

Adaptively Secure Identity-based Identification from Lattices without Random Oracles

Markus Rückert*
markus.rueckert@cased.de

Technische Universität Darmstadt
Department of Computer Science
Cryptography and Computeralgebra
Germany

Abstract. We propose a concurrently secure, identity-based identification scheme from lattices. It offers adaptive-identity security in the standard model, quasi optimal online performance, optimal leakage resilience, and its security is based on mild worst-case assumptions in ideal lattices. Our scheme uses an ideal-lattice interpretation of the Bonsai tree concept in lattices (EUROCRYPT 2010), which we call *convoluted* Bonsai trees. It allows us to build an identity-based identification scheme in a new “static identity” model that is weaker than the standard “adaptive identity” model. We show that both models are equivalent under the existence of Chameleon hash functions.

Keywords Lattice cryptography, identification, identity-based cryptography, security model

1 Introduction

Identification schemes are one of the most important primitives in modern cryptography because typical e-business or e-government applications essentially rely on secure online access control. Their importance is likely to grow in the future as more and more everyday tasks and processes are computerized. With identity-based identification schemes (IBI), motivated by Shamir [Sha84], one can get rid of public-key infrastructures, which are unfavorable in today’s widespread decentralized networks. The public key is replaced with a unique identifier string, such as an e-mail address, and the secret key is “extracted” by a trusted party for this identifier. In hierarchical identity-based identification (HIBI), motivated by Gentry and Silverberg [GS02], this concept is generalized so that each party can act as a key extraction authority for its subordinates. Thus, this concept perfectly models organizational structures in, e.g., a company. Currently, we mainly use schemes based on the factoring or discrete logarithm problem.

Our current knowledge suggests that alternatives for the post-quantum era can be based on the hardness of the decoding problem in error correcting codes,

* This work was supported by CASED (www.cased.de).

on the hardness of solving non-linear multivariate equation systems, or on the hardness of lattice problems. Refer to [BBD08] for an overview of each field. Basically, all three alternatives rely on the hardness of certain *average-case* problems and, at first, it is unclear how to generate hard instances of these problems. More precisely, we always need to know a “hard” distribution of keys that admits efficient key generation. Unlike with multivariate or code-based cryptography, lattice-based constructions have a built-in “trust anchor” in the form of Ajtai’s worst-case to average-case reduction [Ajt96]. This reduction is even stronger than a random self reductions in, e.g., the discrete logarithm problem. It states that solving a certain average-case problem, which is relevant in cryptography, implies a solution to a related worst-case problem in *all* lattices. Although this may sound purely theoretical, it is of great practical value as keys that are chosen uniformly at random already provide *worst-case* security guarantees. The hardness of this underlying worst-case problem is also plausible as the best known algorithm to solve it requires exponential time [AKS01,MV10].

It is well-known that identity-based identification schemes can be realized in the standard model with a so-called certification approach due to Bellare, Neven, and Namprempre [BNN09] but these generic, black-box constructions require a certain computational and bandwidth overhead. The only known direct constructions, which are conjectured to resist quantum computer attacks, are the code-based scheme of Cayrel, Gaborit, Galindo, and Girault [CGGG09] and the lattice-based scheme of Stehlé, Steinfeld, Tanaka, and Xagawa [SSTX09]. However, both are only provably secure in the random oracle model and the code-based scheme merely resists passive attacks, where the adversary may not interact with the secret-key holder before his or her impersonation attempt. Another approach is using identity-based signature schemes, e.g., the lattice-based construction due to Rückert [Rüc10], in a challenge-response protocol for identification. This, however, would make the online phase significantly less efficient compared to the solution in this paper.

Therefore, we fill a gap with our proposal, as it is the first direct construction of an adaptive-identity secure identity-based identification scheme that is secure under active attacks without random oracles. We modify the identification scheme of Lyubashevsky [Lyu08a,Lyu08b] to support key extraction via an ideal-lattice interpretation of the Bonsai tree principle, originally due to Cash, Hofheinz, Kiltz, and Peikert [CHKP10]. Our changes make it necessary to reprove the security of the protocol and we provide a simpler, more modular proof than the one in [Lyu08b] by exploiting the Reset Lemma of Bellare and Palacio [BP02]. This modification also makes the reduction a little tighter. The resulting IBI offers quasi-linear efficiency with respect to secret identification keys, public keys, bandwidth, and computation. The only exception from this quasi-optimality is the quasi-quadratic master public key and the quasi-quadratic complexity of the secret-key extraction procedure. Boyen [Boy10] improves the construction in [CHKP10] to allow for smaller lattice dimensions at the expense of a stronger assumption. We believe that the technique can be adapted for our setting to extract secret identification keys in a smaller dimension.

We also introduce a new, intermediary security model that is akin to the static message attack model for ordinary signatures (SMA), which is sometimes also referred to as “weak unforgeability”. It is well-known that such SMA secure schemes are easier to realize than full CMA secure schemes. We demonstrate that the same holds for identity-based “authentication-type” schemes, e.g., for IBI or identity-based signatures. Therefore, we show a generic conversion from static-identity attack security to full adaptive-identity security of identity-based identification schemes. This transformation carries over to identity-based signature schemes and also holds in the hierarchical setting. This transformation does not hold in identity-based encryption-type schemes because the message flow is reversed. For our transformation, it is crucial that the secret key holder can send a message to the public key holder. In encryption schemes, this is not possible. In signature and identification schemes, however, this is exactly what happens.

With our new model, we greatly simplify the security proofs for direct constructions because the simulator has access to all secret-key extraction queries before the actual simulation and it can therefore “rig” the public key accordingly.

Furthermore, as an aside, our identification scheme is leakage-resilient, supporting a per-identity leakage of a $(1 - o(1))$ fraction of the identities’ secret keys in a model that is inspired by Katz and Vaikuntanathan [KV09].

Organization. After some basic facts about identity-based identification schemes, we introduce our weaker, static-identity security model in Section 2. There, we also provide the necessary background about lattices and Chameleon hash functions. In the next section, Section 3, we explain how static-identity security implies adaptive-identity security if Chameleon hash functions exist. Then, we demonstrate how to construct an identity-based identification scheme from lattices in the weaker model in Section 4. We then conclude the paper in Section 5 and prove additional, supporting lemmas and a result concerning leakage-resilience in the full version.

2 Preliminaries

With n , we always denote the security parameter. The joint execution of two algorithms \mathcal{A} and \mathcal{B} in an interactive protocol with private inputs x to \mathcal{A} and y to \mathcal{B} is written as $b \leftarrow \langle \mathcal{A}(x), \mathcal{B}(y) \rangle$, where b is the result of the interaction. Accordingly, $\langle \mathcal{A}(x), \mathcal{B}(y) \rangle^k$ means that the interaction can take place up to k times. The statement $x \leftarrow_{\S} X$ means that x is chosen uniformly at random from the finite set X . When X is an algorithm, it means that X is probabilistic. Recall that the statistical distance of two random variables X, Y over a discrete domain D is defined as $\Delta(X, Y) = 1/2 \sum_{a \in D} |\text{Prob}[X = a] - \text{Prob}[Y = a]|$. A function is negligible if it vanishes faster than $1/p(n)$ for any polynomial p . All logarithms are base 2, we identify $\{1, \dots, k\}$ with $[k]$, and $\{a_1, \dots, a_k\}$ with $\{a_i\}_1^k$. With $\omega, \Omega, \mathcal{O}$ we denote the usual Landau symbols for asymptotic growth and $\tilde{\mathcal{O}}$ is like \mathcal{O} but it hides poly-logarithmic terms. The concatenation of strings,

vectors, matrices (column-wise) is done with the operator $\|$ and $a \sqsubset b$ means that $b = a\|c$ for some, possibly empty, string c .

2.1 Identity-based Identification

An ID-based identification scheme IBI comprises a triple $(\text{Kg}, \text{Extract}, \text{Protocol})$ of algorithms: The master-key generator Kg outputs a master secret key msk and a master public key mpk ; the key extraction algorithm Extract uses msk to generate a secret key sk_{ID} for a given identity ID ; and the identification protocol Protocol is a joint execution of a prover $\mathcal{P}_{\text{ID}}(\text{mpk}, \text{ID}, \text{sk}_{\text{ID}})$ and a verifier $\mathcal{V}(\text{mpk}, \text{ID})$, where \mathcal{V} outputs 1 iff \mathcal{P} could identify itself correctly.

The security model for identity-based identification [Sha84] was first formalized by Kurosawa and Heng [KH05] and it is also discussed in the recent work of Bellare, Neven, and Namprempre [BNN09]. Security is proven against concurrent identity-based impersonation under adaptive identity attacks as described in the `adapt-id-imp-ca` experiment in Figure 1, where the adversary (impersonator) \mathcal{I}^* works in two modes: `verify` and `impersonate`. In mode `verify` it has access to mpk , to a secret key extraction oracle Extract and to provers \mathcal{P}_{ID} for arbitrary identities ID . At some point, it selects a target identity ID^* , which it tries to impersonate in the second phase. In mode `impersonate`, \mathcal{I}^* has access to provers and secret keys for all identities different from ID^* and it is supposed to convince a verifier that it knows the secret key for ID^* . Obviously, the secret key for ID^* must not have been among the queries to the extraction oracle in the first phase. Also, note that \mathcal{I}^* is allowed to keep a state `st.verify`. The usual security notions for identification schemes, passive (pa), active (aa), and concurrent (ca) apply as well and the experiments can be easily changed to cover these attacks.

