A Forward Secure Identity Based Encryption Scheme with Master Key Update

Peng YANG* and Kanta MATSUURA*

1. INTRODUCTION

1.1 Background

Security of digital systems is becoming increasingly critical in our daily life. One important primitive is the identity based encryption (IBE)\(^7\). IBE has the maximum flexibility for assigning user’s public key, i.e., any arbitrary string (identity) could be the recipient’s public key. IBE has been fairly well researched in recent years\(^3,1,2,8,6,4\). There are (at least) two levels of secret in an IBE scheme. For simplicity, here we only consider single layer IBE. And our following discussion affects HIBE case. They are the top-level secret, which is called the master key, and the end-level secrets, which are the users’ secret keys. In order to minimize damage in case of an adversary successfully expose users’ secret keys, forward security has been introduced into IBE\(^5,9,2\). In a forward secure identity based encryption (FS-IBE) scheme, the adversary can obtain no information about the compromised user’s secret encrypted before the breaking-in time point.

1.2 Related Works and Our Motivation

One inherent weakness of IBE is the key-escrow problem, which means the trusted center, called private key generator (PKG), possesses the master key. Since the master key is used to generate secret keys corresponding to every identity, compromising the master key equals success of breaking the whole IBE scheme. We can naturally consider a sufficiently motivated adversary will try his best to expose the master key.

Although in historical works\(^3,2\) forward secrecy of users’ secret keys has been perfectly achieved, but forward secrecy of the master key was out of concern. Actually, there exists an constant top secret stored in PKG, and that may become the weakest point of the whole scheme. In this paper, we focus on constructing such an FS-IBE scheme with master key update (FS-IBEm) that the top-level secret evolves as same as users’ secret keys do, so that even if at some time point the adversary compromise the master key, he can no longer generate users’ secret keys corresponding to passed time points. Note this attack can be mounted in all the other known works.

In this paper, we focus on how to construct such FS-IBEm in standard model.

1.3 Our contributions

Our first contribution is that we combined Waters’ HIBE (Waters)\(^8\) and Boneh-Boyen’s HIBE (BonehBoyen)\(^1\) to a hierarchical FS-IBE. We employed Waters as the identity hierarchy and BonehBoyen as the time hierarchy.

To achieve FS-IBEm’s property, we simply let the identity hierarchy be two-level, and force PKG to use a level one secret key as the actual functional master key and to delete the original unevolutional master key.

The security of our FS-IBEm could be considered straightforwardly based on Waters and BonehBoyen scheme. Our FS-IBEm is secure in the sense of FS-IND-ID-CPA in standard model. We stress here that the security proof is not the main contribution of this paper. We remark that because Waters and BonehBoyen are based on decisional bilinear Diffie-Hellman (DBDH) assumption and in our scheme no additional assumption is introduced, our FS-IBEm is also provably secure from DBDH assumption.

Comparing with our scheme, 9) is only secure in the sense of FS-OW-ID-CPA in the random oracle model, which means 9) requires ideal cryptographic hash function, and 2) is secure in the sense of FS-IND-sID-CPA, which means 2) is weak against the adaptive chosen identity attack. Although one can raise 2) to fully security, that will sacrifice security reduction.

2. PRELIMINARY

2.1 Notions and Notations

Let \( y \leftarrow x \) denote assignment. If \( F \) is an algorithm, then let \( y \leftarrow F(x_1, \cdots, \cdot) \) denote the experiment of running \( F \) on input \( x_1, \cdot, \cdots \) and letting \( y \) be the output.

2.2 Security Definitions

Let IND-ID-CPA\(^{10}\) denote indistinguishability against adaptive chosen identity and chosen plaintext attack. Let IND-sID-CPA\(^{10}\)
3. SECURITY MODEL

3.1 Algorithms

Definition 1 An FS-IBEm scheme is specified by six ppt algorithms, i.e., FS-IBEm = \{Setup, Ext, mkUpd, skUpd, Enc, Dec\}.

The functionalities of setup algorithm, secret key extraction algorithm, secret key update algorithm, encryption algorithm and decryption algorithm are almost the same as ordinary FS-IBE, thus we omit details. The master key update algorithm \(\text{mkUpd}\) takes input as system parameter \(\text{param}\), current time index \(\tau\) and the corresponding the old master key \(\text{mk}^\tau\), and it evolves the master key to \(\text{mk}^{\tau+1}\) for the next time period.

3.2 Security Notions

Definition 2 An FS-IBEm scheme is FS-IND-ID-CPA secure if for all polynomial \(N(\cdot)\), the advantage of any ppt adversary in the following game is negligible.

