Separable Identity-Based Ring Signatures: Theoretical Foundations For Fighting Phishing Attacks

Ben Adida* Susan Hohenberger*,† Ronald L. Rivest*

February 28, 2005

Abstract

Email *phishing* attacks are one of today's most common and costly forms of digital identity theft, where an adversary tricks a user into revealing their personal information by impersonating an established company. Such attacks could be mitigated with digitally-signed emails, if these signatures did not: (1) destroy the traditional repudiability of email, and (2) require the unrealistic, widespread adoption of a Public-Key Infrastructure (PKI).

In order to overcome these obstacles, we introduce, define, and implement *separable (a.k.a. cross-domain) identity-based ring signatures* (SIBR, pronounced "cyber," signatures). The ring structure of these signatures provides repudiability. With identity-based public keys, a full PKI is no longer required. Separability allows ring constructions across different identity-based master key domains. Together, these properties make SIBR signatures a practical solution to the email spoofing problem.

Our construction yields a number of interesting components. First, we present several novel proofs of knowledge of bilinear map pre-images. We then present new identity-based identification (IBI) and signature (IBS) schemes based on these proofs. We note how our constructions share system parameters with the existing identity-based encryption schemes of Boneh-Franklin and Waters, thereby forming complete identity-based cryptosystems. We finally construct the first SIBR signature schemes by transforming our new signature schemes and certain other signature schemes.

Keywords: phishing, identity-based ring signatures, separable ring signatures, proofs of knowledge of bilinear pre-images.

1 Introduction

Email *phishing* is a form of digital identity theft where an adversary tricks an email recipient into revealing personal information, such as bank account credentials or credit card numbers, by impersonating an established company. Over the past year, phishing attacks were launched pretending to be AOL, eBay, VISA, and many others, with an estimated cost of identity theft to these companies and their consumers surpassing $10 billion dollars [27]. The Anti-Phishing Working Group reports that phishing attacks are increasing at an astounding rate of 28% per month [27]. Due to the huge impact of the phishing problem, many efforts to mitigate it are underway using techniques from machine learning [33, 32, 46], blacklists [47, 34], user interface design [36, 29, 21], and legal recourse.

In this work, we establish the theoretical foundations for *mitigating phishing attacks using cryptography*. Most email-based phishing attacks succeed in large part because they spoof the `from:` field of our email.

*Computer Science and Artifical Intelligence Laboratory; Massachusetts Institute of Technology; 32 Vassar Street; Cambridge, MA 02139, USA. Email: {ben,srhohen,rivest}@mit.edu.

†Supported by an NDSEG Fellowship.

Thus, one common suggestion, as put forward by the Anti-Phishing Working Group [3], is to authenticate all email using standard digital signatures like PGP or S/MIME. Unfortunately, this simple solution gives rise to two problems: (1) adoption is unlikely because it requires a widespread public-key infrastructure (PKI), and (2) traditional repudiability of email is destroyed. Our work seeks the advantages of digital signatures – email recipients are confident of an email's origin – without those practical roadblocks.

For this purpose, we introduce *separable (a.k.a. cross-domain) identity-based ring signatures* (SIBR, pronounced "cyber," signatures). In such a scheme, public keys are derived from character strings (usually email addresses) and a master public key, and a ring can be composed of public keys derived from different master public keys. One of our constructions even allows rings formed from public keys corresponding to different individual signature schemes altogether.

The identity-based and separability properties make deployment practical. Identity-based systems allow a user's public key to be easily computed from the text of his email address. While this improvement removes the PKI requirement, it is not practical enough: the assumption that all users derive their public keys from the same master – and thus that all users trust this single master – is unrealistic. Separability makes an identity-based system truly practical: users select a master of their choice, and cryptographic schemes operate across various masters. In the context of email, a user's master will simply be her email domain: Alice, with email address `alice@wonderland.com`, will derive her public key from `wonderland.com`. The distribution and certification of master public keys is far simpler and easier to coordinate than the management of user-level keys. We have prepared a separate paper [2] which discusses the distribution of these master keys via DNS records, as well as other deployment issues.

Ring signatures with strategically-formed signatory groups – as previously explored in the literature [10, 39] – provide *sender repudiability* based on *recipient forgeability*. More precisely, a user Alice can send an email to Bob with a ring signature including exactly Alice and Bob as signers. Bob can tell that Alice signed the email, since he knows he personally did not. However, all such signatures are forgeable by the recipient, Bob: he cannot convince a third party that Alice was the real signer.

Another avenue in obtaining this specific functionality is deniable signatures, as introduced by Naor [20] and recently extended by Susilo and Mu [44, 45]. This type of signature achieves similar repudiability by designated-recipient forgeability. Though this paper focuses on ring signatures to obtain sender-repudiability, deniable signatures may provide another workable approach.

1.1 Our Results

We provide several distinct results on our way to constructing SIBR signature schemes.

Bilinear Proofs of Knowledge. We present three novel, efficient honest-verifier zero-knowledge (HVZK) proofs of knowledge of various pre-images of bilinear maps. For example, given a mapping $e : G_1 \times G_1 \rightarrow G_2$, an element $g \in G_1$, and an element $X \in G_2$, prove knowledge of a value $\alpha \in G_1$ such that $e(\alpha, g) = X$. Our first protocol is actually a *generalization* of one implicitly presented by Hess [30]. Two of these protocols also work smoothly for mappings of the form $e : G_1 \times \tilde{G} \rightarrow G_2$, where G_1 and $\tilde{G}$ are distinct groups without an efficient mapping between them.

New IBI and IBS Schemes. Using these proofs of knowledge, we: (1) clarify the IBI and IBS schemes due to Hess [30], and (2) build new IBI and IBS schemes that share system parameters with the recent Waters [48] IBE scheme. This second result is non-trivial, because for any one public key in the Waters' IBE scheme there are many possible secret keys. Our IBI scheme is the first identification scheme based on bilinear maps proven secure in the standard model. Our IBS scheme is efficient and combined with the Waters IBE scheme creates a complete public key identity-based system.

SIBR Signature Constructions. We introduce, define, and offer two generic constructinos of *separable identity-based ring signatures*. We show how to build a SIBR signature scheme using the proof of partial knowledge techniques of Cramer, Dåmgard, and Schoenmakers [17] on any one of five existing IBS schemes including our new scheme (see Section 6). We then show how to build a more flexible SIBR signature scheme using the *standard* ring signature compiler due to Abe, Ohkubo, and Suzuki [1] on an even larger set of IBS schemes (i.e., seven schemes). We also show that these signatures are practical by extrapolating from comparable experimental data recently reported by Ateniese et al. [4].

1.2 Related Work

Our work builds on many areas of previous research. We contribute to the set of efficient proof of knowledge protocols, such as those of Schnorr [41] and others [16, 12, 11]. Our work also contributes to the set of over twelve identity-based signature schemes as started by Shamir [43] and recently summarized by Bellare et al. [5]. We also provide a new construction of the ring signature primitive as formalized by Rivest et al. [39], and its identity-based versions [49]. Our SIBR signature schemes also share some properties with *deniable signature schemes* as introduced by Naor [35] and recently extended [44, 45, 38] where a sender can authenticate a message m for a receiver in the public key setting in such a way that a third party is not convinced that an authentication of m took place. Deniable signatures are a special case of the more general *designated verifier proof* primitive introduced by Jakobsson et al. [31].

1.3 Roadmap

In Section 2, we begin with a formalization of the notion of a SIBR signature scheme and explain why it is particularly well-suited to the mitigation of phishing attacks. Then, we begin our construction of SIBR signatures. In Section 3, we discuss some technical preliminaries, including our notation and complexity assumptions. Next, in Section 4, we present three novel HVZK proofs of knowledge of bilinear maps pre-images. These proofs are used to build new IBI and IBS schemes in Section 5. These IBS schemes are then transformed into SIBR signature schemes in Section 6. We discuss an alternative solution in Section 7 before making some concluding remarks in Section 8.

2 Definitions and Motivation

We begin with an overview of the individual properties of SIBR signature schemes. Combined, these properties yield a single security definition for SIBR signatures. We conclude this section by pointing out how our new definition presents a particularly well-suited defense against email-based phishing attacks.