In Figure 1, we also propose a relaxed security model, called security against concurrent identity-based impersonation under static identity attacks (`stat-id-imp-ca`). The model gives \mathcal{I}^* significantly less power as the adversary has to submit all identities (*distinct*) to the oracle Extract before seeing the master public key. It then receives the extracted secret keys together with mpk . The remaining experiment stays unchanged. This new security model is reminiscent of that for weak unforgeability, or unforgeability under static message attacks, of digital signatures. Via a black-box transformation in Section 3, we show that both models are equivalent if Chameleon hash functions (cf. Section 2.3) exist. This transformation in conjunction with our simplified model enables much simpler designs for identity-based identification and it is also applicable to identity-based signature schemes. The resulting schemes are potentially more efficient and their security proofs are greatly simplified because one can prepare for all key extraction queries before handing over the master public key.

All definitions easily carry over the hierarchical setting [GS02], where identities can be concatenated to describe a subordinate identity and its relation in an organizational structure. Here, every entity can act as a key extraction authority for its subordinates.

Experiment $\text{Exp}_{\mathcal{Z}^*, \text{IBI}}^{\text{adapt-id-imp-ca}}(n)$ $(\text{msk}, \text{mpk}) \leftarrow_{\S} \text{IBI.Kg}(1^n)$ $(\text{ID}^*, \text{st_verify}) \leftarrow_{\S} \mathcal{Z}^{*(\mathcal{P}_{\text{ID}, \cdot})^\infty, \text{Extract}(\text{msk}, \cdot)}(\text{verify}, \text{mpk})$ Let $\{\text{ID}_i\}_1^\ell$ be the ID's queried to Extract. $b \leftarrow_{\S} (\mathcal{Z}^{*(\mathcal{P}_{\neq \text{ID}^*, \cdot})^\infty, \text{Extract}_{\neq \text{ID}^*}(\text{msk}, \cdot)}, \mathcal{V})((\text{impersonate}, \text{st_verify}), \text{ID}^*)$ Return b	Experiment $\text{Exp}_{\mathcal{Z}^*, \text{IBI}}^{\text{stat-id-imp-ca}}(n)$ $(\text{ID}_1, \dots, \text{ID}_\ell, \text{st_find}) \leftarrow_{\S} \mathcal{Z}^*(\text{find})$ for distinct ID_i $(\text{msk}, \text{mpk}) \leftarrow_{\S} \text{IBI.Kg}(1^n)$ $\text{sk}_i \leftarrow_{\S} \text{Extract}(\text{msk}, \text{ID}_i)$ for $i \in [\ell]$ $(\text{ID}^*, \text{st_verify}) \leftarrow_{\S} \mathcal{Z}^{*(\mathcal{P}_{\text{ID}, \cdot})^\infty}(\text{verify}, \text{mpk}, \{\text{sk}_i\}_1^\ell, \text{st_find})$ $b \leftarrow_{\S} (\mathcal{Z}^{*(\mathcal{P}_{\neq \text{ID}^*, \cdot})^\infty, \mathcal{V}})((\text{impersonate}, \text{st_verify}), \text{ID}^*)$ Return 1 iff $b = 1 \wedge \text{ID}^* \notin \{\text{ID}_i\}_1^\ell$
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Fig. 1. Security experiments for identity-based identification.

2.2 Lattices

A lattice in \mathbb{R}^n is a discrete set $\Lambda = \{\sum_{i=1}^d x_i \mathbf{b}_i \mid x_i \in \mathbb{Z}\}$, where $\mathbf{b}_1, \dots, \mathbf{b}_d$ are linearly independent over \mathbb{R}^n . The matrix $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_d]$ is a basis of the lattice Λ and we write $\Lambda = \Lambda(\mathbf{B})$. The dimension of the lattice is d . The main computational problem in lattices is the (approximate) shortest vector problem (SVP^p), where an algorithm is given a description, a basis, of a lattice Λ and is supposed to find the shortest vector $\mathbf{v} \in \Lambda \setminus \{\mathbf{0}\}$ with respect to a certain ℓ_p norm (up to an approximation factor). More precisely, find a vector $\mathbf{v} \in \Lambda \setminus \{\mathbf{0}\}$, such that $\|\mathbf{v}\|_p \leq \gamma \|\mathbf{w}\|_p$ for all $\mathbf{w} \in \Lambda \setminus \{\mathbf{0}\}$ for an approximation factor $\gamma \geq 1$.

In this work, we are interested in a special family of lattices related to ideals in the ring $\mathbf{R} = \mathbb{Z}_q[X]/\langle \mathbf{g} \rangle$, where q is prime and $\mathbb{Z}_q = \{-(q-1)/2, \dots, (q-1)/2\}$. We focus on $\mathbf{g} = X^n + 1$ and $n =$ “power of two” for efficiency reasons but it may be replaced with any irreducible polynomial over \mathbb{Z} . Then, our scheme and the analysis become only slightly more involved. We identify $\mathbf{f} \in \mathbf{R}$ with its coefficient vector $\mathbf{f} = (f_0, \dots, f_{n-1}) \in \mathbb{Z}_q^n$. Furthermore, we denote elements of the \mathbf{R} -module \mathbf{R}^m with $\hat{\mathbf{a}} = (\mathbf{a}_0, \dots, \mathbf{a}_{m-1})$ or directly with $(a_0, \dots, a_{mn-1}) \in \mathbb{Z}_q^{mn}$. Consequently, we define $\|\mathbf{f}\|_\infty = \|(f_0, \dots, f_{n-1})\|_\infty$ for $\mathbf{f} \in \mathbb{Z}[X]$. A lattice corresponds to an ideal $I \subset \mathbf{R}$ if and only if every lattice vector is the coefficient vector of a polynomial in I . The SVP problem easily translates to ideal lattices, where we call it ideal-SVP^p (ISVP^p).

The average-case hardness assumption for our construction relies on the problem finding short vectors in the kernel of the family $\mathcal{H}(\mathbf{R}, m)$ of module homomorphisms $h_{\hat{\mathbf{a}} \in \mathbf{R}^m} : \mathbf{R}^m \rightarrow \mathbf{R}, \hat{\mathbf{x}} \mapsto h(\hat{\mathbf{a}}, \hat{\mathbf{x}}) = \hat{\mathbf{a}} \otimes \hat{\mathbf{x}} = \sum_{i=0}^{m-1} \mathbf{a}_i x_i$, when restricting the domain to $D' \subset \mathbf{R}$, i.e., restricting the coefficients in the input vector to $[-2d, 2d] \cap \mathbb{Z}$.¹ This problem can be stated as a short vector problem in the lattice $\Lambda_{\mathbf{R}}^\perp(\hat{\mathbf{a}}) := \{\mathbf{x} \in \mathbb{Z}^{mn} : \mathbf{A}\mathbf{x} \equiv \mathbf{0} \pmod{q}\}$, where \mathbf{A} is structured and represents the multiplication $\otimes \pmod{\mathbf{g}}$. Hence, ideal lattices of the form $\Lambda_{\mathbf{R}}^\perp(\hat{\mathbf{a}})$ are a special case of q -ary lattices $\Lambda_q^\perp(\mathbf{A}) := \{\mathbf{x} \in \mathbb{Z}^{mn} : \mathbf{A}\mathbf{x} \equiv \mathbf{0} \pmod{q}\}$, where \mathbf{A} is unstructured and chosen from $\mathbb{Z}_q^{n \times mn}$.

The main average-case problem is the following collision problem.

Definition 1 (Collision Problem [LM06]). *The problem $\text{Col}(\mathcal{H}(\mathbf{R}, m), D)$ asks to find a distinct pair $(\hat{\mathbf{x}}, \hat{\mathbf{x}}') \in D^m \times D^m$ such that $h(\hat{\mathbf{x}}) = h(\hat{\mathbf{x}}')$ for $h \leftarrow_{\S} \mathcal{H}(\mathbf{R}, m)$.*

¹ For better readability, we use both notations $h_{\hat{\mathbf{a}}}(\cdot)$ and $h(\hat{\mathbf{a}}, \cdot)$

Obviously, the function is linear over \mathbf{R}^m , i.e., $h(\mathbf{a}(\hat{x} + \hat{y})) = \mathbf{a}(h(\hat{x}) + h(\hat{y}))$ for all $\mathbf{a} \in \mathbf{R}$, $\hat{x}, \hat{y} \in \mathbf{R}^m$. In addition, solving $\text{Col}(\mathcal{H}(\mathbf{R}, m), D)$ implies being able to solve ISVP^∞ in *every* lattice that corresponds to an ideal in \mathbf{R} .

