A preliminary version of this paper appears in the proceedings of ESORICS 2007, Lecture Notes in Computer Science Vol. ????, Springer-Verlag, 2007. This is the full version.

Generalized Key Delegation for Hierarchical Identity-Based Encryption

Michel Abdalla[1] Eike Kiltz[2] Gregory Neven[3]

June 2007

Abstract

In this paper, we introduce a new primitive called identity-based encryption with wildcard key derivation (WKD-IBE, or "wicked IBE") that enhances the concept of hierarchical identity-based encryption (HIBE) by allowing more general key delegation patterns. A secret key is derived for a vector of identity strings, where entries can be left blank using a wildcard. This key can then be used to derive keys for any pattern that replaces wildcards with concrete identity strings. For example, one may want to allow the university's head system administrator to derive secret keys (and hence the ability to decrypt) for all departmental sysadmin email addresses `sysadmin@*.univ.edu`, where `*` is a wildcard that can be replaced with any string. We provide appropriate security notions and provably secure instantiations with different tradeoffs in terms of ciphertext size and efficiency. We also present a generic construction of identity-based broadcast encryption (IBBE) from any WKD-IBE scheme. One of our instantiation yields an IBBE scheme with constant ciphertext size.

Keywords: Cryptographic protocols, Hierarchical identity-based encryption, key delegation, broadcast encryption.

1 Introduction

Identity-based encryption. Securely linking users to their public keys is a notorious obstacle in the adoption of public-key encryption schemes in practice. Most commonly, it is overcome by means of a public key infrastructure (PKI) where a trusted authority certifies, by means of a digital signature, the relation between users and their public keys. The high cost of setting up and maintaining such a PKI can be prohibitive for many organizations however. In 1984, Shamir [19] proposed identity-based encryption (IBE) as a cheaper alternative to traditional PKIs. Here, the public key of a user *is* his identity (e.g. his name or email address), while the corresponding private key is handed to him by a trusted key distribution center. It lasted until 2000 however for the first practical IBE schemes [17, 6] to be proposed based on bilinear maps.

[1] Departement d'Informatique, École normale supérieure, 45 Rue d'Ulm, 75230 Paris Cedex 05, France. Email: `Michel.Abdalla@ens.fr`. URL: `http://www.di.ens.fr/~mabdalla`.

[2] CWI Amsterdam, The Netherlands. Email: `kiltz@cwi.nl`. URL: `http://kiltz.net`.

[3] Department of Electrical Engineering, Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, B-3001 Heverlee-Leuven, Belgium. Email: `Gregory.Neven@esat.kuleuven.be`. URL: `http://www.neven.org`.

Hierarchical identity-based encryption (HIBE) schemes [13, 11] are the hierarchical extension of IBEs where user identities are vectors of bit strings. The root entity generates private keys for users at the first level; users at level ℓ can derive keys for their children at level $\ell + 1$. This prevents the distribution center from becoming a bottleneck in the system, and at the same time reflects the hierarchical structure of many organizations and user identities, in particular email addresses. For example, the head of the computer science department of a university could be given the key for identity (`edu`,`univ`,`cs`) allowing him to derive keys for identities (`edu`,`univ`,`cs`,`username`) corresponding to email addresses `username@cs.univ.edu`.

Wildcard key derivation. Hierarchical key derivation is a useful feature, but has its limitations. For example, it would be reasonable to prevent end-users from further deriving keys for identities below them. This feature was referred to before as *limited delegation* by Boneh-Boyen-Goh [5], who show a tweak to their HIBE scheme offering exactly this functionality—albeit without a formal security notion or proof for their approach. In some circumstances, it could also be useful to be able to deviate from the hierarchical structure. For example, one may want to allow the university's head system administrator to derive keys for all departmental sysadmin email addresses `sysadmin@*.univ.edu`, where `*` is a wildcard that can be replaced with any string. As another example, it could be practical to provide a company like Google Inc. that registers its name at all top-level domains with a key for `*@google.*`.

These applications lead us to generalize the concept of HIBE schemes to *identity-based encryption with wildcard key derivation* (WKD-IBE), or more succinctly *wicked IBE*. After defining adequate security notions, we start looking for constructions. First observe that if a HIBE scheme allows a maximal hierarchy depth L to be fixed, then the limited-delegation property of [5] can be achieved generically by padding the identity vector with "dummy" strings at the unused lower levels. (But this may come at the cost of efficiency.) The more general functionality of wildcard key delegation cannot be achieved generically though. Nevertheless, we show that many of the existing HIBE schemes are amenable to a modification that enables wildcard key derivation, including the Gentry-Silverberg [11], Boneh-Boyen [4], Waters [20], and Boneh-Boyen-Goh [5] HIBE schemes. For the former three this may come as a bit of a surprise, because no limited-delegation tweaks were previously proposed for these schemes. We prove the security of the modified schemes under our new notions, thereby providing as a special case formal ground for the intuition of [5] regarding their limited-delegation tweak.

Application to identity-based broadcast encryption. Broadcast encryption [10] allows to encrypt a message to any subset $S \subseteq \{1, \ldots, N\}$ of N users so that only users in S can decrypt the message. A trivial solution consists of concatenating encryptions of the message under the public key of each user in S separately, but this yields ciphertexts of size linear in $|S|$. The most efficient fully collusion-resistant (meaning where the adversary can corrupt all users outside of S) public-key broadcast encryption schemes are due to Boneh et al. [7], who present a first construction with constant-size ciphertexts and private keys but with $O(N)$-size public keys, and a second construction with $O(\sqrt{N})$-size ciphertexts and public keys.

Identity-based broadcast encryption (IBBE) is the natural extension of broadcast encryption to the identity-based setting. It is particularly appealing as a primitive because the total number of users in the system N is limited only by the size of the identity space. We propose a generic construction of an IBBE schemes from any WKD-IBE scheme. The construction inflates the private key size by a factor L being the maximal number of identities in a recipient set, but otherwise shares the same cost as the underlying wicked IBE.

Of all the instantiations of wicked IBE that we propose, the most attractive resulting IBBE scheme is that obtained from the scheme based on [5], because it achieves constant-size ciphertexts. However, it has the disadvantage of having private keys of size $O(L^2)$, where L is the maximum number of recipients in a ciphertext. The other concrete instantiations are less attractive because they have ciphertext size $O(L)$, just like the trivial scheme that concatenates individual ciphertexts. Unlike most other broadcast schemes however, they do have the remarkable feature that knowledge of the recipient set is not required in order to decrypt the message.

Wildcard signatures. Just like the key derivation of an IBE scheme automatically gives rise to a signature scheme [6], a WKD-IBE scheme gives rise to a new primitive that we call a *wildcard signature scheme*. It allows a signer to issue a signature on a message containing wildcards, which anyone can replace with concrete values at a later point without invalidating the signature. Our constructions of wicked identity-based encryption yield a number of wildcard signature schemes with different tradeoffs.

Related work. Wicked identity-based encryption can be seen as the dual notion of identity-based encryption with wildcards [1] (WIBE). There, one can use wildcards in the recipient identity to which a ciphertext is encrypted, so that all users whose identity matches the recipient pattern can decrypt it. In fact, the notions of WKD-IBE and WIBE could be combined into a universal primitive that allows wildcards to be used in both the encryption and key derivation algorithms. Instantiations of this primitive can be obtained from all WKD-IBE schemes presented in this work, except for the one based on Gentry-Silverberg's HIBE [11].

Key-policy attribute-based encryption (KP-ABE) [12] associates to each decryption key an access structure consisting of a logical combination of attribute values using AND and OR gates. A ciphertext is encrypted under a set of descriptive attributes and can only be decrypted with a key whose access structure is satisfied by the set of attributes. As discussed in [12], HIBE schemes are a special case of KP-ABE schemes by mapping the identity vector $(\mathtt{edu}, \mathtt{univ}, \mathtt{cs}, \mathtt{sysadmin})$ to the access structure $(1\|\mathtt{edu} \ \wedge \ 2\|\mathtt{univ} \ \wedge \ 3\|\mathtt{cs} \ \wedge \ 4\|\mathtt{sysadmin})$. Likewise, wicked IBE can be seen as a special case of KP-ABE by letting the key for identity $(\mathtt{edu}, *, *, \mathtt{sysadmin})$ be given by the key for $(1\|\mathtt{edu} \ \wedge \ 4\|\mathtt{sysadmin})$. The wicked IBE scheme obtained through the first construction of [12] has the disadvantage of having public keys linear in the size of the attribute universe. The instantiation obtained from their second, large-universe construction is quite similar to the scheme that we derive from the Boneh-Boyen HIBE scheme [4]. None of the schemes derived from [12] achieve constant ciphertext size though, like our wicked IBE construction based on [5].

The use of HIBE schemes in the design of broadcast encryption schemes was first considered by Dodis and Fazio [9]. Chatterjee and Sarkar [8] gave a direct construction of an IBBE scheme that is closely related to the instantiation of our generic construction with the WKD-IBE scheme based on [5]. Our generic construction provides insight into the design of their scheme, but their construction contains some interesting efficiency-improving tweaks. The schemes are compared in more detail in Section 5.3.

