Blind Identity-Based Encryption and Simulatable Oblivious Transfer

Matthew Green* Susan Hohenberger*

Abstract

In an identity-based encryption (IBE) scheme, there is a *key extraction* protocol where a user submits an identity string to a master authority who then returns the corresponding secret key for that identity. In this work, we describe how this protocol can be performed efficiently and in a *blind* fashion for several known IBE schemes; that is, a user can obtain a secret key for an identity without the master authority learning anything about this identity.

We formalize this notion as *blind IBE* and discuss the many practical applications of such a scheme. In particular, we build upon the recent work of Camenisch, Neven, and shelat in Eurocrypt 2007 to construct oblivious transfer (OT) schemes which achieve full simulatability for both sender and receiver. OT constructions with comparable efficiency prior to Camenisch et al. were proven secure in the weaker half-simulation model. Our OT schemes can be constructed generically from any blind IBE, and thus require only static complexity assumptions (e.g., DBDH) whereas prior comparable schemes require dynamic complexity assumptions (e.g., q-PDDH).

Keywords: identity-based encryption, oblivious transfer, blind key extraction.

1 Introduction

In an oblivious transfer (OT_k^N) protocol, introduced by Rabin [Rab81] and generalized by Even, Goldreich and Lempel [EGL82] and Brassard, Crépeau and Robert [BCR86], a Sender with messages $M_1, \ldots, M_N$ and a Receiver with indices $\sigma_1, \ldots, \sigma_k \in [1, N]$ interact in such a way that at the end the Receiver obtains $M_{\sigma_1}, \ldots, M_{\sigma_k}$ without learning anything about the other messages and the Sender does not learn anything about $\sigma_1, \ldots, \sigma_k$. Naor and Pinkas were the first to consider an *adaptive* setting, $\mathsf{OT}_{k\times 1}^N$, where the sender may obtain M_{i-1} before deciding on σ_i [NP99b]. Oblivious transfer is a useful, interesting primitive in its own right, but it is has even greater significance as OT_1^4 is a key building block for secure multi-party computation [Yao86, GMW87, Kil88]. Realizing efficient protocols under modest complexity assumptions is therefore an important goal.

The definition of security for oblivious transfer has been evolving. Informally, security is defined with respect to an ideal-world experiment in which the Sender and Receiver exchange messages via a trusted party. An OT protocol is secure if, for every real-world cheating Sender (resp. Receiver) we can describe an ideal-world counterpart who gains as much information from the ideal-world interaction as from the real protocol. Bellare and Micali [BM89] presented the first practical OT_1^2 protocol to satisfy this intuition in the honest-but-curious model. This was followed by practical OT protocols due to Naor and Pinkas [NP99a, NP99b, NP01] in the "half-simulation" model where the

*The Johns Hopkins University, {mgreen, susan}@cs.jhu.edu.

simulation-based model (described above) is used only to show Sender security and Receiver security is defined by a simpler game-based definition. Almost all efficient OT protocols are proven secure with respect to the half-simulation model, e.g. [NP99a, NP99b, NP01, DHRS04, Kal05, HK07]. Unfortunately, Naor and Pinkas demonstrated that this model permits *selective-failure* attacks, in which a malicious Sender can induce transfer failures that are dependent on the message that the Receiver requests [NP99b].

Recently, Camenisch, Neven, and shelat [CNS07] proposed several practical $\mathsf{OT}^N_{k\times 1}$ protocols that are secure in the "full-simulation" model, where the security of both the Sender and Receiver are simulation-based. These simulatable OT protocols are particularly nice because they can used to construct other cryptographic protocols in a simulatable fashion. More specifically, Camenisch et al. [CNS07] provide two distinct results. First, they show how to efficiently construct $\mathsf{OT}^N_{k\times 1}$ generically from any unique blind signature scheme in the random oracle model. The two known efficient unique blind signature schemes due to Chaum [Cha82] and Boldyreva [Bol03] both require *interactive* complexity assumptions: one-more-inversion RSA and chosen-target CDH, respectively. Second, Camenisch et al. [CNS07] provide a clever $\mathsf{OT}^N_{k\times 1}$ construction in the standard model based on *dynamic* assumptions q-Power Decisional Diffie-Hellman (i.e., in a bilinear setting $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$, given $(g, g^x, g^{x^2}, \ldots, g^{x^q}, H)$ where $g \leftarrow \mathbb{G}$ and $H \leftarrow \mathbb{G}_T$, distinguish $(H^x, H^{x^2}, \ldots, H^{x^q})$ from random values) and q-Strong Diffie-Hellman (which Cheon has shown require larger than commonly used security parameters [Che06]). These complexity assumptions seem significantly stronger than those, such as DDH and quadratic residuosity, used to construction efficient OT schemes in the half-simulation model. Thus, a well-motivated problem is to find efficient, fully-simulatable OT schemes under weaker complexity assumptions.

Our Contributions. In this work, we provide, to our knowledge, the first efficient and fully-simulatable OT^N_k and $\mathsf{OT}^N_{k\times 1}$ schemes secure under *static* complexity assumptions (e.g., Decisional Bilinear Diffie-Hellman (DBDH), where given (g, g^a, g^b, g^c), it is hard to distinguish $e(g, g)^{abc}$ from random). We summarize our results as follows.

First, we introduce a building block. In identity-based encryption (IBE) [Sha84], there is an *extraction* protocol where a user submits an identity string to a master authority who then returns the corresponding decryption key for that identity. We formalize the notion of *blindly* executing this protocol, in a strong sense; where the authority does not learn the identity nor can she cause failures dependent on this identity, and the user learns nothing beyond the normal extraction protocol. In §3.1, we then describe efficient *blind extraction* protocols satisfying this definition for the IBE schemes due to Boneh and Boyen [BB04a], Waters [Wat05], and Naccache [Nac05]. We call IBE schemes supporting efficient blind extraction protocols: *blind IBE*.

Second, we present an efficient and fully-simulatable OT^N_k protocol generically constructed from any (selective-identity secure) blind IBE scheme. Thus, using any of the blind IBEs mentioned above, our construction is secure under only DBDH. Intuitively, consider the following OT^N_k construction. The Sender runs the IBE setup algorithm and sends the corresponding public parameters to the Receiver. Next, for $i = 1$ to N, the Sender encrypts M_i under identity "i" and sends this ciphertext to the Receiver. To obtain k messages, the Receiver blindly extracts k decryption keys for identities of his choice and uses these keys to decrypt and recover the corresponding messages. While this simple protocol does not appear to be simulatable, we are able to appropriately modify it. Our generic construction from blind IBE is inspired by the Camenisch et al. [CNS07] generic construction from unique blind signatures. Indeed, recall that the secret keys sk_{id} of any

fully-secure IBE can be viewed as signatures by the authority on the message *id* [BF01]. However, [CNS07] require *unique* blind signatures, whereas we do not; also, while signatures can be derived from weakly-secure IBE schemes [CFH+07], we require a scheme that provides at least semantic security.

Third, we present an efficient and fully-simulatable $\mathsf{OT}^N_{k\times 1}$ protocol generically constructed from any (selective-identity secure) blind IBE scheme in the random oracle model. We discuss how to remove these oracles at an additional cost. This improves on the complexity assumptions required by the comparable random-oracle scheme in Camenisch et al. [CNS07], although we leave the same improvement for their (specific) adaptive construction without random oracles as an open problem. Finally, in §5, we point out the independent usefulness of blind IBE to other applications, such as blind signatures, anonymous email, and encrypted keyword search.

2 Technical Preliminaries

Let $\mathsf{BMsetup}$ be an algorithm that, on input the security parameter 1^κ, outputs the parameters for a bilinear mapping as $\gamma = (q, g, \mathbb{G}, \mathbb{G}_T, e)$, where g generates $\mathbb{G}$ and $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$. We will refer to the following complexity assumption made in these groups.

Decisional Bilinear Diffie-Hellman (DBDH) [BF01]: Let $\mathsf{BMsetup}(1^\kappa) \to (q, g, \mathbb{G}, \mathbb{G}_T, e)$. For all p.p.t. adversaries Adv, the following probability is strictly less than $1/2 + 1/\mathrm{poly}(\kappa)$:
$\Pr[a, b, c, d \leftarrow \mathbb{Z}_q;\ x_0 \leftarrow e(g,g)^{abc};\ x_1 \leftarrow e(g,g)^d;\ z \leftarrow \{0,1\};\ z' \leftarrow \mathsf{Adv}(g, g^a, g^b, g^c, x_z) : z = z']$.

Known Discrete-Logarithm-Based, Zero-Knowledge Proofs. We use known techniques for proving statements about discrete logarithms, such as (1) proof of knowledge of a discrete logarithm modulo a prime [Sch91], (2) proof that a committed value lies in a given integer interval [CFT98, CM99, Bou00], and also (3) proof of the disjunction or conjunction of any two of the previous [CDS94]. These protocols are secure under the discrete logarithm assumption, although some implementations of (2) require the Strong RSA assumption [BP97, FO97].

When refering to the proofs above, we will use the notation of Camenisch and Stadler [CS97]. For instance, $PoK\{(x, r) : y = g^x h^r \wedge (1 \leq x \leq n)\}$ denotes a zero-knowledge proof of knowledge of integers x and r such that $y = g^x h^r$ holds and $1 \leq x \leq n$. All values not in enclosed in ()'s are assumed to be known to the verifier. We can apply the Fiat-Shamir heuristic [FS86] to make such proofs non-interactive in the random oracle model.

Commitments. Let $(\mathsf{Commit}, \mathsf{Decommit})$ be a commitment scheme where on input a message M, $\mathsf{Commit}(M)$ outputs a commitment/decommitment pair $(\mathcal{C}, \mathcal{D})$, and $\mathsf{Decommit}(M, \mathcal{C}, \mathcal{D})$ outputs 1, or 0 if $\mathcal{D}$ is invalid with respect to $(M, \mathcal{C})$. Our constructions make use of a secure commitment scheme supporting an efficient protocol for proving knowledge of a valid decommitment with respect to $(\mathcal{C}, M)$. We recommend using the Pedersen commitment scheme [Ped92] based on the discrete logarithm assumption, in which the public parameters are a group of prime order q, and random generators $(g_0, \ldots, g_m)$. In order to commit to the values $(v_1, \ldots, v_m) \in \mathbb{Z}_q^m$, pick a random $r \in \mathbb{Z}_q$ and set $\mathcal{C} = g_0^r \prod_{i=1}^m g_i^{v_i}$ and $\mathcal{D} = r$. We can use the technique of Schnorr [Sch91] to efficiently prove knowledge of the decommitment value r.