Theorem 1 (Worst-case to Average-case Reduction, [LM06, Theorem 2]). *Let $D = \{f \in \mathbf{R} : \|f\|_\infty \leq d\}$, $m > \log(q)/\log(2d)$, and $q \geq 4dmn\sqrt{n}\log(n)$. An adversary \mathcal{C} that solves the $\text{Col}(h, D)$ problem, i.e., finds distinct preimages $\hat{x}, \hat{y} \in D^m$ such that $h(\hat{x}) = h(\hat{y})$, can be used to solve ISVP^∞ with approximation factors $\gamma \geq 16dmn\log^2(n)$ in the worst case.*

2.3 Chameleon Hash Functions

Krawczyk and Rabin [KR00] proposed Chameleon hashes to be collision-resistant hash functions with a trapdoor and the following properties. 1) The function $C : D \times E \rightarrow R$ is chosen from a family \mathcal{C} of Chameleon hashes along with a secret trapdoor C^{-1} . 2) The output distribution is indistinguishable from uniform. 3) In order to sample from the distribution $(d, e, C(d, e)) \in D \times E \times R$, we can do one of two things. Either we run C on the given document d and a randomness $e \sim \Delta(E)$ from an efficiently samplable distribution Δ over E , or we apply an inversion algorithm $e \leftarrow C^{-1}(d, r)$ on a given image $r \in R$ and a target document $d \in D$. Thus, we obtain a randomness e such that $C(d, e) = r$. The resulting distributions are indistinguishable. We will require statistical indistinguishability to facilitate a simpler proof in Section 3. Note that whenever we need the Chameleon hash to map to a certain range $\neq R$, we can compose it with an arbitrary collision resistant hash function. As for their realization, Krawczyk and Rabin claim in [KR98] that Chameleon hash functions exist if there are claw-free trapdoor permutations. Interestingly, they can be easily implemented with the lattice-based trapdoor function in [GPV08] as observed in [CHKP10].

3 From stat-id-imp-ca to adapt-id-imp-ca

To simplify the construction of (hierarchical) ID-based identification schemes, we propose a generic, black-box transformation from static-identity security to adaptive-identity security. The transformation is reminiscent of a generic transformation from static message secure digital signature schemes to adaptively chosen message secure schemes as both involve Chameleon hash functions.

In principle, our transform works for all *authentication-type* ID-based cryptography, e.g., ID-based identification or signatures. For encryption this does not work because there is no message-flow from the secret-key holder to the public-key holder. In other words, the encrypter cannot derive the recipients identity as it does not know the randomness for the Chameleon hash.

Suppose we have a scheme $\text{IBI}^{\text{stat}} = (\text{Kg}, \text{Extract}, \text{Protocol})$ that is secure against static identity attacks, we show how to construct a scheme $\text{IBI}^{\text{adapt}} = (\text{Kg}, \text{Extract}, \text{Protocol})$ that is secure against adaptive identity attacks if there is

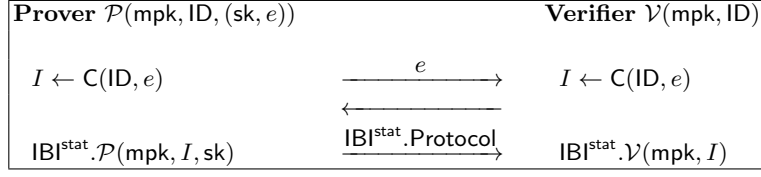


Fig. 2. Identity-based identification protocol for $\text{IBI}^{\text{adapt}}$.

a family of Chameleon hash functions \mathcal{C} . Notice that the inversion algorithm \mathcal{C}^{-1} is not used in the actual scheme, it is merely necessary to simulate the extraction oracle in the security proof.

Master-key Generation. $\text{Kg}(1^n)$ runs $(\text{msk}', \text{mpk}') \leftarrow \text{IBI}^{\text{stat}}.\text{Kg}(1^n)$ and select a Chameleon hash function $(\mathcal{C}, \mathcal{C}^{-1}) \leftarrow_{\mathcal{S}} \mathcal{C}(1^n)$. It returns $(\text{msk}, \text{mpk}) \leftarrow (\text{msk}', (\text{mpk}', \mathcal{C}))$.

Key Extraction. $\text{Extract}(\text{msk}, \text{ID})$. The algorithm selects $e \sim \Delta(E)$ and computes $I \leftarrow \mathcal{C}(\text{ID}, e)$. Then, it computes the secret key for I by calling $\text{sk} \leftarrow \text{IBI}.\text{Extract}(\text{msk}, I)$ and returns the pair (e, sk) .

Identification Protocol. Whenever a prover \mathcal{P} wants to prove its identity ID to a verifier \mathcal{V} , both parties act as per the protocol in Figure 2.

We show a reduction that proves adaptive identity security of $\text{IBI}^{\text{adapt}}$ under two assumptions. First, the underlying IBI^{stat} needs to be secure under static identity attacks. Second, the employed Chameleon hash function needs to be collision resistant. Notice that the reduction is property preserving with regard to the identification scheme, i.e., security under passive, active, and concurrent attacks is preserved under the reduction.

Theorem 2 (Adaptive Identity Security). *Suppose Chameleon hash functions exist. $\text{IBI}^{\text{adapt}}$ is secure under adaptive identity queries in the imp- $\{pa, aa, ca\}$ sense if IBI^{stat} is secure under static identity attacks in the same sense.*

Proof. First of all notice that Chameleon hash functions ensure that there is no efficient adversary that can reuse a given secret key sk for a particular identity ID to impersonate a different identity ID^* . Such an adversary refutes the collision resistance of the family \mathcal{C} of Chameleon hash functions. The reduction is straightforward. Therefore, we focus on the simulation against an impersonator \mathcal{I}^* that does not exploit any weakness in the Chameleon hash function. Suppose that the adversary makes at most Q queries to the extraction oracle.

Setup. On input mpk , the simulator chooses a Chameleon hash function and its trapdoor $(\mathcal{C}, \mathcal{C}^{-1}) \leftarrow_{\mathcal{S}} \mathcal{C}(1^n)$. It prepares a set of random identity strings $I_1, \dots, I_Q \leftarrow_{\mathcal{S}} \mathcal{R}$. Afterwards, the simulator calls its external extraction oracle $\text{IBI}^{\text{stat}}.\text{Extract}$ to obtain the corresponding secret keys $\text{sk}_1, \dots, \text{sk}_Q$ and sets up a counter $i \leftarrow 0$. It runs \mathcal{I}^* on input $(\text{mpk}, \mathcal{C})$.

Extraction Queries. Whenever \mathcal{I}^* queries an identity ID to its extraction oracle, the internal counter i is incremented and the reduction calls $e \leftarrow C^{-1}(\text{ID}, I_i)$. It returns (e, sk_i) .

Prover Queries. The simulator runs the protocol in Figure 2, by using its external prover oracle.

Impersonation Attempt. At some point, \mathcal{I}^* outputs a challenge identity ID^* , which has not been queried to the extraction oracle before. After that, the extraction oracle answers \perp when queried with ID^* . When the adversary instantiates a verifier to prove its identity ID^* with randomness e^* , the simulation forwards all messages to and from its external verifier oracle for $I^* = C(\text{ID}^*, e^*)$.

The environment of \mathcal{I}^* is perfectly simulated if the input-output relation of C can be sampled perfectly. The extraction oracle in the simulation was never called with ID^* , so the simulation never called the external oracle with identity $I^* = C(\text{ID}^*, e^*)$ (but with negligible probability). If \mathcal{I}^* is successful in the impersonation attempt, so is the simulator in the experiment $\text{Exp}_{\mathcal{I}^*, |\mathcal{B}|}^{\text{adapt-id-imp-ca}}$. \square

This transformation can be adapted to work in the hierarchical setting. There, an identity is prefixed with superordinate identities. In the transformation, one simply splits the entire string into sub-identities and computes a Chameleon hash for each of them. The reduction is somewhat looser than in Theorem 2 as the simulator has to prepare identities on all ℓ levels in the hierarchy. In consequence, the reduction only works when $\ell = \tilde{\mathcal{O}}(1)$.

4 A Construction Without Random Oracles

In this section, we show how to instantiate our new static-identity model from lattices. Our construction builds upon Lyubashevsky’s identification scheme. By using the Bonsai-tree technique [CHKP10] in the setting of ideal lattices, we show how to realize secret key extraction in the standard model.

The required lattice-based tools are introduced in Section 4.1 and our main construction is in Section 4.2. An additional discussion of leakage-resilience will appear in the full version. For each aspect, we prove a main theorem. Supporting lemmas are stated before the theorems and proven in the full version.

4.1 Convoluteds Bonsai Trees

For our main construction, we require a certain toolbox for lattices that goes by the name of “Bonsai trees”. In [CHKP10], such a toolbox is constructed from q -ary lattices to implement lattice-based signatures in the standard model as well as identity-based encryption. The authors also point out that Bonsai trees from more efficient ideal lattices seem possible. We confirm this observation by making it explicit, based on a family of trapdoor functions in ideal lattices. Notice that our main construction can be instantiated from q -ary lattices as well and that ideal lattices are merely necessary to achieve quasi-linear efficiency. In addition, making the notion of *ideal Bonsai trees* explicit may be of independent interest.

