Identity-Based Encryption

Nabil Abou Reslan, Cecylia Bocovich, Madhur Kukreti

December 16, 2013

1 Introduction

Public key encryption was invented as a means to provide secure communication to parties separated by space or time. Digital signatures additionally allow parties to confirm the author of a particular message.

The original public key encryption schemes, however, leave something to be desired. If Alice wishes to send a message to Bob, she must first communicate her desire for secure communications with him. She must then attain his public key from either Bob himself or a certificate authority. These extra communication are time-consuming and involve infrastructures that make distribution more difficult.

In 1984, Shamir proposed an encryption scheme to address these difficulties by mimicking an "ideal mail system" [5]. In an ideal mail system, a user need only know the name and address of the person they would like to communicate with. They could then send secure messages to this person without ever having communicated with them before. Additionally, if a document is received by a user in an ideal mail system, they would be able to verify the signature only by knowing the sender's name.

Identity-based encryption accomplishes both of these goals by using identities as a user's public key. This identity can be anything that is tied indisputably to a particular user (e.g. an email address, name, pseudonym, etc.). If Alice wants to send a message to Bob in this scheme, she need only know his email address. Then, the message can be both encrypted and sent to the address and decrypted only by the owner with access to a secret key. Similarly, any messages sent by Bob can be signed with his secret key and verified by Alice to match the email address from which they came. This perfectly resembles the ideal mail system described above — allowing Alice and Bob to communicate securely without the need to exchange keys or rely on a certificate authority.

This paper serves as a literature review of the background and some recent advancements in identity-based encryption. It is organized as follows: first we discuss the basic implementation of an IBE scheme developed by Boneh and Franklin [1]. We then discuss the security of IBE schemes and the application of the Fujisaki-Okamoto conversion[6] to make the basic scheme IND-ID-CCA secure. Finally, we discuss some recent applications and extensions of basic IBE schemes[2, 4, 3].

2 Implementation

The idea of identity-based encryption (IBE) surfaced in 1984, but it wasn't until 2001 that the first fully functional IBE scheme was proposed by Boneh and Franklin[1]. Part of the reason for this long period of time before an implementation was introduced is due to the difficult nature of the mathematical problem surrounding IBE. The problem is as follows: a private key must be computable from any identity, the private key must be computed by trusted key generation centres who know some sort of global secret, and the private key must not be computable by anyone without access to the global secret.

It is clear right away that the mathematics of RSA are not suitable to this problem. The scheme developed by Boneh and Franklin instead relies on bilinear maps. A bilinear map is of the form:

$$\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$$

with the property that $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$. This property, as we will later seem allows us to compute private keys from public keys.

Identity-based encryption schemes have four main algorithms. There is the initial setup during which the master-key and the system parameters are computed, extraction which involves computing private keys for each identity, and the usual encryption and decryption procedurs. Below are in-depth descriptions of each of these algorithms.

Setup

The setup algorithm is run only once to initialize the system. This can be done by the private key generator (PKG). During this phase, the message space is defined and the master-key is decided upon. First, the algorithm takes in a security parameter k and chooses a prime $q > k$. Then, groups $\mathbb{G}_1$ and $\mathbb{G}_2$ or order q are generated and a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ between them. Next, a generator $P \in \mathbb{G}_1$ is chosen and the PKG next picks a random integer $s \in \mathbb{Z}_q^*$ to be the master-key. This key is used to compute $P_{pub} = sP$. Finally, we define two one-way functions:

$$H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1^*$$

$$H_2 : \mathbb{G}_2 \rightarrow \{0,1\}^n$$

These map identities to the public key space ($\mathbb{G}_1$) and elements of $\mathbb{G}_2$ to the message space, respectively. The usage of these functions becomes more clear in the subsequent algorithms.

The master-key is kept secret by the PKG and the rest of the information is published:

$$\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, P_{pub}, H_1, H_2 \rangle$$

The message space consists of strings of length n, $\mathcal{M} = \{0,1\}^n$ and the ciphertext space $\mathcal{C} \subseteq \mathbb{G}_1 \times \{0,1\}^n$.

