Applicability of OR-Proof Techniques to Hierarchical Identity-Based Identification

Atsushi Fujioka[1], Taiichi Saito[2], and Keita Xagawa[1]

[1] NTT Secure Platform Laboratories,
3-9-11 Midori-cho, Musashino-shi, Tokyo 180-8585, Japan
{fujioka.atsushi,xagawa.keita}@lab.ntt.co.jp
[2] Tokyo Denki University,
5 Senju Asahi-cho, Adachi-ku, Tokyo 120-8551, Japan
taiichi@c.dendai.ac.jp

Abstract. We discuss the applicability of the well known OR-proof technique to hierarchical identity-based identification (HIBI) protocols for enhancing their security. We first describe formal security definitions for HIBI protocol not only in the adaptive hierarchical-identity setting but also in both "static" and "weak selective" hierarchical-identity settings. Next, we investigate whether the security enhancement transformations for identity-based identifications presented at ACNS 2012, which is based on the OR-proof technique, can be applied to HIBI protocols. We formally prove that several of these transformations are applicable to HIBI with slight modification. Curiously, the rest do not seem applicable, which stems from hierarchy and delegation. We also present a variant transformation and show that it can enhance the security of HIBI protocols in all three hierarchical-identity settings.

Keywords: hierarchical identity-based identification, OR-proof, impersonation under concurrent attacks.

1 Introduction

Identification is a protocol between a *prover* and a *verifier* through which the prover tries to convince the verifier of his/her identity. The security of identification protocols is defined by an experiment consisting of *learning* and *challenge phases*. In the learning phase, an adversary acts as many verifiers to gather much information and in the challenge phase, acts as a prover to impersonate an entity. A strong security model of identification protocols is formulated as a *model against impersonation under concurrent attacks* [2], in which an adversary is allowed to *concurrently* access entities who prove their identities, even in the challenge phase. On the other hand, in the security *model against impersonation under passive attacks* [2], an adversary is only allowed to eavesdrop on identification communications in the learning phase.

After the proposal of identity-based cryptography [13], identification in the identity-based setting, called *identity-based identification* (IBI), has also been

J. Pieprzyk, A.-R. Sadeghi, and M. Manulis (Eds.): CANS 2012, LNCS 7712, pp. 169–184, 2012.

investigated. In an IBI protocol, the existence of a *private key generator* (PKG) is assumed as well as in other identity-based cryptographic schemes. The PKG generates a secret key corresponding to the inputted identity of an entity, and gives the secret key to the entity. For IBI protocols, we can consider other types of attack models with respect to an adversary's selection of identities. Under *adaptive identity attacks* (i.e., in the adapt-id-imp-atk security model) [1,12], an adversary is allowed to adaptively ask identities to oracles. Under *static identity attacks* (i.e., in the stat-id-imp-atk security model) [12], an adversary declares, only at the beginning of the learning phase, the identities of all entities to be in queries or challenged. Under *weak selective identity attacks* (i.e., in the wsid-imp-atk security model) [14], an adversary requests secret keys of identities only at the beginning of the learning phase. Here, atk denotes a type of attack such that $\mathsf{atk} \in \{\mathsf{pa}, \mathsf{ca}\}$, and pa and ca mean *passive attack* and *concurrent attack*, respectively.

It is natural to extend IBI to identification in the hierarchical identity setting, called *hierarchical identity-based identification* (HIBI). In an HIBI protocol, the single PKG functionality of generating secret keys is divided into partial ones and the divided functionalities are delegated to multiple PKGs. If a PKG is assigned a hierarchical identity, $\mathsf{ID}^{(k-1)} = (I_1, \ldots, I_{k-1})$, and given a secret key, $sk_{\mathsf{ID}^{(k-1)}}$, corresponding to the hierarchical identity, then it can generate a secret key, $sk_{\mathsf{ID}^{(k)}}$, corresponding to a hierarchical identity, $\mathsf{ID}^{(k)} = (I_1, \ldots, I_k)$. We may omit the word "hierarchical" to indicate a hierarchical identity if its meaning is clear and denote a (hierarchical) identity by ID if we do not need to specify its hierarchy depth. Since IBI has three phases, HIBI also has three: Setup, Extract, and Identification.

Security for HIBI. The first security formulation for HIBI was given by Chin, Heng, and Goi [3] with their first proposal of an HIBI protocol. However, they formulated the passive and concurrent security only under adaptive hierarchical-identity attacks. Therefore, we can consider three types of attack models regarding an adversary's selection of hierarchical identities for HIBI protocols, as well as for IBI protocols. One is called *security against impersonation under adaptive hierarchical-identity attacks* (adapt-hid-imp-atk security), which is an extension of adapt-id-imp-atk security. The second one is called *security against impersonation under static hierarchical-identity attacks* (stat-hid-imp-atk security), which is an extension of stat-id-imp-atk security and in which an adversary requests secret keys of hierarchical identities only at the beginning of the learning phase. The third one is called *security against impersonation under weak selective hierarchical-identity attacks* (wshid-imp-atk security), which is an extension of wsid-imp-atk security and in which an adversary declares, only at the beginning of the learning phase, the hierarchical identities of all entities to be in queries or challenged.

Security Enhancement Transformations of IBI. It is well known that the OR-proof technique [5,4] enhances the security of not only standard identification

but also IBI protocols from passive security to the concurrent security [11,6]. Thus, we expect that the OR-proof technique can be also applied to HIBI protocols to enhance their security. Actually, HIBI protocols proposed in [7] utilize the OR-proof technique, and they are concurrently secure.

In [6], Fujioka, Saito, and Xagawa investigated OR-proof techniques for IBI protocols, which are formulated as three transformations, *DIsk*, *MI*, and *DPsk transformations* with double-key variants *DIdk* and *DPdk transformations*.[1] The authors proved that all the transformations can enhance an adapt-id-imp-pa (resp. wsid-imp-pa) secure IBI protocol to an adapt-id-imp-ca (resp. wsid-imp-ca) secure one, and that the DPsk, and DPdk transformations can enhance a stat-id-imp-pa secure IBI protocol to a stat-id-imp-ca secure one [6].

Our Contributions. We formally define *security against impersonation under static hierarchical-identity attacks* (stat-hid-imp-atk security) and *security against impersonation under weak selective hierarchical-identity attacks* (wshid-imp-atk security), along with the existing adapt-hid-imp-atk security, where atk denotes a type of attack such that $\mathsf{atk} \in \{\mathsf{pa}, \mathsf{ca}\}$.

Next we introduce two properties of HIBI protocols, which are extensions of the Σ^+*-type* and Σ^**-type* properties defined for IBI protocols [6]. The requirement of Σ^+-type is weaker than that of Σ^*-type in HIBI.

We examine whether the transformations discussed in [6] can be applied to HIBI protocols to enhance their security, and show the following points:

- To apply the DIdk transformation to a Σ^+-type HIBI protocol, we need a slight modification in choosing (imaginary) identities.
- The DIdk, MI, and DPdk transformations can convert an adapt-hid-imp-pa (resp. wshid-imp-pa) secure Σ^+-type HIBI protocol to an adapt-hid-imp-ca (resp. wshid-imp-ca) secure one.
- The DPdk transformation can convert a stat-hid-imp-pa secure Σ^*-type HIBI protocol to a stat-hid-imp-ca secure one.
- It seems difficult to enhance the passive security of an HIBI protocol in the static hierarchical-identity attack model by the DIdk and MI transformations.
- It seems difficult to enhance the passive security of an HIBI protocol in all the three hierarchical-identity attack models by the DIsk and DPsk transformations.