In independent work, Shacham [18] formalizes the concept of *limited delegation* for HIBE schemes and proves this feature for the HIBE scheme of [5]. As we pointed out above, limited delegation for HIBEs can be seen as a special case of WKD-IBE where wildcards can only appear at the end of the identity vector. Our WKD-IBE scheme based on [5] can therefore be seen as a generalization of the result of [18].

2 Basic Definitions

In this section, we introduce some notation and computational problems that we will use throughout the rest of the paper. In doing so, we adopt the same notation and definition style used in [1].

Notation. Let $\mathbb{N} = \{0, 1, \ldots\}$ be the set of natural numbers. Let ε be the empty string. If $n \in \mathbb{N}$, then $\{0,1\}^n$ denotes the set of n-bit strings, and $\{0,1\}^*$ is the set of all bit strings. More generally, if S is a set, then S^n is the set of n-tuples of elements of S, $S^{\leq n}$ is the set of tuples of length at most n. If S is finite, then $x \stackrel{\$}{\leftarrow} S$ denotes the assignment to x of an element chosen uniformly at random from S. If A is an algorithm, then $y \leftarrow \mathsf{A}(x)$ denotes the assignment to y of the output of A on input x, and if A is randomized, then $y \stackrel{\$}{\leftarrow} \mathsf{A}(x)$ denotes that the output of an execution of $\mathsf{A}(x)$ with fresh coins is assigned to y.

The Decisional Bilinear Diffie-Hellman Assumption [6]. Let $\mathbb{G}, \mathbb{G}_\mathrm{T}$ be multiplicative groups of prime order p with an admissible map $\hat{e} : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_\mathrm{T}$. By admissible we mean that the map is bilinear, non-degenerate and efficiently computable. Bilinearity means that for all $a, b \in \mathbb{Z}_p$ and all $g \in \mathbb{G}$ we have $\hat{e}(g^a, g^b) = \hat{e}(g, g)^{ab}$. By non-degenerate we mean that $\hat{e}(g, g) = 1$ if and only if $g = 1$. Let $g \in \mathbb{G}$ be a generator. In such a setting, the bilinear decisional Diffie-Hellman (BDDH) problem is to determine, given g, $A = g^a$, $B = g^b$, $C = g^c$, and $Z = \hat{e}(g, g)^z$, whether $Z = \hat{e}(g, g)^{abc}$ for hidden values of a, b, c and z. More formally, let $\mathcal{A}$ be an adversary for the BDDH problem. Such an adversary has advantage ϵ in solving the BDDH problem if $\left|\Pr[\mathcal{A}(g, A, B, C, \hat{e}(g, g)^{abc}) = 1] - \Pr[\mathcal{A}(g, A, B, C, \hat{e}(g, g)^z) = 1]\right| \geq \epsilon$, where the probabilities are over the choice of a, b, c, z and over the random coins consumed by $\mathcal{A}$.

Definition 2.1 The (t, ϵ)-BDDH assumption holds if no t-time adversary has at least ϵ advantage in the above game.

We note that throughout this paper we will assume that the time t of an adversary includes its code size, in order to exclude trivial "lookup" adversaries.

The Decisional Bilinear Diffie-Hellman Exponent Assumption (BDHE) [5]. The ℓ-BDHE problem in $\mathbb{G}$ is: given g, h and $g^{(\alpha^i)} \in \mathbb{G}$, for $i = 1, \ldots, \ell-1, \ell+1, \ldots, 2\ell$ as input, output $\hat{e}(g, h)^{(\alpha^\ell)} \in \mathbb{G}_T$. Boneh, Boyen and Goh, conjectured that the ℓ-BDHE is a hard problem, meaning with this that no polynomially bounded adversary $\mathcal{A}$ can solve it with more than negligible probability, over the random choices of $g, h \in \mathbb{G}$, the choice of $\alpha \in \mathbb{Z}_p$, and the random coin tosses of $\mathcal{A}$.

The decisional version of the problem can be defined in the usual manner. Let $\vec{y} = (g^\alpha, g^{(\alpha^2)}, \ldots, g^{(\alpha^{\ell-1})}, g^{(\alpha^{\ell+1})}, \ldots, g^{(\alpha^{2\ell})})$. An algorithm $\mathcal{B}$ that outputs a bit b, has advantage ϵ in solving the decisional ℓ-BDHE problem in $\mathbb{G}$ if $\left|\Pr\left[\mathcal{B}(g, h, \vec{y}, \hat{e}(g, h)^{(\alpha^\ell)}) = 1\right] - \Pr\left[\mathcal{B}(g, h, \vec{y}, T) = 1\right]\right| \geq \epsilon$, where the probabilities are taken over the random choices of $g, h \in \mathbb{G}$, the random choice of $\alpha \in \mathbb{Z}_p$, the random choice of $T \in \mathbb{G}_T$, and the internal coin tosses of $\mathcal{B}$.

Definition 2.2 The decisional (t, ϵ, ℓ)-BDHE assumption holds in $\mathbb{G}$ if no t-time (probabilistic) algorithm has advantage at least ϵ in solving the decisional ℓ-BDHE problem in $\mathbb{G}$.

3 Wicked Identity-Based Encryption

Syntax. A wicked identity-based encryption scheme (WKD-IBE) is a generalization of a HIBE scheme which allows for more general key delegation patterns. In a WKD-IBE scheme, secret keys are associated with patterns rather than identity vectors. A pattern P is a vector $(P_1, \ldots, P_\ell) \in (\{0,1\}^* \cup \{*\})^\ell$ of length $\ell \leq L$, where $*$ is a special wildcard symbol and L is the maximal depth of the WKD-IBE scheme. That is, each component of a pattern P is either a specific identity string or a wildcard. The main idea behind the WKD-IBE notion is that a user in possession of the secret key for a given pattern P can generate secret keys for any pattern P' that matches P. We say that a pattern $P' = (P'_1, \ldots, P'_{\ell'})$ *matches* P, denoted $P' \in_* P$, if and only if $\ell' \leq \ell$; $\forall\, i = 1 \ldots \ell'$, $P'_i = P_i$ or $P_i = *$; and $\forall\, i = \ell' + 1 \ldots \ell$, $P_i = *$.

More formally, a WKD-IBE scheme is a tuple of algorithms $\mathcal{WKD\text{-}IBE} = (\mathsf{Setup}, \mathsf{KeyDer}, \mathsf{Enc}, \mathsf{Dec})$ providing the following functionality. The root authority first generates a master key pair $(mpk, msk) \stackrel{\$}{\leftarrow} \mathsf{Setup}$. Via $sk_{P'} \stackrel{\$}{\leftarrow} \mathsf{KeyDer}(sk_P, P')$, a user possessing the secret key sk_P for a pattern $P = (P_1, \ldots, P_\ell)$ can derive a secret key for any pattern $P' \in_* P$. The secret key of the root identity is $msk = sk_{(*,\ldots,*)}$.

To create a ciphertext of message $m \in \{0,1\}^*$ intended for an identity $ID = (ID_1, \ldots, ID_\ell)$, the sender computes $C \stackrel{\$}{\leftarrow} \mathsf{Enc}(mpk, ID, m)$. Any user in possession of the secret key for a pattern P such that $ID \in_* P$ can decrypt the ciphertext using sk_P as $m \leftarrow \mathsf{Dec}(sk_P, C, ID)$. Correctness requires that for all key pairs (mpk, msk) output by Setup, all messages $m \in \{0,1\}^*$, all $0 \leq \ell \leq L$, all patterns $P \in (\{0,1\}^* \cup \{*\})^\ell$, and all identities $ID \in (\{0,1\}^*)^{\ell'}$ such that $ID \in_* P$, $\mathsf{Dec}(\ \mathsf{KeyDer}(msk, P)\ ,\ \mathsf{Enc}(mpk, ID, m), ID\ ) = m$ with probability one.

Security. We define the security of WKD-IBE schemes in a way that is very similar to the case of HIBE schemes, but where the adversary can query for the secret keys corresponding to arbitrary patterns, rather than specific identity vectors. Of course, the adversary is not allowed to query the key derivation oracle for any pattern matched by the challenge identity.

More specifically, security is defined through the following game with an adversary. In the first phase, the adversary is run on input of the master public key of a freshly generated key pair $(mpk, msk) \stackrel{\$}{\leftarrow} \mathsf{Setup}$. In a chosen-plaintext attack (IND-WKID-CPA), the adversary is given access to a key derivation oracle that on input a pattern $P \in \big(\{0,1\}^* \cup \{*\}\big)^{\leq L}$ returns $sk_P \stackrel{\$}{\leftarrow} \mathsf{KeyDer}(msk, P)$.

At the end of the first phase, the adversary outputs two equal-length challenge messages $m_0^*, m_1^* \in \{0,1\}^*$ and a challenge identity $ID^* = (ID_1^*, \ldots, ID_{\ell^*}^*)$ where $0 \leq \ell^* \leq L$. The adversary is given a challenge ciphertext $C^* \stackrel{\$}{\leftarrow} \mathsf{Enc}(mpk, ID^*, m_b^*)$ for a randomly chosen bit b, and is given access to the same oracles as during the first phase of the attack. The second phase ends when the adversary outputs a bit b'. The adversary is said to win the IND-WKID-CPA game if $b' = b$ and if it never queried the key derivation oracle for the key of any pattern P such that $ID^* \in_* P$. If *Succ* is the event that the adversary wins the above game, then its advantage is defined as $\epsilon = 2 \cdot \Pr[\,Succ\,] - 1$.