3 Blind Identity-Based Encryption

An identity-based encryption (IBE) scheme supports two types of players: a single master authority and multiple users; together with the algorithms Setup, Encrypt, Decrypt and the protocol Extract. Let us provide some input/output specification for these protocols with intuition for what they do.

Notation: Let $\mathcal{I}$ be the identity space and $\mathcal{M}$ be the message space. We write $P(\mathcal{A}(a), \mathcal{B}(b)) \to (c, d)$ to indicate that protocol P is between parties $\mathcal{A}$ and $\mathcal{B}$, where a is $\mathcal{A}$'s input and c is $\mathcal{A}$'s output and b is $\mathcal{B}$'s input and d is $\mathcal{B}$'s output.

- In the $\mathsf{Setup}(1^\kappa, c(\kappa))$ algorithm, on input a security parameter 1^κ and a description of an the identity space $|\mathcal{I}| \leq 2^{c(\kappa)}$ where $c(\cdot)$ is a computable, polynomially-bounded function, the master authority $\mathcal{P}$ outputs master parameters and a master secret key $(params, msk)$.
- In the $\mathsf{Extract}(\mathcal{P}(msk), \mathcal{U}(params, id)) \to (id, sk_{id})$ protocol, an honest user $\mathcal{U}$ with identity $id \in \mathcal{I}$ obtains the corresponding secret key sk_{id} from the master authority $\mathcal{P}$ or outputs an error message. The master authority's output is **the identity** id or an error message.
- In the $\mathsf{Encrypt}(params, id, m)$ algorithm, on input identity $id \in \mathcal{I}$ and message $m \in \mathcal{M}$, any party can output ciphertext C.
- In the $\mathsf{Decrypt}(sk_{id}, C)$ algorithm, on input a ciphertext C, the user with sk_{id} can output a message $m \in \mathcal{M}$ or an error message.

Definition 3.1 (Selective-Identity Security for IBE (IND-sID-CPA) [CHK03, CHK04]) *Let κ be a security parameter, $c(\cdot)$ be a polynomially-bounded function, $|\mathcal{I}| \leq 2^{c(\kappa)}$ and $\mathcal{M}$ be the message space. The IBE scheme is* IND-sID-CPA*-secure if every p.p.t. adversary $\mathcal{A}$ has an advantage negligible in κ for the following game: (1) $\mathcal{A}$ outputs a target identity $id^* \in \mathcal{I}$. (2) Run $\mathsf{Setup}(1^\kappa, c(\kappa))$ to obtain $(params, msk)$, and give params to $\mathcal{A}$. (3) $\mathcal{A}$ may query an oracle $O_{msk}(\cdot)$ polynomially many times, where on any input $id \neq id^*$ in $\mathcal{I}$, the oracle returns sk_{id}, and on any other input, the oracle returns an error message. (4) $\mathcal{A}$ outputs two messages $m_0, m_1 \in \mathcal{M}$ where $|m_0| = |m_1|$. Select a random bit b and give $\mathcal{A}$ the challenge ciphertext $c^* \leftarrow \mathsf{Encrypt}(params, id^*, m_b)$. (5) $\mathcal{A}$ may continue to query oracle $O_{msk}(\cdot)$ under the same conditions as before. (6) $\mathcal{A}$ outputs $b' \in \{0, 1\}$. This is the end of the game. We define $\mathcal{A}$'s advantage in the above game as $|\Pr[b' = b] - 1/2|$.*

On stronger notions of ciphertext security for IBE. A stronger notion of ciphertext security for IBE schemes is adaptive-identity ("full") security (IND-ID-CPA) [BF01]. This notion strengthens the IND-sID-CPA definition by allowing $\mathcal{A}$ to select the target identity id^* at the beginning of step (4) in the above game. Later in this section, we will show Blind IBE constructions satisfying both IND-sID-CPA and IND-ID-CPA security. Fortunately, our oblivious transfer applications in §4 require only the weaker notion: IND-sID-CPA-security (because the "identities" will be fixed integers from 1 to $\text{poly}(\kappa)$), some additional applications in §5 require the stronger IND-ID-CPA-security.

Blind IBE. So far, we have only described traditional IBE schemes. A *blind IBE* scheme consists of the same players, together with the same algorithms Setup, Encrypt, Decrypt and yet we replace the protocol Extract with a new protocol BlindExtract which differs only in the authority's output:

- In the $\mathsf{BlindExtract}(\mathcal{P}(msk), \mathcal{U}(params, id)) \to (\text{nothing}, sk_{id})$ protocol, an honest user $\mathcal{U}$ with identity $id \in \mathcal{I}$ obtains the corresponding secret key sk_{id} from the master authority $\mathcal{P}$ or outputs an error message. The master authority's output is **nothing** or an error message.

We now define security for blind IBE, which informally is any IND-sID-CPA-secure IBE scheme with a BlindExtract protocol that satisfies two properties:

1. **Leak freeness:** a potentially malicious user cannot learn anything by executing the BlindExtract protocol with an honest authority which she could not have learned by executing the Extract protocol with an honest authority; moreover, as in Extract, the user must know the identity for which it is extracting a key.
2. **Selective-failure blindness:** a potentially malicious authority cannot learn anything about the user's choice of identity during the BlindExtract protocol; moreover, the authority cannot cause the BlindExtract protocol to fail in a manner dependent on the user's choice.

Of course, a protocol realizing the functionality $\mathsf{BlindExtract}(\mathcal{P}(msk),\mathcal{U}(params, id)) \rightarrow (\text{nothing}, sk_{id})$ (in a fashion that satisfies the properties above) is a special case of secure two-party computation [Yao86, GMW87, Kil88], as are blind signature protocols, e.g. [JLO97, Oka06]. In fact, recall that sk_{id} in a fully-secure IBE can be viewed as a signature by the authority on message id (see §5). Thus, our BlindExtract protocol (for full-security IBE) *is* a blind signature scheme, but the converse implication is not necessarily true. Our leak freeness property is much stronger than the common *one-more unforgeability* requirement of blind signatures. Moreover, we will not require full-security for the IBE in our OT applications. Let us now formally state these properties.

Definition 3.2 (Leak Freeness) *A protocol* $\mathsf{BlindExtract} = (\mathcal{P},\mathcal{U})$ *associated with an IBE scheme* $\Pi = (\mathsf{Setup}, \mathsf{Extract}, \mathsf{Encrypt}, \mathsf{Decrypt})$ *is* leak free *if for all efficient adversaries* $\mathcal{A}$*, there exists an efficient simulator* $\mathcal{S}$ *such that for every value* κ *and polynomial* $c(\cdot)$*, no efficient distinguisher* D *can distinguish whether* $\mathcal{A}$ *is playing Game Real or Game Ideal with non-negligible advantage:*

Game Real: *Run* $(params, msk) \leftarrow \mathsf{Setup}(1^\kappa, c(\kappa))$*. As many times as* D *wants,* $\mathcal{A}$ *chooses an identity* id *and executes the* BlindExtract *protocol with* $\mathcal{P}$*:* $\mathsf{BlindExtract}(\mathcal{P}(msk), \mathcal{A}(params, id))$.

Game Ideal: *Run* $(params, msk) \leftarrow \mathsf{Setup}(1^\kappa, c(\kappa))$*. As many times as* D *wants,* $\mathcal{S}$ *chooses an identity* id *and queries a trusted party to obtain the output of* $\mathsf{Extract}(msk, id)$*, if* $id \in \mathcal{I}$ *and* $\perp$ *otherwise.*

Here D *and* $\mathcal{A}$ *(or* $\mathcal{S}$*) may communicate at any time. Further, params defines the identity space* $\mathcal{I}$*.*

This definition implies that the identity id (for the key being extracted) is *extractable* from the BlindExtract protocol, since $\mathcal{S}$ must be able to interact with $\mathcal{A}$ to learn which identities to submit to the trusted party. We will make use of this observation later. Another nice property of this definition is that any key extraction protocol with leak-freeness (regardless of whether blindness holds or not) composes into the existing security definitions for IBE. (This would not necessarily be true of a blind signature protocol for the same type of signatures.) We state this formally below.

Lemma 3.3 *If* $\Pi = (\mathsf{Setup}, \mathsf{Extract}, \mathsf{Encrypt}, \mathsf{Decrypt})$ *is an* IND-sID-CPA*-secure IBE scheme and* BlindExtract *is a leak-free protocol, then* $\Pi' = (\mathsf{Setup}, \mathsf{BlindExtract}, \mathsf{Encrypt}, \mathsf{Decrypt})$ *is an* IND-sID-CPA*-secure IBE scheme.*

Next, we must define the second property of *blindness*, guaranteeing that a potentially malicious authority remain blind as to the user's identity choice during the BlindExtract protocol. We use the strongest notion of blindness available called *selective-failure blindness* proposed recently by Camenisch, Neven and shelat [CNS07]. This definition ensures that even a malicious authority is unable to induce BlindExtract protocol failures that are dependent on the identity being extracted.

Definition 3.4 (Selective-Failure Blindness (SFB) [CNS07]) *A protocol $P(\mathcal{A}(\cdot),\mathcal{U}(\cdot,\cdot))$ is said to be selective-failure blind if every p.p.t. adversary $\mathcal{A}$ has a negligible advantage in the following game: First, $\mathcal{A}$ outputs params and a pair of identities $id_0, id_1 \in \mathcal{I}$. A random $b \in \{0,1\}$ is chosen. $\mathcal{A}$ is given black-box access to two oracles $\mathcal{U}(params, id_b)$ and $\mathcal{U}(params, id_{b-1})$. The $\mathcal{U}$ algorithms produce local output sk_b and sk_{b-1} respectively. If $sk_b \neq \perp$ and $sk_{b-1} \neq \perp$ then $\mathcal{A}$ receives (sk_0, sk_1). If $sk_b = \perp$ and $sk_{b-1} \neq \perp$ then $\mathcal{A}$ receives $(\perp, \varepsilon)$. If $sk_b \neq \perp$ and $sk_{b-1} = \perp$ then $\mathcal{A}$ receives $(\varepsilon, \perp)$. If $sk_b = \perp$ and $sk_{b-1} = \perp$ then $\mathcal{A}$ receives $(\perp, \perp)$. Finally, $\mathcal{A}$ outputs its guess b'. We define $\mathcal{A}$'s advantage in the above game as $|\Pr[b' = b] - 1/2|$.*

We thus arrive at our final definition.