We also present a modified version of the DIdk transformation (named *mDIdk transformation*) and show that the mDIdk transformation can also convert a stat-hid-imp-pa secure HIBI protocol to a stat-hid-imp-ca secure one. While it is an open problem [6] whether there exists an OR-proof security enhancement

[1] DIsk, MI and DPsk stand for *Dual-Identity single-key*, *Master-Identity*, and *Double-Parameter single-key*, respectively. Also DIdk and DPsk stand for *Dual-Identity double-key* and *Double-Parameter double-key*, respectively [6]. The "double-parameter" means two master public keys, and "single/double-key" indicates the numbers of user's secret keys.

transformation based on a single master public key that converts a stat-id-imp-pa secure IBI protocol to a stat-id-imp-ca one, there exists an OR-proof security enhancement transformation based on a single master public key for stat-hid-imp-pa secure HIBI protocols.

For the summary and comparison, see Table 1.

Table 1. Applicability of Transformations

	HIBI [this paper]			IBI [6]		
	adapt-hid	stat-hid	wshid	adapt-id	stat-id	wsid
DIsk				✓		✓
DIdk	✓		✓	✓		✓
MI	✓		✓	✓		✓
DPsk				✓	✓	✓
DPdk	✓	✓	✓	✓	✓	✓
mDIdk	✓	✓	✓			

2 Definitions

We formally define hierarchical identity-based identification (HIBI) protocols and their security by following the formal definition of IBI protocols [1].

Hierarchical Identity-Based Identification. Two types of formal definitions for key generation in hierarchical identity-based cryptography have been proposed. One consists of three algorithms, *Root Setup, Lower-level Setup*, and *Extraction*, as in the Gentry-Silverberg hierarchical identity-based encryption (HIBE) scheme [9], and the other consists of two algorithms, *root-key-generation algorithm* and *node-key-generation algorithm*, as in the Horwitz-Lynn HIBE scheme [10]. The two types of definitions are essentially the same. We adopt the formal definition of HIBI protocols proposed by Chin et al. [3]. Note that their key generation is the former type, but we here describe it in the latter type.

Let $\mathsf{HIBI} = (\mathsf{SetUp}, \mathsf{KG}, \mathsf{P}, \mathsf{V})$ be an HIBI protocol, and κ denote the security parameter. In HIBI, SetUp is the root-key-generation algorithm that on input 1^κ outputs mpk and msk. To simplify notation, we set $sk_{\mathsf{ID}^{(0)}} = msk$. KG is the node-key-generation algorithm that on input $(mpk, sk_{\mathsf{ID}^{(k-1)}}, \mathsf{ID}^{(k)})$ outputs $sk_{\mathsf{ID}^{(k)}}$, P is the prover algorithm that takes mpk, ID, and sk_{ID} as inputs and interacts with V, and V is the verifier algorithm that takes mpk and ID as inputs, interacts with P, and finally outputs $dec \in \{accept, reject\}$, where $\mathsf{ID}^{(k-1)} = (I_1, \ldots, I_{k-1})$, $\mathsf{ID}^{(k)} = (I_1, \ldots, I_k)$. Thus, SetUp is used in Setup, KG is used in Extract, and P and V are used in Identification. Throughout this paper, we denote $\mathsf{pref}(\mathsf{ID})$ as the set of all prefixes of ID, i.e., $\mathsf{pref}(\mathsf{ID}) = \{(I_1), (I_1, I_2), \ldots, (I_1, \ldots, I_{k-1}), (I_1, \ldots, I_k)\}$ when $\mathsf{ID} = (I_1, \ldots, I_k)$. Note that $\mathsf{pref}(\mathsf{ID})$ includes ID.

We describe the formal definitions of the security of HIBI based on the following experiment $\mathbf{Exp}^{\mathsf{adapt\text{-}hid\text{-}imp\text{-}atk}}_{\mathsf{HIBI},\mathcal{I}}(\kappa)$ between a challenger and an impersonator $\mathcal{I} = (\mathsf{CV}, \mathsf{CP})$, where atk denotes a type of attack such that $\mathsf{atk} \in \{\mathsf{pa}, \mathsf{ca}\}$.

Experiment $\mathbf{Exp}^{\mathsf{adapt\text{-}hid\text{-}imp\text{-}atk}}_{\mathsf{HIBI},\mathcal{I}}(\kappa)$**:**

Setup Phase: The challenger obtains $(mpk,\ msk) \leftarrow \mathsf{SetUp}(1^\kappa)$ and initializes $HU,\ CU,\ TU,\ PS \leftarrow \emptyset$, where HU, CU, and TU denote the sets of honest users, corrupted users, and target users, respectively, and PS denotes the set of provers' sessions. The impersonator CV is given the security parameter 1^κ and the master public key mpk.

Learning Phase: The CV can ask queries to the Init, Corr, and Conv oracles when $\mathsf{atk} = \mathsf{pa}$ and also to Prov when $\mathsf{atk} = \mathsf{ca}$. Note that $\mathsf{ID} \notin HU \setminus TU$ means that ID is a target identity, a prefix of target identity, corrupted identity, or non-initiated identity.

- The oracle Init receives input $\mathsf{ID}^{(k)}$. If $\mathsf{ID}^{(k)} \in HU \cup CU \cup TU$, then it returns $\perp$. Otherwise, it computes k secret keys $sk_{\mathsf{ID}^{(1)}}, \ldots, sk_{\mathsf{ID}^{(k)}}$ by running $sk_{\mathsf{ID}^{(i)}} \leftarrow \mathsf{KG}(mpk,\ sk_{\mathsf{ID}^{(i-1)}},\ \mathsf{ID}^{(i)})$ if all prefixes of $\mathsf{ID}^{(k)}$ are not in HU, adds $\mathsf{pref}(\mathsf{ID}^{(k)})$ to HU, and provides the CV with $\mathsf{ID}^{(k)}$. If some prefixes of $\mathsf{ID}^{(k)}$ are in HU and if $\mathsf{ID}^{(j)}$ is the longest one in the prefixes, it computes $k-j$ secret keys $sk_{\mathsf{ID}^{(j+1)}}, \ldots, sk_{\mathsf{ID}^{(k)}}$.
- The oracle Corr receives input ID. If $\mathsf{ID} \notin HU \setminus TU$, then it returns $\perp$. Otherwise, it deletes all ID's in HU such that $\mathsf{ID} \in \mathsf{pref}(\mathsf{ID}')$, adds them to CU, and returns sk_{ID} to the CV.
- The oracle Conv receives input ID. If $\mathsf{ID} \notin HU$, then it returns $\perp$. Otherwise it returns a transcript of a transaction between the prover with identity ID and a verifier.
- (only when $\mathsf{atk} = \mathsf{ca}$) The oracle Prov receives inputs ID, s, and M_{in}. If $\mathsf{ID} \notin HU \setminus TU$, then it returns $\perp$. If $(\mathsf{ID},\ s) \notin PS$, then it adds $(\mathsf{ID},\ s)$ to PS, selects a random coin ρ, and sets a state of the prover $st_{\mathsf{P}}[(\mathsf{ID}, s)] \leftarrow (mpk,\ sk_{\mathsf{ID}},\ \rho)$. Next, it obtains $(M_{out},\ st_{\mathsf{P}}[(\mathsf{ID}, s)]) \leftarrow \mathsf{P}(M_{in},\ st_{\mathsf{P}}[(\mathsf{ID}, s)])$. Finally, it returns M_{out}. Note that we require that $\mathsf{ID} \notin HU \setminus TU$ since we do not consider man-in-the-middle attacks [8] in this paper.

Challenge Phase: The CV outputs a target identity ID^* and state information st_{CP}. If ID^* is not in HU, then the challenger outputs *reject* and halts. Otherwise, the challenger sets $TU \leftarrow \mathsf{pref}(\mathsf{ID}^*)$ and gives st_{CP} to CP. CP can ask queries to Init, Corr, and Conv, (and Prov when $\mathsf{atk} = \mathsf{ca}$) as in the learning phase. Finally, the challenger obtains $(tr,\ dec) \leftarrow \mathbf{Run}[\mathsf{CP}(st_{\mathsf{CP}})^{\textsc{Init},\textsc{Corr},\textsc{Conv}(,\textsc{Prov})} \leftrightarrow \mathsf{V}(mpk,\ \mathsf{ID}^*)]$ and outputs dec.