Definition 3.1 A WKD-IBE scheme is $(t, q_{\mathrm{K}}, \epsilon)$ IND-WKID-CPA-secure if all t-time adversaries making at most q_{K} queries to the key derivation oracle have at most advantage ϵ in the IND-WKID-CPA game described above.

Selective-identity Security. As for the case of HIBEs, we also define the weaker selective-identity (sWKID) security notion IND-sWKID-CPA. The IND-sWKID-CPA definition is analogous to the IND-WKID-CPA one given above except that the adversary has to commit to the challenge identity at the beginning of the game, before the master public key is made available.

4 Constructions of Wicked Identity-Based Encryption

4.1 Constructions with Linear-Size Ciphertexts

A construction from Gentry-Silverberg's HIBE scheme. In the following, we present a wicked IBE scheme based on the Gentry-Silverberg HIBE scheme [11]. The scheme uses L independent random oracles $H_i : \{0,1\}^* \to \mathbb{G}$ for $1 \le i \le L$. These can be derived from a single random oracle via standard techniques [3].

We provide some intuition into our construction by taking a closer look at the key derivation of (a slight variant of) the original Gentry-Silverberg HIBE scheme. For master secret key $\alpha \xleftarrow{\$} \mathbb{Z}_p$ and master public key $g_1 \leftarrow g^\alpha$, the decryption key of an identity (ID_0) at the top level is given by $sk_{(ID_0)} \leftarrow H_0(ID_0)^\alpha$. The key for a lower-level identity $(ID_0, \ldots, ID_\ell)$ is given by

$$sk_{(ID_0,\ldots,ID_\ell)} \leftarrow \left(H_0(ID_0)^\alpha \cdot \prod_{i=1}^{\ell} H_i(ID_i)^{r_i} \;,\; g^{r_1} \;,\; \ldots \;,\; g^{r_\ell} \right)$$

for random $r_1, \ldots, r_\ell \xleftarrow{\$} \mathbb{Z}_p$. One could "insert a wildcard" at level $1 \le j \le \ell$ by omitting the factor $H_j(ID_j)^{r_j}$ from the product in the first component and omitting the entry g^{r_j} in the vector; any value for ID_j can then be filled in later by choosing r_j, multiplying $H_j(ID_j)^{r_j}$ into the first component and inserting a new component g^{r_j}. Inserting a wildcard at the top level is not so easy though, as knowledge of the master key α is required to compute the factor $H_0(ID_0)^\alpha$. We therefore "disable" the top level by fixing it to identity $\perp$, or equivalently, by including $h_0 = H_0(\perp)$ in the public key. A similar fix can be used to prevent a user at level $\ell < L$ to further derive keys for users at levels $\ell+1, \ldots, L$. Namely, the key is computed as if it were for the identity at level L with the components at levels $\ell+1, \ldots, L$ fixed to $\perp$. Equivalently, one can include the elements $h_i = H_i(\perp)$ for $1 \le i \le L$ in the public key.

Before presenting the scheme, we first need to introduce some additional notation. If $P = (P_1, \ldots, P_\ell)$ is a pattern, then let $|P| = \ell$ be the length of P, let $\mathrm{W}(P)$ be the set containing all wildcard indices in P, i.e. the indices $1 \le i \le \ell$ such that $P_i = *$, and let $\overline{\mathrm{W}}(P)$ be the complementary set containing all non-wildcard indices. Clearly, $\mathrm{W}(P) \cap \overline{\mathrm{W}}(P) = \emptyset$ and $\mathrm{W}(P) \cup \overline{\mathrm{W}}(P) = \{1, \ldots, \ell\}$. We also extend the notations $P|_{\le i}$, $P|_{> i}$ and $P|_I$ that we introduced for identity vectors to patterns in the natural way. We are now ready to present the $\mathcal{GS}$-$\mathcal{WKD}$-$\mathcal{IBE}$ scheme in full details:

Setup. The root identity chooses random generators $g, h_0, \ldots, h_L \xleftarrow{\$} \mathbb{G}^*$. It chooses $\alpha \xleftarrow{\$} \mathbb{Z}_p$ and computes $g_1 \leftarrow g^\alpha$. It publishes $mpk \leftarrow (g, g_1, h_0, \ldots, h_L)$ as the master public key and keeps $msk \leftarrow h_0^\alpha$ secret.

Key Derivation. To compute a secret key for a pattern $P = (P_1, \ldots, P_\ell)$ directly from the master secret key, the root proceeds as follows. Let $I = \overline{\mathrm{W}}(P) \cup \{\ell+1, \ldots, L\}$. For all $i \in I$ the root chooses $r_i \xleftarrow{\$} \mathbb{Z}_p$ and lets $b_i \leftarrow g^{r_i}$. It then computes $a \leftarrow msk \cdot \prod_{i \in \overline{\mathrm{W}}(P)} H_i(P_i)^{r_i} \cdot \prod_{i=\ell+1,\ldots,L} h_i^{r_i}$. The secret key for pattern P is $sk_P \leftarrow (a, (b_i)_{i \in I})$.

Anyone knowing this secret key can generate a key for a pattern $P' = (P'_1, \ldots, P'_{\ell'}) \in_* P$ as follows. Let $I' = \overline{\mathrm{W}}(P') \cup \{\ell'+1, \ldots, L\}$. Note that $P' \in_* P$ implies that $I \subseteq I'$. For all $i \in I$, choose $r_i \xleftarrow{\$} \mathbb{Z}_p$ and compute $b'_i \leftarrow b_i \cdot g^{r_i}$; for all $i \in I' \setminus I$, choose $r_i \xleftarrow{\$} \mathbb{Z}_p$ and compute $b'_i \leftarrow g^{r_i}$. Finally, compute $a' \leftarrow a \cdot \prod_{i \in \overline{\mathrm{W}}(P')} H_i(P'_i)^{r_i} \cdot \prod_{i=\ell'+1}^{L} h_i^{r_i}$ and return the secret key $sk_{P'} \leftarrow (a', (b'_i)_{i \in I'})$.

Encryption. To encrypt a message $m \in \mathbb{G}_\mathrm{T}$ to identity $ID = (ID_1, \ldots, ID_\ell)$ under $mpk =$

$(g, g_1, h_0, \ldots, h_L)$, the sender chooses $t \stackrel{\$}{\leftarrow} \mathbb{Z}_p$, computes

$$\begin{aligned} C_0 &\leftarrow g^t \\ C_i &\leftarrow H_i(ID_i)^t \quad \text{for } i = 1, \ldots, \ell \\ C_i &\leftarrow h_i^t \quad \text{for } i = \ell+1, \ldots, L \\ C_{L+1} &\leftarrow \hat{e}(g_1, h_0)^t \cdot m \end{aligned}$$

and outputs the ciphertext $C = (C_0, \ldots, C_{L+1})$.

Decryption. A recipient knowing the secret key sk_P for a pattern $P = (P_1, \ldots, P_\ell)$ can decrypt a ciphertext $(C_0, \ldots, C_{L+1})$ intended to any identity $ID \in_* P$ as follows. Let $I = \overline{\mathrm{W}}(P) \cup \{\ell+1, \ldots, L\}$ and let $a_P = (a, (b_i)_{i \in I})$. The recipient recovers the plaintext as

$$m \leftarrow C_{L+1} \cdot \frac{\prod_{i \in I} \hat{e}(b_i, C_i)}{\hat{e}(C_0, a)} .$$

Note that the recipient need not even know the exact identity under which the message was encrypted.

The fact that decryption works can be seen as follows. Let $P = (P_1, \ldots, P_\ell)$ be a pattern, let $I = \overline{\mathrm{W}}(P) \cup \{\ell+1, \ldots, L\}$ and let $sk_P = (a, (b_i)_{i \in I})$ be a secret key for P. For all $i \in I$, let r_i be the discrete logarithm of b_i with respect to g, i.e. $b_i = g^{r_i}$. From the key derivation algorithm one can see that $a = h_0^\alpha \cdot \prod_{i \in \overline{\mathrm{W}}(P)} H_i(ID_i)^{r_i} \cdot \prod_{i=\ell+1}^{L} h_i^{r_i}$. When $(C_0, \ldots, C_{L+1})$ is a ciphertext intended for $ID = (ID_1, \ldots, ID_{\ell'}) \in_* P$, we have that

$$\begin{aligned} \hat{e}(C_0, a) &= \hat{e}\left(g^t \,,\, h_0^\alpha \cdot \textstyle\prod_{i \in \overline{\mathrm{W}}(P)} H_i(P_i)^{r_i} \cdot \prod_{i=\ell+1}^{L} h_i^{r_i}\right) \\ &= \hat{e}(g^t, h_0^\alpha) \cdot \prod_{i \in \overline{\mathrm{W}}(P)} \hat{e}\left(g^{r_i}, H_i(P_i)^t\right) \cdot \prod_{i=\ell+1}^{L} \hat{e}\left(g^{r_i}, h_i^t\right) \\ &= \hat{e}(g_1, h_0)^t \cdot \prod_{i \in I} \hat{e}(b_i, C_i) \ , \end{aligned}$$

where the last equality holds because $P_i = ID_i$ for all $i \in \overline{\mathrm{W}}(P)$ if $ID \in_* P$. Hence, the value of K at decryption is exactly the argument of H_2 at encryption, and the correct message is recovered.