Definition 3.5 (Secure Blind IBE) *A blind IBE scheme* Π = (Setup, BlindExtract, Encrypt, Decrypt) *is called* IND-sID-CPA-secure *(resp.* IND-ID-CPA*) if and only if: (1)* Π *is* IND-sID-CPA*-secure (resp.* IND-ID-CPA*), and (2)* BlindExtract *is leak free and selective-failure blind.*

3.1 IBE Schemes with Efficient BlindExtract Protocols

In this section, we describe efficient BlindExtract protocols for: (1) the IND-sID-CPA-secure IBE due to Boneh and Boyen [BB04a] and (2) the IND-ID-CPA-secure IBE due to Naccache [Nac05] which is a generalized version of Waters IBE [Wat05]. Indeed, these two schemes share a similar structure, so we'll begin by describing their common elements.

Setup$(1^\kappa, c(k))$: Let $\gamma = (q, g, \mathbb{G}, \mathbb{G}_T, e)$ be the output of BMsetup(1^κ). Choose random elements $h, g_2 \in \mathbb{G}$ and a random value $\alpha \in \mathbb{Z}_q$. Set $g_1 = g^\alpha$. Finally, select a function $F : \mathcal{I} \to \mathbb{G}$ that maps identities to group elements. (The descriptions of F and $\mathcal{I}$ will be defined specific to the schemes below.) Output $params = (g, g_1, g_2, h, F)$ and $msk = g_2^\alpha$.

Later, we will perform proofs of the form $PoK\{(msk) : (params, msk) \in \text{Setup}(1^\kappa, c(\kappa))\}$. For all the above IBE schemes, this proof can be done as: $PoK\{(msk) : e(g_1, g_2) = e(g, msk)\}$.

For these schemes, identity secret keys are of the form: $sk_{id} = (d_0, d_1) = (g_2^\alpha \cdot F(id)^r, g^r)$, where $r \in \mathbb{Z}_q$ is randomly chosen by the master authority. Note that given a secret key $sk_{id} = (d_0, d_1)$ for identity id, one can check that it is correct by verifying: $e(d_0, g)/e(d_1, F(id)) = e(g_1, g_2)$.

Encrypt$(params, id, M)$: Given an identity $id \in \mathcal{I}$, and a message $M \in \mathbb{G}_T$, select a random value $s \in \mathbb{Z}_q$ and output the ciphertext $C = (e(g_1, g_2)^s \cdot M, g^s, F(id)^s)$.

Decrypt$(params, sk_{id}, c_{id})$: On input a decryption key $sk_{id} = (d_0, d_1) \in \mathbb{G}^2$ and a ciphertext $C = (A, B, C) \in \mathbb{G}_T \times \mathbb{G}^2$, output $M = A \cdot e(C, d_1)/e(B, d_0)$.

Next, we'll describe the format of the secret keys sk_{id} and corresponding protocol BlindExtract.

3.1.1 A BlindExtract Protocol for the IND-sID-CPA-Secure Boneh-Boyen IBE

In the Boneh-Boyen IBE [BB04a], $\mathcal{I} \subseteq \mathbb{Z}_q$ and the function $F : \mathcal{I} \to \mathbb{G}$ is defined as $F(id) = h \cdot g_1^{id}$. A secret key for identity id is defined as follows, where $r \in \mathbb{Z}_q$ is chosen randomly by the authority:

$$sk_{id} = (d_0, d_1) = (g_2^\alpha \cdot F(id)^r, g^r) = (g_2^\alpha \cdot (h \cdot g_1^{id})^r, g^r).$$

The protocol BlindExtract($\mathcal{P}(msk)$,$\mathcal{U}(params, id)$) is described in Figure 1. Recall that $\mathcal{U}$ wants to obtain sk_{id} without revealing id, and $\mathcal{P}$ wants to reveal no more than sk_{id}. Let Π_1 be the blind IBE that combines algorithms Setup, Encrypt, Decrypt with the protocol BlindExtract in Figure 1.

$\underline{\mathcal{U}(params, id)}$	$\underline{\mathcal{P}(msk)}$
1. Choose $y \xleftarrow{\$} \mathbb{Z}_q$.	
2. Compute $h' \leftarrow g^y g_1^{id}$ and send h' to $\mathcal{P}$.	
3. Execute $PoK\{(y, id) : h' = g^y g_1^{id}\}$.	
	4. If the proof fails to verify, abort.
	5. Choose $r \xleftarrow{\$} \mathbb{Z}_q$.
	6. Compute $d_0' \leftarrow g_2^{\alpha} \cdot (h'h)^r$.
	7. Compute $d_1' \leftarrow g^r$.
	8. Send (d_0', d_1') to $\mathcal{U}$.
9. Check that $e(g_1, g_2) \cdot e(d_1', h \cdot h) = e(d_0', g)$.	
10. If the check passes, choose $z \xleftarrow{\$} \mathbb{Z}_q$; otherwise, output $\perp$ and abort.	
11. Compute $d_0 \leftarrow (d_0'/(d_1')^y) \cdot F(id)^z$ and $d_1 \leftarrow d_1' \cdot g^z$.	
12. Output $sk_{id} = (d_0, d_1)$.	

Figure 1: A BlindExtract protocol for the Boneh-Boyen IBE above. To form a BlindExtract-protocol for the Waters-Naccache IBE, make the following alterations. Parse the identity as $id = (a_1, \ldots, a_n)$, where each a_i is ℓ bits. In line 2, compute h' as $g^y \cdot \prod_{j=1}^{n} u_j^{a_j}$. In line 3, execute the proof $PoK\{(y, a_1, \ldots, a_n) : h' = g^y \cdot \prod_{j=1}^{n} u_j^{a_j} \wedge 0 \leq a_i < 2^{\ell}$, for $i = 1$ to $n\}$.

Theorem 3.6 *Under the Decisional Bilinear Diffie-Hellman assumption, blind IBE* Π_1 *is secure (according to Definition 3.5); i.e., protocol* BlindExtract *is both leak-free and selective-failure blind.*

See Appendix A for proof of Theorem 3.6.

3.1.2 A BlindExtract Protocol for the IND-ID-CPA-Secure Waters-Naccache IBE

In the Naccache IBE [Nac05] which is a generalized version of the Waters IBE [Wat05], the identity space $\mathcal{I}$ is the set of bit strings of length N, where N is polynomial in κ, represented by n blocks of ℓ bits each. The function $F : \{0,1\}^N \rightarrow \mathbb{G}$ is defined as $F(id) = h \cdot \prod_{j=1}^{n} u_j^{a_j}$, where each $u_j \in \mathbb{G}$ is randomly selected by the master authority and each a_j is an ℓ-bit segment of id. Naccache discusses practical IBE deployment with $N = 160$ and $\ell = 32$ [Nac05]. A secret key for identity id is defined as follows, where $r \in \mathbb{Z}_q$ is chosen randomly by the authority:

$$sk_{id} = (d_0, d_1) = (g_2^{\alpha} \cdot F(id)^r, g^r) = (g_2^{\alpha} \cdot (h \cdot \prod_{j=1}^{n} u_j^{a_j})^r, g^r).$$

The protocol BlindExtract($\mathcal{P}(msk)$,$\mathcal{U}(params, id)$) is described in Figure 1, with the following alterations. Parse the identity as $id = (a_1, \ldots, a_n)$, where each a_i is ℓ bits. In line 2, compute h' as $g^y \cdot \prod_{j=1}^{n} u_j^{a_j}$. In line 3, execute the proof $PoK\{(y, a_1, \ldots, a_n) : h' = g^y \cdot \prod_{j=1}^{n} u_j^{a_j} \wedge 0 \leq a_i < 2^{\ell}$, for $i = 1$ to $n\}$. The range part of this proof (e.g., $0 \leq a_i < 2^{\ell}$) can be performed exactly or, by shortening each a_i by a few bits, can be done at almost no additional cost [CFT98, CM99, Bou00]. Follow the rest of the protocol as is. Let Π_2 be the blind IBE that combines algorithms Setup, Encrypt, Decrypt with the protocol BlindExtract described above.

Theorem 3.7 *Under the Decisional Bilinear Diffie-Hellman assumption, blind IBE* Π_2 *is secure (according to Definition 3.5); i.e., protocol* BlindExtract *is both leak-free and selective-failure blind.*

See Appendix A for proof of Theorem 3.7.

3.1.3 On Other IBEs and HIBEs

Let us briefly summarize what we know about efficient BlindExtract-protocols for other IBE schemes and hierarchical IBE (HIBE) schemes. First, the well-known random oracle based IBEs [BF01, Coc01] actually seem less suited to developing efficient BlindExtract-protocols than their standard model successors. This is in part due to the fact that the identity string is hashed into an element in $\mathbb{G}$ in these schemes, instead of represented as an integer exponent, which makes our proof of knowledge techniques unwieldy. We were not able to find BlindExtract-protocols for the Boneh and Franklin [BF01], Cocks [Coc01], or the recent Boneh-Gentry-Hamburg [BGH07] IBEs with running time better than $O(|\mathcal{I}|)$, where $\mathcal{I}$ is the identity space. A suitable BlindExtract-protocol for the Cocks or Boneh-Gentry-Hamburg IBE would be particularly interesting, because then our generic OT constructions (in the full-simulation model) could be implemented without bilinear maps.

Boneh and Boyen [BB04a] and Waters [Wat05] proposed HIBEs. Chatterjee and Sarkar [CS06] described the HIBE version of Naccache's optimizations to Waters scheme [Nac05]. For all of these HIBEs, the number of elements comprising a user's secret key grow with the depth of the hierarchy, but each piece is similar in format to the original keys and our same techniques would apply.