In these experiments, the impersonator is allowed to obtain a secret key of an adaptively chosen (hierarchical) identity and a transcript of a transaction between the prover of an adaptively chosen (hierarchical) identity and a verifier. In the case of $\mathsf{atk} = \mathsf{ca}$, the Prov oracle allows multiple sessions at the same time.

Definition 2.1. *Let* $\mathsf{HIBI} = (\mathsf{SetUp}, \mathsf{KG}, \mathsf{P}, \mathsf{V})$ *be an HIBI protocol and* $\mathcal{I} = (\mathsf{CV}, \mathsf{CP})$ *an impersonator. Let* κ *be a security parameter. The advantage of* $\mathcal{I}$ *in attacking* HIBI *is defined by*

$$\mathbf{Adv}^{\mathsf{adapt\text{-}hid\text{-}imp\text{-}atk}}_{\mathsf{HIBI},\mathcal{I}}(\kappa) := \Pr\left[\ \mathbf{Exp}^{\mathsf{adapt\text{-}hid\text{-}imp\text{-}atk}}_{\mathsf{HIBI},\mathcal{I}}(\kappa) = \mathit{accept}\ \right].$$

We say that HIBI *is* secure against impersonation under adaptive hierarchical-identity and concurrent attacks (adapt-hid-imp-ca secure) *if* $\mathbf{Adv}^{\mathsf{adapt\text{-}hid\text{-}imp\text{-}ca}}_{\mathsf{HIBI},\mathcal{I}}(\kappa)$ *is negligible for every polynomial-time* $\mathcal{I}$ *and is* secure against impersonation under adaptive hierarchical-identity and passive attacks (adapt-hid-imp-pa secure) *if* $\mathbf{Adv}^{\mathsf{adapt\text{-}hid\text{-}imp\text{-}pa}}_{\mathsf{HIBI},\mathcal{I}}(\kappa)$ *is negligible for every polynomial-time* $\mathcal{I}$.

Static and Weak Selective Hierarchical-Identity Attack Models. Following Rückert [12] and Yang et al. [14], we describe two other security definitions, which are weaker than the adapt-hid-imp-atk security, of HIBI based on the following experiments, $\mathbf{Exp}^{\mathsf{stat\text{-}hid\text{-}imp\text{-}atk}}_{\mathsf{HIBI},\mathcal{I}}(\kappa)$ and $\mathbf{Exp}^{\mathsf{wshid\text{-}imp\text{-}atk}}_{\mathsf{HIBI},\mathcal{I}}(\kappa)$ ($\mathsf{atk} \in \{\mathsf{pa}, \mathsf{ca}\}$), between a challenger and impersonator $\mathcal{I} = (\mathsf{CV}, \mathsf{CP})$.

Experiment $\mathbf{Exp}^{\mathsf{stat\text{-}hid\text{-}imp\text{-}atk}}_{\mathsf{HIBI},\mathcal{I}}(\kappa)$**:**

- **Setup Phase:** At the beginning of this phase, the CV on input 1^κ issues a single corrupt query $(\mathsf{ID}_1, \ldots, \mathsf{ID}_t)$ to the challenger before receiving the master public key. The challenger is given the security parameter 1^κ, obtains $(mpk, msk) \leftarrow \mathsf{SetUp}(1^\kappa)$, and computes $sk_{\mathsf{ID}_i} \leftarrow \mathsf{KG}(mpk, msk, \mathsf{ID}_i)$ $(1 \le i \le t)$. It sets $CU \leftarrow \{\mathsf{ID}_1, \mathsf{ID}_2, \ldots, \mathsf{ID}_t\}$ and then returns $(sk_{\mathsf{ID}_1}, \ldots, sk_{\mathsf{ID}_t})$ to the CV. The challenger initializes HU, TU, and $PS \leftarrow \emptyset$. The CV is given the master public key mpk.
- **Learning and Challenge Phases:** The learning and challenge phases are defined the same as those in experiment $\mathbf{Exp}^{\mathsf{adapt\text{-}hid\text{-}imp\text{-}atk}}_{\mathsf{HIBI},\mathcal{I}}(\kappa)$, except that impersonator $\mathcal{I}$ is not allowed additional queries to Corr during these phases.

Experiment $\mathbf{Exp}^{\mathsf{wshid\text{-}imp\text{-}atk}}_{\mathsf{HIBI},\mathcal{I}}(\kappa)$**:**

- **Setup Phase:** At the beginning of this phase, the CV on input 1^κ issues a single initialization query $(\mathsf{ID}_1, \ldots, \mathsf{ID}_t)$ to the challenger before receiving the master public key. The challenger is given the security parameter 1^κ and obtains $(mpk, msk) \leftarrow \mathsf{SetUp}(1^\kappa)$. It sets $HU \leftarrow \bigcup_{i=1}^{t} \mathsf{pref}(\mathsf{ID}_1)$ and provides the CV with $(\mathsf{ID}_1, \ldots, \mathsf{ID}_t)$. The challenger initializes CU, TU, and $PS \leftarrow \emptyset$. The CV is given the master public key mpk.
- **Learning and Challenge Phases:** The learning and challenge phases are defined the same as those in experiment $\mathbf{Exp}^{\mathsf{adapt\text{-}hid\text{-}imp\text{-}atk}}_{\mathsf{HIBI},\mathcal{I}}(\kappa)$, except that $\mathcal{I}$ is not allowed additional queries to the Init oracle during these phases.

In the stat-hid-imp-atk experiment, the impersonator has to choose all (hierarchical) identities that it wants to corrupt at the beginning of the experiment. After that, it is allowed to access oracles except for Corr. In the wshid-imp-atk experiment, the impersonator has to select all (hierarchical) identities that it

wants to initialize at the beginning of the experiment. Then, it is allowed to send queries of only the (hierarchical) identities chosen at the beginning,

Let $\mathsf{HIBI} = (\mathsf{SetUp}, \mathsf{KG}, \mathsf{P}, \mathsf{V})$ be an HIBI protocol and $\mathcal{I} = (\mathsf{CV}, \mathsf{CP})$ an impersonator. Let κ be a security parameter. The advantages of $\mathcal{I}$ in attacking HIBI are defined as

$$\mathbf{Adv}^{\mathsf{stat\text{-}hid\text{-}imp\text{-}atk}}_{\mathsf{HIBI},\mathcal{I}}(\kappa) := \Pr\left[\ \mathbf{Exp}^{\mathsf{stat\text{-}hid\text{-}imp\text{-}atk}}_{\mathsf{HIBI},\mathcal{I}}(\kappa) = \mathit{accept}\ \right] \text{ and}$$

$$\mathbf{Adv}^{\mathsf{wshid\text{-}imp\text{-}atk}}_{\mathsf{HIBI},\mathcal{I}}(\kappa) := \Pr\left[\ \mathbf{Exp}^{\mathsf{wshid\text{-}imp\text{-}atk}}_{\mathsf{HIBI},\mathcal{I}}(\kappa) = \mathit{accept}\ \right].$$

We say that HIBI is *secure against impersonation under static (resp. weak selective) hierarchical-identity and concurrent attacks (*$\mathsf{stat\text{-}hid\text{-}imp\text{-}ca}$ *(resp.* $\mathsf{wshid\text{-}imp\text{-}ca}$*) secure)* if $\mathbf{Adv}^{\mathsf{stat\text{-}hid\text{-}imp\text{-}ca}}_{\mathsf{HIBI},\mathcal{I}}(\kappa)$ (resp. $\mathbf{Adv}^{\mathsf{wshid\text{-}imp\text{-}ca}}_{\mathsf{HIBI},\mathcal{I}}(\kappa)$) is negligible for every polynomial-time $\mathcal{I}$, and is *secure against impersonation under static (resp. weak selective) hierarchical-identity and passive attacks (*$\mathsf{stat\text{-}hid\text{-}imp\text{-}pa}$ *(resp.* $\mathsf{wshid\text{-}imp\text{-}ca}$*) secure)* if $\mathbf{Adv}^{\mathsf{stat\text{-}hid\text{-}imp\text{-}pa}}_{\mathsf{HIBI},\mathcal{I}}(\kappa)$ (resp. $\mathbf{Adv}^{\mathsf{wshid\text{-}imp\text{-}pa}}_{\mathsf{HIBI},\mathcal{I}}(\kappa)$) is negligible for every polynomial-time $\mathcal{I}$.