The following theorem states the security of the above scheme in the selective-identity notion under the BDDH assumption in the random oracle model. Security in the full-identity notion can be obtained at the cost of losing a factor $O(q_{\mathrm{H}}^L)$ in the reduction.

Theorem 4.1 Under the (t', ϵ') BDDH assumption, the $\mathcal{GS}$-$\mathcal{WKD}$-$\mathcal{IBE}$ scheme described above is $(t, q_{\mathrm{K}}, q_{\mathrm{H}}, \epsilon)$ IND-sWKID-CPA-secure in the random oracle model for $\epsilon \geq 2\epsilon'$ and $t \leq t' - (q_{\mathrm{H}} + (q_{\mathrm{K}} + 3)L)t_{\exp}$, where $t_{\exp}$ is the time required to perform an exponentiation in $\mathbb{G}$.

Proof of Theorem 4.1: Given an adversary A against the IND-sWKID-CPAsecurity of $\mathcal{GS}$-$\mathcal{WKD}$-$\mathcal{IBE}$, consider the following BDH algorithm B. On input $(g, A = g^a, B = g^b, C = g^c, Z)$, B first runs A to obtain the target identity $ID^* = (ID_1^*, \ldots, ID_{\ell^*}^*)$. It then chooses random integers $\alpha_1, \ldots, \alpha_L \stackrel{\$}{\leftarrow} \mathbb{Z}_p$, sets $g_1 \leftarrow A$, $h_0 \leftarrow B$, $h_i \leftarrow g^{\alpha_i} B^{-1}$ for $1 \leq i \leq \ell^*$ and $h_i \leftarrow g^{\alpha_i}$ for $\ell^* + 1 \leq i \leq L$, and runs A again on input $mpk = (g, g_1, h_0, \ldots, h_L)$. Note that from A's point of view, the public key consists entirely of random elements of $\mathbb{G}$, as required. To answer A's oracle queries, B keeps a set Q and associative arrays $L_{1,i}[\cdot]$, $s_i[\cdot]$ and $L_2[\cdot]$ for $i = 1, \ldots, L$. It initializes these by choosing $s_i[ID_i^*] \xleftarrow{\$} \mathbb{Z}_p$ and $L_{1,i}[ID_i^*] \leftarrow g^{s_i[ID_i^*]}$ for all $i = 1, \ldots, L$; all other entries are left undefined. Algorithm B responds to A's oracle queries as follows:

Random oracle queries $H_i(ID_i)$: If $L_{1,i}[ID_i]$ is defined, then B returns this value. Otherwise, it chooses $s_i[ID_i] \xleftarrow{\$} \mathbb{Z}_p$, sets $L_{1,i}[ID_i] \leftarrow g^{s_i[ID_i]} B^{-1}$ and returns $L_{1,i}[ID_i]$. Note that the response is uniformly distributed over $\mathbb{G}$, as required, due to the random choice of $s_i[ID_i]$.

Key derivation queries for $P = (P_1, \ldots, P_\ell)$: Before going into the details of how keys are simulated, let's have a closer look at the way they are supposed to be distributed. If $I = \overline{\mathrm{W}}(P) \cup \{\ell + 1, \ldots, L\}$, then the secret key for pattern P is a tuple $sk_P = (a, (b_i)_{i \in I})$ where the b_i are uniformly distributed over $\mathbb{G}$ and a is the unique element such that

$$\hat{e}(a, g) \;=\; \hat{e}(g_1, h_0) \cdot \textstyle\prod_{i \in \overline{\mathrm{W}}(P)} \hat{e}(b_i, H_i(P_i)) \cdot \prod_{i=\ell+1}^{L} \hat{e}(b_i, h_i) \;. \tag{1}$$

Given a key $(a, (b_i)_{i \in I})$ for which the above relation for a holds but the b_i are not uniformly distributed, it is easy to see that one can "re-randomize" the key into a correctly distributed one $sk'_P = (a', (b'_i)_{i \in I})$ by choosing $r_i \xleftarrow{\$} \mathbb{Z}_p$ for all $i \in I$, setting $b'_i \leftarrow b_i \cdot g^{r_i}$ and computing $a' \leftarrow a \cdot \prod_{i \in \overline{\mathrm{W}}(P)} H_i(P_i)^{r_i} \cdot \prod_{i=\ell+1}^{L} h_i^{r_i}$.

Since $ID^* \notin_* P$, at least one of the following three conditions must be satisfied:

1. $\ell < \ell^*$. In this case, B lets $b_{\ell^*} \leftarrow g^x$, $b_i \leftarrow 1$ for $i \in I \setminus \{\ell^*\}$ and $a \leftarrow (g^x)^{\alpha_{\ell^*}}$. Since $\hat{e}(1, \cdot) = \hat{e}(\cdot, 1) = 1$, filling these values into the right-hand side of Equation (1) gives

$$\hat{e}(g_1, h_0) \cdot \hat{e}(b_{\ell^*}, h_{\ell^*}) \;=\; \hat{e}(g^x, g^y) \cdot \hat{e}(g^x, g^{\alpha_{\ell^*}} g^{-y}) \;=\; \hat{e}(g^x, g^{\alpha_{\ell^*}}) \;=\; \hat{e}(a, g) \;.$$

   Re-randomizing this key as explained above gives a correctly distributed key for P.

2. There exists $1 \le j \le \ell^*$ such that $P_j \ne *$ and $P_j \ne ID_j^*$. In this case, B sets $b_j \leftarrow g^x$, $b_i \leftarrow 1$ for $i \in I \setminus \{j\}$, $a \leftarrow (g^x)^{s_j[P_j]}$ and re-randomizes $(a, (b_i)_{i \in I})$ as explained above. This key satisfies Equation (1) because

$$\hat{e}(g_1, h_0) \cdot \hat{e}(b_j, H_{1,j}(P_j)) \;=\; \hat{e}(g^x, g^y) \cdot \hat{e}(g^x, g^{s_j[P_j]} g^{-y}) \;=\; \hat{e}(g^x, g^{s_j[P_j]}) \;=\; \hat{e}(a, g) \;.$$

3. There exists $\ell^* + 1 \le j \le L$ such that $P_i \ne *$. In this case, B proceeds exactly as in the previous case. The correctness of the key holds by the same arguments.

At some point, A outputs challenge messages m_0, m_1. Algorithm B chooses a random bit $b \xleftarrow{\$} \{0, 1\}$ and generates the challenge ciphertext $(C_0, \ldots, C_{L+1})$ that it feeds to A as

$$\begin{aligned} C_0 &\leftarrow C \\ C_i &\leftarrow C^{s_i[ID_i^*]} \quad \text{for } 1 \le i \le \ell^* \\ C_i &\leftarrow C^{\alpha_i} \quad \text{for } \ell^* + 1 \le i \le L \\ C_{L+1} &\leftarrow m_b \cdot Z \;. \end{aligned}$$

If A outputs $b' = b$, then B outputs 1; else, it outputs 0.

One can see from the way that $H_i(ID_i^*)$ and h_i were simulated for $1 \leq i \leq \ell^*$ and for $\ell^* + 1 \leq i \leq L$, respectively, that the challenge ciphertext is correctly distributed if $Z = \hat{e}(g,g)^{abc}$, so in this case A outputs $b' = b$ with probability $1/2 \pm \epsilon/2$. However, if Z is a random element from $\mathbb{G}_{\mathrm{T}}$, then A's view is independent of B's choice of b, so its probability of outputting $b' = b$ is $1/2$. Therefore, we have that the BDDH advantage of B is $\epsilon' = |1/2 \pm \epsilon/2 - 1/2| = \epsilon/2$.

The running time of B is roughly that of A plus $2L$ exponentiations at the initialization phase, $q_{\mathrm{H}} + q_{\mathrm{K}} L$ exponentiations during the simulation, and L exponentiations to generate the challenge ciphertext. ▌

Constructions from Boneh-Boyen's and Waters' HIBE schemes. The attentive reader will have noticed the resemblance of the above scheme with the HIBE schemes of Boneh-Boyen [4] and Waters [20]. Indeed, if identity strings are elements of $\mathbb{Z}_p^*$, then one can obtain a wicked IBE variant of [4] by setting $H_i(ID_i) = h_{i,0} h_{i,1}^{ID_i}$, where $h_{i,0}, h_{i,1}$ are random elements of $\mathbb{G}$ that are fixed in the master public key. This scheme can be proved IND-sWKID-CPA secure under the BDDH assumption in the standard (i.e., non-random oracle) model using a proof quite similar to the above analysis. Likewise, one can obtain a variant based on Waters' HIBE scheme when identities are n-bit strings by setting $H_i(ID_i = ID_{i,1} \ldots ID_{i,n}) = h_{i,0} \prod_{ID_{i,j}=1} h_{i,j}$. An analysis similar to the one in [20] can be used to prove this scheme IND-WKID-CPA secure under the BDDH assumption in the standard model at the cost of losing a factor $O((nq_{\mathrm{K}})^L)$ in the reduction.

