good* pair of sequences with probability at least $(adv)^2 - 2^{-t/18}$. Multiplying this by the probability that the pair of sequences we randomly chose is *good*, we get

$$Pr[(\star)] > \left((adv)^2 - 2^{-t/18}\right)\left(1 - 2^{-t/18}\right) > (adv)^2 - 2 \cdot 2^{-t/18} \qquad \square$$

[6] Recall that an adversary is allowed to answer incorrectly up to $.35t$ times and still be accepted, and this is why having just one i for which $c_i \neq c'_i$ is not enough.

Algorithm 1. (Attack on ID scheme given public keys $\boldsymbol{A} \in \mathbb{Z}_p^{n \times m}, \boldsymbol{w} \in \mathbb{Z}_p^n$)

1: Find $\tilde{\boldsymbol{w}}' \in \{-5m, \ldots, -1, 0, 1, \ldots, 5m-1\}^m$ such that $\boldsymbol{A}\tilde{\boldsymbol{w}}' \bmod p = \boldsymbol{w}$
2: **for** $i = 1$ to t (performed concurrently for all i) **do**
3: Pick random $\tilde{\boldsymbol{y}}_i' \in \{0,1\}^m$. Set $\boldsymbol{y}_i \leftarrow \boldsymbol{A}\tilde{\boldsymbol{y}}_i' \bmod p$
4: Send $\boldsymbol{y}_i$ to the Verifier
5: Receive $c_i \in \{0,1\}$ from the Verifier
6: Set $\boldsymbol{z}_i \leftarrow c\tilde{\boldsymbol{w}}_i' + \tilde{\boldsymbol{y}}_i'$
7: Send $\boldsymbol{z}_i$ to the Verifier
8: **end for**

4 Ideal Lattices

In this section, we discuss how the identification scheme can be sped up by almost a factor n if we base its security on the hardness of finding the shortest vector in *ideal* lattices. The main savings in efficiency, and the only difference in the protocol, is that the matrix $\boldsymbol{A} \in \mathbb{Z}_p^{n \times m}$ will no longer be chosen at random from $\mathbb{Z}_p^{n \times m}$, but instead from $\mathrm{ROT}(n, m, p)$. Everything else in the identification scheme in Figure 2 remains exactly the same. Notice that the most expensive operation in the protocol is the multiplication $\boldsymbol{A}\tilde{\boldsymbol{y}} \bmod p$ for the prover and $\boldsymbol{A}\boldsymbol{z} \bmod p$ for the verifier, which involves $O(mn)$ multiplications of integers of bit length $\log p = O(\log n)$. But it's possible to exploit the algebraic structure of $\boldsymbol{A} \in \mathrm{ROT}(n, m, p)$, and perform that same matrix-vector multiplication by using the Fast Fourier Transform, and thus require only $O(m \log n)$ operations. The proof of security for the new protocol is extremely similar to the one already provided for general lattices. Thus, rather than providing complete proofs, we briefly sketch the necessary modifications.

It is still be true that each round of the protocol remains witness indistinguishable, and the proof of witness indistinguishability is almost the same. The only difference is that we have to be careful to make sure that Corollary 6 remains valid when the matrix $\boldsymbol{A}$ is chosen from $\mathrm{ROT}(n, m, p)$ rather than from all of $\mathbb{Z}_p^{n \times m}$. A condition that is sufficient for this is that we choose the parameter p in a way that makes the ring $\mathbb{Z}_p[x]/\langle x^n + 1\rangle$ a field (i.e. every element in the ring should have an inverse). We point out that it's also possible to prove witness-indistinguishability when $\mathbb{Z}_p[x]/\langle x^n + 1\rangle$ is not a field, but then we can no longer use the leftover hash lemma, and we would instead need to use a lemma very similar to Micciancio's regularity lemma [17, Theorem 4.2].

5 Attacks

We have shown that our identification schemes are provably secure in an asymptotic sense, but as we'll show in this section, they unfortunately cannot yet be put into practice because they are insecure for parameters that one might conceivably use in applications. The core issue behind our schemes' vulnerabilities is

that lattice-reduction algorithms seem to work better in practice than in theory. See Algorithm 1 for the description of the attack.