Σ^+- and Σ^*-Type HIBI Protocols. We define two analogues of Σ-type IBI-protocols [6] in the context of HIBI protocols. Let $\mathsf{HIBI} = (\mathsf{SetUp}, \mathsf{KG}, \mathsf{P}, \mathsf{V})$ be an HIBI protocol. Suppose that P and V interact by using four probabilistic polynomial-time algorithms $(\Sigma_{\mathsf{hibi\text{-}com}}, \Sigma_{\mathsf{hibi\text{-}ch}}, \Sigma_{\mathsf{hibi\text{-}res}}, \Sigma_{\mathsf{hibi\text{-}vrfy}})$ as follows:

$\mathsf{P} \to \mathsf{V}$: P computes $(a, st) \leftarrow \Sigma_{\mathsf{hibi\text{-}com}}(mpk, \mathsf{ID}, sk_{\mathsf{ID}})$ and sends a to V.
$\mathsf{V} \to \mathsf{P}$: V computes $c \leftarrow \Sigma_{\mathsf{hibi\text{-}ch}}(mpk, \mathsf{ID})$ and sends c to P.
$\mathsf{P} \to \mathsf{V}$: P computes $z \leftarrow \Sigma_{\mathsf{hibi\text{-}res}}(mpk, \mathsf{ID}, sk_{\mathsf{ID}}, a, c, st)$ and sends z to V.
V: V computes $dec \leftarrow \Sigma_{\mathsf{hibi\text{-}vrfy}}(mpk, \mathsf{ID}, a, c, z)$ and outputs $dec \in \{\mathit{accept}, \mathit{reject}\}$.

We call this type of three-move HIBI protocol *canonical* [2]. We also call an HIBI protocol HIBI *Σ^+-type* if it is canonical and satisfies the following three properties: *special zero-knowledge*, *special soundness*, and *special challenge*:

Special Zero-Knowledge: We can obtain an accepting transcript from a challenge c, mpk, and ID. That is, there is a probabilistic polynomial-time algorithm $\Sigma_{\mathsf{hibi\text{-}sim}}$ that takes on input mpk, ID, and c such that $c \leftarrow \Sigma_{\mathsf{hibi\text{-}ch}}(mpk, \mathsf{ID})$ and outputs (a, z) such that $\mathit{accept} = \Sigma_{\mathsf{hibi\text{-}vrfy}}(mpk, \mathsf{ID}, a, c, z)$. The distribution of transcripts generated by $\Sigma_{\mathsf{hibi\text{-}ch}}$ and $\Sigma_{\mathsf{hibi\text{-}sim}}$ is indistinguishable from that of real transcripts.

Special Soundness: We can compute the user secret key sk_{ID} for an identity ID from mpk, ID, and two accepting transcripts (a, c, z) and $(a, \tilde{c}, \tilde{z})$ such that $c \neq \tilde{c}$. That is, there is a probabilistic polynomial-time algorithm $\Sigma_{\mathsf{hibi\text{-}ext}}$ that takes as input mpk, ID, and two transcripts (a, c, z) and $(a, \tilde{c}, \tilde{z})$ satisfying $\mathit{accept} = \Sigma_{\mathsf{hibi\text{-}vrfy}}(mpk, \mathsf{ID}, a, c, z) = \Sigma_{\mathsf{hibi\text{-}vrfy}}(mpk, \mathsf{ID}, a, \tilde{c}, \tilde{z})$ and $c \neq \tilde{c}$, and outputs sk_{ID}.

Special Challenge: $\Sigma_{\mathsf{hibi\text{-}ch}}$ depends only on mpk, not on (mpk, ID), and the output c is uniformly distributed over a commutative group $\mathbb{G}$. In addition, the

group operation + is computable in polynomial time, and $\mathbb{G}$ is determined only by mpk, not by (mpk, ID). That is, there is a probabilistic polynomial time algorithm $\Sigma^{+}_{\mathsf{hibi\text{-}ch}}$ such that it takes as input mpk (without ID) and outputs c, and c is uniformly distributed over $\mathbb{G}$.

We call an HIBI protocol HIBI Σ^{*}-*type* if the special challenge property is replaced with the following property:

Strongly Special Challenge: $\Sigma_{\mathsf{hibi\text{-}ch}}$ depends only on 1^{κ}, not on mpk, and the output c is uniformly distributed over $\mathbb{G}$. In addition, + is computable in polynomial time, and $\mathbb{G}$ is determined only by 1^{κ}, not by mpk. That is, there is a probabilistic polynomial time algorithm $\Sigma^{*}_{\mathsf{hibi\text{-}ch}}$ such that it takes as input 1^{κ} (not mpk) and outputs c, and c is uniformly distributed over $\mathbb{G}$.

3 Security Enhancement Transformations

In this section, we describe three security enhancement transformations, DIdk, MI, and DPdk, and show that they can enhance the passive security of an HIBI protocol to the concurrent security in both adaptive and weak selective hierarchical-identity settings. Next, we present a modified version of the DIdk transformation (mDIdk) and show that the DPdk and mDIdk transformation can enhance a $\mathsf{stat\text{-}hid\text{-}imp\text{-}pa}$ secure HIBI protocol to a $\mathsf{stat\text{-}hid\text{-}imp\text{-}ca}$ secure one. In addition, we discuss the difficulty in enhancing security in the static hierarchical-identity models by the DIdk, DIsk, MI, DPsk, or a single key variant of the mDIdk (mDIsk) transformation.

Let $\mathsf{HIBI}' = (\mathsf{SetUp}', \mathsf{KG}', \mathsf{P}', \mathsf{V}')$ be a Σ^{+}-type (Σ^{*}-type) HIBI protocol in which $(\mathsf{P}', \mathsf{V}')$ use four probabilistic polynomial time algorithms $\Sigma_{\mathsf{hibi\text{-}com}}$, $\Sigma^{+}_{\mathsf{hibi\text{-}ch}}$ ($\Sigma^{*}_{\mathsf{hibi\text{-}ch}}$), $\Sigma_{\mathsf{hibi\text{-}res}}$, and $\Sigma_{\mathsf{hibi\text{-}vrfy}}$, and have the special zero-knowledge property with a probabilistic polynomial time algorithm $\Sigma_{\mathsf{hibi\text{-}sim}}$ and the special soundness with a probabilistic polynomial time algorithm $\Sigma_{\mathsf{hibi\text{-}ext}}$.

3.1 Dual-Identity Double-Key Transformation

We show a security enhancement transformation based on the OR-proof technique, applicable to the Σ^{+}-type HIBI protocol. We call this transformation *DIdk transformation*.

Let $i.\mathsf{ID}^{(k)}$ denote $(i||I_1, I_2, \ldots, I_k)$ when $\mathsf{ID}^{(k)} = (I_1, \ldots, I_k)$. In an HIBI protocol generated by the DIdk transformation, an entity of an identity ID is given two secret keys corresponding to imaginary (hierarchical) identities $0.\mathsf{ID}$ and $1.\mathsf{ID}$ for the underlying HIBI protocol, and shows his/her identity by proving that the imaginary (hierarchical) identity is either $0.\mathsf{ID}$ or $1.\mathsf{ID}$.