2.1 Ring Signatures

Ring signatures are a particular type of ad-hoc group signature, where a user Alice may create a signature σ on a message m with an arbitrarily selected signatory group $\mathcal{G}$, as long as $\mathcal{G}$ includes Alice (of course). Verification of σ does not yield any information as to the exact identity of the signer within $\mathcal{G}$. We adopt the formal definition of ring signatures given by Rivest et al. [39].

2.2 Identity-Based Signatures

Identity-based signatures are signature schemes where a user's public key PK is derived from a character string representing the user's identity, usually his email address, and a master public key MPK. The private key is then computed by a master authority in possession of the master secret MSK (corresponding to MPK), and delivered to the proper user after proper authentication, usually via a separate channel. The

concept was first introduced and defined by Shamir in 1984 [43]. We adopt the definition for identity-based signature (IBS) schemes secure against adaptive chosen-message attacks as summarized by Bellare et al. [5].

2.3 Separability

Separability was introduced by Camenisch and Michels [12] to quantify the amount of common system parameters necessary to create a group signature. The term is further refined across three subcategories:

- *Weakly Separable:* all parties must use the same signature scheme (e.g., El Gamal, Schnorr, 1024-bit RSA) and system parameters (e.g., g, p, q).
- *Medium Separable:* all parties must use the same signature scheme, but may possess independent key pairs.
- *Strongly Separable:* parties may use potentially different signature schemes, where the group signature adopts roughly the minimum security parameter 1^k of the schemes.

Since our anti-phishing application requires the realistic cooperation of many parties, we focus on medium and strongly separable schemes.

2.4 SIBR Signatures

A *(strongly) separable identity-based ring (SIBR) signature scheme* is defined as a set of identity-based signature schemes Π together with algorithms RSign and $\mathsf{RVerify}$ such that:

- $\mathsf{RSign}(gp, \mathcal{R}, sk, m)$ produces a ring signature σ, where gp is the set of global parameters corresponding to each IBS scheme in Π, $\mathcal{R}$ is a set of user public keys generated according to each user's respective scheme in Π, sk is a secret key corresponding to an element of $\mathcal{R}$, and m is an arbitrary bit string.
- $\mathsf{RVerify}(gp, \mathcal{R}, m, \sigma)$ returns 1 if σ is a valid ring signature for signatory group $\mathcal{R}$ on message m according to global parameters gp; and 0 otherwise.

We require adaptive chosen-message attack [5], and *unconditional signer-ambiguity*: even an infinitely powerful adversary given access to the secret keys of all users and an unbounded number of chosen-message signatures cannot guess the true signer of m with probability better than $1/|\mathcal{R}|$.

2.5 Why Use These Signatures For Anti-Phishing?

Email-based phishing attacks generally exploit the lack of any deployed email authentication mechanism: official-looking emails are spoofed, thus appearing to originate from official-looking addresses. The immediate cryptographic defense is some form of digital signature. Unlike existing email signature schemes, SIBR signatures are well adapted to the problem at hand: they preserve existing email usage habits and offer a flexible, low-overhead deployment mechanism.

Repudiability from Ring Signatures. Email is currently repudiable. The widespread use of digital signatures on email could harm this property and strip email users of their privacy: emails might become legally binding. The use of ring signatures, where the signatory ring includes the sender and recipient, provides the perfect level of repudiability: the recipient of an email might have forged it.

Deployment from Identity-Based Infrastructure. A ring signature scheme is not enough, as it requires a Public-Key Infrastructure for distributing user public keys. The applied cryptography literature repeatedly mentions the impracticality of such a requirement, both because public keys do not exist until users generate them, and because finding a user's public key requires a new infrastructure. The natural property we seek is thus an identity-based ring signature scheme.

More Deployment from Separable Schemes. A simple identity-based ring signature scheme is still not enough, though, as it assumes that all users will derive their public keys from the same master public key. In practice, this is almost as impractical as a PKI. We cannot assume that all users will live within the same trust domain, where a single entity computes private keys for all Internet users. With *separability*, signatory rings can span multiple trust domains, each with its own, independently selected, master keypair.

Efficiency from Signature Design. Our SIBR signature constructions are reasonably efficient. We provide some performance estimates in Section 6.4.

3 Preliminaries

We briefly review some technical preliminaries which will be used from this point on in the paper.

3.1 Bilinear Maps

Let Setup be an algorithm that, on input the security parameter 1^k, outputs $\gamma = (q, g, G_1, G_2, e)$, where e is a non-degenerate, efficiently computable bilinear map from $G_1 \times G_1$ to G_2, where both G_1 and G_2 are groups of prime order $q = \Theta(2^k)$. We assume that each group element has a unique binary representation. More formally, $e : G_1 \times G_1 \rightarrow G_2$ is a function that is:

- *Bilinear*: for all $g, h \in G_1$, for all $a, b \in \mathbb{Z}_q$, $e(g^a, h^b) = e(g, h)^{ab}$,
- *Non-degenerate*: if g is a generator of G_1, then $e(g, g)$ generates G_2, and
- *Efficient*: computing $e(g, h)$ is efficient for all $g, h \in G_1$.

By writing, $G_1 = \langle g \rangle$, we mean that g generates G_1. We recognize that, for some instantiations of the mappings, it is more efficient to let $e : G_1 \times \tilde{G} \rightarrow G_2$, where G_1 and $\tilde{G}$ are distinct groups of size q. Most of our constructions will work in this setting as well.

3.2 Notation

By writing $x \stackrel{R}{\leftarrow} S$, we denote that x is chosen uniformly at random from a set S. By writing $P\left(\mathcal{A}(x), \mathcal{B}(y)\right)$, we denote that P is a protocol between two parties $\mathcal{A}$ and $\mathcal{B}$, where $\mathcal{A}$ takes input x and $\mathcal{B}$ takes input y.

When discussing various proofs of knowledge of pre-images of bilinear maps, we will follow the notation introduced by Camenisch and Stadler [13] for proofs of knowledge of discrete logarithms. For example,

$$PK\left\{(\alpha, \beta, \gamma) : e(\alpha, \beta) = X \wedge e(\gamma, g) = Y\right\}$$

denotes a "*zero-knowledge proof of knowledge of values α, β, and γ such that $e(\alpha, \beta) = X$ and $e(\gamma, g) = Y$*" where g is an element of some group $\langle g \rangle = G_1$ and X, Y are elements of some group G_2 such that e is a bilinear mapping from $G_1 \times G_1$ to G_2. The convention is that Greek letters denote quantities of which the knowledge is being proven, while other parameters are public.

3.3 Complexity Assumptions

We use the following complexity assumptions based on bilinear maps generated according to $\mathsf{Setup}(1^k) \rightarrow params = (q, g, G_1, G_2, e)$. To our knowledge, the second assumption has not previously appeared in the literature.

Computational Diffie-Hellman (CDH) Assumption Given (h, h^a, h^b) for random $h \in G_1$ and $a, b \in \mathbb{Z}_q$, the probability that any PPT adversary outputs h^{ab} is negligible in k.

Bilinear One-Way Assumption Given $(params, Q)$ for random $Q \in G_2$, the probability that any PPT adversary outputs a pair $x, y \in G_1$ such that $e(x, y) = Q$ is negligible in k.

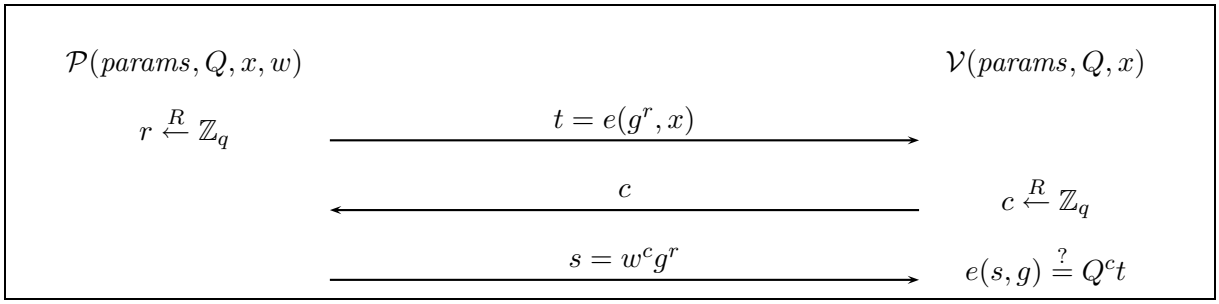

Figure 1: Description of the HVZK proof of knowledge protocol $\mathcal{T}_1$ for showing knowledge of the canonical pre-image of Q with respect to element x (i.e., $w \in G_1$ such that $Q = e(w, x)$). First, the prover sends $t = e(g^r, x)$ as a commitment to a random value $g^r \in G_1$. Next, the verifier issues a random challenge $c \in \mathbb{Z}_{\shortparallel}$. Finally, the prover responds with $s = w^c g^r$. The verifier accepts if and only if $e(s, x) = Q^c t$. Note that the prover need not know the value of r during this protocol; it is enough to know g^r, which can be selected at random from G_1.