Notice that the vectors $\boldsymbol{z}_i$ will always have coordinates in the range between $-5m$ and $5m$, and so $\|\boldsymbol{z}_i\| \leq 5m^{1.5}$. Also notice that the adversary has no need to hide his "secret key" and so he never has to respond with $\perp$, and thus the verifier will always accept this interaction. The hard part is performing step 1 of the above attack. In fact, performing this step is as hard as approximating the shortest vector in all lattices to within a factor of $\tilde{O}(n^{1.5})$. As n grows large, this is believed to be a hard problem, but for small parameters, it is feasible to solve and we will explain this next.

The problem of finding the $\tilde{\boldsymbol{w}}'$ in step 1 is the problem of finding a vector $\boldsymbol{x}$ with small coefficients such that $\boldsymbol{A}\boldsymbol{x} \bmod p = \boldsymbol{y}$ where $\boldsymbol{A}$ is random matrix in $\mathbb{Z}_p^{n\times m}$ (or in $\mathrm{ROT}(n, m, p)$) and $\boldsymbol{y}$ is a random vector in $\mathbb{Z}_p^n$. We want to phrase this problem as a lattice reduction, and so we first construct the matrix $\boldsymbol{A}' = [\boldsymbol{A}|\boldsymbol{y}]$ and consider the problem of finding a vector $\boldsymbol{x}' \in \mathbb{Z}^{m+1}$ such that $\boldsymbol{A}'\boldsymbol{x}' \bmod p = \boldsymbol{0}$. Notice that if we are able to find such an $\boldsymbol{x}'$ all of whose coefficients are small and the last coefficient is -1, then we are able to find an $\boldsymbol{x}$ that solves the original problem. Also notice that all the $\boldsymbol{x}' \in \mathbb{Z}^{m+1}$ that satisfy $\boldsymbol{A}'\boldsymbol{x}' \bmod p = \boldsymbol{0}$ form an additive subgroup of $\mathbb{Z}^{m+1}$, and thus an integer lattice of dimension $m+1$. So what we need to do is first construct a basis of this lattice and then find a vector in it with coordinates between $-5m$ and $5m-1$ (and have the last coordinate be -1).

Constructing a basis for this lattice can be done in polynomial time by viewing $\boldsymbol{A}'$ as a linear function mapping $\mathbb{Z}^{m+1}$ to $\mathbb{Z}_p^n$ and computing the basis for its kernel. This basis is exactly the basis of the lattice we referred to above. It's not hard to see that by the pigeonhole principle the lattice has a vector all of whose coefficients are either $-1, 0$, or 1, and so finding a vector that has coefficients between $-5m$ and $5m-1$ roughly equates to finding a short vector within a factor of m of the shortest one. This becomes a hard problem as m gets large, but for small and medium-sized m that could potentially be used in practice (around 1000), lattice reduction algorithms can find such vectors fairly efficiently. And finding such a vector whose last coordinate is -1 is heuristically feasible.

6 Conclusions and Open Problems

We have presented a framework for constructing identification schemes that are secure in the active attack model based on the worst-case hardness of lattice problems. A lot of open questions remain, though. The most significant of these is whether the ideas presented in this paper can be used for the construction of an identification protocol that can be instantiated with practical-sized parameters. Recent results that provide practical instantiations [16] of collision resistant lattice-based hash functions based on theoretical ideas in [22,15] makes us optimistic that with some new ideas the same could be done for the identification schemes presented here.

A possible approach would be to see whether it is somehow plausible to pick the values $\tilde{\boldsymbol{y}}$ from a smaller set. Notice that the set that $\tilde{\boldsymbol{y}}$'s got picked from was designed so that for a random $\tilde{\boldsymbol{y}}$, the value of $\tilde{\boldsymbol{y}} + \tilde{\boldsymbol{w}}$ could be safely revealed with a high enough probability. Since the size of this set played a critical role in the attack, reducing it would make the attack more difficult to mount. Another open problem is to somehow modify the ID scheme so that it has perfect completeness. Having perfect completeness would allow us to reduce the number of rounds t in the protocol.