We describe an HIBI protocol $\mathsf{HIBI} = (\mathsf{SetUp}, \mathsf{KG}, \mathsf{P}, \mathsf{V})$ produced by applying the DIdk transformation to HIBI' in Fig. 1. Note that in the EXTRACT algorithm in this figure, we let $sk_{\mathsf{ID}^{(0)}} = (msk', msk')$.

Due to the special challenge property, c is an element in $\mathbb{G}$ determined by mpk', so are c_0 and c_1 since $c = c_0 + c_1$ and the operation + is defined in $\mathbb{G}$. Then, c_0 and c_1 are possible challenges under mpk'.

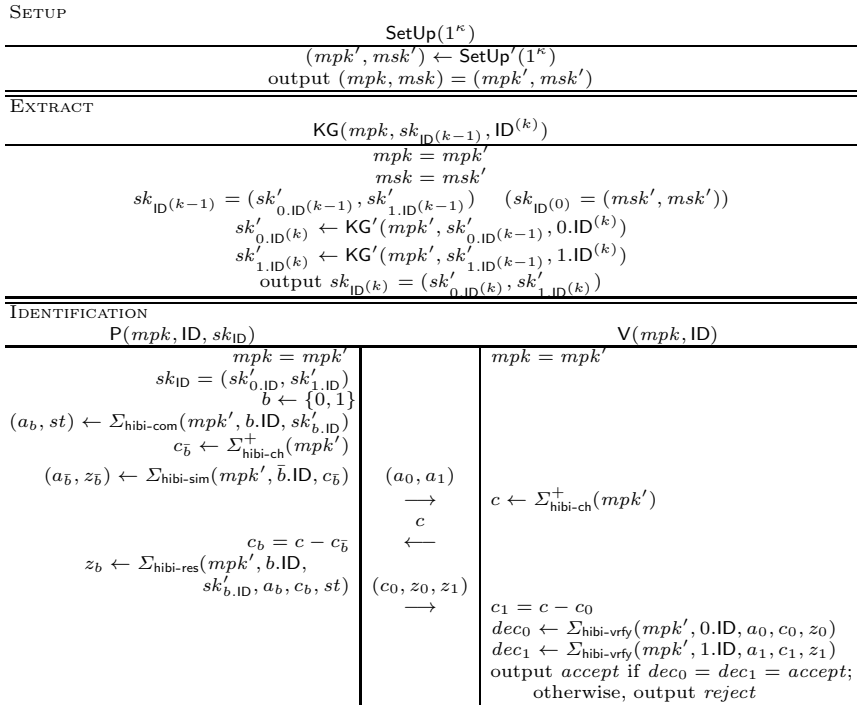

Fig. 1. DIdk Transformation

It is easy to have a variant of the DIdk transformation (DIsk transformation) such that each entity is given only either secret key of identities 0.ID or 1.ID, and the entity shows that it has either the secret key of 0.ID or 1.ID. However, it seems difficult to enhance the passive security of an HIBI protocol in any hierarchical-identity attack model with this variant.

3.2 Master-Identity Transformation

We show another security enhancement transformation based on the OR-proof technique, applicable to the Σ^+-type HIBI protocol. We call this transformation *MI transformation*.

In an HIBI protocol generated by the MI transformation, an entity with an identity ID is simply given a secret key corresponding to the identity ID for the underlying HIBI protocol, and the entity of the identity ID proves that his/her identity is either ID or an (imaginary) master identity.

We describe an HIBI protocol $\mathsf{HIBI} = (\mathsf{SetUp}, \mathsf{KG}, \mathsf{P}, \mathsf{V})$ produced by applying the MI transformation to HIBI' in Fig. 2. In the EXTRACT algorithm in this figure, we let $sk_{\mathsf{ID}^{(0)}} = msk'$.

Due to the special challenge property, c is an element in $\mathbb{G}$ determined by mpk', so are c_0 and c_1 since $c = c_0 + c_1$ and the operation $+$ is defined in $\mathbb{G}$. Then, c_0 and c_1 are possible challenges under mpk'.

SETUP

$\mathsf{SetUp}(1^\kappa)$

$(mpk', msk') \leftarrow \mathsf{SetUp}'(1^\kappa)$
choose a master identity ID_{master} from the set of identities of depth 1
output $(mpk, msk) = ((mpk', \mathsf{ID}_{master}), msk')$

EXTRACT

$\mathsf{KG}(mpk, sk_{\mathsf{ID}^{(k-1)}}, \mathsf{ID}^{(k)})$

$mpk = (mpk', \mathsf{ID}_{master})$
output $\perp$ if $\mathsf{ID}_{master} \in \mathsf{pref}(\mathsf{ID}^{(k)})$
$sk_{\mathsf{ID}^{(k-1)}} = sk'_{\mathsf{ID}^{(k-1)}} \quad (sk_{\mathsf{ID}^{(0)}} = msk')$
$sk'_{\mathsf{ID}^{(k)}} \leftarrow \mathsf{KG}'(mpk', sk'_{\mathsf{ID}^{(k-1)}}, \mathsf{ID}^{(k)})$
output $sk_{\mathsf{ID}^{(k)}} = sk'_{\mathsf{ID}^{(k)}}$

IDENTIFICATION

$\mathsf{P}(mpk, \mathsf{ID}, sk_{\mathsf{ID}})$		$\mathsf{V}(mpk, \mathsf{ID})$
$mpk = (mpk', \mathsf{ID}_{master})$		$mpk = (mpk', \mathsf{ID}_{master})$
$sk_{\mathsf{ID}} = sk'_{\mathsf{ID}}$		
$(a_0, st) \leftarrow \Sigma_{\mathsf{hibi\text{-}com}}(mpk', \mathsf{ID}, sk'_{\mathsf{ID}})$		
$c_1 \leftarrow \Sigma^+_{\mathsf{hibi\text{-}ch}}(mpk')$		
$(a_1, z_1) \leftarrow \Sigma_{\mathsf{hibi\text{-}sim}}(mpk', \mathsf{ID}_{master}, c_1)$	(a_0, a_1)	
	$\longrightarrow$	$c \leftarrow \Sigma^+_{\mathsf{hibi\text{-}ch}}(mpk')$
	c	
$c_0 = c - c_1$	$\longleftarrow$	
$z_0 \leftarrow \Sigma_{\mathsf{hibi\text{-}res}}(mpk', \mathsf{ID}, sk'_{\mathsf{ID}}, a_0, c_0, st)$	(c_0, z_0, z_1)	
	$\longrightarrow$	$c_1 = c - c_0$
		$dec_0 \leftarrow \Sigma_{\mathsf{hibi\text{-}vrfy}}(mpk', \mathsf{ID}, a_0, c_0, z_0)$
		$dec_1 \leftarrow \Sigma_{\mathsf{hibi\text{-}vrfy}}(mpk', \mathsf{ID}_{master}, a_1, c_1, z_1)$
		output *accept* if $dec_0 = dec_1 = accept$s; otherwise, output *reject*

Fig. 2. MI Transformation

Here, ID_{master} is randomly chosen from the set of identities but does not coincide with any identities of real entities. It is clear that an impersonator should not be allowed to obtain the secret key of ID_{master}. In the construction of KG, $\mathsf{KG}(mpk, sk_{\mathsf{ID}^{(k-1)}}, \mathsf{ID}^{(k)})$ outputs $\perp$ if $\mathsf{ID}_{master} \in \mathsf{pref}(\mathsf{ID}^{(k)})$. This means that the space of all possible identities of entities does not include ID_{master}. Note that since ID_{master} is randomly chosen from the set of identities, it is a hierarchical identity of depth 1.