4.2 Constructions with Constant-Size Ciphertexts

In this section, we describe efficient wicked IBE schemes with constant-size ciphertexts based on the Boneh-Boyen-Goh [5] and Waters [20] HIBE schemes. We build the wicked IBE scheme $\mathcal{BBG}\text{-}\mathcal{WKD}\text{-}\mathcal{IBE} = (\mathsf{Setup}, \mathsf{KeyDer}, \mathsf{Enc}, \mathsf{Dec})$ described as follows:

Setup. The trusted authority chooses random generators g from $\mathbb{G}$, a random $\alpha \in \mathbb{Z}_p$ and sets $g_1 \leftarrow g^\alpha$. Next, it picks random elements $g_2, g_3, h_1, \ldots, h_L$ from $\mathbb{G}$ and sets $g_4 \leftarrow g_2^\alpha$. The master public key is $mpk = (g, g_1, g_2, g_3, h_1, \ldots, h_L)$. The corresponding master secret key is $msk = g_4$.

Key Derivation. Let $P' = (P'_1, \ldots, P'_\ell) \in \left(\mathbb{Z}_p^* \cup \{*\}\right)^{\leq L}$ be the pattern for which a secret key needs to be generated. To compute the secret key for P' from the master secret key, first a random $r \stackrel{\$}{\leftarrow} \mathbb{Z}_p$ is chosen, then the secret key $sk_{P'} = (a'_1, a'_2, b')$ for P' is constructed as

$$a'_1 = g_4 \cdot \left(g_3 \prod_{i \in \overline{\mathrm{W}}(P')} h_i^{P'_i}\right)^r \; ; \quad a'_2 = g^r \; ; \quad b' = (b_i = h_i^r)_{i \in \mathrm{W}(P')} \; .$$

In order to generate the secret key $sk_{P'}$ for pattern P' from the secret key $sk_P = (a_1, a_2, b)$ for pattern P such that $P' \in_* P$, ones simply chooses a random $r' \stackrel{\$}{\leftarrow} \mathbb{Z}_p$ and outputs $sk_{P'} = (a'_1, a'_2, b')$, where

$$\begin{aligned} a'_1 &= a_1 \cdot \left(g_3 \prod_{i \in \overline{\mathrm{W}}(P')} h_i^{P'_i}\right)^{r'} \cdot \left(\prod_{i \in \overline{\mathrm{W}}(P') \bigcap \mathrm{W}(P)} b_i^{P'_i}\right) \\ a'_2 &= a_2 \cdot g^{r'} \\ b' &= \left(b'_i = b_i \cdot h_i^{r'}\right)_{i \in \mathrm{W}(P')} \end{aligned}$$

Encryption. To encrypt a message $m \in \mathbb{G}_T$ for an identity $ID = (ID_1, \ldots, ID_\ell)$, the sender first chooses $t \xleftarrow{\$} \mathbb{Z}_p$ and outputs the ciphertext $C = (C_1, C_2, C_3) \in \mathbb{G} \times \mathbb{G} \times \mathbb{G}_T$, where

$$C_1 = g^t \; ; \quad C_2 = \left(g_3 \prod_{i=1}^{\ell} h_i^{ID_i} \right)^t \; ; \quad C_3 = m \cdot \hat{e}(g_1, g_2)^t \; .$$

Decryption. Let be the $C = (C_1, C_2, C_3)$ and $ID = (ID_1, \ldots, ID_\ell)$ be the identity to which the ciphertext was created. If the receiver is the root authority holding the master key msk, then he can recover the message by computing $C_3/\hat{e}(C_1, msk)$. Any other receiver holding a secret key for pattern P such that $ID \in_* P$ can decrypt the ciphertext as follows. Let $sk_P = (a_1, a_2, b)$ be the decryption key for the receiver. He can recover the message by computing

$$a_1' \leftarrow a_1 \cdot \left(\prod_{i \in \mathrm{W}(P)|_{\leq \ell}} b_i^{ID_i} \right) \quad \text{and} \quad m \leftarrow C_3 \cdot \frac{\hat{e}(a_2, C_2)}{\hat{e}(C_1, a_1')}.$$

The fact that decryption works can be seen as follows. Since $ID \in_* P$, we have that $P_i = ID_i$ for all $i \in \overline{\mathrm{W}}(P)|_{\leq \ell}$. Thus the quantity $\frac{\hat{e}(a_2, C_2)}{\hat{e}(C_1, a_1')}$ becomes:

$$\frac{\hat{e}(a_2, C_2)}{\hat{e}(C_1, a_1 \prod_{i \in \mathrm{W}(P)|_{\leq \ell}} b_i^{ID_i})} = \frac{\hat{e}(g^r, (g_3 \prod_{i=1}^{\ell} h_i^{ID_i})^r)}{\hat{e}(g^t, g_4 \cdot (g_3 \prod_{i \in \overline{\mathrm{W}}(P)} h_i^{P_i})^r \cdot \prod_{i \in \mathrm{W}(P)|_{\leq \ell}} b_i^{ID_i})}$$

$$= \frac{\hat{e}(g^r, (g_3 \prod_{i=1}^{\ell} h_i^{ID_i})^t)}{\hat{e}(g^t, g_4) \; \hat{e}(g^t, (g_3 \prod_{i=1}^{\ell} h_i^{ID_i})^r)} = \frac{1}{\hat{e}(g^t, g_4)} = \frac{1}{\hat{e}(g_1, g_2)^t}$$

The following theorem states the security of the above scheme in the selective-identity notion under the ℓ-BDHE assumption in the standard model. We remark that, interestingly, we can only prove security of the scheme based on the ℓ-BDHE assumption, whereas the weaker ℓ-BDHI assumption was sufficient for the security proof of the HIBE scheme [5].

Theorem 4.2 Let $\mathcal{BBG}$-$\mathcal{WKD}$-$\mathcal{IBE}$ be the WKD-IBE scheme as described above. Under the decisional (t, ϵ, ℓ)-BDHE assumption, the $\mathcal{BBG}$-$\mathcal{WKD}$-$\mathcal{IBE}$ scheme of depth $L = \ell - 1$ is $(t', q_{\mathrm{K}}, 2\epsilon)$ IND-sWKID-CPA-secure where $t' = t - O(Lq_{\mathrm{K}}') \cdot t_{\exp}$ and $t_{\exp}$ is the time it takes to perform an exponentiation in $\mathbb{G}$.

Proof: We assume that there exist an adversary $\mathcal{A}$ that breaks the IND-WKID-CPA-security of WKD-IBE scheme $\mathcal{BBG}$-$\mathcal{WKD}$-$\mathcal{IBE}$ and then we show how to efficiently build another adversary $\mathcal{B}$ that, using $\mathcal{A}$ as a black box, manages to break the ℓ-BDHE problem in $\mathbb{G}$.

Let g, h, $\vec{y} = (y_1, \ldots, y_{\ell-1}, y_{\ell+1}, \ldots, y_{2\ell})$ with $y_i = g^{\alpha^i} \in \mathbb{G}$, and $T \in \mathbb{G}_T$ be given to $\mathcal{B}$ as input. Adversary $\mathcal{B}'s$ task is to decide if $T = \hat{e}(g, h)^{\alpha^\ell}$ or if T is uniform in $\mathbb{G}$.

Initialization. Adversary $\mathcal{A}$ first outputs an identity vector $ID^* = (ID_1^*, \ldots, ID_m^*) \in (\mathbb{Z}_p^*)^m$ for $m \leq L$ she intends to attack. By padding ID^* with zero identities we can assume that ID^* is a vector of size $L = \ell - 1$.

Setup. Adversary $\mathcal{A}$ prepares a correctly distributed master public key $mpk = (g, g_1, g_2, g_3, h_1, \ldots, h_L)$ as follows. Initially it picks random values $\gamma, \gamma_1, \ldots, \gamma_L, \delta$ from $\mathbb{Z}_p^*$ and defines

$$g_1 \leftarrow y_1 = g^\alpha \; ; \quad g_2 \xleftarrow{\$} y_{\ell-1} \cdot g^\gamma = g^{\alpha^{\ell-1} + \gamma} \; ; \quad g_3 \xleftarrow{\$} g^\delta \prod_{i=1}^{L} y_{\ell-i}^{ID_i^*}$$

and

$$h_i \leftarrow g^{\gamma_i}/y_{\ell-i}, 1 \le i \le L = \ell - 1.$$

Note that the master secret key is implicitly defined as $g_4 = g_2^\alpha = g^{\alpha^\ell+\gamma\alpha} = y_\ell\, y_1^\gamma$ which is unknown to $\mathcal{A}$ since $y_\ell = g^{\alpha^\ell}$ is unknown.