4 Oblivious Transfer from Blind IBE

We now construct efficient and *fully-simulatable* oblivious transfer protocols generically from blind IBE. In particular, we focus on (non-adaptive) OT_k^N and (adaptive) $\mathsf{OT}_{k\times 1}^N$ protocols, in which a Sender and Receiver transfer up to k messages out of an N-message set. In the non-adaptive model [BCR86, NP99a], the Receiver obtains all k messages simultaneously. In the adaptive model [NP99b] the Receiver may request the messages one at a time. Intuitively, the Receiver should learn only the messages it selects (and nothing about the remaining messages), while the Sender should gain no information about *which* messages the Receiver selected.

Full-simulation vs. half-simulation security. Security for oblivious transfer is defined via simulation. Informally, a protocol is secure if, for every real-world cheating Sender (resp. Receiver) we can describe an ideal-world counterpart who gains as much information from the ideal-world interaction as from the real protocol. Surprisingly, much of the oblivious transfer literature uses the simulation-based definition only to show *Sender* security, choosing to define Receiver security by a simpler game-based definition. Naor and Pinkas demonstrated that this weaker "half-simulation" approach permits *selective-failure* attacks, in which a malicious Sender induces transfer failures that are dependent on the message that the Receiver requests [NP99b]. Unfortunately, most practical OT protocols have only been proven secure in the half simulation model. Recently, Camenisch et al. [CNS07] proposed several practical $\mathsf{OT}_{k\times 1}^N$ protocols that are secure under a stronger "full-simulation" definition, using a variety of adaptive (e.g., q-PDDH) or interactive (e.g., one-more-inversion RSA) complexity assumptions. We now enhance their results by demonstrating efficient full-simulation OT_k^N and $\mathsf{OT}_{k\times 1}^N$ protocols secure under static complexity assumptions (e.g., DBDH).

4.1 Definitions

We now provide formal definitions for both the non-adaptive and adaptive protocols. For consistency with earlier work, we adopt some notation due to Camenisch *et al.* [CNS07].

Definition 4.1 (k-out-of-N Oblivious Transfer (OT_k^N, $\mathsf{OT}_{k\times 1}^N$)) *We generalize an* OT *scheme as a tuple of algorithms* $(\mathsf{S_I}, \mathsf{R_I}, \mathsf{S_T}, \mathsf{R_T})$. *These algorithms are used in matched pairs. During the initialization phase the Sender runs* $\mathsf{S_I}(M_1, \ldots, M_N)$ *to obtain state value* S_0, *and the Receiver runs* $\mathsf{R_I}()$ *to obtain state value* R_0. *The Sender and Receiver execute* $\mathsf{S_T}, \mathsf{R_T}$ k *times as described below.*

Adaptive OT. *In the adaptive* $\mathsf{OT}_{k\times 1}^N$ *case, for* $1 \leq i \leq k$, *the* i^{th} *transfer proceeds as follows: the Sender runs* $\mathsf{S_T}(S_{i-1})$ *to obtain state value* S_i, *and the Receiver runs* $\mathsf{R_T}(R_{i-1}, \sigma_i)$ *where* $1 \leq \sigma_i \leq N$ *is the index of the message to be received. This produces state information* R_i *and the message* M_{σ_i} *or* $\perp$ *indicating failure.*

Non-adaptive OT. *In the non-adaptive* OT_k^N *case the parties execute the protocol as above; however, for round* $i < k$ *the algorithm* $\mathsf{R_T}(R_{i-1}, \sigma_i)$ **does not** *output a message. At the end of the the* k^{th} *transfer* $\mathsf{R_T}(R_{k-1}, \sigma_k)$ *outputs the messages* $(M'_{\sigma_1}, \ldots, M'_{\sigma_k})$ *where for* $j = 1, \ldots, N$ *each* $M'_{\sigma_j} = M_{\sigma_j}$ *or* $\perp$. *(Note that in a non-adaptive scheme, the initialization and* k *transfers do not necessarily require a corresponding number of communication rounds.)*

Definition 4.2 (Full Simulation Security.) *Security for oblivious transfer is defined according to a simulation-based definition.*

Real experiment. *In experiment* $\mathbf{Real}_{\hat{\mathsf{S}},\hat{\mathsf{R}}}(N, k, M_1, \ldots, M_N, \Sigma)$ *the possibly cheating sender* $\hat{\mathsf{S}}$ *is given messages* $(M_1, \ldots, M_N)$ *as input and interacts with possibly cheating receiver* $\hat{\mathsf{R}}(\Sigma)$, *where* Σ *is a selection algorithm that on input messages* $(M_{\sigma_1}, \ldots, M_{\sigma_{i-1}})$ *outputs the index* σ_i *of the next message to be queried. At the beginning of the experiment, both* $\hat{\mathsf{S}}$ *and* $\hat{\mathsf{R}}$ *output initial states* (S_0, R_0). *In the adaptive case, for* $1 \leq i \leq k$ *the sender computes* $S_i \leftarrow \hat{\mathsf{S}}(S_{i-1})$, *and the receiver computes* $(R_i, M'_i) \leftarrow \hat{\mathsf{R}}(R_{i-1})$, *where* M'_i *may or may not be equal to* M_i. *In the non-adaptive case, the Receiver obtains no messages until the* k^{th} *round, and therefore the selection strategy* Σ *must be non-adaptive. At the end of the* k^{th} *transfer the output of the experiment is* (S_k, R_k).

Ideal experiment. *In experiment* $\mathbf{Ideal}_{\hat{\mathsf{S}}',\hat{\mathsf{R}}'}(N, k, M_1, \ldots, M_N, \Sigma)$ *the possibly cheating sender algorithm* $\hat{\mathsf{S}}'$ *generates messages* $(M_1^*, \ldots, M_N^*)$ *and transmits them to a trusted party* T. *In the* i^{th} *round* $\hat{\mathsf{S}}'$ *sends a bit* b_i *to* T; *the possibly cheating receiver* $\hat{\mathsf{R}}'(\Sigma)$ *transmits* σ_i^* *to* T. *In the adaptive case, if* $b_i = 1$ *and* $\sigma_i^* \in (1, \ldots, N)$ *then* T *hands* $M_{\sigma_i^*}$ *to* $\hat{\mathsf{R}}'$. *If* $b_i = 0$ *then* T *hands* $\perp$ *to* $\hat{\mathsf{R}}'$. *Note that in the non-adaptive case,* T *does not give* $\hat{\mathsf{R}}'$ *any response until the* k^{th} *round. At the end of the* k^{th} *transfer the output of the experiment is* (S_k, R_k).

Sender Security. $\mathsf{OT}_{k\times 1}^N$ *provides Sender security if for every real-world p.p.t. receiver* $\hat{\mathsf{R}}$ *there exists an ideal-world receiver* $\hat{\mathsf{R}}'$ *such that* $\forall N = \ell(\kappa)$, $k \in \{1, \ldots, N\}$, $(M_1, \ldots, M_N)$, Σ, *and every p.p.t. distinguisher:* $\mathbf{Real}_{\mathsf{S},\hat{\mathsf{R}}}(N, k, M_1, \ldots, M_N, \Sigma) \overset{c}{\approx} \mathbf{Ideal}_{\mathsf{S}',\hat{\mathsf{R}}'}(N, k, M_1, \ldots, M_N, \Sigma)$.

Receiver Security. $\mathsf{OT}_{k\times 1}^N$ *provides Receiver security if for every real-world p.p.t. sender* $\hat{\mathsf{S}}$ *there exists an ideal-world sender* $\hat{\mathsf{S}}'$ *such that* $\forall N = \ell(\kappa)$, $k \in \{1, \ldots, N\}$, $(M_1, \ldots, M_N)$, Σ, *and every p.p.t. distinguisher:* $\mathbf{Real}_{\hat{\mathsf{S}},\mathsf{R}}(N, k, M_1, \ldots, M_N, \Sigma) \overset{c}{\approx} \mathbf{Ideal}_{\hat{\mathsf{S}}',\mathsf{R}'}(N, k, M_1, \ldots, M_N, \Sigma)$.

$\underline{\mathsf{S}_\mathsf{I}(M_1,\ldots,M_N),\mathsf{S}_\mathsf{T}()}$	$\underline{\mathsf{R}_\mathsf{I}(),\mathsf{R}_\mathsf{T}(\sigma_1,\ldots,\sigma_k)}$
Sender and Receiver agree on parameters for a commitment scheme.[a]	
1. Select $(params, msk) \leftarrow \mathsf{Setup}(1^\kappa, c(\kappa))$ and a collision-resistant hash function H.	
2. For $j = 1,\ldots,N$ set $C_j \leftarrow \mathsf{Encrypt}(params, j, M_j)$.	
3. Compute $(\mathcal{C},\mathcal{D}) \leftarrow \mathsf{Commit}\,(H(C_1,\ldots,C_N))$.	
4. Send $(params,\mathcal{C})$ to Receiver.	
5. Execute $PoK\{msk : (params, msk) \in \mathsf{Setup}(1^\kappa, c(\kappa))\}$.	
	6. If the proof fails to verify, abort.
For $i = 1,\ldots,k$, Sender and Receiver execute $\mathsf{BlindExtract}$ for identity σ_i.	
After the k^{th} execution:	
1. Send the ciphertexts $(C_1,\ldots,C_N)$.	
2. Conduct $PoK\{\mathcal{D} : \mathsf{Decommit}\,(H(C_1,\ldots,C_N),\mathcal{D}) = 1\}$.	
	4. If the proof fails to verify, abort.
	5. For $j = 1,\ldots,k$: $M_{\sigma_j} \leftarrow \mathsf{Decrypt}(sk_{\sigma_j}, C_{\sigma_j})$ or $M_{\sigma_j} = \perp$ if $\mathsf{BlindExtract}$ failed.
Output $S_i = S_{i-1}$	Output $R_i = (R_{i-1}, sk_{\sigma_i})$ and any messages.

[a] In the case of a computationally-binding scheme such as Pedersen's, these parameters may be generated by the Receiver. Alternatively, they may be common shared parameters.

Figure 2: A non-adaptive OT_k^N protocol from blind IBE. We present the $\mathsf{S}_\mathsf{I}, \mathsf{R}_\mathsf{I}, \mathsf{S}_\mathsf{T}, \mathsf{R}_\mathsf{T}$ algorithms in a single protocol flow.