4 Proofs of Knowledge of Bilinear Pre-Images

The first step towards our SIBR constructions is the development of new proofs of knowledge involving bilinear maps. We will use these proofs to build new IBS schemes in Section 5; we will then use these IBS schemes to build SIBR schemes in Section 6.

In this section, we work in the common parameters model, where the global parameters $params = (q, g, G_1, G_2, e)$ are obtained by running the Setup algorithm on input the security parameter 1^k. Let $x \in G_1$ and $Q, Q^* \in G_2$ be publicly known values, where $Q^* = e(y, y)$ for some $y \in G_1$.

We provide the following novel proof-of-knowledge protocols, which are all public coin and honest-verifier zero-knowledge (HVZK):

1. PoK of canonical bilinear pre-image of Q with respect to x: $PK\{(\alpha) : e(\alpha, x) = Q\}$.
2. PoK of any bilinear pre-image of Q: $PK\{(\alpha, \beta) : e(\alpha, \beta) = Q\}$.
3. PoK of the symmetric bilinear pre-image of Q^*: $PK\{(\alpha) : e(\alpha, \alpha) = Q^*\}$.

Each of these protocols naturally extends the Schnorr protocol for proving knowledge of a discrete logarithm [41] into protocols for proving knowledge of special pre-images of bilinear mappings. These protocols form the basis for our new IBI and IBS schemes in Section 5.

Non-interactive proofs: All of the protocols presented here can be made non-interactive by applying the Fiat-Shamir heuristic [23]. Though certain theoretical concerns have been expressed about this approach [26], most implemented signature schemes continue to employ it, because no practical attack has yet been exhibited.

4.1 Proving Knowledge of the Canonical Pre-image of a Bilinear Mapping

For global parameters $params = (q, g, G_1, G_2, e)$ and any values $x \in G_1$ and $Q \in G_2$, the *canonical* pre-image of Q with respect to the bilinear mapping e and element x is the value $\alpha \in G_1$ such that $e(\alpha, x) = Q$. We provide a protocol $\mathcal{T}_1$ in Figure 1 for proving knowledge of the *canonical* pre-image of Q with respect to the bilinear mapping e and element x; that is:

$$PK\{(\alpha) : e(\alpha, x) = Q\}.$$

Alternatively, this protocol can be thought of as contributing to the set of existing protocols for proving knowledge of a committed value [41, 25, 18], in a new (computationally-hiding, perfectly binding) commitment scheme where one commits to a value $y \in G_1$ by publishing $(e(y, x), x)$, and decommits by publishing y.

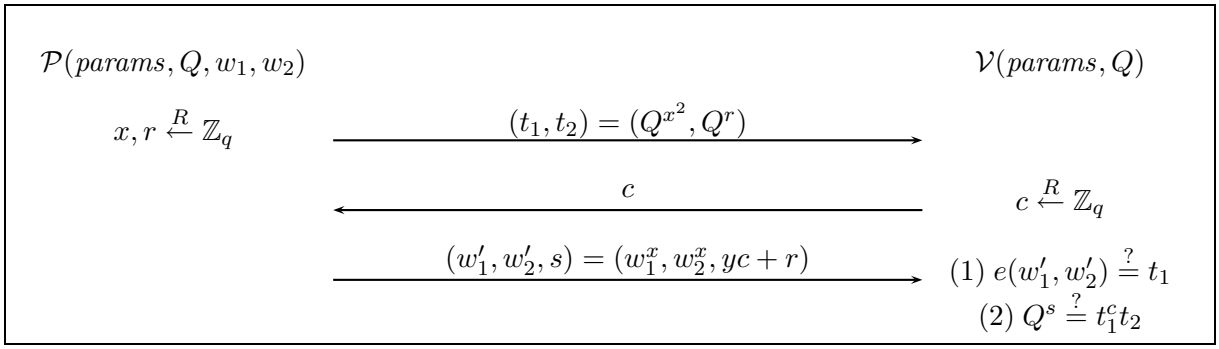

Figure 2: Description of the HVZK proof of knowledge protocol $\mathcal{T}_2$ for showing knowledge of any pre-image of Q (i.e., $w_1, w_2 \in G_1$ such that $Q = e(w_1, w_2)$). First, the prover sends $t_1 = Q^y$ and $t_2 = Q^r$, where x, r are random values in $\mathbb{Z}_q$ and $y = x^2 \pmod q$. Next, the verifier sends a random challenge $c \in \mathbb{Z}_q$. Finally, the prover responds with $w_1' = w_1^x$, $w_2' = w_2^x$, and $s = yc + r$. The verifier accepts if and only if the following relations hold: (1) $e(w_1', w_2') = t_1$ and (2) $Q^s = t_1^c t_2$.

Observe that protocol $\mathcal{T}_1$ also works smoothly for mappings of the form $e : G_1 \times \tilde{G} \rightarrow G_2$. When the value x is set to g from the global parameters, this protocol exactly corresponds with the IBI scheme which Bellare et al. [5] derived from an IBS scheme which was proposed by Hess [30] and proven secure by Dodis et al. [19].

Lemma 4.1 *Protocol $\mathcal{T}_1$ is an honest-verifier, zero-knowledge proof of knowledge of the canonical pre-image of Q with respect to the global parameters $params = (q, g, G_1, G_2, e)$ and element x under the Computational Diffie-Hellman (CDH) Assumption in G_1.* (Proof in Appendix A.)

4.2 Proving Knowledge of *any* Pre-image of a Bilinear Mapping

For global parameters $params = (q, g, G_1, G_2, e)$ and any value $Q \in G_2$, we provide a protocol $\mathcal{T}_2$ in Figure 2 for proving knowledge of *any* pre-image of Q with respect to the bilinear mapping e; that is:

$$PK\{(\alpha, \beta) : e(\alpha, \beta) = Q\}.$$

This is a generalization of the canonical pre-image protocol. Though slightly less efficient, this protocol is interesting because, unlike the previous one, there are many possible witnesses.

Observe that protocol $\mathcal{T}_2$ also works smoothly for mappings of the form $e : G_1 \times \tilde{G} \rightarrow G_2$. For a witness indistinguishable proof of knowledge, the prover may simply choose a random $c \in \mathbb{Z}_q^*$ and send the verifier $(w_1^c, w_2^{1/c})$. However, this has the potentially negative side-effect of providing the verifier with a valid pre-image of Q. Thus, we achieve something stronger in $\mathcal{T}_2$.

Lemma 4.2 *Protocol $\mathcal{T}_2$ is an honest-verifier, zero-knowledge proof of knowledge of a pre-image of Q with respect to the global parameters $params = (q, g, G_1, G_2, e)$ under the Bilinear One-Way Assumption.* (Proof in Appendix A.)