3.3 Double-Parameter Double-Key Transformation

We show the other security enhancement transformation based on the OR-proof technique, applicable only to the Σ^*-type HIBI protocol, not to the Σ^+-type. We call this transformation *DPdk transformation*.

In an HIBI protocol generated by the DPdk transformation, an entity of an identity ID is given secret keys corresponding to the identity based on both master public keys in the underlying HIBI protocol and proves that his/her (hierarchical) identity is ID under either master public key.

We describe an HIBI protocol $\mathsf{HIBI} = (\mathsf{SetUp}, \mathsf{KG}, \mathsf{P}, \mathsf{V})$ produced by applying the DIdk transformation to HIBI' in Fig. 3. In the EXTRACT algorithm in this figure, we let $sk_{\mathsf{ID}^{(0)}} = (msk'_0, msk'_1)$.

Due to the strongly special challenge property, c is an element in $\mathbb{G}$ determined only by 1^κ, so are c_0 and c_1 since $c = c_0 + c_1$ and the operation $+$ is defined in $\mathbb{G}$. Therefore, c_0 and c_1 are possible challenges under mpk'_0 and mpk'_1, respectively.

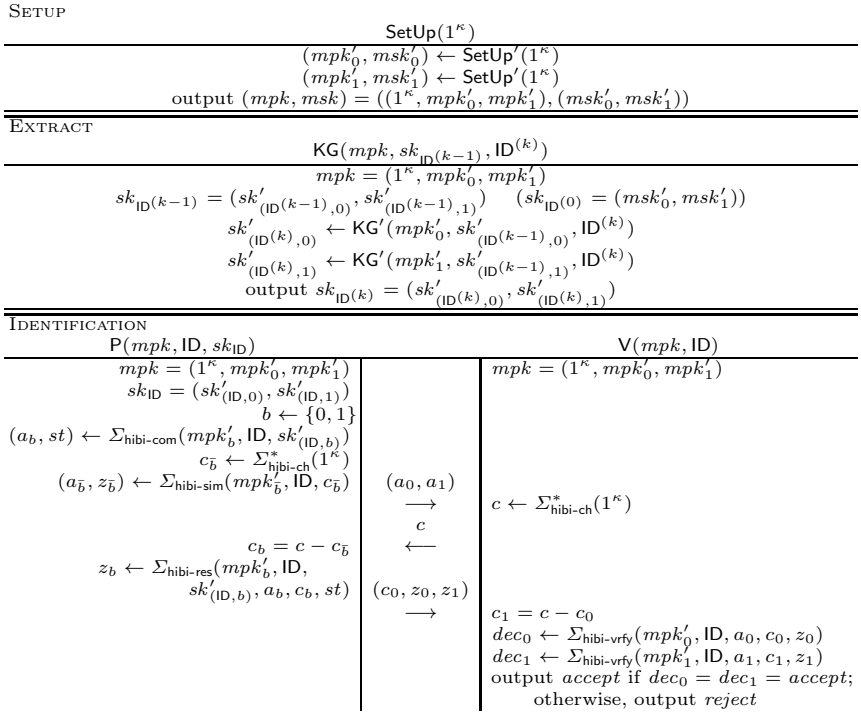

Fig. 3. DPdk Transformation

Although the DPdk transformation requires two master public keys and is less efficient than the DIdk and MI transformations, the transformation can enhance the security of a Σ^*-type HIBI protocol even in the static hierarchical-identity attack model.

It is easy to have a variant of the DPdk transformation (DPsk transformation) such that each entity is given either a secret key based on mpk'_0 or mpk'_1, and the entity shows that it has either the secret key in mpk'_0 or mpk'_1. However, it seems difficult to enhance the passive security of HIBI protocols in any hierarchical-identity attack mode with this variant.

3.4 Modified Dual-Identity Double-Key Transformation

We modify the DIdk transformation and call the modified transformation *mDIdk transformation*, which is also applicable to the Σ^+-type HIBI protocol.

We have seen in the previous subsection that, in the DIdk transformation, a user of identity $\mathsf{ID}^{(k)}$ $(= (I_1, I_2, \ldots, I_k))$ is given two secret keys corresponding to imaginary identities $0.\mathsf{ID}^{(k)} = (0||I_1, I_2, \ldots, I_k)$ and $1.\mathsf{ID}^{(k)} = (1||I_1, I_2, \ldots, I_k)$ in the underlying HIBI protocol. We modify the DIdk transformation in a way that a user of identity $\mathsf{ID}^{(k)}$ is given two secret keys corresponding to imaginary

identities $(0, I_1, I_2, \ldots, I_k)$ and $(1, I_1, I_2, \ldots, I_k)$ in the underlying HIBI protocol. We then see that in this mDIdk transformation, if the underlying HIBI protocol is ℓ-level (i.e., the maximum length of hierarchical identities is ℓ), the resulting HIBI protocol is $(\ell - 1)$-level. We let $i \circ \mathsf{ID}^{(k)}$ denote $(i, I_1, I_2, \ldots, I_k)$ when $\mathsf{ID}^{(k)} = (I_1, \ldots, I_k)$.

We describe an HIBI protocol, $\mathsf{HIBI} = (\mathsf{SetUp}, \mathsf{KG}, \mathsf{P}, \mathsf{V})$ produced by applying the mDIdk transformation to HIBI' in Fig. 4. In the EXTRACT algorithm in this figure, we let $sk_{\mathsf{ID}^{(0)}} = (sk'_{(0)}, sk'_{(1)})$.

SETUP

$\mathsf{SetUp}(1^\kappa)$

$(mpk', msk') \leftarrow \mathsf{SetUp}'(1^\kappa)$
$sk'_{(0)} \leftarrow \mathsf{KG}'(mpk', msk', (0))$
$sk'_{(1)} \leftarrow \mathsf{KG}'(mpk', msk', (1))$
output $(mpk, msk) = (mpk', (sk'_{(0)}, sk'_{(1)}))$

EXTRACT

$\mathsf{KG}(mpk, sk_{\mathsf{ID}(k-1)}, \mathsf{ID}^{(k)})$

$mpk = mpk'$
$sk_{\mathsf{ID}(k-1)} = (sk'_{0\circ\mathsf{ID}(k-1)}, sk'_{1\circ\mathsf{ID}(k-1)})$ $\quad (sk_{\mathsf{ID}^{(0)}} = (sk'_{(0)}, sk'_{(1)}))$
$sk'_{0\circ\mathsf{ID}^{(k)}} \leftarrow \mathsf{KG}'(mpk', sk'_{0\circ\mathsf{ID}^{(k-1)}}, 0 \circ \mathsf{ID}^{(k)})$
$sk'_{1\circ\mathsf{ID}^{(k)}} \leftarrow \mathsf{KG}'(mpk', sk'_{1\circ\mathsf{ID}^{(k-1)}}, 1 \circ \mathsf{ID}^{(k)})$
output $sk_{\mathsf{ID}^{(k)}} = (sk'_{0\circ\mathsf{ID}^{(k)}}, sk'_{1\circ\mathsf{ID}^{(k)}})$