4.1.1 Non-adaptive OT_k^N without Random Oracles

Given a blind IBE scheme Π, it is tempting to consider the following "intuitive" protocol: First, the Sender runs the IBE Setup algorithm and sends *params* to the Receiver. Next, for $i = 1,\ldots,N$ the Sender transmits an encryption of message M_i under identity "i". To obtain k messages, the Receiver extracts decryption keys for identities $(\sigma_1,\ldots,\sigma_k)$ via k distinct executions of $\mathsf{BlindExtract}$, and uses these keys to decrypt the corresponding ciphertexts. If Π is a blind IBE scheme secure in the sense of definition 3.5, then intuitively a cheating Receiver gains no information about the messages corresponding to secret keys he did not extract. Similarly, a cheating Sender does not learn the identities extracted. While we are not aware of any practical attack on this protocol, we do not know how to prove it secure in the full simulation model.

Fortunately, we are able to convert this intuition into the fully-simulatable OT_k^N protocol shown in Figure 2. We require only the following modifications: first, we have the Sender prove knowledge of the value *msk* using appropriate zero-knowledge techniques.[1] Then, rather than transmitting the ciphertext vector during the first phase of the protocol, the Sender transmits only a *commitment* to a collision-resistant hash of the ciphertext vector, and sends the actual ciphertexts at the end of the k^{th} round, along with a proof that she can open the commitment to the hash of the ciphertexts.

Theorem 4.3 (Full Security of OT_k^N Scheme) *If Π is a secure blind IBE scheme in the sense*

[1] In §3.1, we describe how these proofs would work for the practical blind IBE constructions we consider.

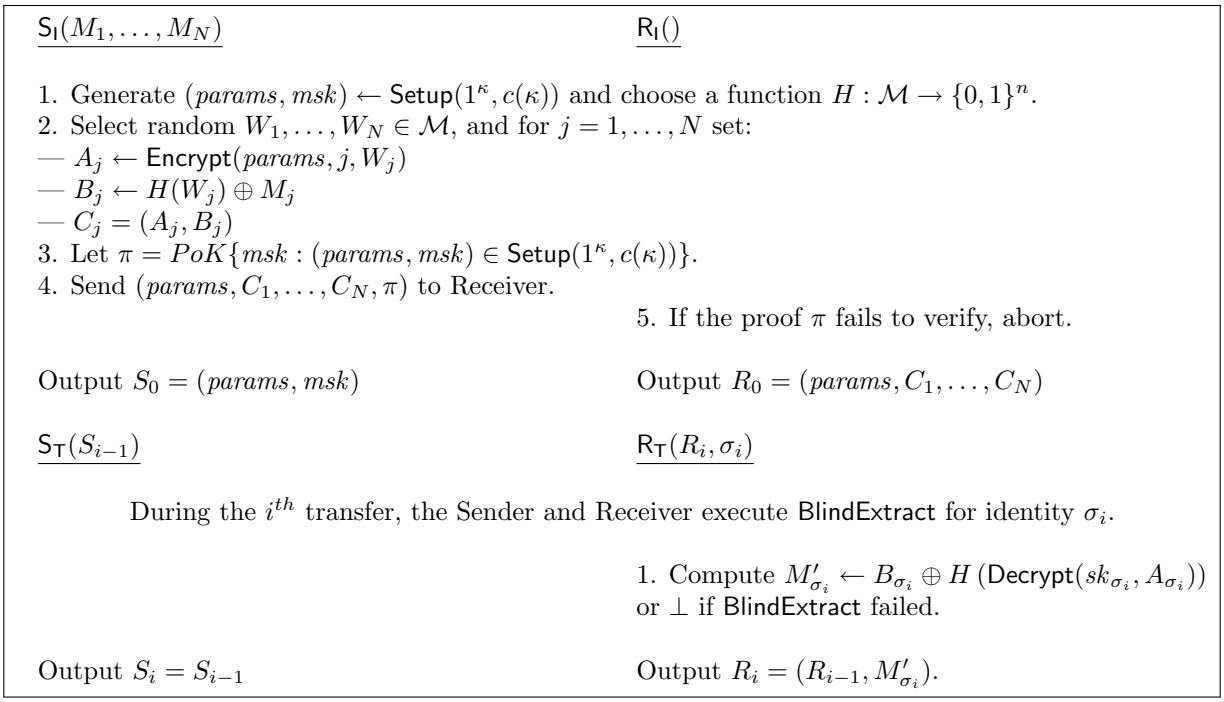

Figure 3: Adaptive $\mathsf{OT}^N_{k\times 1}$ from blind IBE secure in the random oracle model, with $M_1,\ldots,M_N \in \{0,1\}^n$.

of definition 3.5, and $(\mathsf{Commit}, \mathsf{Decommit})$ *is a secure commitment scheme, then the* OT^N_k *protocol of figure 2 is sender-secure and receiver-secure in the full-simulation model.*

We present a full proof of Theorem 4.3 in Appendix B. When Π is either of the blind IBE schemes presented in §3.1, then our OT protocol is secure under DBDH.

4.1.2 Adaptive $\mathsf{OT}^N_{k\times 1}$ in the Random Oracle Model

While our first protocol is efficient and full-simulation secure, it permits only *non-adaptive* queries. For many practical applications (e.g., oblivious retrieval from a large database), we desire a protocol that supports an adaptive query pattern. We approach this goal by first proposing an efficient $\mathsf{OT}^N_{k\times 1}$ protocol secure in the random oracle model. The protocol, which we present in Figure 3, requires an IBE scheme with a super-polynomial message space (as in the constructions of §3.1), and has approximately the same efficiency as the construction with random oracles of Camenisch et al. [CNS07]. However, their construction requires unique blind signatures and the two known options due to Chaum [Cha82] and Boldyreva [Bol03] both require interactive complexity assumptions. By using the blind IBE schemes in §3.1, our protocols can be based on the DBDH assumption.

Theorem 4.4 (Full Security of $\mathsf{OT}^N_{k\times 1}$ Scheme) *If* Π *is a secure blind IBE scheme in the sense of definition 3.5, then the* $\mathsf{OT}^N_{k\times 1}$ *protocol of figure 3 is sender-secure and receiver-secure in the full-simulation model and the random oracle model.*

4.1.3 Adaptive $\mathsf{OT}^N_{k\times 1}$ without Random Oracles

The random-oracle $\mathsf{OT}^N_{k\times 1}$ presented above is extremely efficient both in terms of communications cost and round-efficiency. More importantly, it is generic, i.e., can be instantiated using any secure blind IBE scheme. Ideally, we would like to construct a generic protocol of comparable efficiency in the standard model. The two approaches available to us can be summarized as follows. We can construct a generic $\mathsf{OT}^N_{k\times 1}$ by compiling k instances of the non-adaptive OT^N_k from §4.1.1. Each protocol round would consist of a 1-out-of-N instance of the protocol, with new IBE parameters and new a collection of ciphertexts $(C_1, \ldots, C_N)$. To ensure that each round is consistent with the previous rounds, the Sender would additionally prove that the underlying plaintexts remain the same from round to round. This can be achieved using standard proof techniques, but is impractical for large values of k or N.

Alternatively, we can abandon generic constructions, and instead develop an efficient protocol from scratch. Camenisch et al. [CNS07] successfully took this approach: providing efficient adaptive schemes generically in the random oracle model and one specific scheme in the standard model. Their efficient $\mathsf{OT}^N_{k\times 1}$, for example, incurs only a constant cost per transfer phase. However, the protocol relies on the dynamic q-Strong DH [BB04b] and q-Power Decisional DH assumptions, the former of which may not hold when q is sufficiently large [Che06]. Fortunately, one might be able to keep q small (on the order of k rather than N) by combining the Camenisch et al. scheme [CNS07] with ours as follows: in their initialization, the Sender releases N values corresponding to the messages that require $q = N$. This seems risky when N is very large. Instead, we could use a blind IBE scheme to encrypt these N values during initialization, and then during the adaptive transfer phase, a Receiver could request the decryption key of his choice along with the information rquired in the Camenisch et al. scheme. Thus, reducing the values available to an adversary to $q = k$.

5 Other Applications of Blind IBE

Privacy-preserving delegated keyword search. Several works use IBE as a building-block for *public-key searchable encryption* [BCOP04, WBDS04]. These schemes permit a keyholder to delegate search capability to other parties. For example, Waters et al. [WBDS04] describe a searchable encrypted audit log in which a third party auditor is granted the ability to independently search the encrypted log for specific keywords. To enable this function, a central authority generates "trapdoors" for the keywords that the auditor wishes to search on. In this approach, the trapdoor generation authority necessarily learns each of the search terms. This may be problematic in circumstances where the pattern of trapdoor requests reveals sensitive information (e.g., the name of user under suspicion). By using blind and partially-blind IBE, we permit the authority to generate trapdoors while learning no information (or only partial information) about the search terms.[2]

Blind and partially-blind signature. Naor pointed out that every fully-secure IBE scheme is implicitly an existentially unforgeable signature scheme [BF01, CFH+07]. By the same token, a fully-secure blind IBE scheme with a BlindExtract protocol implies an unforgeable, selective-failure blind signature scheme. This result applies to the fully-secure Waters-Naccache-based protocol of §3.1.2, and to the Boneh-Boyen-based protocol of §3.1.1 when that scheme is instantiated with

[2] Boneh *et al.* [BCOP04] note that keyword search schemes can be constructed from any *key anonymous* IBE scheme. While the standard Waters and Naccache schemes are not key anonymous, Waters remarks that key anonymity might be acheived by implementing his scheme in *asymmetric* bilinear groups [BW06].

appropriately-sized parameters and a hash function (see §7 of [BB04a]). The efficient BlindExtract protocol for the Waters-Naccache scheme can also be used to construct a *partially-blind* signature, by allowing the signer (PKG) to supply a portion of the input string. Partially-blind signatures have a number of practical applications, such as document timestamping and electronic cash [MS98].

Temporary anonymous identities. In a typical IBE deployment, the PKG can link each extracted identity to the user who requested it. This can be undesirable when users wish to remain anonymous or pseudonymous. By employing (partially-)blind IBE, a PKG can grant temporary credentials anonymously, without even learning which identities are in use.

Acknowledgments. We are grateful to abhi shelat for helpful discussions.