4.3 Proving Knowledge of the Symmetric Pre-image of a Bilinear Mapping

For global parameters $params = (q, g, G_1, G_2, e)$ and special values of $W \in G_2$, where W is of the form $e(g, g)^{a^2}$ for some $a \in \mathbb{Z}_q$, the above protocol can be adjusted to become a protocol $\mathcal{T}_3$ in Figure 3 for proving knowledge of the *symmetric* pre-image of Q with respect to the bilinear mapping e; that is:

$$PK\{(\alpha) : e(\alpha, \alpha) = Q\}.$$

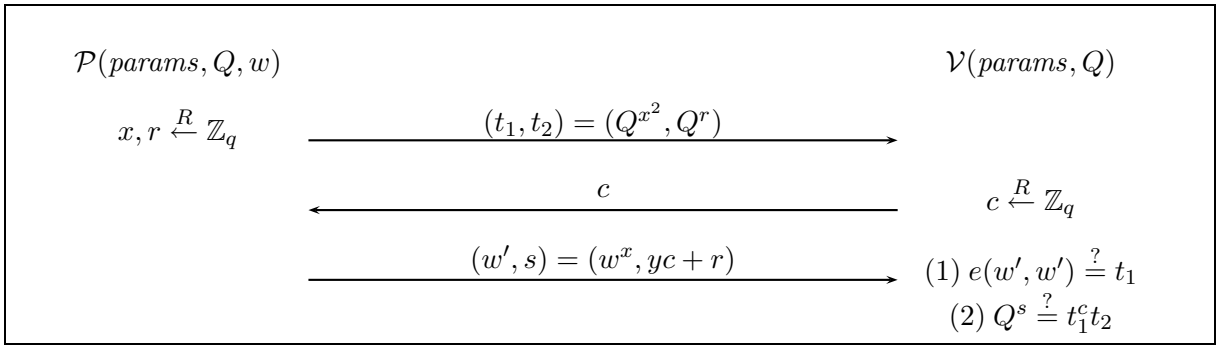

Figure 3: Description of the HVZK proof of knowledge protocol $\mathcal{T}_3$ for showing knowledge of a symmetric pre-image of Q (i.e., $w \in G_1$ such that $Q = e(w, w)$). $\mathcal{P}$ and $\mathcal{V}$ follow the $\mathcal{T}_2$ protocol; except in round three, $\mathcal{P}$ only sends two values $w' = w^x$ and $s = yc + r$, and $\mathcal{V}$ accepts if and only if the following relations hold: (1) $e(w', w') = t_1$ and (2) $Q^s = t_1^c t_2$.

Lemma 4.3 *Protocol $\mathcal{T}_3$ is an honest-verifier, zero-knowledge proof of knowledge of the symmetric pre-image of Q with respect to the global parameters $params = (q, g, G_1, G_2, e)$ under the Bilinear One-Way Assumption.*

5 Two Identity-Based Identification and Signature Schemes

Using the proofs of knowledge from the previous section, we now build two different IBI and IBS schemes. In the next section, we will use these IBS schemes to implement SIBR signature schemes.

Our IBI and IBS schemes share keypairs with the Boneh-Franklin [8] and Waters [48] identity-based encryption (IBE) schemes, respectively. Thus, combining these IBS schemes with their corresponding IBE scheme provides a complete identity-based public key solution. The IBI schemes are proofs of knowledge of a secret key given a corresponding public key, and the IBS schemes are constructed by applying (an extended version of) the Fiat-Shamir [23] heuristic to these proofs.

Our first construction, using Boneh-Franklin [8] keys, is relatively straightforward and was previously proposed by Hess [30]. Viewing the Hess scheme in our proof of knowledge framework (as presented below) offers clarity that will be useful for our SIBR constructions in the next section. Our second identity-based construction is novel and more complicated. In particular, there are many possible secret keys associated with any one public key in the Waters [48] IBE scheme. Thus, we are proving knowledge of a *pair of values with a certain form*, rather than knowledge of some specific value.

5.1 Simple IBI and IBS Schemes Using Boneh-Franklin IBE Keys

The following identity-based identification scheme due to Hess [30] borrows its *Setup* and *Extract* algorithms from the Boneh-Franklin IBE scheme [9, 8], which is secure in the random oracle model. Interestingly, most of the known IBS schemes based on bilinear maps (e.g., [40, 15]) have identical *Setup* and *Extract* algorithms. The following protocol, however, offers the best efficiency.

Setup: On input the security parameter 1^k, run the bilinear map generation algorithm $\mathsf{Setup}(1^k) \rightarrow (q, g, G_1, G_2, e) = params$. Select a hash function $H : \{0,1\}^* \rightarrow G_1$. Select a random element $s \in \mathbb{Z}_q$. Output the master public key $MPK = (q, g, G_1, G_2, e, H, g^s)$ and store the master secret key $MSK = (MPK, s)$.

Extract: On input an identity $ID \in \{0,1\}^*$, the master computes the secret key for identity ID as $SK_{ID} = H(ID)^s$, where anyone can compute the corresponding public key $PK_{ID} = H(ID)$.

Identification Protocol: On common input MPK and ID, where the prover $\mathcal{P}$ has additional input SK_{ID}, the prover $\mathcal{P}$ and verifier $\mathcal{V}$ both locally compute the value $Q = e(g^s, PK_{ID})$ and then execute the $\mathcal{T}_1$ protocol from Section 4.1 as $\mathcal{T}_1(\mathcal{P}(params, Q, g, SK_{ID}), \mathcal{V}(params, Q, g))$.

Let us explain why the identification protocol works. The prover is being asked to prove knowledge of the canonical pre-image of $Q = e(g^s, PK_{ID})$ with respect to element g. There is only one solution for this and it is $SK_{ID} = H(ID)^s = (PK_{ID})^s$. Interestingly, when one views the value $H(ID)$ as g^b for some $b \in \mathbb{Z}_q$, it becomes apparent that the above identification protocol is actually proving knowledge of the *completion* of a DDH tuple in G_1. To see this, note that we have public values $g, g^s, H(ID) = g^b$, and the prover must show that he knows the value $g^{sb} = H(ID)^s$. Thus, for any $g, X, Y \in G_1$, we have the following proof protocol: $PK\{(\alpha) : X = g^x \wedge Y = g^y \wedge \alpha = g^{xy}\}$.

Theorem 5.1 *The above IBI scheme is secure against* passive impersonation attacks (**imp-pa**) *under the Computational Diffie-Hellman (CDH) Assumption in G_1 in the random oracle model.*

The above result follows from Lemma 4.1. This scheme was also shown to be secure against active and concurrent impersonation attacks under the one-more CDH problem in G_1 by Bellare, Namprempre, and Neven [5].

Let $H' : \{0,1\}^* \rightarrow \mathbb{Z}_q$ be a hash function. We apply the extended Fiat-Shamir heuristic due to Bellare et al. [5], which includes the identity in the challenge hash, to our *canonical* IBI to obtain the following simple IBS scheme. By canonical, we mean a three round, public coin IBI scheme that consists of the prover sending a commitment, the verifier sending a random challenge, and the prover sending back a response. Bellare et al. [5] argued that applying this extended transformation to any canonical **imp-pa** secure IBI scheme results in an IBS scheme secure against existential forgery under chosen message attack (**uf-cma**) in the random oracle model.

Sign: On input a secret key SK_{ID} and a message $m \in \{0,1\}^*$, select a random $g^r \in G_1$ and output a signature $\sigma = (R, S)$ where $R = e(g^r, g)$ and $S = SK_{ID}^{H'(m||R||ID)} g^r$.

Verify: On input a purported signature $\sigma = (R, S)$, a message m, and an identity ID, accept if and only if $e(S, g) = e(g^s, H(ID))^{H'(m||R||ID)} R$.

5.2 New IBI and IBS Schemes Using Waters IBE Keys

The following identity-based identification scheme borrows its *Setup* and *Extract* algorithms from the Waters IBE scheme [48], which is secure in the standard model.

Setup: On input the security parameter 1^k, run $\mathsf{Setup}(1^k) \rightarrow (q, g, G_1, G_2, e)$. Select random elements $h, u', u_1, \ldots, u_n \leftarrow G_1$. Select a random element $s \in \mathbb{Z}_q$. Output the master public key $MPK = (q, g, G_1, G_2, e, h, u', u_1, \ldots, u_n, g^s)$ and store the master secret key $MSK = (MPK, h^s)$. (Note that this scheme supports IDs of length $n \in \mathrm{poly}(k)$.)

Extract: On input an identity $ID \in \{0,1\}^*$, the master computes the secret key for identity ID by selecting a random $r \in \mathbb{Z}_q$ and outputting $SK_{ID} = (h^s F(ID)^r, g^r)$, where the function $F(ID) = u' \prod_{ID_i=1} u_i$. Anyone can compute the corresponding public key $PK_{ID} = F(ID)$.