Key Derivation queries. Suppose adversary $\mathcal{B}$ makes a key derivation query for pattern $P = (P_1, \dots, P_u) \in (\mathbb{Z}_p^* \cup \{*\})^u$ of length $u \le L$. By padding P with zero identities we can assume that P is a vector of size $L = \ell - 1$. By the definition of the security experiment, we know that $ID^* \notin_* P$. That means that there exists an index $k \in \overline{\mathrm{W}}(P)$ such that $P_k \neq ID_k^*$. We define k to be the smallest of all possible indices. $\mathcal{B}$ picks a random $\tilde{r} \in \mathbb{Z}_p^*$ and (implicitly) sets $r \leftarrow -\frac{\alpha^k}{ID_k^*-P_k} + \tilde{r} \in \mathbb{Z}_p$. The secret key $sk_P = (a_1, a_2, b)$ for P is constructed as

$$a_1 = g_4 \cdot \left(g_3 \prod_{i \in \overline{\mathrm{W}}(P)} h_i^{P_i} \right)^r ; \quad a_2 = g^r ; \quad (b_i = h_i^r)_{i \in \mathrm{W}(P)}$$

We have

$$\begin{aligned} \left(g_3 \prod_{i \in \overline{\mathrm{W}}(P)} h_i^{P_i} \right)^r &= \left(g^\delta \prod_{i=1}^{L} y_{\ell-i}^{ID_i^*} \prod_{i \in \overline{\mathrm{W}}(P)} g^{\gamma_i P_i} y_{\ell-i}^{-P_i} \right)^r \\ &= \left(g^{\delta + \sum_{\in \overline{\mathrm{W}}(P)} P_i \gamma_i} \cdot \prod_{i \in \overline{\mathrm{W}}(P) \setminus \{k\}} y_{\ell-i}^{ID_i^* - P_i} \cdot y_{\ell-k}^{ID_k^* - P_k} \cdot \prod_{i \in \mathrm{W}(P)} y_{\ell-i}^{ID_i^*} \right)^r \end{aligned}$$

We split this term up into the two factors $A \cdot Z$, where $A = (y_{\ell-k}^{ID_k^*-P_k})^r$ is the third product only. It can be checked that Z can be computed by $\mathcal{A}$, i.e. the terms y_i only appear with indices $i \in \{1, \dots, \ell-1, \ell+1, \dots, 2\ell\}$. The term A can be expressed as

$$A = g^{\alpha^{\ell-k}(ID_k^*-P_k)(-\frac{\alpha^k}{ID_k^*-P_k}+\tilde{r})} = y_\ell^{-1} \cdot y_{\ell-k}^{(ID_k^*-P_k)\tilde{r}}$$

Hence,

$$a_1 = g_4 \cdot A \cdot Z = y_\ell y_1^\gamma \cdot y_\ell^{-1} y_{\ell-k}^{(ID_k^*-P_k)\tilde{r}} \cdot Z = y_1^\gamma \cdot y_{\ell-k}^{(ID_k^*-P_k)\tilde{r}} \cdot Z$$

can be computed by $\mathcal{A}$. Furthermore,

$$g^r = g^{-\frac{\alpha^k}{ID_k^*-P_k}+\tilde{r}} = y_k^{-\frac{1}{ID_k^*-P_k}} \cdot g^{\tilde{r}}$$

and for each $i \in \mathrm{W}(P)$,

$$h_i^r = (g^{\gamma_i}/y_{\ell-i})^{-\frac{\alpha^k}{ID_k^*-P_k}+\tilde{r}} = y_k^{-\frac{\gamma_i}{ID^*-P_k}} \cdot y_{k+\ell-i}^{\frac{1}{ID_k^*-P_k}} \cdot g^{\gamma_i \tilde{r}} \cdot y_{\ell-i}^{-\tilde{r}}$$

can be computed since $k \notin \mathrm{W}(P)$.

Challenge. Eventually $\mathcal{A}$ returns two messages m_0 and m_1 on which she wants to be challenged on. Adversary $\mathcal{B}$ flips a coin b and creates a challenge ciphertext $C^* = (C_1^*, C_2^*, C_3^*)$ for target identity ID^* as follows

$$C_1^* = h ; \quad C_2^* = h^{\delta + \sum_{i=1}^{L} ID_i^* \gamma_i} ; \quad C_3^* = m_b \cdot T \cdot \hat{e}(y_1, h)^\gamma$$

Define $t \in \mathbb{Z}_p$ such that $h = g^t$. We show that in case $T = \hat{e}(g,h)^{\alpha^\ell}$, C^* is a correctly distributed ciphertext with (unknown) randomness t. For C_2^* we have

$$\begin{aligned}
\left(g_3 \prod_{i=1}^{L} h_i^{ID_i}\right)^t &= \left(g^\delta \prod_{i=1}^{L} y_{\ell-i}^{ID_i^*} \prod_{i=1}^{L} g^{\gamma_i ID_i^*} y_{\ell-i}^{-ID_i^*}\right)^t \\
&= \left(g^\delta \prod_{i=1}^{L} g^{\gamma_i ID_i^*}\right)^t \\
&= h^{\delta + \sum_{i=1}^{L} \gamma_i ID_i^*} \\
&= C_2^*
\end{aligned}$$

For C_3^* we use $\hat{e}(y_i, y_j) = \hat{e}(g, y_{i+j})$ to show

$$\begin{aligned}
m_b \cdot \hat{e}(g_1, g_2)^t &= m_b \cdot \hat{e}(y_1, y_{\ell-1} \cdot g^\gamma)^t \\
&= m_b \cdot \hat{e}(h, y_\ell) \cdot \hat{e}(y_1, h)^\gamma \\
&= C_3^*
\end{aligned}$$

Guess. Eventually, adversary $\mathcal{A}$ outputs a bit b'. Finally, adversary $\mathcal{B}$ terminates her game and returns 1 (meaning $T = \hat{e}(g,h)^{\alpha^\ell}$) if $b = b'$, and returns 0 (meaning $T \in \mathbb{G}$ is random) otherwise.

When $T = \hat{e}(g,h)^{\alpha^\ell}$ from adversary $\mathcal{B}$'s input then $\mathcal{A}$'s view is identical to its view in a real attack game and therefore $\mathcal{A}$ satisfies $|\Pr[b = b'] - 1/2| \geq \varepsilon/2$. When the T is uniform in $\mathbb{G}$ then $\mathcal{A}$'s view is independent of the bit b and therefore $\Pr[b = b'] = 1/2$. Therefore,

$$\left|\Pr\left[\mathcal{B}(g, h, \vec{y}, \hat{e}(g,h)^{(\alpha^\ell)}) = 1\right] - \Pr\left[\mathcal{B}(g, h, \vec{y}, T) = 1\right]\right| \geq |(1/2 \pm \varepsilon/2) - 1/2| = \varepsilon/2 ,$$

from which the theorem follows. ▮

Full security in the standard model. It is mentioned in [5] that using techniques from Waters [20] one can construct a variant of their HIBE scheme that achieves full security in the standard model. The same techniques can be also used to achieve full IND-WKID-CPA security in the standard model for the $\mathcal{BBG}$-$\mathcal{WKD}$-$\mathcal{IBE}$ scheme, at the cost of increasing the master public key size to $(n+1)L+3$ group elements, where n is the length of an identity string.

4.3 Full Security in the Random Oracle Model

As in the case of IBE and HIBE schemes [4, 5], any WKD-IBE scheme $\mathcal{WKD}$-$\mathcal{IBE}$ that is IND-sWKID-CPA-secure can be transformed into a WKD-IBE scheme $\mathcal{WKD}$-$\mathcal{IBE}'$ that is IND-WKID-CPA-secure in the random oracle model, by replacing every pattern (or identity) at key derivation or encryption with the hash of that pattern, if that pattern is not a wildcard. That is, any given pattern $P = (P_1, \ldots, P_\ell)$ in $\mathcal{WKD}$-$\mathcal{IBE}$ is mapped onto a pattern $P' = (P'_1, \ldots, P'_\ell)$ in $\mathcal{WKD}$-$\mathcal{IBE}'$, where $P'_i = H_i(P_i)$ if $P_i \neq *$ or $P'_i = *$ otherwise, and H_i, $1 \leq i \leq L$ are independent random oracles mapping arbitrary bit strings into an appropriate range $\mathcal{ID}$ corresponding to the identity space of $\mathcal{WKD}$-$\mathcal{IBE}$. As in the cases of HIBE schemes, this transformation only works if the depth L is logarithmic in the security parameter due to the loss of a factor $O(q_{\mathrm{H}}^L)$ in the reduction. Moreover, $\mathcal{ID}$ needs to be sufficiently large to make the probability of collisions in the output of the hash function negligible.

5 Application to Identity-Based Broadcast Encryption

5.1 Definitions

An identity-based broadcast encryption (IBBE) scheme is a tuple of algorithms $\mathcal{IBBE} = (\mathsf{Setup}, \mathsf{KeyDer}, \mathsf{Enc}, \mathsf{Dec})$ providing the following functionality. The trusted authority runs Setup to generate a master key pair (mpk, msk). It publishes the master public key mpk and keeps the master secret key msk private. When a user with identity ID wishes to become part of the system, the trusted authority generates a user decryption key $sk_{ID} \xleftarrow{\$} \mathsf{KeyDer}(msk, ID)$, and sends this key over a secure and authenticated channel to the user. To broadcast an encrypted message m to a set of users with identities $S = \{ID_1, \ldots, ID_k\}$ of cardinality $k \leq L$, the sender computes the ciphertext $C \xleftarrow{\$} \mathsf{Enc}(mpk, S, m)$, which can be decrypted by a user holding sk_{ID} for any $ID \in S$ as $m \leftarrow \mathsf{Dec}(sk_{ID}, C, S)$. Here the value L is an upper bound on the maximal number of distinct receivers for a broadcast encryption.