IDENTIFICATION

$\mathsf{P}(mpk, \mathsf{ID}, sk_{\mathsf{ID}})$		$\mathsf{V}(mpk, \mathsf{ID})$
$mpk = mpk'$		$mpk = mpk'$
$sk_{\mathsf{ID}} = (sk'_{0\circ\mathsf{ID}}, sk'_{1\circ\mathsf{ID}})$		
$b \leftarrow \{0, 1\}$		
$(a_b, st) \leftarrow \Sigma_{\mathsf{hibi\text{-}com}}(mpk', b \circ \mathsf{ID}, sk'_{b\circ\mathsf{ID}})$		
$c_{\bar{b}} \leftarrow \Sigma^{+}_{\mathsf{hibi\text{-}ch}}(mpk')$		
$(a_{\bar{b}}, z_{\bar{b}}) \leftarrow \Sigma_{\mathsf{hibi\text{-}sim}}(mpk', \bar{b} \circ \mathsf{ID}, c_{\bar{b}})$	(a_0, a_1)	
	$\longrightarrow$	$c \leftarrow \Sigma^{+}_{\mathsf{hibi\text{-}ch}}(mpk')$
	c	
$c_b = c - c_{\bar{b}}$	$\longleftarrow$	
$z_b \leftarrow \Sigma_{\mathsf{hibi\text{-}res}}(mpk', b \circ \mathsf{ID},$		
$sk'_{b\circ\mathsf{ID}}, a_b, c_b, st)$	(c_0, z_0, z_1)	
	$\longrightarrow$	$c_1 = c - c_0$
		$dec_0 \leftarrow \Sigma_{\mathsf{hibi\text{-}vrfy}}(mpk', 0 \circ \mathsf{ID}, a_0, c_0, z_0)$
		$dec_1 \leftarrow \Sigma_{\mathsf{hibi\text{-}vrfy}}(mpk', 1 \circ \mathsf{ID}, a_1, c_1, z_1)$
		output *accept* if $dec_0 = dec_1 = accept$; otherwise, output *reject*

Fig. 4. mDIdk Transformation

Due to the special challenge property, c is an element in $\mathbb{G}$ determined by mpk', so are c_0 and c_1 since $c = c_0 + c_1$ and the operation $+$ is defined in $\mathbb{G}$. Therefore, c_0 and c_1 are possible challenges under mpk'.

It is easy to have a variant of the mDIdk transformation (mDIsk transformation) such that each entity is given only either secret key of identities $0 \circ \mathsf{ID}$ or $1 \circ \mathsf{ID}$, and the entity shows that it has either the secret key of $0 \circ \mathsf{ID}$ or $1 \circ \mathsf{ID}$. However, it seems difficult to enhance the passive security of HIBI protocols in any hierarchical-identity attack model with this variant.

3.5 Security of DIdk, MI, DPdk, and mDIdk Transformations

We prove that the DIdk, MI, DPdk, and mDIdk transformations can convert an adapt-hid-imp-pa secure HIBI protocol to an adapt-hid-imp-ca secure one. We construct an adapt-hid-imp-pa impersonator, $\mathcal{I}'$, from an adapt-hid-imp-ca impersonator, $\mathcal{I}$.

Theorem 3.1. *The DIdk transformation converts an* adapt-hid-imp-pa *secure Σ^+-type HIBI protocol into an* adapt-hid-imp-ca *secure one.*

In the reduction from an adapt-hid-imp-ca experiment to an adapt-hid-imp-pa experiment, $\mathcal{I}'$ needs to simulate the PROV oracle. In the DIdk transformation, $\mathcal{I}'$ can have either secret key of an (imaginary) identity 0.ID or 1.ID; thus, $\mathcal{I}'$ can simulate all oracles.

Theorem 3.2. *The MI transformation converts an* adapt-hid-imp-pa *secure Σ^+-type HIBI protocol into an* adapt-hid-imp-ca *secure one.*

We assume two types of impersonators and show that there exist reductions from each impersonator to $\mathcal{I}'$. We let $\mathcal{I}_{master}$ be an impersonator from which $\mathcal{I}'$ derives the secret key corresponding to ID_{master}, and $\mathcal{I}_{user}$ be the other impersonator from which $\mathcal{I}'$ derives a secret key of a user. In the reduction from $\mathcal{I}_{master}$, $\mathcal{I}'$ can perfectly simulate the PROV oracle by obtaining secret keys of users from the external CORR oracle. Thus, $\mathcal{I}'$ can extract a secret key of ID_{master} from $\mathcal{I}_{master}$ (by using the Reset Lemma), and can impersonate ID_{master}. Note that since ID_{master} is randomly chosen from the set of identities, it is a hierarchical identity of depth 1. In the reduction from $\mathcal{I}_{user}$, $\mathcal{I}'$ can perfectly simulate the PROV oracle with a secret key of ID_{master} obtained from the external CORR oracle, and extract a secret key of the target identity ID^* from $\mathcal{I}_{user}$ (by using the Reset Lemma). Thus, it can impersonate ID^*.

Theorem 3.3. *The DPdk transformation converts an* adapt-hid-imp-pa *secure Σ^*-type HIBI protocol into an* adapt-hid-imp-ca *secure one.*

After $\mathcal{I}'$ receives mpk' from the challenger, $\mathcal{I}'$ internally generates another key pair (mpk'_*, msk'_*), and can perfectly simulate all oracles since it obtains the secret keys of all users with this msk'_*. Thus, $\mathcal{I}'$ can extract from $\mathcal{I}$ a secret key of the target identity ID^* either for mpk' or mpk'_* (by using the Reset Lemma). If $\mathcal{I}'$ obtains secret key for ID^* in mpk', it can impersonate ID^* in mpk'.

Theorem 3.4. *The mDIdk transformation converts an* adapt-hid-imp-pa *secure Σ^+-type HIBI protocol into an* adapt-hid-imp-ca *secure one.*

This theorem is proved in almost the same way as **Theorem 3.1**.

We see in the following theorems that the security enhancement transformations can be also applied to wshid-imp-pa secure HIBI protocols. We construct an wshid-imp-pa impersonator, $\mathcal{I}'$, from an wshid-imp-ca impersonator, $\mathcal{I}$.

Theorem 3.5. *The DIdk transformation converts a* wshid-imp-pa *secure Σ^+-type HIBI protocol into a* wshid-imp-ca *secure one.*

In the Setup phase, $\mathcal{I}$ issues $(\mathsf{ID}_1, \ldots, \mathsf{ID}_t)$ to $\mathcal{I}'$ acting as the challenger in the wshid-imp-ca experiment. Then, $\mathcal{I}'$ issues $(0.\mathsf{ID}_1,\ 1.\mathsf{ID}_1, \ldots,\ 0.\mathsf{ID}_t,\ 1.\mathsf{ID}_t)$ to the external challenger. After receiving mpk', $\mathcal{I}'$ sends $(b^*||I_1^{(i)})$ $(1 \leq i \leq t)$ to the external CORR where b^* is a random bit and $\mathsf{ID}_i = (I_1^{(i)}, \ldots, I_{k_i}^{(i)})$, and receives the keys $sk'_{(b^*||I_1^{(i)})}$. $\mathcal{I}'$ can simulate the oracles in the same way as in the proof of **Theorem 3.1**.

Theorem 3.6. *The MI transformation converts a* wshid-imp-pa *secure Σ^+-type HIBI protocol into a* wshid-imp-ca *secure one.*

Theorem 3.7. *The DPdk transformation converts a* wshid-imp-pa *secure Σ^*-type HIBI protocol into a* wshid-imp-ca *secure one.*

Theorem 3.8. *The mDIdk transformation converts a* wshid-imp-pa *secure Σ^+-type HIBI protocol into a* wshid-imp-ca *secure one.*

On the other hand, the DPdk and mDIdk transformations can also convert a passively secure HIBI protocol to a concurrently secure one in the static hierarchical-identity attack model. We construct an stat-hid-imp-pa impersonator, $\mathcal{I}'$, from an stat-hid-imp-ca impersonator, $\mathcal{I}$.

Theorem 3.9. *The DPdk transformation converts a* stat-hid-imp-pa *secure Σ^*-type HIBI protocol into a* stat-hid-imp-ca *secure one.*

In the reduction from a stat-hid-imp-ca experiment to a stat-hid-imp-pa experiment, $\mathcal{I}'$ needs to simulate the PROV oracle. In the DPdk transformation, $\mathcal{I}'$ can have either the master secret key of the master public key mpk'_0 or mpk'_1, can generate a secret key of any identity; thus, $\mathcal{I}'$ can simulate all oracles.