Identification Protocol: On common input MPK and ID, where the prover $\mathcal{P}$ has additional input $SK_{ID} = (S_1, S_2)$, the prover $\mathcal{P}$ and the verifier $\mathcal{V}$ each locally compute $X = e(g^s, h)$ and then execute the proof protocol $PK\{(\alpha, \beta) : Xe(\beta, PK_{ID}) = e(\alpha, g)\}$ as:

1. $\mathcal{P}$ simultaneously sends $\mathcal{V}$ the value $A = e(S_1, g)$ with the first messages of the procotols:

$$P_1 = \mathcal{T}_1(\mathcal{P}(params, A, g, S_1), \mathcal{V}(params, A, g))$$
$$P_2 = \mathcal{T}_1(\mathcal{P}(params, A/X, PK_{ID}, S_2), \mathcal{V}(params, A/X, PK_{ID}))$$

   Intuitively, the prover is simultaneously showing:

$$PK\{(\alpha, \beta) : e(\alpha, g) = A \quad \wedge \quad e(\beta, PK_{ID}) = A/X\}.$$

2. $\mathcal{V}$ issues a challenge that can be parsed as (c_1, c_2).
3. $\mathcal{P}$ jointly sends the response of protocol P_1 on challenge c_1 with that of protocol P_2 on challenge c_2.

This appears to be the first IBI scheme based on bilinear maps which is secure in the standard model.

Theorem 5.2 *The above IBI scheme is secure against* passive impersonation attacks (**imp-pa**) *under the Computational Diffie-Hellman (CDH) Assumption in G_1 in the standard model.* (Proof in Appendix A.)

Let $H : \{0,1\}^* \rightarrow \mathbb{Z}_q$ be a hash function. As before, we apply the extended Fiat-Shamir heuristic to this canonical **imp-pa** secure IBI to obtain a **uf-cma** secure IBS scheme in the random oracle model. It has been observed that any IBE scheme that can be extended to a two-level heirarchical IBE (HIBE) scheme, such as Waters' IBE can, allows for a generic one-time IBS construction [14]. Since the Waters' HIBE does not require random oracles, neither would the generic Waters' IBS. However, for our constructions in the next section, we will want a (many-time) signature scheme in the standard three-move (i.e., commit, challenge, response) format to which the Fiat-Shamir heuristic can be applied with a random oracle. We obtain just such an IBS scheme as follows.

Sign: On input a secret key $SK_{ID} = (S_1, S_2)$ and a message $m \in \{0,1\}^*$, do the following:

1. Compute $T = (t_1, t_2, A)$ by selecting random values $r_1, r_2 \in \mathbb{Z}_q$ and setting $t_1 = e(g^{r_1}, PK_{ID})$, $t_2 = e(g^{r_2}, g)$, and $A = e(S_1, g)$.
2. Compute $c = (c_1, c_2)$ as $c_1 = H(m||T||ID)$ and $c_2 = H(m||T||ID)$.
3. Compute $J = (j_1, j_2)$ as $j_1 = S_2^{c_1} g^{r_1}$ and $j_2 = S_1^{c_2} g^{r_2}$.

Output the signature $\sigma = (T, J)$.

Verify: On input a purported signature $\sigma = (T, J)$, a message m, and an identity ID, accept if and only if the following relations hold: (1) $e(j_2, g) = A^{c_2} t_2$ and (2) $e(j_1, PK_{ID}) = (A/X)^{c_1} t_1$.

6 Two SIBR Signature Constructions

In Section 2, we defined separable identity-based ring signatures. We will now describe two generic techniques for constructing them using the IBS schemes presented in Section 5 and other IBS schemes. To determine which IBS schemes are suitable for our SIBR constructions, we will first categorize the existing IBS schemes. We conclude this section by providing some performance estimates for these constructions.

6.1 Categories of Identity-Based Signature Schemes

We will construct our (strong) SIBR signature schemes by combining several IBS that are of **type-CCR**; that is, canonical three-round (i.e., commit, challenge, response) schemes. As defined by Abe et al. [1], a **type-CCR** signature scheme conforms to the outline:

$\text{Sign}(sk, m) =$		$\text{Verify}(pk, m, \sigma) =$	
	$t \leftarrow C(sk; r)$		σ parses as (c, s)
	$c = H(m, t)$		$z = V(c, s, pk)$
	$s = Z(sk, r, c)$		$e = H(m, z)$
	Return $\sigma = (c, s)$.		Return 1 if $c = e$; otherwise 0.

where H, Z and V are deterministic algorithms and algorithm C takes as input randomness r.

Fortunately, we have many IBS schemes to choose from. All of the IBS schemes listed below are secure in the random oracle model using the Fiat-Shamir paradigm; we point out which ones also conform to the **type-CCR** style. We note that the only bilinear map based IBS scheme that is **type-CCR** is the one presented in Section 5.1.

type-CCR IBS (7 schemes): (*factoring based*) Feige, Fiat, and Shamir [22], Dodis, Katz, Xu, and Yung [19], Fischlin and Fischlin [24]; (*RSA based*) Guillou and Quisquater [28], Okamoto [37]; (*DL based*) Beth [7]; (*Bilinear map based*) Hess [30] in Section 5.1.

Not type-CCR IBS (5 schemes): (*RSA based*) Shamir [43], Bellare, Namprempre, and Neven [5]; (*Bilinear map based*) Sakai, Ohgishi, and Kasahara [40], Cha and Cheon [15], New scheme in Section 5.2.

Although the bilinear IBS schemes are currently slower in practice (roughly eight times slower than the others), they may be preferable given that they share keys with existing IBE schemes. *With some care* the same keys could be used for both encryption and signing.

6.2 Weakly and Medium Separable Identity-Based Ring Signatures

First, we observe that any **type-CCR** signature scheme can be transformed into a *weakly* SIBR signature scheme by applying the proof of partial knowledge techniques of Cramer, Damgård, and Schoenmakers [17]. To form a ring signature on message m between herself and Bob (who is using the *same* IBS scheme), Alice can non-interactively prove that she knows *either* sk_A or sk_B. If she knew both secret keys, the transcripts would look like (t_1, c_1, s_1) and (t_2, c_2, s_2) for $c_1 = H(L_1, m, t_1)$ and $c_2 = H(L_2, m, t_2)$, where L_1, L_2 include the global parameters and public keys of Alice and Bob respectively. To convince Carol that she knows at least one of the secrets, the transcripts are instead of the form (t_1, c, s_1) and (t_2, c', s_2) where c, c' are secret shares of $H(L_1, L_2, m, t_1, t_2)$ according to some scheme.

The above works because the IBS schemes of Alice and Bob have the same *challenge set* (i.e., range of H, for example, $\mathbb{Z}_q$). Suppose Alice is using the IBS scheme in Section 5.1 with challenge set $\mathbb{Z}_q$, and Bob is using that of Section 5.2 with challenge set $\mathbb{Z}_{q'}$. In this case, we can obtain a *medium* SIBR signature scheme by drawing a challenge from the smaller of the two challenge sets $\mathbb{Z}_{\min\{q,q'\}}$; that is, we use the hash function from the IBS scheme with the smaller set, and then we proceed as in the weak SIBR signature case. This works for any IBS scheme where the challenge set is an additive group of prime order.

If the component IBS schemes are **uf-cma** secure, then both of the above methods result in ring signatures that are **uf-cma** and unconditionally signer ambiguous in the random oracle model.

6.3 Strongly Separable Identity-Based Ring Signatures

A *strongly* SIBR signature scheme can be constructed from any combination of **type-CCR** IBS schemes[1] by applying the ring signature compiler of Abe, Ohkubo, and Suzuki [1]. This compiler was designed for joining keys from many *standard* signature schemes into a ring signature. However, it is easy to see the extension to *identity-based* signature schemes, when one views these schemes as IBIs with a Fiat-Shamir component as in Section 5.1 and Bellare et al. [5]. If all adjoining IBS schemes are **uf-cma** secure, then

[1] Abe et al.[1] also allow for hash-and-sign style signatures, however, no known IBS schemes have this form.

the resulting AOS ring signature is also **uf-cma** secure and unconditionally signer-ambiguous in the random oracle model [1]. Indeed, it seems that random oracles are a necessary part of the AOS construction for linking the (separable) signatures into a ring.