Setup: The challenger runs Setup on security parameter \(\lambda\) and a random n-length vector \(U_j\), where \(j \in \{1, 2\}\) and the elements of \(U_j\) are uniformly distributed. Let \(h_{\text{mk}}\) denote running Waters’ hash on the root identity, e.g., the domain of a company. Thus, \(h_{\text{mk}} = u^*_i \Pi_{i \in \alpha} u_{i,j}\), where \(\alpha\) is the set of indices for which the root identity is set to 1. It generates the master key \(\text{mk}^\tau\) at time-period \(\tau = \langle|0| \cdots |0|\rangle\) as in Fig. 1. Finally, it deletes \(\alpha\) and publishes information, except the master key, as system parameter.

\[
\text{mk}^\tau = \begin{bmatrix}
g^{i_0} \cdot h^\tau \cdot (g^{i_1} \cdot f_1) \cdot (g^{i_2} \cdot f_2) \cdots (g^{i_{N-1}} \cdot f_{N-1}) \\
g^0 \\
g^{i_0} \cdot h^\tau \cdot (g^{i_1} \cdot f_1) \cdots (g^{i_t} \cdot f_{t+1}) \cdots (g^{i_{N-1}} \cdot f_{N-1})
\end{bmatrix}
\]

Figure 1: General form of master key at time period \(\tau\).

Ext (param, \(\text{mk}^\tau\), id, \(\tau\)): The secret key extraction algorithm generates secret key \(\text{sk}_{\text{id}}^\tau\) for certified user with identity id. Let \(h_{\text{id}}\) denote running Waters’ hash on the user’s identity. It works as follows: (1) Pick up random \(r^* \leftarrow Z_p\) and compute \(y^* = h^\tau \cdot a_1 \leftarrow g^{i_t}\). (2) Parse \(\text{mk}^\tau\) as \((a_0, a_1, b_1, \ldots, b_t, c_1, \ldots, c_t, d_1, \ldots, d_t)\), where unless \(\tau_i = 0\), set \(c_i = 1\), \(d_i = 1\). (3) Output \(\text{sk}_{\text{id}}^\tau = (a_0, y^*, a_1, b_1, \ldots, b_t, c_1, y^*, \ldots, c_t, y^*, d_1, \ldots, d_t)\).

skUpd (param, \(\text{mk}^\tau\), \(\tau\)): It evolves the master key to \(\text{mk}^{\tau+1}\) for the next time period. The essential part is to use the current BonehBoyen time point to generate cover set for time periods \(\tau + 1, \tau + 2\).

skUpd (param, \(\text{mk}^\tau\), id, \(\tau\)): This algorithm computes and returns the evolved user’s secret key \(\text{sk}_{\text{id}}^{\tau+1}\). The computation is essentially as same as mkUpd.

Enc (param, id, \(\text{mk}^\tau\), M): To encrypt a plaintext \(M \in G_1\) using \(\text{mk}^\tau\) at time-period \(\tau \leq \tau^*\), first parse \(\tau\) as \(\tau = \tau_1 \cdots \tau_t\), and then pick a random vector \(s \leftarrow Z_p\) and compute \(C = (s (g^{i_0} \cdot g_s) \cdot M, g^0, h^\tau, h_{\text{mk}} (g^{i_1} \cdot f_1)^{\tau_1} \cdots (g^{i_t} \cdot f_{t+1})^{\tau_t}) \in G \times G^{t+1}\).

Dec (param, \(\text{sk}_{\text{id}}^{\tau+1}\), C, \(\tau\)): To decrypt a ciphertext \(C\) for id at time \(\tau\), first parse \(C\) as \(<A, B, D_1, D_2, E_1, \ldots, E_{\tau_t}>\), and then pick a random vector \(s \leftarrow Z_p\) and compute \(\text{sk}_{\text{id}}^{\tau+1} = (a_0, a_1, a_2, b_1, \ldots, b_t, c_1, \ldots, c_t, d_1, \ldots, d_t)\). Compute the plaintext as follows: \(M = A \cdot \text{enc}(a_0, D_1) \cdot \text{enc}(a_1, D_2) \cdot \text{enc}(a_2, B)\).
and Boneh-Boyen is secure in the sense of IND-sID-CPA.

Proof. The security of our scheme can be proved straightforwardly from the security of Waters’ scheme (Waters)\(^1\) and Boneh-Boyen’s scheme (BonehBoyen)\(^2\).

Roughly speaking, the identity-axis and the time-axis of each hierarchy evolve independently in most of the operations. The two axes only meet together in the Dec algorithm.

5. Conclusions

In this paper, we proposed a forward security identity based encryption scheme whose top-level secret information, the master key, can be evolved by time. The security of our scheme is based on standard decisional bilinear Diffie-Hellman assumption.

Future work can be considered as designing FS-IBE scheme with master key evolution functionality with more efficient encryption, more efficient decryption algorithms, and shorter keys.

Acknowledgements

This work was supported by the special fund of Institute of Industrial Science, The University of Tokyo, and the Mitsubishi Foundation (Research Grants in the Natural Sciences).

(Manuscript received, July 24, 2008)

References