Preimage-samplable Trapdoor Functions. Gentry et al. [GPV08] introduce a family of preimage samplable functions $\text{GPV} = (\text{TrapGen}, \text{Eval}, \text{SamplePre})$ on lattices, which was later on adapted to ideal lattices by Stehlé et al. [SSTX09]. Its parameters $q, m, \tilde{L}, s = \omega(\sqrt{\log(n)})\tilde{L}$ are functions of n . We define the set $D_d := \{\mathbf{x} \in \mathbf{R} \setminus \{\mathbf{0}\} : \|\mathbf{x}\|_\infty \leq d\}$ for $d > 0$.

The algorithm $\text{TrapGen}(1^n)$ outputs a public description $\hat{\mathbf{a}} \in \mathbf{R}^m$ for the lattice $\Lambda_{\mathbf{R}}^\perp(\hat{\mathbf{a}})$ together with a secret trapdoor $\mathbf{T} \in \mathbb{Z}^{mn \times mn}$, $\|\tilde{\mathbf{T}}\| = \max_{i=1, \dots, mn} \{\|\tilde{\mathbf{t}}_i\|_2\} \leq \tilde{L}$, where $\tilde{\mathbf{T}}$ is the Gram-Schmidt orthogonalization of \mathbf{T} . Evaluation of the trapdoor function $h_{\hat{\mathbf{a}}} : \mathbf{R}^m \rightarrow \mathbf{R}$ is performed by the *convolution* product $\text{Eval}(\hat{\mathbf{a}}, \hat{\mathbf{x}}) = \hat{\mathbf{a}} \circledast \hat{\mathbf{x}}$. The inversion algorithm $\text{SamplePre}(\mathbf{T}, s, \mathbf{Y})$ samples from the set of preimages $\{\hat{\mathbf{x}} \in D_d^m : h(\hat{\mathbf{a}}, \hat{\mathbf{x}}) = \mathbf{y}\}$ for any $d = s\omega(\sqrt{\log(mn)})$. By construction, the function compresses the input and therefore admits collisions, but they are hard to find unless finding short vectors in ideal lattices is easy.

The following proposition is our adaptation of [GPV08] for ideal lattices.

Proposition 1. *Given a basis \mathbf{T} for $\Lambda_{\mathbf{R}}^\perp(\hat{\mathbf{a}})$ with $\|\tilde{\mathbf{T}}\| \leq \tilde{T}$ and a Gaussian parameter $s = \omega(\sqrt{\log(n)})\tilde{L}$, there is a polynomial-time algorithm SamplePre that, for any $\mathbf{Y} \in \mathbf{R}$, outputs $\hat{\mathbf{x}} \in \mathbf{R}^m$ with $\text{Eval}(\hat{\mathbf{a}}, \hat{\mathbf{x}}) = \mathbf{Y}$ and $\hat{\mathbf{x}} \in D_d$ for $d = s\omega(\sqrt{\log(mn)})$ with overwhelming probability. Furthermore, $\hat{\mathbf{x}}$ has a conditional min-entropy of $\omega(\log(n))$, conditioned on $h(\hat{\mathbf{a}}, \hat{\mathbf{x}}) = \mathbf{Y}$.*

Bonsai Trees from Ideal Lattices. We explicitly describe the functionalities of Bonsai trees in the language of ideal lattices. A central ingredient is the work of Stehlé et al. [SSTX09] because they show how to generate an ideal lattice together with a basis of short vectors of that lattice.

The notion of Bonsai trees on lattices is an analogy to arboriculture. An arborist always starts with a certain amount of *undirected*, i.e., random, natural growth that he cannot control. Then, he applies his tools and starts cultivating individual branches to achieve the desired looks via *directed* growth. The arborist is successful if the resulting tree still looks sufficiently natural to the observer. Once cultivated, a branch can easily be *extended* to form more directed growth without too much additional care. Instead of extending directed growth, the arborist can also generate *randomized* offsprings, which can be given to another arborist that can easily cultivate them by *extending* growth. The offsprings hide the first arborist's work and the employed techniques. We formalize these concepts in the context of ideal lattices. A (binary) bonsai tree is generated out of a root $\hat{\mathbf{a}}^*$ and branches $\hat{\mathbf{b}}_i^{(b)} \in \mathbf{R}^{m_i}$, $b \in \{0, 1\}$, $i \leq k \leq \text{poly}(n)$, that are statistically close to uniform. The entire tree is the set $\{\hat{\mathbf{a}}^* \|\hat{\mathbf{b}}_1^{(x_1)}\| \dots \|\hat{\mathbf{b}}_k^{(x_k)}\| : \mathbf{x} \in \{0, 1\}^{\leq k}\}$. The core of the Bonsai-tree technique is that we can append two vectors of polynomials $\hat{\mathbf{a}} \in \mathbf{R}^{m_1}$ and $\hat{\mathbf{b}} \in \mathbf{R}^{m_2}$ to form $\hat{\mathbf{c}} = \hat{\mathbf{a}} \|\hat{\mathbf{b}} \in \mathbf{R}^{m_1+m_2}$. Now, knowing a solution $\hat{\mathbf{x}} \in \mathbf{R}^{m_1}$ to the equation $\hat{\mathbf{a}} \otimes \hat{\mathbf{x}} \equiv \mathbf{0} \in \mathbf{R}$, we immediately obtain a solution $\hat{\mathbf{y}} \in \mathbf{R}^{m_1+m_2}$ to the equation $\hat{\mathbf{c}} \otimes \hat{\mathbf{y}} \equiv \mathbf{0}$ by setting $\hat{\mathbf{y}} = \hat{\mathbf{x}} \|\hat{\mathbf{0}} \in \mathbf{R}^{m_1+m_2}$ with $\|\hat{\mathbf{x}}\| = \|\hat{\mathbf{y}}\|$ for any norm. To see this, we directly apply the definition of \circledast and obtain $\hat{\mathbf{c}} \circledast \hat{\mathbf{y}} = \hat{\mathbf{a}} \circledast \hat{\mathbf{x}} + \hat{\mathbf{b}} \circledast \hat{\mathbf{0}} = \mathbf{0}$.

Proposition 2 (Directed Growth). *Let $n, \sigma, r \in \mathbb{N}_{>0}$, $q = q(n) \geq 3$ be a prime, $f \in \mathbb{Z}[X]$ be monic and irreducible over \mathbb{Z} . Let $\mathbf{R} = \mathbb{Z}_q[X]/\langle f \rangle$. There is a polynomial time algorithm $\text{ExtLattice}(\hat{\mathbf{a}}, m)$ that, given a uniformly random $\hat{\mathbf{a}} \in \mathbf{R}^{m_1}$, $m = m_1 + m_2 \geq (\lceil \log(q) \rceil + 1)(\sigma + r)$, $m_1 \geq \sigma$, generates $\hat{\mathbf{b}} \in \mathbf{R}^{m_2}$ with $m_2 = m - m_1$ together with a basis $\mathbf{S} = [\hat{\mathbf{s}}_1, \dots, \hat{\mathbf{s}}_m] \in \mathbf{R}^{m \times m}$ such that $(\hat{\mathbf{a}} \parallel \hat{\mathbf{b}}) \otimes \hat{\mathbf{s}}_i \equiv \mathbf{0}$ for $i \in [m]$.*

Let $f = \prod_{i \leq t} f_i$ be the factorization of f over \mathbb{Z}_q . The algorithm succeeds with probability $\geq 1 - p(n)$, where $p(n) = 1 - \prod_{i \leq t} (1 - q^{-\deg(f_i)\sigma})$. When it does, we have 1. $\Delta(\hat{\mathbf{a}} \parallel \hat{\mathbf{b}}, \text{unif}(\mathbf{R}^m)) \leq p + \frac{m}{2} \sqrt{\prod_{i \leq t} \left(1 + \frac{q}{3^r}^{\deg(f_i)}\right) - 1}$; 2. $\|\mathbf{S}\| \leq L = \sqrt{2} \sqrt{n(9r + \sigma)}$; 3. $\|\tilde{\mathbf{S}}\| \leq \tilde{L} \leq L$.

The proposition follows [SSTX09, Theorem 3.1].

Given $f = X^n + 1$ and a prime $q \equiv 3 \pmod{8}$, we know that f splits into $f_1 f_2$ for $f_i = X^{n/2} + z_i X^{n/4} - 1$, $z_i \in \mathbb{Z}_q$. This seems to be the best choice for q as the success probability and the uniformity of the output in Proposition 2 depend on f having only a small number of factors. But, even if f splits completely over \mathbb{Z}_q , it is still possible to find suitable (larger) parameters r, σ . For our choice of q , we can set $\sigma = 1$ and $r = \lceil \log_3(q) + 1 \rceil$ and repeat the process when it fails.

The interpretation in terms of arboriculture is generating “directed growth” out of “undirected growth” because one starts with some random growth $\hat{\mathbf{a}}$ and cultivates a branch $\hat{\mathbf{a}} \parallel \hat{\mathbf{b}}$ along with a trapdoor \mathbf{S} , which is the arborist’s journal or a trace of his work. However, the observer cannot distinguish undirected growth from directed growth.