Due to space considerations, we only sketch how the AOS compiler is applied to form a two-party ring signature from distinct **type-CCR** IBS schemes Φ_0 and Φ_1 with global parameters $params_0$ and $params_1$ that include the appropriate hash functions H_0 and H_1. Let $L = (params_0, params_1)$ and $K = (pk_0, pk_1)$ for some pk_0 and pk_1, where secret key sk_b is known.

- $\mathsf{RSign}(L, K, sk_b, m)$: start the ring by computing $e_b = C_b(sk_b; r)$, where r is a random string, and $c_{\bar{b}} = H_{\bar{b}}(L, K, m, e_b)$. Then choose $s_{\bar{b}}$ at random and compute $e_{\bar{b}} = V_{\bar{b}}(c_{\bar{b}}, s_{\bar{b}}, pk_{\bar{b}})$. Finally, close the ring by computing $c_b = H_b(L, K, m, e_{\bar{b}})$. The resulting signature for (L, K, m) is $\sigma = (c_0, s_0, s_1)$.
- $\mathsf{RVerify}(L, K, m, \sigma)$: we verify the signature as follows. Compute $e_0 = V_0(c_0, s_0, pk_0)$, $c_1 = H_1(L, K, m, e_0)$, and $e_1 = V_1(c_1, s_1, pk_1)$. Output 1 if the ring closes as $c_0 = H_0(L, K, m, e_1)$ and 0 otherwise.

6.4 Efficiency of SIBR Signature Constructions

Our SIBR signature constructions are reasonably efficient. We provide some performance estimates in Appendix B, which we summarize here. Recently, Ateniese et al. reported that a bilinear mapping took on average 8.6ms, while an exponentiation took an average of 1.5ms on a client machine using an AMD Athlon 1.8 GHz with 1 GB RAM [4]. By extrapolating from these measurements, and disregarding operations that are negligible compared to exponentiations, we arrive at the following *optimized* performance estimates:

SIBR Construction	Signing	Verification
Weak (2 IBS Sec 5.1 w/CDS)	9ms	20.2ms
Medium (IBS Sec 5.1,5.2 w/CDS)	15ms	30.3ms
Strong (2 IBS Sec 5.1 w/AOS)	20.2ms	20.2ms

7 An Alternative Solution: Deniable Signatures

In SIBR signature schemes, a sender can deny having authored a message m because the signature on m could have been created by either the sender or the receiver. Thus, σ is repudiable because the *identity of the signer* is in question. Conversely, in a *deniable* signature scheme [35, 44, 45], the sender creates a signature σ on message m using some public information of the receiver, in such a way that the receiver can later compute a message $m' \neq m$ where (σ, m') also verifies. Thus, σ is repudiable because the *authenticated content* is in question. We point out separable identity-based deniable signatures (no known constructions) as a possible alternative to mitigate phishing.

8 Conclusion

In this paper, we introduced, defined, and provided two constructions of *separable, identity-based ring signatures*. We showed how these signatures are particularly well-suited to the real-world task of mitigating email spoofing attacks such as phishing.

Our constructions yielded three novel and efficient HVZK proofs-of-knowledge of bilinear map pre-images. We clarified, in this new context, the signature scheme of Hess [30], and presented new identity-based identification and signature schemes based on Waters [48] keys. All of these components may well prove useful in other applications.

This work strives to assemble theoretical components into a solution for a pressing real-world problem. We believe that many such subtle practical problems may benefit from advanced cryptographic techniques.

Acknowledgements. We thank Brent Waters for useful discussions about his IBE scheme. We also thank Steve Weis for comments on an earlier version of this paper and for suggesting the name "cyber" signatures.

References

[1] Masayuki Abe, Miyako Ohkubo, and Koutarou Suzuki. 1-out-of-n signatures from a variety of keys. In Yuliang Zheng, editor, *Advances in Cryptology — ASIACRYPT '02*, volume 2501 of *LNCS*, pages 415–432. Springer Verlag, 2002.

[2] Ben Adida, Susan Hohenberger, and Ronald L. Rivest. Fighting Phishing Attacks: A Lightweight Trust Architecture for Detecting Spoofed Emails (to appear), 2005. Available at `http://theory.lcs.mit.edu/~rivest/publications.html`.

[3] Anti-Phishing Working Group. Digital Signatures to Fight Phishing Attacks. `http://www.antiphishing.org/smim-dig-sig.htm`.

[4] Giuseppe Ateniese, Kevin Fu, Matthew Green, and Susan Hohenberger. Improved Proxy Re-encryption Schemes with Applications to Secure Distributed Storage. In *the 12th Annual Network and Distributed System Security Symposium*, pages 29–43, 2005. Full version available at http://eprint.iacr.org/2005/028.

[5] Mihir Bellare, Chanathip Namprempre, and Gregory Neven. Security proofs for identity-based identification and signature schemes. In Christian Cachin and Jan Camenisch, editors, *Advances in Cryptology — EUROCRYPT '04*, volume 3027 of *LNCS*, pages 268–286. Springer Verlag, 1999.

[6] Mihir Bellare and Adriana Palacio. GQ and Schnorr Identification Schemes: Proofs of Security against Impersonation under Active and Concurrent Attacks. In Moti Yung, editor, *Advances in Cryptology — CRYPTO 2002*, volume 2442 of *LNCS*, pages 162–177. Springer Verlag, 2002.

[7] Thomas Beth. Efficient zero-knowledge identification scheme for smart cards. In C. Gunther, editor, *Advances in Cryptology — EUROCRYPT '88*, volume 330 of *LNCS*, pages 77–84. Springer Verlag, 1988.

[8] Dan Boneh and Matt Franklin. Identity-based Encryption from the Weil Pairing. *SIAM Journal of Computing*, 32(3):586–615, 2003.

[9] Dan Boneh and Matthew Franklin. Identity-based encryption from the Weil pairing. In Joe Kilian, editor, *Advances in Cryptology — CRYPTO 2001*, volume 2139 of *LNCS*, pages 213–229. Springer Verlag, 2001.

[10] N. Borisov, I. Goldberg, and E. Brewer. Off-the-record communication, or, why not to use PGP. In *WPES '04: Proceedings of the 2004 ACM workshop on Privacy in the electronic society*, pages 77–84. ACM Press, 2004.

[11] Jan Camenisch and Markus Michels. Proving in zero knowledge that a number n is the product of two safe primes. In *proceedings of Eurocrypt '99*, volume 1592 of LNCS, pages 107–122, 1999.

[12] Jan Camenisch and Markus Michels. Separability and efficiency for generic group signature schemes. In *proceedings of Crypto '99*, volume 1666 of LNCS, pages 413–430, 1999.

[13] Jan Camenisch and M. Stadler. Efficient group signature schemes for large groups. In *proceeding of Crypto '97*, volume 1296 of LNCS, pages 410–424, 1997.

[14] Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity-based encryption. In *Advances in Cryptology – EUROCRYPT '04*, volume 3027 of *LNCS*, pages 207–222, 2004.

[15] Jae Choon Cha and Jung Hee Cheon. An Identity-Based Signature from Gap Diffie-Hellman Groups. In Y.G. Desmedt, editor, *PKC 2003*, volume 2567 of *LNCS*, pages 18–30. Springer-Verlag, 2003.

[16] David Chaum and T.P. Pedersen. Wallet Databases with Observers. In *proceedings of Crypto '92*, volume 740 of LNCS, pages 89–105, 1992.

[17] Ronald Cramer, Ivan Damgård, and Berry Schoenmakers. Proofs of partial knowledge and simplified design of witness hiding protocols. In Yvo G. Desmedt, editor, *Advances in Cryptology — CRYPTO '94*, volume 839 of *LNCS*, pages 174–187. Springer Verlag, 1994.

[18] Ivan Damgård and Eiichiro Fujisaki. An integer commitment scheme based on groups with hidden order. In *Advances in Cryptology — ASIACRYPT 2002*, volume 2501 of *LNCS*. Springer, 2002.