The security of an IBBE scheme is defined through the following game. In a first phase, the adversary is given as input the master public key mpk of a freshly generated key pair $(mpk, msk) \xleftarrow{\$} \mathsf{Setup}$. In a chosen-plaintext attack (IND-ID-CPA), the adversary is given access to a key derivation oracle that on input of an identity ID, returns the secret key $sk_{ID} \xleftarrow{\$} \mathsf{KeyDer}(msk, ID)$ corresponding to identity ID. At the end of the first phase, the adversary outputs two equal-length challenge messages $m_0^*, m_1^* \in \{0,1\}^*$ and a challenge set of identities $S^* = (ID_1^*, \ldots, ID_{k^*}^*)$, where $0 \leq k^* \leq L$. The game chooses a random bit $b \xleftarrow{\$} \{0,1\}^*$, generates a challenge ciphertext $C^* \xleftarrow{\$} \mathsf{Enc}(mpk, S^*, m_b^*)$ and gives C^* as input to the adversary for the second phase, during which it gets access to the same oracles as during the first phase. Assume that during the attack the adversary made key derivation queries for identities $ID_1, \ldots, ID_{q_\mathrm{K}}$. The adversary wins the game if it outputs a bit $b' = b$ and $S^* \cap \{ID_1, \ldots, ID_{q_\mathrm{K}}\} = \emptyset$.

Definition 5.1 An IBBE scheme is $(t, q_\mathrm{K}, \epsilon)$-IND-ID-CPA-secure if all t-time adversaries making at most q_K queries to the key derivation oracle have at most advantage ϵ in winning the IND-ID-CPA game described above.

Selective-identity Security. As for the previous primitives, we further define the weaker (sID) security notion IND-sID-CPA. The IND-sID-CPA definition is analogous to the IND-ID-CPA one except that the adversary has to commit to the challenge set of identities $S^* = (ID_1^*, \ldots, ID_{k^*}^*)$ at the beginning of the game, before even seeing the public-key.

5.2 A Construction from any Wicked Identity-Based Encryption Scheme

First, observe that an IBBE scheme can be trivially constructed from any IBE scheme by concatenating ciphertext. Meaning, the IBBE encryption for the identity set $ID = \{ID_1, \ldots, ID_k\}$ is simply the concatenation of k separate ciphertexts, one for each identity ID_i in the set ID. This leads to IBBE ciphertext sizes that are a factor of k longer than the original IBE ciphertexts.

We now present a generic construction from any WKD-IBE scheme that, depending on the instantiation, can offer advantages over the trivial one. To any WKD-IBE scheme $\mathcal{WKD\text{-}IBE} = (\mathsf{Setup}, \mathsf{KeyDer}, \mathsf{Enc}, \mathsf{Dec})$, we associate an IBBE scheme $\mathcal{IBBE} = (\mathsf{Setup}, \mathsf{KeyDer}', \mathsf{Enc}', \mathsf{Dec}')$. For an identity $ID \in \{0,1\}^*$, define

$$P_i(ID) = (*, \ldots, *, \underbrace{ID}_{i\text{th position}}, *, \ldots, *)$$

as a pattern of length L that has ID at its ith position and the rest consists of wildcards.

Setup. Setup outputs whatever the WKD-IBE setup outputs.

Key Derivation. Let ID be the identity for which the user secret key sk_{ID} needs to be generated. The user secret key is defined as the set of L distinct WKD-IBE user secret keys

$$sk_{ID} = \{sk_{P_1(ID)}, \ldots, sk_{P_L(ID)}\},$$

where $sk_{P_i(ID)}$ can be computed by calling $\mathsf{KeyDer}(msk, P_i(ID))$.

Encryption. Let m be the message and let $S = \{ID_1, \ldots, ID_k\}$ be the set of broadcast recipients of cardinality $k \le L$ that we assume to be ordered with respect to some unique standard ordering. The IBBE ciphertext is defined as the WKD-IBE encryption of message m and identity vector $ID = (ID_1, \ldots, ID_k)$.

Decryption. Let $sk_{ID} = \{sk_{P_1(ID)}, \ldots, sk_{P_L(ID)}\}$ be the user secret key of identity ID. Let $S = \{ID_1, \ldots, ID_k\}$ be the set of $k \le L$ recipients to whom the ciphertext C was encrypted, and let index $1 \le j \le k$ be such that $ID = ID_j \in S$. It is clear that $(ID_1, \ldots, ID, \ldots, ID_k) \in_* P_j(ID)$, and therefore that the ciphertext can be decrypted as $m \leftarrow \mathsf{Dec}(sk_{P_j(ID)}, C, ID)$.

Theorem 5.2 Assume an WKD-IBE scheme $\mathcal{WKD\text{-}IBE}$ is $(t, q_{\mathrm{K}}, \epsilon)$ IND-sWKID-CPA-secure (resp. IND-WKID-CPA-secure). Then the IBBE scheme $\mathcal{IBBE}$ described above is $(t, q_{\mathrm{K}}, \epsilon)$-IND-sID-CPA-secure (resp. IND-ID-CPA-secure).

The crucial observation is the following. Let $S^* = \{ID_1^*, \ldots, ID_{k^*}^*\}$ be the set of challenge broadcast receivers and let $ID_1, \ldots, ID_{q_{\mathrm{K}}}$ be the identities an adversary attacking the IBBE scheme queries the user secret key for. The imposed requirement is that $S^* \cap \{ID_1, \ldots, ID_{q_{\mathrm{K}}}\} = \emptyset$. For $1 \le i \le q_{\mathrm{K}}$ and $1 \le j \le L$ consider the user secret keys for the patterns $P_j(ID_i) = (*, \ldots, *, ID_i, *, \ldots, *)$ (i.e., ID_i is at the jth position) that are established by the transformation when simulating the IBBE key derivation oracle. For a successful simulation we have to show that $ID^* = (ID_1^*, \ldots, ID_{k^*}^*) \notin_* P_j(ID_i)$. But this is the case since by $S^* \cap \{ID_1, \ldots, ID_{q_{\mathrm{K}}}\} = \emptyset$ and we can guarantee that $ID_i \neq ID_l^*$ for all $1 \le i \le q_{\mathrm{K}}$ and $1 \le l \le k$.

The above construction allows for the following trade off between ciphertext size and key size. If $L = L'A$, then one can obtain an IBBE scheme with ciphertext size of A times that of the WKD-IBE scheme, while having a key length that is only L' times that of the WKD-IBE scheme. The new scheme creates master public keys to allow for broadcast encryption to sets of maximal cardinality L'. To encrypt a message to a set of broadcast identities $S = \{ID_1, \ldots, ID_k\}$ of cardinality $k \le L$ split the set S into A smaller sets $S_1, \ldots, S_A$, each of cardinality $L/A \le L'$ and define the new broadcast ciphertext to be $(C_1, \ldots, C_A)$, where C_i is the encryption of the message m to the set S_i.

5.3 Instantiations

Among all the instantiations of IBBE schemes based on WKD-IBE schemes, the most attractive one is that obtained from the WKD-IBE scheme based on [5] because it achieves constant-size ciphertexts. However, it has the disadvantage of having private keys of size $O(L^2)$. Instantiations with any of the other WKD-IBE schemes that we proposed are less attractive because they have ciphertext size $O(L)$, just like the trivial ciphertext-concatenation scheme. Unlike most other (public-key) broadcast schemes however, these instantiations do have the remarkable advantage that knowledge of the set of recipients is not required in order to decrypt the message.

Chatterjee and Sarkar [8] recently proposed a direct IBBE scheme that is closely related to our generic construction when instantiated with the WKD-IBE scheme based on [5]. Their scheme does not impose an a priori maximum on the number of recipients ℓ, but makes clever use of a non-cryptographic hash function to achieve an average ciphertext size $O(\ell/L)$ and private key size $O(L)$, where the "average" is taken over the recipients' identities. This means that when $\ell \leq L$, their scheme has constant ciphertext size on average. Worst-case however, their scheme has ciphertext size $O(\ell)$, which is worse than our construction.

6 Wicked and Wildcard Signatures

As observed by Naor [6], any IBE scheme automatically gives rise to a signature scheme by using as a signature on message m the decryption key for identity $ID = m$. Verification can be done by encrypting a random message to identity $ID = m$ and testing whether it decrypts correctly, but most concrete schemes have a more natural and efficient verification test. Likewise, one can construct an L-level hierarchical identity-based signature (HIBS) scheme from an $(L+1)$-level HIBE [11] by letting the signature on message m by identity $(ID_1, \ldots, ID_\ell)$ be given by the decryption key for identity $(0\|ID_1, \ldots, 0\|ID_\ell, 1\|m)$. The same technique can be used to construct *wicked identity-based signatures* (WKD-IBS), the signing analogue to wicked IBE. Here, a root authority derives secret signing keys for identity patterns with wildcards, from which anyone can further derive signing keys for matching patterns. An L-level WKD-IBS is constructed from an $(L+1)$-level WKD-IBE by letting the signature on message m by identity $(ID_1, \ldots, ID_\ell)$ be given by the decryption key for identity $(0\|ID_1, \ldots, 0\|ID_\ell, 1\|m)$.