References

[BB04a] Dan Boneh and Xavier Boyen. Efficient selective-ID secure Identity-Based Encryption without random oracles. In *EUROCRYPT '04*, volume 3027 of LNCS, pages 223–238, 2004.

[BB04b] Dan Boneh and Xavier Boyen. Short signatures without random oracles. In *EUROCRYPT '04*, volume 3027 of LNCS, pages 382–400, 2004.

[BCOP04] Dan Boneh, Giovanni Di Crescenzo, Rafail Ostrovsky, and Giuseppe Persiano. Public key encryption with keyword search. In *EUROCRYPT '04*, volume 3027 of LNCS, pages 506–522, 2004.

[BCR86] Gilles Brassard, Claude Crépeau, and Jean-Marc Robert. All-or-nothing disclosure of secrets. In *CRYPTO '86*, volume 263 of LNCS, pages 234–238, 1986.

[BF01] Dan Boneh and Matthew K. Franklin. Identity-based encryption from the Weil Pairing. In *CRYPTO '01*, volume 2139 of LNCS, pages 213–229, 2001.

[BGH07] Dan Boneh, Craig Gentry, and Michael Hamburg. Space-efficient identity based encryption without pairings, 2007. Available at `http://crypto.stanford.edu/~dabo/pubs.html`.

[BM89] Mihir Bellare and Silvio Micali. Non-interactive oblivious transfer and applications. In *CRYPTO '89*, volume 435 of LNCS, pages 547–557, 1989.

[Bol03] Alexandra Boldyreva. Threshold, Multisignature and Blind Signature Schemes Based on the Gap-Diffie-Hellman-Group Signature Scheme. In *PKC '03*, volume 2139 of LNCS, pages 31–46, 2003.

[Bou00] Fabrice Boudot. Efficient proofs that a committed number lies in an interval. In *EUROCRYPT '00*, volume 1807 of LNCS, pages 431–444, 2000.

[BP97] Niko Barić and Birgit Pfitzmann. Collision-free accumulators and fail-stop signature schemes without trees. In *EUROCRYPT '97*, volume 1233 of LNCS, pages 480–494, 1997.

[BW06] Xavier Boyen and Brent Waters. Anonymous hierarchical identity-based encryption (without random oracles). In *CRYPTO '06*, volume 4117 of LNCS, pages 290–307, 2006.

[CDS94] Ronald Cramer, Ivan Damgård, and Berry Schoenmakers. Proofs of partial knowledge and simplified design of witness hiding protocols. In *CRYPTO '94*, volume 839 of LNCS, pages 174–187, 1994.

[CFH+07] Yang Cui, Eiichiro Fujisaki, Goichiro Hanaoka, Hideki Imai, and Rui Zhang. Formal security treatments for ibe-to-signature transformation: Relations among security notions. Cryptology ePrint Archive, Report 2007/030, 2007. `http://eprint.iacr.org/`.

[CFT98] Agnes Chan, Yair Frankel, and Yiannis Tsiounis. Easy come – easy go divisible cash. In *EUROCRYPT '98*, volume 1403 of LNCS, pages 561–575, 1998.

[Cha82] David Chaum. Blind signatures for untraceable payments. In *CRYPTO '82*, pages 199–203. Plenum Press, 1982.

[Che06] Jung Hee Cheon. Security analysis of the strong diffie-hellman problem. In *EUROCRYPT '06*, volume 4004 of LNCS, pages 1–11, 2006.

[CHK03] Ran Canetti, Shai Halevi, and Jonathan Katz. A forward-secure public-key encryption scheme. In *EUROCRYPT '03*, volume 3027 of LNCS, pages 255–271, 2003.

[CHK04] Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from Identity Based Encryption. In *EUROCRYPT '04*, volume 3027 of LNCS, pages 207–222, 2004.

[CM99] Jan Camenisch and Markus Michels. Proving in zero-knowledge that a number n is the product of two safe primes. In *EUROCRYPT '99*, volume 1592 of LNCS, pages 107–122, 1999.

[CNS07] Jan Camenisch, Gregory Neven, and Abhi Shelat. Simulatable adaptive oblivious transfer. In *EUROCRYPT '07*, volume 4515 of LNCS, pages 573–590, 2007.

[Coc01] Clifford Cocks. An identity based encryption scheme based on Quadratic Residues. In *Cryptography and Coding, IMA International Conference*, volume 2260 of LNCS, pages 360–363, 2001.

[CS97] Jan Camenisch and M. Stadler. Efficient group signature schemes for large groups. In *CRYPTO '97*, volume 1296 of LNCS, pages 410–424, 1997.

[CS06] Sanjit Chatterjee and Palash Sarkar. HIBE with Short Public Parameters without Random Oracle. In *ASIACRYPT '06*, volume 4284 of LNCS, pages 145–160, 2006.

[DHRS04] Yan Zong Ding, Danny Harnik, Alon Rosen, and Ronen Shaltiel. Constant-round oblivious transfer in the bounded storage model. In *TCC '04*, volume 2951 of LNCS, pages 446–472, 2004.

[EGL82] Shimon Even, Oded Goldreich, and Abraham Lempel. A randomized protocol for signing contracts. In *CRYPTO '82*, pages 205–210, 1982.

[FO97] Eiichiro Fujisaki and Tatsuaki Okamoto. Statistical zero knowledge protocols to prove modular polynomial relations. In *CRYPTO '97*, volume 1294 of LNCS, pages 16–30, 1997.

[FS86] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *CRYPTO '86*, volume 263 of LNCS, pages 186–194, 1986.

[GMW87] Oded Goldreich, Silvio Micali, and Avi Wigderson. How to play any mental game or a completeness theorem for protocols with honest majority. In *STOC '87*, pages 218–229, 1987.

[HK07] Shai Halevi and Yael Tauman Kalai. Smooth projective hashing and two-message oblivious transfer. Cryptology ePrint Archive, Report 2007/118, 2007. `http://eprint.iacr.org/`.

[JLO97] Ari Juels, Michael Luby, and Rafail Ostrovsky. Security of blind digital signatures (extended abstract). In *CRYPTO '97*, volume 1294 of LNCS, pages 150–164, 1997.

[Kal05] Yael Tauman Kalai. Smooth projective hashing and two-message oblivious transfer. In *EUROCRYPT '05*, volume 3494 of LNCS, pages 78–95, 2005.

[Kil88] Joe Kilian. Founding cryptography on oblivious transfer. In *STOC '88*, pages 20–31, 1988.

[MS98] Shingo Miyazaki and Kouichi Sakurai. A more efficient untraceable e-cash system with partially blind signatures based on the discrete logarithm problem. In *Financial Cryptography '98*, volume 1465 of LNCS, pages 296–308, 1998.

[Nac05] David Naccache. Secure and *practical* identity-based encryption. Cryptology ePrint Archive, Report 2005/369, 2005. `http://eprint.iacr.org/`.

[NP99a] Moni Naor and Benny Pinkas. Oblivious transfer and polynomial evaluation. In *STOC '99*, pages 245—254, 1999.

[NP99b] Moni Naor and Benny Pinkas. Oblivious transfer with adaptive queries. In *CRYPTO '99*, volume 1666 of LNCS, pages 573–590, 1999.

[NP01] Moni Naor and Benny Pinkas. Efficient oblivious transfer protocols. In *SODA '01*, pages 448–457, 2001.

[Oka06] Tatsuaki Okamoto. Efficient blind and partially blind signatures without random oracles. In *Theory of Cryptography (TCC)*, volume 3876 of LNCS, pages 80–99, 2006.

[Ped92] Torben Pryds Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. In *CRYPTO '92*, volume 576 of LNCS, pages 129–140, 1992.

[Rab81] Michael Rabin. How to exchange secrets by oblivious transfer. Technical Report TR-81, Aiken Computation Laboratory, Harvard University, 1981.

[Sch91] Claus-Peter Schnorr. Efficient signature generation for smart cards. *Journal of Cryptology*, 4(3):239–252, 1991.

[Sha84] Adi Shamir. Identity-based cryptosystems and signature schemes. In *CRYPTO '84*, volume 196 of LNCS, pages 47–53, 1984.

[Wat05] Brent Waters. Efficient Identity-Based Encryption without random oracles. In *EUROCRYPT '05*, volume 3494 of LNCS, pages 114–127, 2005.

[WBDS04] Brent R. Waters, Dirk Balfanz, Glenn Durfee, and D. K. Smetters. Building an encrypted and searchable audit log. In *NDSS'04*, 2004.

[Yao86] Andrew Yao. How to generate and exchange secrets. In *FOCS '86*, pages 162–167, 1986.

A Security Proofs for Blind IBE Schemes

A.1 Proof of Theorem 3.6

Proof sketch. We must show that the BlindExtract protocol in Figure 1 is both leak free and selective-failure blind. We begin with leak freeness, which requires the existence of an efficient simulator $\mathcal{S}$ such that no efficient distinguisher D can distinguish Game Real (where $\mathcal{A}$ is interacting with an honest $\mathcal{P}$ running the BlindExtract protocol) from Game Ideal (where the ideal adversary $\mathcal{S}$ is given access to a trusted party executing the ideal Extract protocol).

We describe the ideal adversary $\mathcal{S}$ as follows:

1. On input *params* from the trusted party, $\mathcal{S}$ hands *params* to a copy of $\mathcal{A}$ it runs internally.
2. Each time $\mathcal{A}$ engages $\mathcal{S}$ in a BlindExtract protocol, $\mathcal{S}$ behaves as follows. In the first message of the protocol, $\mathcal{A}$ must send to $\mathcal{S}$ a value h' and prove knowledge of values (y, id) such that $h' = g^y \cdot g_1^{id}$. If the proof fails to verify, $\mathcal{S}$ aborts. Since this proof of knowledge is implemented using the *extractable* techniques mentioned in §2, $\mathcal{S}$ can efficiently extract the values (y, id).
3. Next, $\mathcal{S}$ submits id to the trusted party, who returns the valid secret key for this identity $sk_{id} = (d_0, d_1) = (g_2^\alpha \cdot F(id)^r, g^r)$ for some random $r \in \mathbb{Z}_q$.
4. Finally, $\mathcal{S}$ computes the pair $(d_0', d_1') = (d_0 \cdot d_1^y, d_1)$ and returns these values to $\mathcal{A}$.