[19] Yevgeniy Dodis, Jonathan Katz, Shouhuai Xu, and Moti Yung. Strong key-insulated signature schemes. In Y. Desmedt, editor, *PKC 2003*, volume 2567 of *LNCS*, pages 130–144. Springer Verlag, 2003.

[20] Cynthia Dwork, Moni Naor, and Amit Sahai. Concurrent zero-knowledge. In *the 30th ACM Symposium on the Theory of Computing*, pages 409–418, 1998.

[21] Eudora. ScamWatch. `http://www.eudora.com/email/features/scamwatch.html`.

[22] Uriel Feige, Amos Fiat, and Adi Shamir. Zero-knowledge proofs of identity. *Journal of Cryptology*, 1:77–94, 1988.

[23] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *proceedings of Crypto '86*, volume 263 of LNCS, pages 186–194, 1986.

[24] M. Fischlin and R. Fischlin. The representation problem based on factoring. In *proceedings of CT-RSA '02*, volume 2271 of LNCS, pages 96–113, 2002.

[25] Eiichiro Fujisaki and Tatsuaki Okamoto. Statistical zero knowledge protocols to prove modular polynomial relations. In Burt Kaliski, editor, *Advances in Cryptology — CRYPTO '97*, volume 1294 of *LNCS*, pages 16–30. Springer Verlag, 1997.

[26] Shafi Goldwasser and Yael Tauman Kalai. On the (in)security of the Fiat-Shamir paradigm. In *Proc. 44th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 102–115. IEEE Computer Society Press, 2003.

[27] Anti-Phishing Working Group. Phishing activity trends report, November, 2004. Available at http://www.antiphishing.org.

[28] Louis C. Guillou and Jean-Jacques Quisquater. A "paradoxical" identity-based signature scheme resulting from zero-knowledge. In Shafi Goldwasser, editor, *Advances in Cryptology — CRYPTO '88*, volume 403 of *LNCS*, pages 216–231. Springer Verlag, 1988.

[29] Amir Herzberg and Ahmad Gbara. Trustbar: Protecting (even naive) web users from spoofing and phishing attacks. Cryptology ePrint Archive, Report 2004/155, 2004. `http://eprint.iacr.org/2004/155`.

[30] Florian Hess. Efficient identity based signature schemes on pairings. In K. Nyberg and H. Heys, editors, *Selected Areas in Cryptography — SAC '02*, volume 2595 of *LNCS*, pages 310–324. Springer Verlag, 2002.

[31] Markus Jakobsson, Kazue Sako, and Russell Impagliazzo. Designated verifier proofs and their applications. In Ueli Maurer, editor, *Advances in Cryptology — EUROCRYPT '96*, volume 1233 of *LNCS*. Springer, 1996.

[32] Justin Mason. Filtering Spam with SpamAssassin. In *proceedings of HEANet Annual Conference*, 2002.

[33] M. Sahami and S. Dumais and D. Heckerman and E. Horvitz. A Bayesian Approach to Filtering Junk E-Mail. In *Learning for Text Categorization: Papers from the 1998 Workshop*, May 1998.

[34] MAPS. RBL - Realtime Blackhole List, 1996. `http://www.mail-abuse.com/services/mds_rbl.html`.

[35] Moni Naor. Deniable ring authentication. In *Proceedings of Advances in Cryptology – CRYPTO '02*, volume 2442 of *LNCS*, pages 481–498. Springer, 2002.

[36] Netcraft. Anti-Phishing Toolbar. `http://news.netcraft.com/archives/2004/12/28/netcraft_antiphishing_tool%bar_available_for_download.html`.

[37] Tatsuaki Okamoto. Provably secure and practical identification schemes and corresponding signature schemes. In *proceedings of CRYPTO'92*, volume 740 of LNCS, pages 31–53, 1992.

[38] Mario Di Raimondo and Rosario Gennaro. New approaches for deniable authentication. In *Workshop on Provable Security*, 2004.

[39] Ronald L. Rivest, Adi Shamir, and Yael Tauman. How to leak a secret. In Colin Boyd, editor, *Advances in Cryptology — ASIACRYPT '01*, volume 2248 of *LNCS*, pages 552–565. Springer Verlag, 2001.

[40] R. Sakai, K. Ohgishi, and M. Kasahara. Cryptosystems based on pairing. In *Proceedings of the Symposium on Cryptography and Information Security — SCIS 2000*, 2000.

[41] Claus-Peter Schnorr. Efficient Signature Generation by Smart Cards. *Journal of Cryptology*, 4(3):161–174, 1991.

[42] Michael Scott. MIRACL library. Indigo Software. http://indigo.ie/∼mscott/#download.

[43] Adi Shamir. Identity-based cryptosystems and signature schemes. In George Robert Blakley and David Chaum, editors, *Advances in Cryptology — CRYPTO '84*, volume 196 of *LNCS*, pages 47–53. Springer Verlag, 1985.

[44] Willy Susilo and Yi Mu. Non-interactive deniable ring signatures. In *the 6th International Conference on Information Security and Cryptology (ICISC) '03*, pages 397–412, 2003.

[45] Willy Susilo and Yi Mu. Deniable ring authentication revisited. In *Applied Cryptography and Network Security (ANCS) '04*, volume 3089 of LNCS, pages 149–163, 2004.

[46] T.A. Meyer and B. Whateley. SpamBayes: Effective open-source, Bayesian based, email classification system. In *in proceedings of Conference on Email and Anti-Spam 2004*, July 2004.

[47] The Spamhaus Project. The Spamhaus Block List. `http://www.spamhaus.org/sbl/`.

[48] Brent Waters. Efficient Identity-Based Encryption Without Random Oracles. In *proceedings of EUROCRYPT 2005 (to appear)*, July 2004. Available at http://eprint.iacr.org/2004/180.

[49] Fangguo Zhang and Kwangjo Kim. ID-Based Blind Signature and Ring Signature from Pairings. In Yuliang Zheng, editor, *Advances in Cryptology — ASIACRYPT '02*, volume 2501 of *LNCS*, pages 533–547. Springer Verlag, 2002.

A Proofs of Security

We note that Bellare and Palacio's Reset Lemma [6] could be applied to the proofs below to obtain a tighter concrete security guarantee.

Lemma 4.1: The following is the proof of Lemma 4.1.

Proof. We first show that $\mathcal{T}_1$ is a proof of knowledge, and then show that it also has the honest-verifier zero-knowledge property.

For $\mathcal{T}_1$ to be a proof of knowledge, there must exist a PPT extractor $\mathcal{E}$ that, after interacting with any prover $\mathcal{P}$ which can convince an honest $\mathcal{V}$ to accept with probability $> 1/\mathrm{poly}(k)$, can produce the witness w with probability $> 1/\mathrm{poly}(k)$. $\mathcal{E}$ works as follows: (Step 1) execute $\mathcal{T}_1$ with $\mathcal{P}$ exactly as an honest $\mathcal{V}$ would to obtain the transcript (t_1, c_1, s_1); (Step 2) rewind $\mathcal{P}$ until just after it sends t_1 and reply with a new random challenge c_2, and receive $\mathcal{P}$'s response s_2 to obtain the transcript (t_1, c_2, s_2). If an honest $\mathcal{V}$ would have accepted these transcripts, then $\mathcal{E}$ now holds the values $s_1 = w^{c_1} t_1$ and $s_2 = w^{c_2} t_1$, where $c_1 \neq c_2$, and thus $\mathcal{E}$ can compute the witness as $(s_1/s_2)^{1/(c_1-c_2)} = (w^{c_1} t_1 / w^{c_2} t_1)^{1/(c_1-c_2)} = (w^{c_1-c_2})^{1/(c_1-c_2)} = w$. Thus, $\mathcal{P}$ must know a witness, or she breaks the computational Diffie-Hellman assumption in G_1.

(To see this, consider that on input (g, g^a, g^b), one sets $x = g^r$ for a random $r \in \mathbb{Z}_q$ and $Q = e(g^a, g^b)$, and then carries out the proof with a cheating prover. When the witness w is extracted, as above, it must be the case that $w^r = g^{ab}$ since $e(w, x) = Q$.)