Alternatively, and perhaps more interestingly, one could also use the wildcard functionality as a homomorphism on the message being signed, rather than for the signers' identities. This yields a new primitive that we call *wildcard signatures*, that allow to sign message patterns instead of simple messages, possibly containing wildcards at certain positions. Given such a signature, anyone can compute a valid signature for any message created by replacing wildcards with concrete values. This could be used for example to implement signed fill-out forms, where each input field is represented by a wildcard in the message.

The construction from a WKD-IBE scheme is straightforward: the key pair is given by the master key pair of the WKD-IBE scheme. The signature on a message pattern P is given by the decryption key for P. Deriving a valid signature for a message pattern $P' \in_* P$ can be done by deriving a decryption key for P'. Verification is done by filling up the remaining wildcards with random messages to create a vector of messages M, encrypting a random message under identity M, and checking whether decryption using the signature as secret key returns the correct message. In fact, one can easily see that the schemes discussed here allow for more efficient deterministic verification algorithms.

Wildcard signatures can be seen as a special instance of homomorphic signatures [16, 14, 2, 15]. Their relation to wicked IBE is particularly reminiscent of the relation between HIBS schemes and append-only signatures [15]. They can also be seen as the dual of redactable signatures [14] that allow anyone to erase parts of a signed message without invalidating the signature.

A fairly simple, generic construction from standard signatures also exists. Namely, for each wildcard in the message the signer generates a fresh key pair, and then signs the message together with all generated public keys. The overall signature also contains the public and secret keys corresponding to all wildcards. To replace a wildcard at position i with a concrete value, the i-th secret key is replaced with a signature on the new value under the i-th public key. The disadvantage of this generic construction is that signature length and verification time are both linear in the number of *original* wildcards in the message, even after these wildcards have been replaced with original values. The signature length and verification time of the scheme derived from the $\mathcal{BBG}$-$\mathcal{WKD}$-$\mathcal{IBE}$ scheme on the other hand is only linear in the number of wildcards that are still present in the message. Also, signatures generated by the generic construction are linkable in the sense that one can check whether a given signature was derived from a second one by filling in wildcards. The decryption keys of the $\mathcal{BBG}$-$\mathcal{WKD}$-$\mathcal{IBE}$ scheme, and therefore the signatures of the associated wildcard signature scheme, can be re-randomized to prevent this type of linkability.

Finally, one could even imagine *wicked wildcard signatures* that allow for wildcards in *both* the signers' identities and the messages being signed. Such schemes are easily constructed from a WKD-IBE scheme by using a different encoding for identity strings and messages, as was done in the construction of WKD-IBS schemes above.

Acknowledgements

The authors have been supported in part by the European Commission through the IST Program under Contract IST-2002-507932 ECRYPT. The first author was supported in part by France Telecom R&D as part of the contract CIDRE, between France Telecom R&D and École normale supérieure. The second author was supported in part by the research program Sentinels (`http://www.sentinels.nl`). Sentinels is being financed by Technology Foundation STW, the Netherlands Organization for Scientific Research (NWO), and the Dutch Ministry of Economic Affairs. The third author is a Postdoctoral Fellow of the Research Foundation – Flanders (FWO-Vlaanderen), and was supported in part by the European Commission through IST Program under Contract IST-2006-034238 SPEED, and in part by the IAP Programme P6/26 BCRYPT of the Belgian State (Belgian Science Policy).

References

[1] Michel Abdalla, Dario Catalano, Alex Dent, John Malone-Lee, Gregory Neven, and Nigel Smart. Identity-based encryption gone wild. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *ICALP 2006: 33rd International Colloquium on Automata, Languages and Programming, Part II*, volume 4052 of *Lecture Notes in Computer Science*, pages 300–311. Springer-Verlag, Berlin, Germany, July 9–16, 2006.

[2] Mihir Bellare and Gregory Neven. Transitive signatures: new schemes and proofs. *IEEE Transactions on Information Theory*, 51(6):2133–2151, 2005.

[3] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *ACM CCS 93: 1st Conference on Computer and Communications Security*, pages 62–73, Fairfax, Virginia, USA, November 3–5, 1993. ACM Press.

[4] Dan Boneh and Xavier Boyen. Efficient selective-ID secure identity based encryption without random oracles. In Christian Cachin and Jan Camenisch, editors, *Advances in Cryptology – EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 223–238, Interlaken, Switzerland, May 2–6, 2004. Springer-Verlag, Berlin, Germany.

[5] Dan Boneh, Xavier Boyen, and Eu-Jin Goh. Hierarchical identity based encryption with constant size ciphertext. In Ronald Cramer, editor, *Advances in Cryptology – EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 440–456, Aarhus, Denmark, May 22–26, 2005. Springer-Verlag, Berlin, Germany.

[6] Dan Boneh and Matthew K. Franklin. Identity based encryption from the Weil pairing. *SIAM Journal on Computing*, 32(3):586–615, 2003.

[7] Dan Boneh, Craig Gentry, and Brent Waters. Collusion resistant broadcast encryption with short ciphertexts and private keys. In Victor Shoup, editor, *Advances in Cryptology – CRYPTO 2005*, volume 3621 of *Lecture Notes in Computer Science*, pages 258–275, Santa Barbara, CA, USA, August 14–18, 2005. Springer-Verlag, Berlin, Germany.

[8] Sanjit Chatterjee and Palash Sarkar. Multi-receiver identity-based key encapsulation with shortened ciphertext. In Rana Barua and Tanja Lange, editors, *Progress in Cryptology – INDOCRYPT 2006*, volume 4329 of *Lecture Notes in Computer Science*, pages 394–408, Kolkata, India, December 11–13, 2006. Springer-Verlag, Berlin, Germany.

[9] Yevgeniy Dodis and Nelly Fazio. Public key broadcast encryption for stateless receivers. In *Digital Rights Management Workshop*, volume 2696 of *Lecture Notes in Computer Science*, pages 61–80. Springer, November 2002.

[10] Amos Fiat and Moni Naor. Broadcast encryption. In Douglas R. Stinson, editor, *Advances in Cryptology – CRYPTO'93*, volume 773 of *Lecture Notes in Computer Science*, pages 480–491, Santa Barbara, CA, USA, August 22–26, 1994. Springer-Verlag, Berlin, Germany.

[11] Craig Gentry and Alice Silverberg. Hierarchical ID-based cryptography. In Yuliang Zheng, editor, *Advances in Cryptology – ASIACRYPT 2002*, volume 2501 of *Lecture Notes in Computer Science*, pages 548–566, Queenstown, New Zealand, December 1–5, 2002. Springer-Verlag, Berlin, Germany.

[12] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In *ACM CCS 06: 13th Conference on Computer and Communications Security*. ACM Press, 2006. Available as Cryptology ePrint Archive Report 2006/309.

[13] Jeremy Horwitz and Ben Lynn. Toward hierarchical identity-based encryption. In Lars R. Knudsen, editor, *Advances in Cryptology – EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 466–481, Amsterdam, The Netherlands, April 28 – May 2, 2002. Springer-Verlag, Berlin, Germany.

[14] Robert Johnson, David Molnar, Dawn Xiaodong Song, and David Wagner. Homomorphic signature schemes. In Bart Preneel, editor, *Topics in Cryptology – CT-RSA 2002*, volume 2271 of *Lecture Notes in Computer Science*, pages 244–262, San Jose, CA, USA, February 18–22, 2002. Springer-Verlag, Berlin, Germany.

[15] Eike Kiltz, Anton Mityagin, Saurabh Panjwani, and Barath Raghavan. Append-only signatures. In Luís Caires, Giuseppe F. Italiano, Luís Monteiro, Catuscia Palamidessi, and Moti Yung, editors, *ICALP 2005: 32nd International Colloquium on Automata, Languages and Programming*, volume 3580 of *Lecture Notes in Computer Science*, pages 434–445, Lisbon, Portugal, July 11–15, 2005. Springer-Verlag, Berlin, Germany.

[16] Ronald Rivest. Two signature schemes. Slides from talk given at Cambridge University, October 17, 2000., 2000.

[17] Ryuichi Sakai, Kiyoshi Ohgishi, and Masao Kasahara. Cryptosystems based on pairing. In *SCIS 2000*, Okinawa, Japan, January 2000.

[18] Hovav Shacham. The BBG HIBE has limited delegation. Cryptology ePrint Archive, Report 2007/201, 2007. http://eprint.iacr.org/.

[19] Adi Shamir. Identity-based cryptosystems and signature schemes. In G. R. Blakley and David Chaum, editors, *Advances in Cryptology – CRYPTO'84*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53, Santa Barbara, CA, USA, August 19–23, 1985. Springer-Verlag, Berlin, Germany.

[20] Brent R. Waters. Efficient identity-based encryption without random oracles. In Ronald Cramer, editor, *Advances in Cryptology – EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 114–127, Aarhus, Denmark, May 22–26, 2005. Springer-Verlag, Berlin, Germany.