Observe that the responses of $\mathcal{S}$ are always correctly formed (as $\mathcal{A}$ can verify) and drawn from the same distribution as those of $\mathcal{P}$. Thus, Game Real and Game Ideal are indistinguishable to both $\mathcal{A}$ and D. We also note (as above) that the identity id being requested by $\mathcal{A}$ is efficiently *extractable* (by an extractor with special rewind capabilities not available to $\mathcal{P}$).

Next, we turn out attention to selective-failure blindness for protocol BlindExtract $= (\mathcal{P}, \mathcal{U})$. Here $\mathcal{A}$ outputs *params* and two identities $id_0, id_1 \in \mathcal{I}$. Then a random bit b is chosen. Next, $\mathcal{A}$ is given black-box access to two oracles $\mathcal{U}(params, id_b)$ and $\mathcal{U}(params, id_{b-1})$. The $\mathcal{U}$ algorithms produce local output sk_b and sk_{b-1} respectively. If $sk_b \neq \perp$ and $sk_{b-1} \neq \perp$ then $\mathcal{A}$ receives (sk_0, sk_1). If $sk_b = \perp$ and $sk_{b-1} \neq \perp$ then $\mathcal{A}$ receives $(\perp, \varepsilon)$. If $sk_b \neq \perp$ and $sk_{b-1} = \perp$ then $\mathcal{A}$ receives $(\varepsilon, \perp)$. If $sk_b = \perp$ and $sk_{b-1} = \perp$ then $\mathcal{A}$ receives $(\perp, \perp)$. $\mathcal{A}$ then tries to predict b, which we want to argue he cannot do with non-negligible advantage over guessing.

First, we observe that in this protocol, $\mathcal{U}$ speaks first and sends to $\mathcal{A}$ a value h' uniformly distributed in $\mathbb{G}$ and then performs a zero-knowledge proof of knowledge $PoK\{(y, id) : h' = g^y \cdot g_1^{id}\}$. Suppose that $\mathcal{A}$ runs one or both of his oracles up to this point. Now, it is $\mathcal{A}$'s turn to speak, and at this point, his views so far are computationally indistinguishable. $\mathcal{A}$ must now return two values $(d_0', d_1') \in \mathbb{G}^2$ to the first oracle. Suppose $\mathcal{A}$ chooses this pair using any strategy he wishes. At the point $\mathcal{A}$ fixes on two values, he is able to *predict* the output sk_i of this oracle $\mathcal{U}(params, id_i)$ as follows:

1. $\mathcal{A}$ checks if $e(g_1, g_2) \cdot e(d_1', h' \cdot h) = e(d_0', g)$ holds. If the test fails, record $sk_0 = \perp$. If the test succeeds, $\mathcal{A}$ (who chose the master keys) can temporarily record $sk_0 = \mathsf{Extract}(msk, id_0)$.
2. Next, $\mathcal{A}$ choses any two values $(d_0', d_1') \in \mathbb{G}^2$ for the second oracle, and performs the same check and recording above for sk_1 and id_1 respectively.
3. Finally, if both tests failed or both tests succeeded, output (sk_0, sk_1). If $sk_0 = \perp$ and $sk_1 \neq \perp$, output $(\perp, \varepsilon)$. If $sk_0 \neq \perp$ and $sk_1 = \perp$, output $(\varepsilon, \perp)$.

This prediction is correct, because $\mathcal{A}$ is performing the same check as the honest $\mathcal{U}$, and when both tests succeed, outputting a valid secret key randomly drawn from the set $\mathsf{Extract}(msk, id)$, as does $\mathcal{U}$. But at a higher-level, note that if $\mathcal{A}$ is able to predict the final output of its oracles accurately, then $\mathcal{A}$'s advantage in distinguishing $\mathcal{U}(params, id_b)$ and $\mathcal{U}(params, id_{b-1})$ is the same without this final output. Thus, all of $\mathcal{A}$'s advantage must come from distinguishing the earlier messages of the oracles. Since these oracles only send one uniformly random value $h' \in \mathbb{G}$ and then perform a zero-knowledge proof of knowledge about the representation of h' with respect to public values, we know from the security of the underlying proof that $\mathcal{A}$ cannot distinguish between them with non-negligible probability. □

A.2 Proof of Theorem 3.7

Proof sketch. This proof follows the outline of the proof of Theorem 3.6 almost identically. To satisfy leak freeness, the simulator $\mathcal{S}$ operates exactly as before: starting up an internal copy of $\mathcal{A}$ in step (1), extracting the values (y, id) from $\mathcal{A}$ in step (2), querying the trusted party for $sk_{id} = (d_0, d_1) \leftarrow \mathsf{Extract}(msk, id)$ in step (3), and returning the pair $(d_0', d_1') = (d_0 \cdot d_1^y, d_1)$ to $\mathcal{A}$ in step (4). Although the internal structure of the secret keys in the Naccache-Waters IBE differ from those of the Boneh-Boyen IBE, the key observation here is that $\mathcal{S}$ doesn't need to know anything about this structure to compute the correct response in step (4).

To satisfy selective-failure blindness, we first observe that the prediction of $\mathcal{U}$'s final output is done exactly as before. Thus, $\mathcal{A}$ must be able to distinguish the oracles after seeing only a value h' again uniformly distributed in $\mathbb{G}$ and a zero-knowledge proof of knowledge about the representation of h' with respect to public values. We conclude that this advantage must be negligible. □

B Proof of Security for OT_k^N

Proof of Sender Security (Theorem 4.3). For any real-world cheating receiver $\hat{\mathsf{R}}$ we can construct an ideal-world receiver $\hat{\mathsf{R}}'$ such that no p.p.t. algorithm D can distinguish the distributions $\mathbf{Real}_{\mathsf{S},\hat{\mathsf{R}}}(N, k, M_1, \ldots, M_N, \Sigma)$ and $\mathbf{Ideal}_{\mathsf{S}',\hat{\mathsf{R}}'}(N, k, M_1, \ldots, M_N, \Sigma)$. As in [CNS07] we arrive at the ideal-world receiver via a series of games. Let $\mathbf{Adv}\,[\,\mathbf{Game\ i}\,]$ be D's advantage in distinguishing the output of **Game i** from the **Real** distribution.

> **Game 0**. In this game the honest real-world sender $\mathsf{S}(M_1, \ldots, M_N)$ interacts with the real-world cheating receiver $\hat{\mathsf{R}}$. Clearly $\mathbf{Adv}\,[\,\mathbf{Game\ 0}\,] = 0$.
>
> **Game 1**. In this game, we employ the knowledge extractor for $\mathsf{BlindExtract}$ to extract from $\hat{\mathsf{R}}$ each the identities $(\sigma_1, \ldots, \sigma_k)$ from the k sequential executions of the $\mathsf{BlindExtract}$ protocol. If the knowledge extractor fails for any execution, output $\perp$. Let $\mathbf{Pr}\,[\,error\,]$ be the probability that the knowledge extractor fails. $\mathbf{Adv}\,[\,\mathbf{Game\ 1}\,] - \mathbf{Adv}\,[\,\mathbf{Game\ 0}\,] \leq (k \cdot \mathbf{Pr}\,[\,error\,])$.

Game 2. In this game, the commitment $\mathcal{C}$ is replaced with a commitment to a random value, and the final proof-of-knowledge for decommitment is replaced with a simulated proof. The difference between this game and **Game 2** is equal to the advantage that a p.p.t. algorithm has in correctly distinguishing $\mathcal{C}$ from a valid commitment on $H(C_1, \ldots, C_N)$, and the proof simulation from a valid proof. We define this probability as $\mathbf{Adv}\,[\,dec\,]$, and note that $\mathbf{Adv}\,[\,dec\,]$ is negligible for a secure commitment scheme and zero-knowledge proof.

Game 3. In the final game, we alter the ciphertext vector $(C_1, \ldots, C_N)$, so that for $j = 1, \ldots, N$ and $j \notin (\sigma_1, \ldots, \sigma_k)$, the ciphertext C_j is replaced by the encryption of a random element from $\mathcal{M}$. Let $\mathbf{Adv}\,[\,ibe\,] = \mathbf{Adv}\,[\,\mathbf{Game\ 3}\,] - \mathbf{Adv}\,[\,\mathbf{Game\ 2}\,]$. We claim below that if $\mathbf{Adv}\,[\,ibe\,]$ is negligible if Π is secure in the sense of definition 3.5. Specifically if $\mathbf{Adv}\,[\,ibe\,]$ is non-negligible then there exists a p.p.t. algorithm that breaks the security of the blind IBE scheme Π.

Summing the differences between the above games, it is clear that $\mathbf{Adv}\,[\,\mathbf{Game\ 3}\,] - \mathbf{Adv}\,[\,\mathbf{Game\ 0}\,]$ is negligible, and therefore no p.p.t. algorithm can distinguish the distribution of **Game 3** from **Game 0**. The ideal-world receiver $\hat{\mathsf{R}}'$ is an algorithm that performs all of the changes between the games above, and on learning the values $(\sigma_1, \ldots, \sigma_k)$ transmits these values to the trusted party T to receive $(M_{\sigma_1}, \ldots, M_{\sigma_k})$. For $i = 1, \ldots, k$, $\hat{\mathsf{R}}'$ sets $C_{\sigma_i} = \mathsf{Encrypt}(\mathit{params}, \sigma_i, M_{\sigma_i})$ and proceeds with the protocol.