Proposition 3 (Extending Control). *There is a polynomial time algorithm $\text{ExtBasis}(\mathbf{T}, \hat{\mathbf{c}} = \hat{\mathbf{a}} \parallel \hat{\mathbf{b}})$ that takes a basis \mathbf{S} of $\Lambda_{\mathbf{R}}^\perp(\hat{\mathbf{a}})$ and an extension $\hat{\mathbf{c}}$ with $\mathbf{R}^{m_1} \ni \hat{\mathbf{a}} \sqsubset \hat{\mathbf{c}} \in \mathbf{R}^{m_1+m_2}$ as input. If $\hat{\mathbf{a}}$ generates \mathbf{R} , the algorithm outputs a basis \mathbf{T}' for $\Lambda_{\mathbf{R}}^\perp(\hat{\mathbf{c}})$ with $\|\tilde{\mathbf{T}}'\| = \|\tilde{\mathbf{T}}\|$.*

The proposition is an adaptation of the respective proposition for q -ary lattices.

The resulting trapdoor is $\begin{pmatrix} \mathbf{S} & \mathbf{V} \\ \mathbf{0} & \mathbf{I}_{m_2} \end{pmatrix}$, where the columns $\hat{\mathbf{v}}_i$ of $\mathbf{V} \in \mathbf{R}^{m_2 \times m_2}$ are arbitrary (not necessarily short) solutions of the equations $\hat{\mathbf{a}} \otimes \hat{\mathbf{v}}_i \equiv -\mathbf{b}_i$.

Whenever trapdoor delegation is required, one cannot simply use extending control and hand over the resulting basis as it leaks information about the original trapdoor. Here, we can use tree propagation to obtain a *randomized* offspring with a new, random trapdoor.

Proposition 4 (Randomizing Control). *On input a basis \mathbf{T} of the lattice $\Lambda_{\mathbf{R}}^\perp(\hat{\mathbf{a}})$ of dimension m and a Gaussian parameter $s \geq \|\tilde{\mathbf{T}}\| \omega(\sqrt{\log(n)})$, the polynomial time algorithm $\text{RandBasis}(\mathbf{T}, s)$ outputs a basis \mathbf{T}' of $\Lambda_{\mathbf{R}}^\perp(\hat{\mathbf{a}})$ with $\|\tilde{\mathbf{T}}'\| \leq s\sqrt{m}$. The basis is independent of \mathbf{T} in the sense that for any two bases $\mathbf{T}_0, \mathbf{T}_1$ of $\Lambda_{\mathbf{R}}^\perp(\hat{\mathbf{a}})$ and $s \geq \max\{\|\tilde{\mathbf{T}}_0\|, \|\tilde{\mathbf{T}}_1\|\} \omega(\sqrt{\log(n)})$, $\text{RandBasis}(\mathbf{T}_0, s)$ is within negligible statistical distance of $\text{RandBasis}(\mathbf{T}_1, s)$.*

The proposition is a direct adaptation of the randomizing control algorithm in [CHKP10]. A more efficient alternative is in [Pei10].

For that concept to be used later on, we require a method of transforming a full-rank set of vectors in a lattice into a basis.

Proposition 5 (Full-rank Set to Basis [MG02, Lemma 7.1]). *Let $\Lambda = \Lambda(\mathbf{B})$ be a lattice generated by the basis $\mathbf{B} \in \mathbb{Z}^{m \times m}$. There is a polynomial time algorithm $\text{ToBasis}(\mathbf{S}, \mathbf{B})$ that takes as input a full-rank set of lattice vectors \mathbf{S} with $\mathbf{S} \subset \Lambda(\mathbf{B})$ and a basis \mathbf{B} . It outputs a basis \mathbf{T} of Λ with 1. $\|\mathbf{T}\| \leq \sqrt{mn}/2 \|\mathbf{S}\|$; 2. $\|\tilde{\mathbf{T}}\| \leq \|\tilde{\mathbf{S}}\|$.*

The idea is to use the oblivious sampler for lattices [GPV08, Pei10] to sample mn linearly independent vectors using the set \mathbf{T} as input. The result is a full-rank set of lattice vectors \mathbf{S} that does not reveal any information about \mathbf{T} . The final step entails calling $\text{ToBasis}(\mathbf{S}, \text{HNF}(\mathbf{T}))$ to obtain a basis. HNF is the unique Hermite normal form of \mathbf{T} , which is necessary to make the input to ToBasis completely independent of \mathbf{T} .

4.2 Our Construction

We construct a lattice-based identity-based identification scheme. It is secure in the standard model under a worst-case assumption in ideal lattices and its time and space complexity is quasi-optimal, i.e., $\tilde{\mathcal{O}}(n)$, in the online phase. The road map for this section is as follows: We describe the 3-move identification scheme IBI , including an informal description of the protocol. Then, we prove completeness and soundness in the static-identity attack model. Full, adaptive-identity security is established by the generic construction in Section 3. Proving completeness is non-trivial as we need to address an inevitable completeness defect. In the course of the discussion, we show that it neither harms security nor efficiency. In particular, the protocol remains statistically witness-indistinguishable and sound unless the collision problem $\text{Col}(\mathcal{H}(\mathbf{R}, m), D)$ is easy. Thus, security can be based on the worst-case hardness of the ISVP.

Observe that the scheme requires lots of parameters that need to be carefully worked out. Their definition in Table 1 will be justified later in the analysis.

Informal Description. We give a detailed, slightly informal description of the protocol Steps 1-4 in Figure 3. For each step, we need a set of carefully chosen parameters from Table 2 to achieve completeness and security.

Basically, the protocol follows the structure of the 3-move identification scheme in [Lyu08a, Lyu08b], which provides a witness-indistinguishable proof of knowledge. The prover proves knowledge of $\hat{\mathbf{s}} \in D_{\mathbf{s}}^m$ such that $h(\hat{\mathbf{a}}, \hat{\mathbf{s}}) = \mathbf{S}$ with $(\hat{\mathbf{a}}, \mathbf{S})$ being the public key.

In the *first step*, the prover \mathcal{P} selects the randomness $\hat{\mathbf{y}} \leftarrow_{\mathcal{S}} D_{\mathbf{y}}^m$ for this protocol run, where m depends on the size of the identity space. Then, \mathcal{P} commits to $\hat{\mathbf{y}}$ by sending $Y = h(\hat{\mathbf{a}}_{\text{ID}}, \hat{\mathbf{y}})$ to the verifier \mathcal{V} . The key $\hat{\mathbf{a}}_{\text{ID}}$ to h is unique for each identity $\text{ID} \in \{0, 1\}^\lambda$ and it can be computed from the master public key $(\hat{\mathbf{a}}^*, \langle \hat{\mathbf{b}} \rangle, \mathbf{S})$.

Parameter	Value	Asymptotic bound
\mathbf{R}	$\mathbb{Z}_q[X]/\langle X^n + 1 \rangle$, q prime	-
n	power of 2	-
m_1, σ	1	$\mathcal{O}(1)$
r	$\lceil \log_3(q) + 1 \rceil$	$\tilde{\mathcal{O}}(1)$
m_2	$\lceil \log(q) \rceil (\sigma + r) + r$	$\tilde{\mathcal{O}}(1)$
m	$m_1 + (\lambda + 1)m_2$	$\tilde{\mathcal{O}}(\lambda)$
D_s	$\{\mathbf{f} \in \mathbf{R} : \ \mathbf{f}\ _\infty \leq \tilde{L}\omega(\sqrt{\log(m)}) =: d_s\}$	$\tilde{\mathcal{O}}(\sqrt{n})$
D_c	$\{\mathbf{f} \in \mathbf{R} : \ \mathbf{f}\ _\infty \leq 1 =: d_c\}$	$\mathcal{O}(1)$
ϕ	positive integer constant ≥ 1	$\mathcal{O}(1)$
D_y	$\{\mathbf{f} \in \mathbf{R} : \ \mathbf{f}\ _\infty \leq \phi mn^2 d_s =: d_y\}$	$\tilde{\mathcal{O}}(n^2 \sqrt{n})$
G	$\{\mathbf{f} \in \mathbf{R} : \ \mathbf{f}\ _\infty \leq d_y - nd_s d_c =: d_G\}$	$\tilde{\mathcal{O}}(n^2 \sqrt{n})$
D	$\{\mathbf{f} \in \mathbf{R} : \ \mathbf{f}\ _\infty \leq d_G + nd_s d_c =: d_D\}$	$\tilde{\mathcal{O}}(n^2 \sqrt{n})$
q (prime)	$\geq 4mn\sqrt{n} \log(n) d_D$	$\tilde{\Theta}(n^4)$

The table defines all parameters for our scheme. The parameters $\sigma, r, \tilde{L}, m_1, m_2$ are as per Proposition 2. The constant ϕ governs the completeness error and λ is the bit length of the identities. The third column contains the asymptotic growth for the respective norm bound or parameter with respect to the main security parameter n .

Table 1. Parameters the identity-based identification scheme IBI.

In the *second step*, \mathcal{V} challenges \mathcal{P} with a challenge \mathbf{c} from the set D_c^m .

The *third step* entails the computation of the response $\hat{\mathbf{z}}$ and checking whether it falls into a safe set G^m of responses. If the coefficients of $\hat{\mathbf{z}}$ fall outside G , the protocol has to be restarted to ensure witness indistinguishability. Otherwise, $\hat{\mathbf{z}}$ is sent to the verifier.