References

1. Ajtai, M.: Generating hard instances of lattice problems. In: STOC, pp. 99–108 (1996)
2. Ajtai, M.: The shortest vector problem in ℓ_2 is NP-hard for randomized reductions. In: STOC, pp. 10–19 (1998)
3. Ajtai, M., Dwork, C.: A public-key cryptosystem with worst-case/average-case equivalence. In: STOC, pp. 284–293 (1997)
4. Ajtai, M., Kumar, R., Sivakumar, D.: A sieve algorithm for the shortest lattice vector problem. In: STOC, pp. 601–610 (2001)
5. Blömer, J., Naewe, S.: Sampling methods for shortest vectors, closest vectors and successive minima. In: Arge, L., Cachin, C., Jurdziński, T., Tarlecki, A. (eds.) ICALP 2007. LNCS, vol. 4596, pp. 65–77. Springer, Heidelberg (2007)
6. Blömer, J., Seifert, J.-P.: On the complexity of computing short linearly independent vectors and short bases in a lattice. In: STOC, pp. 711–720 (1999)
7. Feige, U., Fiat, A., Shamir, A.: Zero-knowledge proofs of identity. J. Cryptology 1(2), 77–94 (1988)
8. Feige, U., Shamir, A.: Witness indistinguishable and witness hiding protocols. In: STOC, pp. 416–426 (1990)
9. Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg (1987)
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. Guillou, L., Quisquater, J.J.: A "paradoxical" indentity-based signature scheme resulting from zero-knowledge. In: Goldwasser, S. (ed.) CRYPTO 1988. LNCS, vol. 403, pp. 216–231. Springer, Heidelberg (1990)
12. Impagliazzo, R., Zuckerman, D.: How to recycle random bits. In: FOCS, pp. 248–253 (1989)
13. Khot, S.: Hardness of approximating the shortest vector problem in lattices. In: FOCS, pp. 126–135 (2004)
14. Kumar, R., Sivakumar, D.: On polynomial-factor approximations to the shortest lattice vector length. SIAM J. Discrete Math. 16(3), 422–425 (2003)
15. Lyubashevsky, V., Micciancio, D.: Generalized compact knapsacks are collision resistant. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4052, pp. 144–155. Springer, Heidelberg (2006)
16. Lyubashevsky, V., Micciancio, D., Peikert, C., Rosen, A.: SWIFFT: A modest proposal for FFT hashing. In: Fast Software Encryption (FSE) (2008); Preliminary version appeared at the 2nd NIST Cryptographic Hash Function Workshop (to appear)

17. Micciancio, D.: Generalized compact knapsacks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions. In: Computational Complexity (2002); Preliminary version in FOCS 2002 (to appear)
18. Micciancio, D.: Efficient reductions among lattice problems. In: SODA (to appear, 2008)
19. Micciancio, D., Regev, O.: Worst-case to average-case reductions based on Gaussian measures. SIAM J. on Computing 37(1), 267–302 (2007)
20. Micciancio, D., Vadhan, S.: Statistical zero-knowledge proofs with efficient provers: Lattice problems and more. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 282–298. Springer, Heidelberg (2003)
21. Okamoto, T.: Provably secure and practical identification schemes and corresponding signature schemes. In: Brickell, E.F. (ed.) CRYPTO 1992. LNCS, vol. 740, pp. 31–53. Springer, Heidelberg (1993)
22. Peikert, C., Rosen, A.: Efficient collision-resistant hashing from worst-case assumptions on cyclic lattices. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, Springer, Heidelberg (2006)
23. Pointcheval, D.: The composite discrete logarithm and secure authentication. In: Public Key Cryptography, pp. 113–128 (2000)
24. Regev, O.: New lattice based cryptographic constructions. In: STOC, pp. 407–416 (2003)
25. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: STOC (2005)
26. Schnorr, C.P.: Efficient signature generation by smart cards. J. Cryptology 4(3), 161–174 (1991)
27. Shamir, A.: An efficient identification scheme based on permuted kernels (extended abstract). In: Brassard, G. (ed.) CRYPTO 1989. LNCS, vol. 435, pp. 606–609. Springer, Heidelberg (1990)
28. Shor, P.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM J. Comput. 26(5), 1484–1509 (1997)
29. Shoup, V.: On the security of a practical identification scheme. J. Cryptology 12(4), 247–260 (1999)
30. Stern, J.: A new paradigm for public key identification. IEEE Transactions on Information Theory 42 (1996)