For $\mathcal{T}_1$ to be an honest-verifier zero-knowledge proof, there must exist a PPT simulator $\mathcal{S}$ that can simulate a prover for any honest $\mathcal{V}$ without knowing the witness w. By saying $\mathcal{V}$ is honest, we mean that $\mathcal{V}$ has a fixed random tape from which he selects his challenges. $\mathcal{S}$ works as follows: (Step 1) send an arbitrary value in G_2, wait for $\mathcal{V}$ to send a challenge c; (Step 2) pick a random $s \in G_1$, compute $t = e(s, x)/Q^c$; (Step 3) rewind $\mathcal{V}$, send t as the first message, and s as the response to $\mathcal{V}$'s challenge. Since $\mathcal{V}$ is honest, he will send the same challenge c and thus accept the transcript (t, c, s). One can observe that this simulation produces perfectly distributed transcripts. □

Lemma 4.2: The following is the proof of Lemma 4.2.

Proof. We first show that $\mathcal{T}_2$ is a proof of knowledge, and then show that it also has the honest-verifier zero-knowledge property.

For $\mathcal{T}_2$, the proof of knowledge extractor $\mathcal{E}$ works as follows: (Step 1) execute $\mathcal{T}_2$ with $\mathcal{P}$ exactly as an honest $\mathcal{V}$ would to obtain the transcript $((t_1, t_2), c, (w_1, w_2, s))$; (Step 2) rewind $\mathcal{P}$ until just after it sends (t_1, t_2) and reply with a new random challenge c', and receive $\mathcal{P}$'s response to obtain the transcript $((t_1, t_2), c', (w_1', w_2', s'))$. If an honest $\mathcal{V}$ would have accepted these transcripts, then $\mathcal{E}$ now holds the values $s = yc + r$ and $s' = yc' + r$, where $c \neq c'$, and thus $\mathcal{E}$ can recover the value y as $(s - s')/(c - c') = y$. Next, $\mathcal{E}$ computes x as the square root of y modulo q. Finally, $\mathcal{E}$ can output the witness $(w_1^{1/x}, w_2^{1/x})$. Thus, $\mathcal{P}$ must know a witness, or she breaks the Bilinear One-Way Assumption.

For $\mathcal{T}_2$, the honest-verifier zero-knowledge simulator $\mathcal{S}$ works as follows: (Step 1) select random values $a, b \in G_1$ and $d \in G_2$, send to $\mathcal{V}$ the pair of values $(t_1 = e(a, b), d)$, and wait for $\mathcal{V}$ to send a challenge c;

(Step 2) pick a random $s \in G_1$, compute $t_2 = Q^s/t_1^c$; (Step 3) rewind $\mathcal{V}$ to the beginning, send (t_1, t_2) as the first message, and s as the response to $\mathcal{V}$'s challenge. Since $\mathcal{V}$ is honest, he will send the same challenge c and thus accept the transcript $((t_1, t_2), c, s)$. One can observe that this simulation produces perfectly distributed transcripts. □

Theorem 5.2: The following if the proof of Theorem 5.2.

Proof. In this proof we will actually show something stronger than the fact that the IBI scheme in Section 5.2 is secure against passive impersonation attacks. We will show that the three round IBI protocol is actually an honest verifier zero-knowledge proof of knowledge of an IBI user secret key.

The proof of knowledge extractor $\mathcal{E}$ works as follows: (Step 1) execute the protocol with $\mathcal{P}$ exactly as an honest $\mathcal{V}$ would to obtain the transcript $((A, t_1, t_2), (c_1, c_2), (j_1, j_2))$; (Step 2) rewind $\mathcal{P}$ until just after it sends (A, t_1, t_2) and reply with a new random challenge (c_1', c_2'), and receive $\mathcal{P}$'s response to obtain the transcript $((A, t_1, t_2), (c_1', c_2'), (j_1', j_2'))$. If an honest $\mathcal{V}$ would have accepted these transcripts, then $\mathcal{E}$ can extract the witness (i.e., user's secret key) as follows.

Since $c_1 \neq c_1'$ and $c_2 \neq c_2'$, $\mathcal{E}$ can recover secret key (S_1, S_2) as $(j_2/j_2')^{1/(c_2 - c_2')} = S_1$ and $(j_1/j_1')^{1/(c_1 - c_1')} = S_2$. Thus, $\mathcal{P}$ must know a the secret key, or she can fake $\mathcal{T}_1$ proof protocols and thereby break the CDH Assumption in G_1.

The honest-verifier zero-knowledge simulator $\mathcal{S}$ works as follows: (Step 1) select a random value $r \in \mathbb{Z}_q$ and $t_1', t_2' \in G_2$, compute $A = e(g^s, h)e(g, f(ID))^r$ (where g^s and h are part of MPK), and send to $\mathcal{V}$ the tuple (A, t_1', t_2'), and wait for $\mathcal{V}$ to send a challenge (c_1, c_2); (Step 2) pick two random values $j_1, j_2 \in G_1$, compute $t_1 = e(j_1, f(ID))/(A/X)^{c_1}$ and $t_2 = e(j_2, g)/A^{c_2}$; (Step 3) rewind $\mathcal{V}$ to the beginning, send (A, t_1, t_2) as the first message, and (j_1, j_2) as the response to $\mathcal{V}$'s challenge. Since $\mathcal{V}$ is honest, he will send the same challenge (c_1, c_2) and thus accept this conversation. Again, we observe that the transcripts produced by $\mathcal{S}$ are perfectly distributed. This follows in part from the fact that a real prover can randomize his witness $(h^s f(ID)^r, g^r)$ as $(h^s f(ID)^r f(ID)^{r'}, g^r g^{r'}) = (h^s f(ID)^{r+r'}, g^{r+r'})$ for any value of $(r + r') \in \mathbb{Z}_q$. Thus, *any* random value from $\mathbb{Z}_q$ used by $\mathcal{S}$ to create A in step one is valid and from the correct distribution. □

B Detailed Performance Estimates

We will first look at the *strong* SIBR signature where both Alice and Bob are using the Hess IBS scheme [30] (see Section 5.1) and their two-party ring signature is formed according to the generic method due to Abe et al. [1]. This requires at most the following computations:

- *Signing:* 5 mappings, 5 exponentiations, 3 multiplications, and 2 hashes
- *Verification:* 4 mappings, 2 exponentiations, 2 multiplications, and 2 hashes

Fortunately, the Hess IBS scheme allows for some further optimizations. By adding an additional element (i.e., $e(g, g)$) to the master public key MPK, the signing algorithm can be reduced by 1 mapping and 1 exponentiation. Further, half of the remaining mappings for both the signing and verification need only be performed once for each new identity (e.g., $e(g, pk_A)$) encountered. Thus, if each user keeps a table of frequently used identities, this scheme optimizes to:

- *Signing:* 2 mappings, 2 exponentiations, 3 multiplications, and 2 hashes
- *Verification:* 2 mappings, 2 exponentiations, 2 multiplications, and 2 hashes

Recently, Ateniese et al. [4] implemented a secure file system using the bilinear mappings from version 4.83 of the MIRACL cryptographic library [42]. Ateniese et al. [4] report that a mapping took on average

8.6ms, while an exponentiation took an average of 1.5ms. These results are based on a client machine using an AMD Athlon 2100+ 1.8 GHz with 1 GB RAM and an IBM 7200 RPM, 40 GB, Ultra ATA/100 hard drive [4].

By extrapolating from these measurements, and disregarding operations that are negligible compared to exponentiations, we arrive at the following optimized performance estimates:

- *Signing:* 2 mappings x (8.6ms) + 2 exps x (1.5ms) = 20.2ms
- *Verification:* 2 mappings x (8.6ms) + 2 exps x (1.5ms) = 20.2ms

The *weak* (between two Hess IBS schemes) and *medium* SIBR signatures (between one Hess IBS and one IBS from Section 5.2) are more efficient in terms of signature generation. For the optimized version of the weak SIBR signature, we have:

- *Signing:* 6 exps x (1.5ms) = 9ms
- *Verification:* 2 mappings x (8.6ms) + 2 exps x (1.5ms) = 20.2ms

And for the optimized medium SIBR signature, we have:

- *Signing:* 10 exps x (1.5ms) = 15ms
- *Verification:* 3 mappings x (8.6ms) + 3 exps x (1.5ms) = 30.3ms