Finally, the verifier performs the actual verification in the *fourth step*. It involves testing that the coefficients of $\hat{\mathbf{z}}$ are within the correct interval and that the prover has used a correct secret key $\hat{\mathbf{s}}_{\text{ID}}$, such that $h(\hat{\mathbf{a}}_{\text{ID}}, \hat{\mathbf{s}}) = \mathbf{S}$, when computing $\hat{\mathbf{z}}$. This last check is possible due to the linearity of h .

Concerning the abort ($\hat{\mathbf{z}} \leftarrow \perp$) in Step 3, we will show that it happens with probability at most $1 - e^{-1/\phi}$ if the set of D_y is set up properly.

Now, we explain how the secret key $\hat{\mathbf{s}}_{\text{ID}}$ is extracted for a given identity ID. Let $\hat{\mathbf{a}}^*$ be the root of a Bonsai tree and let $\langle \hat{\mathbf{b}} \rangle = \left\{ (\hat{\mathbf{b}}_i^{(0)}, \hat{\mathbf{b}}_i^{(1)}) \right\}_1^\lambda$ be the set of branches. Each identity $\text{ID} = \text{ID}_1 || \dots || \text{ID}_\lambda$ defines a unique path $\hat{\mathbf{a}}_{\text{ID}} := \hat{\mathbf{a}}^* || \hat{\mathbf{b}}_1^{(\text{ID}_1)} || \dots || \hat{\mathbf{b}}_\lambda^{(\text{ID}_\lambda)}$ in the tree. Given a trapdoor \mathbf{S} for the master lattice $A_{\mathbf{R}}^\perp(\hat{\mathbf{a}}^*)$, we can find short vectors in the coset $\{\hat{\mathbf{x}} : h(\hat{\mathbf{a}}_{\text{ID}}, \hat{\mathbf{x}}) \equiv \mathbf{S}\}$ of any super lattice $A_{\mathbf{R}}^\perp(\hat{\mathbf{a}}_{\text{ID}})$. The short elements of the coset correspond $\hat{\mathbf{s}}_{\text{ID}}$.

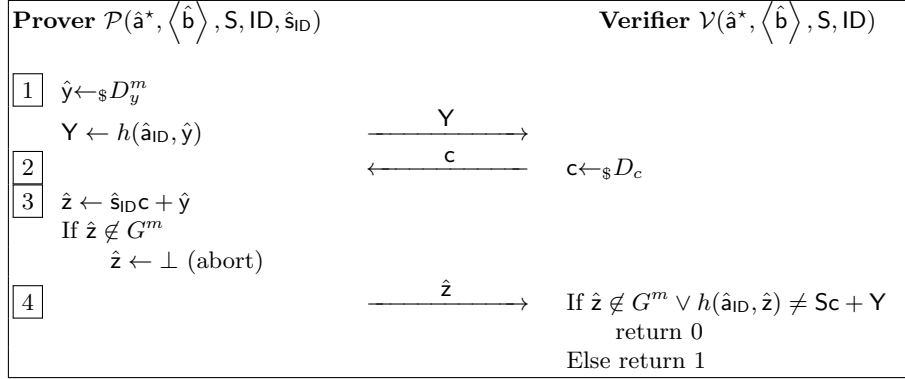


Fig. 3. Identity-based identification protocol.

The Bonsai trees allow the simulation of the extraction oracle for a polynomial number of identities in the security proof, while the attacked identity is likely to overlap only with branches of uncontrolled growth. There, the simulator will embed the challenge.

The simulation of the provers will be possible by using a single secret key $\hat{\mathbf{s}} \in R^{m_1+m_2}$, such that $h(\hat{\mathbf{a}}^*, \hat{\mathbf{s}}) = \mathbf{S}$, for all identities. The individual provers only need to pad $\hat{\mathbf{s}}$ with λm_2 zero polynomials to make the objects compatible. The witness indistinguishability hides this deviation. Thus, we demonstrate that sampling from a coset of $A_{\mathbf{R}}^\perp$ instead of from the lattice itself seems to be much more versatile. A related technique was used in [Rüc10] to achieve *strongly* unforgeable signatures from lattices in the standard model.

Master-key Generation. Let the parameters $q, f, \tilde{L}, m_1, m_2$ be as per Proposition 2 and let $d = s\omega(\sqrt{\log(nm_1 + (\lambda + 1)nm_2)})$ for a Gaussian parameter $s \geq \tilde{L}\omega(\sqrt{\log(n)})$. These parameters may be excluded from the public key as they are the same for all users. Use **ExtLattice** to generate a description $\hat{\mathbf{a}}^* \in \mathbf{R}^{m_1+m_2}$ of the master lattice $A_{\mathbf{R}}^\perp(\hat{\mathbf{a}}^*)$ together with a trapdoor \mathbf{S}^* such that $\|\tilde{\mathbf{S}}^*\| \leq \tilde{L}$. Furthermore, generate the sets $\langle \hat{\mathbf{b}} \rangle := \left\{ \left(\hat{\mathbf{b}}_i^{(0)}, \hat{\mathbf{b}}_i^{(1)} \right) \right\}_1^\lambda$ of random elements in \mathbf{R}^{m_2} . Then, the algorithm chooses $\mathbf{S} \leftarrow_{\mathcal{S}} \mathbf{R}$. Finally, output the secret key \mathbf{S}^* and the public key $(\hat{\mathbf{a}}^*, \langle \hat{\mathbf{b}} \rangle, \mathbf{S})$.

Key Extraction. On input $\mathbf{S}^*, \text{ID} \in \{0, 1\}^*$, we define the module element $\hat{\mathbf{a}}_{\text{ID}} := \hat{\mathbf{a}}^* \parallel \hat{\mathbf{b}}_1^{(\text{ID}_1)} \parallel \dots \parallel \hat{\mathbf{b}}_\lambda^{(\text{ID}_\lambda)} \in \mathbf{R}^{m_1+m_2+\lambda m_2}$. The algorithm samples $\hat{\mathbf{s}}_1, \dots, \hat{\mathbf{s}}_\lambda$ via **SampleDom**(s) and calls $\hat{\mathbf{s}}_0 \leftarrow \text{SamplePre}(\mathbf{S}^*, s, \mathbf{S} - \sum_{i=1}^\lambda \hat{\mathbf{b}}_i^{(\text{ID}_i)} \otimes \hat{\mathbf{s}}_i)$. The output is $\hat{\mathbf{s}}_{\text{ID}} \in D_s^m$ with overwhelming probability. In the event that $\hat{\mathbf{s}}_{\text{ID}} \notin D_s^m$, the algorithm re-samples $\hat{\mathbf{s}}_0$.

Identification Protocol. See Figure 3. Let $g(n) = \omega(\log(n))$. Upon an abort, the protocol is repeated, at most $g(n)$ times.

Notice that our scheme can be also be adapted to support a hierarchy of identities, each acting as the key extraction authority for its subordinates. Thus,

each user receives a secret key, a trapdoor for a super lattice, that can be used to generate the secret key for the identification scheme. This adaptation involves adding more layers to the Bonsai tree and applying `RandBasis` during basis delegation to prevent leaking information about the master trapdoor.

Completeness of `IBI` is a non-trivial issue due to the eventual restarts and the many parameters involved. The next lemma ensures that the number of restarts is small, effectively constant.

Lemma 1. *Let $k = \Omega(n)$, $\mathbf{a}, \mathbf{b} \in \mathbb{Z}^k$ with arbitrary $\mathbf{a} \in \{\mathbf{v} \in \mathbb{Z}^k : \|\mathbf{v}\|_\infty \leq A\}$ and random $\mathbf{b} \leftarrow_{\mathcal{S}} \{\mathbf{v} \in \mathbb{Z}^k : \|\mathbf{v}\|_\infty \leq B\}$. Given $B \geq \phi k A$ for $\phi \in \mathbb{N}_{>0}$, we have $\text{Prob}[\|\mathbf{a} - \mathbf{b}\|_\infty \leq B - A] > \frac{1}{e^{1/\phi}} - o(1)$.*

The multiplication of two polynomials modulo $X^n + 1$ plays a major role in the analysis. Therefore, we need the following lemma, which is a special case of [Lyu08b, Lemma 2.8].

Lemma 2. *For any two polynomials $\mathbf{a}, \mathbf{b} \in \mathbf{R}$, we have $\|\mathbf{a}\mathbf{b} \bmod (X^n + 1)\|_\infty \leq n \|\mathbf{a}\|_\infty \|\mathbf{b}\|_\infty$.*

Theorem 3 (Completeness). *The scheme `IBI` is complete.*

Proof. For all honestly generated master-key pairs $(\mathbf{S}^*, (\hat{\mathbf{a}}^*, \langle \hat{\mathbf{b}} \rangle), \mathcal{S})$, and all identities $\text{ID} \in \{0, 1\}^\lambda$, the key extraction algorithm outputs a secret key $\hat{\mathbf{s}}_{\text{ID}} = \hat{\mathbf{s}}_0 \parallel \dots \parallel \hat{\mathbf{s}}_\lambda \in D_s^m$ with $h(\hat{\mathbf{a}}_{\text{ID}}, \hat{\mathbf{s}}_{\text{ID}}) \equiv h(\hat{\mathbf{a}}^*, \hat{\mathbf{s}}_0) + \sum_{i=0}^\lambda h(\hat{\mathbf{b}}_i^{(\text{ID}_i)}, \hat{\mathbf{s}}_i) \equiv \mathcal{S} - \sum_{i=0}^\lambda h(\hat{\mathbf{b}}_i^{(\text{ID}_i)}, \hat{\mathbf{s}}_i) + \sum_{i=0}^\lambda h(\hat{\mathbf{b}}_i^{(\text{ID}_i)}, \hat{\mathbf{s}}_i) \equiv \mathcal{S}$ and $\|\hat{\mathbf{s}}_{\text{ID}}\|_\infty \leq d_s$ for $d_s = s\omega(\sqrt{\log(m)})$ and $s = \tilde{L}\omega(\sqrt{\log(n)})$ according to Proposition 1.