Theorem 3.10. *The mDIdk transformation converts a* stat-hid-imp-pa *secure Σ^+-type HIBI protocol into a* stat-hid-imp-ca *secure one.*

This theorem is proven by a reduction similar to that of **Theorem 3.9**. At the beginning of the Setup phase, $\mathcal{I}'$ randomly chooses $b^* \in \{0, 1\}$, sends a single corrupt query including the 1-level hierarchical identity (b^*) to the external challenger and obtains a secret key $sk'_{(b^*)}$. After this, $\mathcal{I}'$ can generate secret keys of the underlying HIBI protocol for any hierarchical identities of the form $(b^*, I_1, I_2, \ldots, I_k)$ (i.e., any hierarchical identities in which each identity at first level is b^*) and can simulate the PROV oracle.

3.6 Discussion

The DPdk transformation can convert a stat-hid-imp-pa secure Σ^*-type HIBI protocol to a stat-hid-imp-ca secure one, and the mDIdk transformation can convert a stat-hid-imp-pa secure Σ^+-type HIBI protocol to a stat-hid-imp-ca

secure one. On the other hand, the DIdk and MI transformations seem not to be able to do so. See **Table 1** for a summary.

In the DPdk transformation, two master public keys in the underlying HIBI protocol, mpk'_0 and mpk'_1, compose a master public key in the resulting HIBI protocol, and the secret key of each entity in the resulting HIBI protocol is computed with either the master secret keys, msk'_0 or msk'_1, in the underlying HIBI protocol. Even in the stat-hid-imp-atk security model, since the simulator has a master secret key msk'_{b^*} for the master public key mpk'_{b^*}, it can generate a secret key for any entity and then simulate the PROV oracle.

The mDIdk transformation can be applied only to HIBI protocols with a hierarchical depth larger than one. That is, it is not applicable to IBI protocols. Though it is similar to the DIdk transformation, its security proof is similar to that of the DPdk transformation. In the mDIdk transformation, if we obtain $sk'_{(b^*)}$ ($b^* = 0$ or 1), we can compute secret keys for any descendant of the 1-level identity (b^*) and simulate the PROV oracle, even in the stat-hid-imp-atk security model.

On the other hand, we face a problem in simulating the PROV oracle in the DIdk and MI transformations. In the DIdk transformation, the master public key in the resulting HIBI protocol is set with the master public key in the underlying HIBI protocol, mpk', and a secret key of an entity corresponding to identity ID in the resulting HIBI protocol is a secret key corresponding to either identity, $0.\mathsf{ID}$ or $1.\mathsf{ID}$, in the underlying HIBI protocol. The secret key is computed with the master secret key, msk', corresponding to mpk'. Since the simulator does not have msk' in the proof for the stat-hid-imp-atk security and can no longer corrupt users in the learning phase, it cannot obtain secret keys for identities queried to the PROV oracle and fails to simulate it.

In the MI transformation, the master public and secret keys and the secret keys of entities in the resulting HIBI protocol are the same as those in the underlying HIBI protocol. In the proof for the stat-hid-imp-atk security, the simulator might obtain the secret key for the master identity, ID_{master}, at the Setup phase and simulate the CONV oracle. In the challenge phase, however, if the secret key extracted from transcripts coincides with the key for ID_{master}, the simulator would not obtain the non-trivial secret key and the impersonation would fail. We do not know how to address the problem.

We finally observe that the single-key variants may fail to enhance security because corruption may leak information of a secret key. In an HIBI protocol by the DIsk transformation, an entity of ID has only either $sk'_{0.\mathsf{ID}}$ or $sk'_{1.\mathsf{ID}}$, which determines b of its descendants. Precisely speaking, if an impersonator obtains a secret key $sk'_{b.\mathsf{ID}'}$ such that $\mathsf{ID} \in \mathsf{pref}(\mathsf{ID}')$ by issuing a CORR query ID', it may know b for $sk'_{b.\mathsf{ID}'}$ and then for $sk'_{b.\mathsf{ID}}$. A similar discussion with DIsk can be applied to the mDIsk transformation, Also in the DPsk transformation, an entity of ID has only either $sk'_{(\mathsf{ID},0)}$ or $sk'_{(\mathsf{ID},1)}$, and an impersonator may guess b for $sk'_{(\mathsf{ID},b)}$. Thus, in all three cases, from the impersonator that knows b, a simulator in proof may extract only the trivial secret key corresponding to b, and constructing their security proofs seems to be difficult.

References

1. Bellare, M., Namprempre, C., Neven, G.: Security proofs for identity-based identification and signature schemes. Journal of Cryptology 22(1), 1–61 (2009)
2. Bellare, M., Palacio, A.: GQ and Schnorr Identification Schemes: Proofs of Security against Impersonation under Active and Concurrent Attacks. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 162–177. Springer, Heidelberg (2002)
3. Chin, J.-J., Heng, S.-H., Goi, B.-M.: Hierarchical Identity-Based Identification Schemes. In: Ślęzak, D., Kim, T.-H., Fang, W.-C., Arnett, K.P. (eds.) SecTech 2009. CCIS, vol. 58, pp. 93–99. Springer, Heidelberg (2009)
4. Cramer, R., Damgård, I., Schoenmakers, B.: Proof of Partial Knowledge and Simplified Design of Witness Hiding Protocols. In: Desmedt, Y.G. (ed.) CRYPTO 1994. LNCS, vol. 839, pp. 174–187. Springer, Heidelberg (1994)
5. Feige, U., Shamir, A.: Witness indistinguishable and witness hiding protocols. In: STOC 1990, pp. 416–426. ACM (1990)
6. Fujioka, A., Saito, T., Xagawa, K.: Security Enhancements by OR-Proof in Identity-Based Identification. In: Bao, F., Samarati, P., Zhou, J. (eds.) ACNS 2012. LNCS, vol. 7341, pp. 135–152. Springer, Heidelberg (2012)
7. Fujioka, A., Saito, T., Xagawa, K.: Secure Hierarchical Identity-Based Identification without Random Oracles. In: Gollmann, D., Freiling, F.C. (eds.) ISC 2012. LNCS, vol. 7483, pp. 258–273. Springer, Heidelberg (2012)
8. Gennaro, R.: Multi-trapdoor Commitments and Their Applications to Proofs of Knowledge Secure Under Concurrent Man-in-the-Middle Attacks. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 220–236. Springer, Heidelberg (2004)
9. Gentry, C., Silverberg, A.: Hierarchical ID-Based Cryptography. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 548–566. Springer, Heidelberg (2002)
10. Horwitz, J., Lynn, B.: Toward Hierarchical Identity-Based Encryption. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 466–481. Springer, Heidelberg (2002)
11. Kurosawa, K., Heng, S.-H.: Identity-Based Identification Without Random Oracles. In: Gervasi, O., Gavrilova, M.L., Kumar, V., Laganà, A., Lee, H.P., Mun, Y., Taniar, D., Tan, C.J.K. (eds.) ICCSA 2005, Part II. LNCS, vol. 3481, pp. 603–613. Springer, Heidelberg (2005)
12. Rückert, M.: Adaptively Secure Identity-Based Identification from Lattices without Random Oracles. In: Garay, J.A., De Prisco, R. (eds.) SCN 2010. LNCS, vol. 6280, pp. 345–362. Springer, Heidelberg (2010)
13. Shamir, A.: Identity-Based Cryptosystems and Signature Schemes. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, pp. 47–53. Springer, Heidelberg (1985)
14. Yang, G., Chen, J., Wong, D.S., Deng, X., Wang, D.: A new framework for the design and analysis of identity-based identification schemes. Theoretical Computer Science 407(1-3), 370–388 (2008); A preliminary version appeared ACNS 2007 (2007)