Claim. Let $\mathcal{A}$ be a p.p.t. algorithm that uses $\hat{\mathsf{R}}$ to distinguish the distributions of **Game 3** and **Game 2**, *i.e.,* distinguishes two ciphertext vectors of the form $\vec{C}_1 = (C_{1,1}, \ldots, C_{1,N}), \vec{C}_2 = (C_{2,1}, \ldots, C_{2,N})$ where (a) for $i = 1, \ldots, N$ both $C_{1,i}, C_{2,i}$ are valid ciphertexts under identity i, (b) one of the underlying plaintexts differs in the same position within $\vec{C}_1$ and $\vec{C}_2$, (c) if i is the index where the plaintexts differ, then no key-extraction has been initiated for identity $id = i$. We argue that $\mathcal{A}$ can defeat the $\mathsf{IND\text{-}sID\text{-}CPA}$ security of IBE scheme Π with non-negligible probability. To show this, we construct an adversary $\mathcal{B}$ that plays the $\mathsf{IND\text{-}sID\text{-}CPA}$ game with Π as follows: $\mathcal{B}$ outputs to Π a random identity $id^* \in 1, \ldots, N$, and forwards *params* to $\mathcal{A}$. $\mathcal{B}$ permits $\mathcal{A}$ to blindly extract keys for up to k identities $(\sigma_1, \ldots, \sigma_k)$, answering the queries by using the knowledge extractor for $\mathsf{BlindExtract}$ to learn these values and running separate $\mathsf{BlindExtract}$ sessions with Π (if $\mathcal{A}$ requests a key for identity id^*, $\mathcal{B}$ aborts the simulation). Next, $\mathcal{B}$ selects a pair of values $(M_0, M_1) \in \mathcal{M}^2$ (where $M_0 \neq M_1$) and sets $C_{1,id^*} = \mathsf{Encrypt}(\mathit{params}, i, M_0)$. $\mathcal{B}$ queries Π on the challenge pair (M_0, M_1) and when Π outputs a challenge ciphertext C^*, sets $C_{2,id^*} = C^*$. For all other $j = 1, \ldots, N$ (and $j \neq id$) let $C_{1,j}$ and $C_{2,j}$ be two distinct encryptions of the same arbitrary plaintext. Clearly if C^* is the encryption of M_0, then the plaintext distributions are identical, and therefore indistinguishable. Thus, when $\mathcal{A}$ indicates that it distinguishes the ciphertext vectors $\mathcal{B}$ can send the guess $b = 1$ as the response to Π and win the $\mathsf{IND\text{-}sID\text{-}CPA}$ game with non-negligible advantage. Note that the simulation aborts with probability at most $\frac{N}{k}$. We may extend this argument to encompass up to $N - k$ differing ciphertexts via a standard hybrid argument. □

Proof of Receiver Security (Theorem 4.3). For any real-world cheating sender $\hat{\mathsf{S}}$ we can construct an ideal-world sender $\hat{\mathsf{S}}'$ such that no p.p.t. algorithm D can distinguish the distributions $\mathbf{Real}_{\hat{\mathsf{S}},\mathsf{R}}(N, k, M_1, \ldots, M_N, \Sigma)$ and $\mathbf{Ideal}_{\hat{\mathsf{S}}',\mathsf{R}'}(N, k, M_1, \ldots, M_N, \Sigma)$. We arrive at the ideal-world sender via a series of games. Again let $\mathbf{Adv}\,[\,\mathbf{Game\ i}\,]$ be D's advantage in distinguishing the output of **Game i** from the **Real** distribution.

Game 0. In this game the honest real-world receiver $\mathsf{R}(\Sigma)$ interacts with the real-world cheating sender $\hat{\mathsf{S}}$. Clearly $\mathbf{Adv}\,[\,\mathbf{Game\ 0}\,] = 0$.

Game 1. In this game, use the knowledge extractor for $PoK\{msk : (params, msk) \in \mathsf{Setup}(1^\kappa, c(\kappa))\}$ to extract msk. If the extractor fails or outputs an invalid msk, output $\perp$. Let $\mathbf{Pr}\,[\,error\,]$ be an upper bound on the probability that the extractor fails on a valid proof. By definition, $\mathbf{Adv}\,[\,\mathbf{Game\ 1}\,] - \mathbf{Adv}\,[\,\mathbf{Game\ 0}\,] \leq \mathbf{Pr}\,[\,error\,]$.

Game 2. In this game, replace the k executions of $\mathsf{BlindExtract}$ with executions on random identities $(\sigma'_1, \ldots, \sigma'_k)$. Let $\mathbf{Adv}\,[\,blind\,] = \mathbf{Adv}\,[\,\mathbf{Game\ 2}\,] - \mathbf{Adv}\,[\,\mathbf{Game\ 1}\,]$. We claim (below) that $\mathbf{Adv}\,[\,blind\,]$ is negligible due to the selective-failure blindness of the IBE scheme Π.

Game 3. In this game, for $i = 1, \ldots, k$ we compare the value $\mathsf{Decrypt}(sk_{\sigma_i}, C_{\sigma_i})$ with $\mathsf{Decrypt}(\mathsf{Extract}(msk, \sigma_i), M_{\sigma_i})$ and if the two values differ, output $\perp$. Based on the correctness guarantee of an IBE scheme, the difference $\mathbf{Adv}\,[\,\mathbf{Game\ 3}\,] - \mathbf{Adv}\,[\,\mathbf{Game\ 2}\,]$ is negligible.

Summing the differences beteen the above games, it is clear that $\mathbf{Adv}\,[\,\mathbf{Game\ 3}\,] - \mathbf{Adv}\,[\,\mathbf{Game\ 0}\,]$ is negligible, and therefore no p.p.t. algorithm can distinguish the distribution of **Game 2** from **Game 0**. The ideal-world sender $\hat{\mathsf{S}}'$ is an algorithm that performs all of the changes between the games above, and on learning $(M_1, \ldots, M_N, b_1, \ldots, b_k)$ transmits these values to the trusted party T.

Claim. Let $\mathcal{A}$ be a p.p.t. algorithm that uses $\hat{\mathsf{S}}$ to distinguish the distributions of **Game 2** and **Game 1**, *i.e.,* distinguishes two runs of k executions of the $\mathsf{BlindExtract}$ protocol where one or more extracted identities differ. We construct an adversary $\mathcal{B}$ with non-negligible advantage in winning the selective-failure blindness game. $\mathcal{B}$ learns $(params, C_1, \ldots, C_N)$ from $\mathcal{A}$, extracts msk from the PoK, selects a pair of random identities $(id_0, id_1) \in \mathbb{Z}_N$, and outputs $(params, id_0, id_1)$ to the oracles for the selective-failure blindness game. $\mathcal{B}$ selects arbitrary identities $(\sigma'_1, \ldots, \sigma'_{k-1})$ and for two distinct executions plays the role of the receiver to blindly extract these $k-1$ identities from $\mathcal{A}$. At the k^{th} blind extraction: in one run $\mathcal{B}$ forwards the oracle's extraction of identity id_b to $\mathcal{A}$ (and answers the extraction of id_{b-1} itself using msk). In the other run, $\mathcal{B}$ initiates to $\mathcal{A}$ an extraction on id_0. If the selective-failure oracles output failure for the extraction of id_b, then $\mathcal{B}$ outputs $\perp$ as the result of the k^{th} transfer. Otherwise $\mathcal{B}$ decrypts C_{id_0} using msk and outputs the result. Clearly if $b = 0$ then the two runs were conducted on the same set of identities, and therefore $\mathcal{A}$ should not distinguish the runs. Therefore if $\mathcal{A}$ distinguishes with non-negligible probability it must be the case that $b = 1$, thus $\mathcal{B}$ outputs the correct guess with non-negligible probability. This sketch may be extended via a hybrid argument. □

C Proof of Security for $\mathsf{OT}^N_{k\times 1}$ in the Random Oracle Model

Below, we sketch a proof of Theorem 4.4 in the random oracle model.

Proof sketch. **Sender Security.** For any real-world cheating receiver $\hat{\mathsf{R}}$ we can construct an ideal-world receiver $\hat{\mathsf{R}}'$ such that no p.p.t. algorithm D can distinguish the distributions $\mathbf{Real}_{\mathsf{S},\hat{\mathsf{R}}}(N, k, M_1, \ldots, M_N, \Sigma)$ and $\mathbf{Ideal}_{\mathsf{S}',\hat{\mathsf{R}}'}(N, k, M_1, \ldots, M_N, \Sigma)$. We model the function

$H : \mathcal{M} \rightarrow \{0,1\}^{|M_1|}$ as a random oracle. $\hat{\mathsf{R}}'$ interacts with $\hat{\mathsf{R}}$ and the trusted party as follows. $\hat{\mathsf{R}}'$ first runs $\mathsf{Setup}(1^\kappa, c(\kappa))$ to generate the scheme parameters, proves knowledge of msk, and sends $(C_1, \ldots, C_N)$ formed by setting $(B_1, \ldots, B_N)$ to be random bitstrings and computing $(A_1, \ldots, A_N)$ as usual. When $\hat{\mathsf{R}}$ blindly extracts a decryption key for some identity σ_i, $\hat{\mathsf{R}}'$ uses the knowledge extractor for $\mathsf{BlindExtract}$ to obtain σ_i, and queries the trusted party to obtain M_{σ_i}. $\hat{\mathsf{R}}'$ programs the random oracle so that $H(W_{\sigma_i}) = B_{\sigma_i} \oplus M_{\sigma_i}$. If a p.p.t. algorithm can distinguish the real and ideal-world distributions then it must be the case that either (a) $\hat{\mathsf{R}}$ violates the $\mathsf{IND\text{-}sID\text{-}CPA}$ or Leak-Free security of the IBE scheme Π, or (b) $\hat{\mathsf{R}}$ has some advantage against the zero-knowledge property of the proofs used.

Receiver Security. For any real-world cheating sender $\hat{\mathsf{S}}$ we can construct an ideal-world sender $\hat{\mathsf{S}}'$ such that no p.p.t. algorithm can distinguish the distributions $\mathbf{Real}_{\hat{\mathsf{S}},\mathsf{R}}(N, k, M_1, \ldots, M_N, \Sigma)$ and $\mathbf{Ideal}_{\hat{\mathsf{S}}',\mathsf{R}'}(N, k, M_1, \ldots, M_N, \Sigma)$. $\hat{\mathsf{S}}'$ interacts with $\hat{\mathsf{S}}$ and the trusted party as follows. When $\hat{\mathsf{S}}$ proves knowledge of the value msk, use the appropriate knowledge extractor to obtain msk. Use msk to decrypt the ciphertext vector $(C_1, \ldots, C_N)$ as per the protocol, and transmit the resulting messages $(M_1, \ldots, M_N)$ to the trusted party T. At the i^{th} protocol round, run $\mathsf{BlindExtract}$ on a random identity σ_i'. If $\mathsf{BlindExtract}$ fails, send $b_i = 0$ to T, otherwise send $b_i = 1$. Based on the selective-failure blindness property of the IBE scheme Π, any failures in the $\mathsf{BlindExtract}$ protocol are independent of the values $(\sigma_1, \ldots, \sigma_k)$ actually extracted by an ideal-world honest receiver. $\square$