For all challenges $c \in D_c$ and all random coins $\hat{y} \in D_y^m$, we have $\|\hat{\mathbf{z}}\|_\infty = \|\hat{\mathbf{s}}_{\text{ID}}c + \hat{y}\|_\infty \leq d_y - n \|\hat{\mathbf{s}}_{\text{ID}}\|_\infty \|c\|_\infty = d_y - n = d_G$ with probability $\geq e^{-1/\phi} - o(1)$ because of Lemma 2 and Lemma 1 ($k = mn, A = nd_s d_c, B = d_y$). Hence, the verifier accepts because $h(\hat{\mathbf{a}}_{\text{ID}}, \hat{\mathbf{z}}) = h(\hat{\mathbf{a}}_{\text{ID}}, \hat{\mathbf{s}}_{\text{ID}})c + h(\hat{\mathbf{a}}_{\text{ID}}, \hat{y}) = \mathcal{S}c + \mathcal{Y}$.

Repeating the protocol $\omega(\log(n))$ times in parallel establishes completeness. In practice, a small and constant number $e^{1/\phi}$ of retries is sufficient. \square

Observe that in any case, all operations (including eventual restarts) in `IBI.Protocol` have $\tilde{\mathcal{O}}(n)$ complexity and that private keys, public keys, protocol messages, as well as the master public key have size $\tilde{\mathcal{O}}(n)$. The only exceptions from this optimality are the master secret key size, which is $\tilde{\mathcal{O}}(n^2)$ bits, and the key extraction algorithm `Extract`, which requires $\tilde{\mathcal{O}}(n^2)$ bit operations. Fortunately, the online phase merely requires quasi-linear operations and a quasi-linear bandwidth.

4.3 Security

Since the function family $\mathcal{H}(\mathbf{R}, m)$ compresses the domain D_s^m , it is easy to show that all secret keys collide with at least one other secret key.

Lemma 3. *Let m_1 and m_2 as per Proposition 2, $m \geq m_1 + m_2$, $h_{\hat{\mathbf{a}}} \in \mathcal{H}(\mathbf{R}, m)$, and $\mathcal{S} \in \mathbf{R}$. For every $\hat{\mathbf{s}} \in D_s^m$, there is a second $\hat{\mathbf{s}}' \in D_s^m \setminus \{\hat{\mathbf{s}}\}$ with $h(\hat{\mathbf{a}}, \hat{\mathbf{s}}) = h(\hat{\mathbf{a}}, \hat{\mathbf{s}}') = \mathcal{S}$ (with overwhelming probability).*

The next lemma establishes witness indistinguishability of the protocol. Witness indistinguishability ensures that the malicious verifier cannot distinguish whether the prover uses one of two possible secret keys $\hat{s}, \hat{s}' \in h_{\hat{a}}^{-1}(S) \cap D_s^m$.

Lemma 4. *Let m_1 and m_2 as per Proposition 2, $m \geq m_1 + m_2$, $h_{\hat{a}} \in \mathcal{H}(\mathbf{R}, m)$, and $S \in \mathbf{R}$. For every distinct $\hat{s}, \hat{s}' \in D_s^m$ with $h(\hat{a}, \hat{s}) = S = h(\hat{a}, \hat{s}')$, the resulting protocol views (Y, c, \hat{z}) and (Y', c, \hat{z}) are statistically indistinguishable.*

Using lemmas 3 and 4, we can exploit witness indistinguishability to simulate all provers with a single secret key \hat{s} and at the same time expect the adversary to use a different secret key \hat{s}' with non-negligible probability. Then, we use the Reset Lemma to extract this knowledge to break the collision problem.

Since the protocol is witness-indistinguishable, we can securely use parallel composition of multiple independent instances.

Theorem 4 (Soundness). *IBI is secure in the stat-id-imp-ca model if the collision problem $Col(\mathcal{H}(\mathbf{R}, m), D)$ is hard.*

Proof. The core idea of the proof is that we can simulate all provers with a single secret key $\hat{s} = \hat{s}^* || \hat{0} || \dots || \hat{0} \in \mathbf{R}^{m_1+m_2+\lambda m_2}$, where $\hat{a}^* \otimes \hat{s}^* \equiv S$, which can be prepared during the simulation.

Extraction queries can be prepared in the static identity attack model. We can prepare the set $\langle \hat{\mathbf{b}} \rangle$ so that we know a trapdoor for certain branches of the tree, while others are embedded with the external challenge from the collision problem. These “rigged” branches correspond to the target identity in the impersonation attempt of the adversary with non-negligible probability.

During this phase of the attack, we run the knowledge extractor of the underlying proof of knowledge to obtain $\hat{s}' \neq \hat{s}$. Hence, we solve the collision problem.

Setup. The reduction receives the input $\hat{\mathbf{a}} = \hat{\mathbf{a}}^* || \hat{u}_1^{(0)} || \hat{u}_1^{(1)} || \dots || \hat{u}_\lambda^{(0)} || \hat{u}_\lambda^{(1)} \in \mathbf{R}^{m_1+(2\lambda+1)m_2}$ together with the parameters $n, q, m = m_1 + m_2$, and the norm bound ν . It invokes $\mathcal{I}^*(\text{find})$ to obtain *distinct* $ID_1, \dots, ID_{q_E} \in \{0, 1\}^n$. Let $\langle \pi \rangle := \{\pi_i\}_1^p$ be the set of all strings $\pi \in \{0, 1\}^\lambda$ such that $\pi \not\sqsubseteq ID_j$ for $j \in \{1, \dots, q_E\}$ and $\pi_i \not\sqsubseteq \pi_j$ for all distinct pairs (π_i, π_j) in $\langle \pi \rangle$. The set $\langle \pi \rangle$ contains at most λq_E elements. Now, randomly select an element $\pi \leftarrow_{\$} \langle \pi \rangle$, which represents the challenge subtree. Let $|\pi| = l_\pi$. Setup of the public key:

- $\hat{\mathbf{b}}_i^{(\pi_i)} \leftarrow \hat{u}_i^{(0)}$ for $i = 1, \dots, l_\pi$;
- $\hat{\mathbf{b}}_i^{(b)} \leftarrow \hat{u}_i^{(b)}$ for $b \in \{0, 1\}$ and $i = l_\pi + 1, \dots, \lambda$;
- $\hat{\mathbf{b}}_i^{1-\pi_i}$ and \mathbf{S}_i via $\text{ExtLattice}(\hat{\mathbf{a}}^* || \hat{\mathbf{b}}_1^{(\pi_1)} || \dots || \hat{\mathbf{b}}_{i-1}^{(\pi_{i-1})}, m_2)$ for $i = 1, \dots, l_\pi$.

For the trapdoors, we have $\|\tilde{\mathbf{S}}_i\| \leq \tilde{L}$. Use SampleDom with $s = \omega(\sqrt{\log(n)})\tilde{L}$ to sample an element $\hat{\mathbf{s}} \in \mathbf{R}^{m_1+m_2}$ and compute $S \leftarrow h(\hat{\mathbf{a}}^*, \hat{\mathbf{s}})$. For each identity $I^{(i)} = ID_i$, let j be the smallest index with $I_j^{(i)} \neq \pi_j$. Since $\|\tilde{\mathbf{S}}_i\| \leq \tilde{L}$, we let $s = \omega(\sqrt{\log(n)})\tilde{L}$ and compute the secret key $\hat{\mathbf{s}}_i \leftarrow \text{SamplePre}(\text{ExtBasis}(\mathbf{S}_j, \hat{\mathbf{a}}_{I^{(i)}}), s, S)$. The public key comprises $\hat{\mathbf{A}}^*$, S , and $\langle \hat{\mathbf{b}} \rangle := \left\{ \left(\hat{\mathbf{b}}_i^{(0)}, \hat{\mathbf{b}}_i^{(1)} \right) \right\}_1^\lambda$ and the reduction returns the public key and the list of secret keys to \mathcal{I}^* .

Prover Queries. \mathcal{I}^* (verify) may challenge the reduction with any identity ID .

The simulator acts as per the protocol in Figure 3 but uses the same secret \hat{s} for all identities.

Impersonation Attempt. At some point, \mathcal{I}^* outputs a challenge identity ID^* , which has not been queried to the extraction oracle before. After that, the extraction oracle answer \perp when queried with ID^* . After the algorithm \mathcal{I}^* (impersonate) submits a commitment Y , it is challenged with a random $c_1 \leftarrow_{\mathcal{S}} D_c$, and outputs \hat{z}_1 . Then, the reduction rewinds \mathcal{I}^* to the end of Step 1 and challenges the adversary with a fresh $c_2 \leftarrow_{\mathcal{S}} D_c \setminus \{c_1\}$ to obtain the answer \hat{z}_2 . The reduction suitably rearranges and pads (with $\hat{0}$) the pair $(\hat{z}_1 - \hat{s}c_1, \hat{z}_1 - \hat{s}c_2)$ and outputs the result as its solution to the problem Col .

Analysis. First of all, observe that mpk in the simulation is statistically indistinguishable from mpk in the real scheme. Furthermore, note that the simulator can answer all extraction queries correctly because it knows a trapdoor for a prefix of all requested identities. As for the prover queries, we require that the protocol is witness indistinguishable w.r.t. the secret key (Lemma 4). Let us assume that the reset during \mathcal{I}^* 's impersonation attempt yields another valid response without aborting. Then, we certainly have $h(\hat{a}, \hat{z}_1 - \hat{s}c_1) = Y = h(\hat{a}, \hat{z}_2 - \hat{s}c_2)$ with $\max\{\|\hat{z}_1 - \hat{s}c_1\|_{\infty}, \|\hat{z}_2 - \hat{s}c_2\|_{\infty}\} \leq d_G + nd_s d_c = d_D$. What is left to show is that $\hat{z}_1 - \hat{s}c_1 \neq \hat{z}_2 - \hat{s}c_2$. Lemma 3 guarantees the existence of at least two distinct valid secret keys \hat{s} and \hat{s}' . Now, for one of them, we obtain a valid collision. Assuming the contrary, $\hat{z}_1 - \hat{s}c_1 = \hat{z}_2 - \hat{s}c_2$ and $\hat{z}_1 - \hat{s}'c_1 = \hat{z}_2 - \hat{s}'c_2$ yields $c_1(\hat{s}' - \hat{s}) = c_2(\hat{s}' - \hat{s})$ and therefore $(c_1 - c_2)(\hat{s}' - \hat{s}) = \hat{0}$. This only holds if $\hat{s}' = \hat{s}$ because $\max\{\|\hat{s}\|_{\infty}, \|\hat{s}'\|_{\infty}\} \leq q/2$ and $\mathbb{Z}[X]/\langle X^n + 1 \rangle$ is an integral domain.

Thus, with probability $\geq 1/2$, the simulator can use \mathcal{I}^* 's output to solve $Col(\mathcal{H}(\mathbf{R}, m), D)$. Concerning the success probability of the reset, assume that \mathcal{I}^* is successful with non-negligible probability $\epsilon(n)$. Then, \mathcal{I}^* is successful with non-negligible probability $\geq (\epsilon(n) - 1/|D_c|)^2$ by the Reset Lemma [BP02] in the second run. Then, we need to account for the inherent completeness defect, which makes the second run abort with probability $\leq (1 - e^{-1/\phi})$. All in all, the success probability of the simulator against the collision problems stays non-negligible if $\epsilon(n)$ is non-negligible. \square

5 Conclusions

Using a new, weaker security model for identity-based identification and a generic transformation to full security, we have shown how to construct an identity-based identification scheme from lattices that is secure against concurrent impersonation and adaptive-identity attacks in the standard model. Via a worst-case to average-case reduction, it is provably as hard to break as certain worst-case lattice problems in ideal lattices. Our scheme offers quasi-optimal performance and it is leakage-resilient in an almost optimal sense. Therefore, we expect our construction to withstand even subexponential-time and quantum computers attacks, as well as limited side-channel attacks against the secret key.

Acknowledgments

The author thanks the anonymous reviewers of SCN 2010 for their valuable, detailed, and encouraging comments.

References

- [Ajt96] Miklós Ajtai. Generating hard instances of lattice problems (extended abstract). In *STOC*, pages 99–108. ACM, 1996.
- [AKS01] Miklós Ajtai, Ravi Kumar, and D. Sivakumar. A sieve algorithm for the shortest lattice vector problem. In *STOC*, pages 601–610. ACM, 2001.
- [BBD08] Daniel J. Bernstein, Johannes A. Buchmann, and Erik Dahmen, editors. *Post-Quantum Cryptography*. Springer, 2008.
- [BNN09] Mihir Bellare, Chanathip Namprempre, and Gregory Neven. Security proofs for identity-based identification and signature schemes. *J. Cryptology*, 22(1):1–61, 2009.
- [Boy10] Xavier Boyen. Lattice mixing and vanishing trapdoors: A framework for fully secure short signatures and more. In Phong Q. Nguyen and David Pointcheval, editors, *Public Key Cryptography*, volume 6056 of *Lecture Notes in Computer Science*, pages 499–517. Springer, 2010.
- [BP02] Mihir Bellare and Adriana Palacio. Gq and schnorr identification schemes: Proofs of security against impersonation under active and concurrent attacks. In Moti Yung, editor, *CRYPTO*, volume 2442 of *LNCS*, pages 162–177. Springer, 2002.
- [CGGG09] Pierre-Louis Cayrel, Philippe Gaborit, David Galindo, and Marc Girault. Improved identity-based identification using correcting codes. *CoRR*, abs/0903.0069, 2009.
- [CHKP10] David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert. Bonsai trees, or how to delegate a lattice basis. In Gilbert [Gil10], pages 523–552.
- [Gil10] Henri Gilbert, editor. *Advances in Cryptology - EUROCRYPT 2010, 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, French Riviera, May 30 - June 3, 2010. Proceedings*, volume 6110 of *Lecture Notes in Computer Science*. Springer, 2010.
- [GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In Richard E. Ladner and Cynthia Dwork, editors, *STOC*, pages 197–206. ACM, 2008.
- [GS02] Craig Gentry and Alice Silverberg. Hierarchical id-based cryptography. In Yuliang Zheng, editor, *ASIACRYPT*, volume 2501 of *Lecture Notes in Computer Science*, pages 548–566. Springer, 2002.
- [KH05] Kaoru Kurosawa and Swee-Huay Heng. Identity-based identification without random oracles. In Osvaldo Gervasi, Marina L. Gavrilova, Vipin Kumar, Antonio Laganà, Heow Pueh Lee, Youngsong Mun, David Taniar, and Chih Jeng Kenneth Tan, editors, *ICCSA (2)*, volume 3481 of *Lecture Notes in Computer Science*, pages 603–613. Springer, 2005.
- [KR98] Hugo Krawczyk and Tal Rabin. Chameleon hashing and signatures. Cryptology ePrint Archive, Report 1998/010, 1998. <http://eprint.iacr.org/>.
- [KR00] Hugo Krawczyk and Tal Rabin. Chameleon signatures. In *NDSS*. The Internet Society, 2000.

- [KV09] Jonathan Katz and Vinod Vaikuntanathan. Signature schemes with bounded leakage resilience. In Matsui [Mat09], pages 703–720.
- [LM06] Vadim Lyubashevsky and Daniele Micciancio. Generalized compact knapsacks are collision resistant. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *ICALP (2)*, volume 4052 of *LNCS*, pages 144–155. Springer, 2006.
- [Lyu08a] Vadim Lyubashevsky. Lattice-based identification schemes secure under active attacks. In Ronald Cramer, editor, *Public Key Cryptography*, volume 4939 of *LNCS*, pages 162–179. Springer, 2008.
- [Lyu08b] Vadim Lyubashevsky. *Towards Practical Lattice-Based Cryptography*. PhD thesis, 2008.
- [Mat09] Mitsuru Matsui, editor. *Advances in Cryptology - ASIACRYPT 2009, 15th International Conference on the Theory and Application of Cryptology and Information Security, Tokyo, Japan, December 6-10, 2009. Proceedings*, volume 5912 of *Lecture Notes in Computer Science*. Springer, 2009.
- [MG02] Daniele Micciancio and Shafi Goldwasser. *Complexity of Lattice Problems: a cryptographic perspective*, volume 671 of *The Kluwer International Series in Engineering and Computer Science*. Kluwer Academic Publishers, Boston, Massachusetts, March 2002.
- [MV10] Daniele Micciancio and Panagiotis Voulgaris. A deterministic single exponential time algorithm for most lattice problems based on voronoi cell computations. In Michael Mitzenmacher and Leonard J. Schulman, editors, *STOC*, pages 351–358. ACM, 2010.
- [Pei10] Chris Peikert. An efficient and parallel gaussian sampler for lattices. Manuscript: <http://www.cc.gatech.edu/~cpeikert/>, 2010.
- [Rüc10] Markus Rückert. Strongly unforgeable signatures and hierarchical identity-based signatures from lattices without random oracles. In Nicolas Sendrier, editor, *PQCrypto*, volume 6061 of *Lecture Notes in Computer Science*, pages 182–200. Springer, 2010.
- [Sha84] Adi Shamir. Identity-based cryptosystems and signature schemes. In *CRYPTO*, pages 47–53, 1984.
- [SSTX09] Damien Stehlé, Ron Steinfeld, Keisuke Tanaka, and Keita Xagawa. Efficient public key encryption based on ideal lattices. In Matsui [Mat09], pages 617–635.