Extract

Once setup is complete, the extract algorithm may be performed by the PKG to compute private keys. This algorithm takes an identity, $\mathrm{ID} \in \{0,1\}^*$, and outputs the corresponding private key $d_{\mathrm{ID}} \in \mathbb{G}_1$.

Given an identity, which may be an email address or some other form of identifying information encoded in a string of 0s and 1s, the public key, Q_{ID}, is computed as an element of the group $\mathbb{G}_1^*$ by using the one-way function H_1:

$$Q_{\text{ID}} = H_1(ID)$$

Note that the function H_1 is public and therefore, any user can compute another user's public key only by knowing their identity.

The PKG then uses its secret key s to compute the private key:

$$d_{\text{ID}} = sQ_{\text{ID}}$$

and sends this key to the user with identity ID.

Encrypt

Once a user has their public key, they can receive encrypted messages from others in the system. The encryption algorithm, e_{ID}, takes in a plaintext message $M \in \mathcal{M}$, the identity of the recipient, ID, and outputs a ciphertext $C \in \mathcal{C}$.

Consider a scenario in which Alice composes a message M that she wishes to send to Bob. She first computes Bob's public key as $Q_{\text{ID}_{Bob}} = H_1(\text{ID}_{Bob})$. Then, she chooses a random integer $r \in \mathbb{Z}_q^*$. Now she is ready to compute the ciphertext:

$$C = \langle rP, M \oplus H_2(\hat{e}(Q_{\text{ID}_{Bob}}, P_{pub})^r)\rangle$$

Note that the function H_2 is also public and available to anyone in the system. Alice then proceeds to send this ciphertext to Bob.

Decrypt

Upon receiving a ciphertext $C = \langle U, V\rangle \in \mathcal{C}$, Bob performs a decryption procedure using his private key $d_{\text{ID}_{Bob}}$.

The original plaintext message is computed as:

$$M = V \oplus H_2(\hat{e}(d_{\text{ID}_{Bob}}, U))$$

Note again that the function H_2 is publically available to anyone in the system.

As with any cryptosystem, the one defined above must satisfy the property that $d_K(e_K(M)) = M$ for every possible plaintext $M \in \mathbb{M}$

Theorem 1. *In the IBE scheme proposed above, for any receipient ID, and any plaintext message $M \in \mathcal{M}$, $e_{ID}(d_{ID}(M)) = M$*

Proof. Let $M \in \mathcal{M}$ be an arbitrary plaintext and ID$\in \{0, 1\}$ be an arbitrary identity. According to the encryption algorithm, the ciphertext for this message is computed as

$$e_{\text{ID}}(M) = \langle rP, M \oplus H_2(\hat{e}(Q_{\text{ID}}, P_{pub})^r)\rangle$$

where $Q_{\text{ID}} = H_1(\text{ID})$.

The decryption procedure then computes

$$d_{\mathrm{ID}}(e_{\mathrm{ID}}(M)) = M \oplus H_2(\hat{e}(Q_{\mathrm{ID}}, P_{pub})^r) \oplus H_2(\hat{e}(d_{\mathrm{ID}}, rP))$$

which can be expanded to

$$d_{\mathrm{ID}}(e_{\mathrm{ID}}(M)) = M \oplus H_2(\hat{e}(Q_{\mathrm{ID}}, sP)^r) \oplus H_2(\hat{e}(sQ_{\mathrm{ID}}, rP)).$$

From here, the property of the bilinear map $\hat{e}$ allows us the rewrite this equation as

$$d_{\mathrm{ID}}(e_{\mathrm{ID}}(M)) = M \oplus H_2(\hat{e}(Q_{\mathrm{ID}}, P)^{rs}) \oplus H_2(\hat{e}(Q_{\mathrm{ID}}, P)^{rs})$$

which immediately simplifies to M. □

While this scheme can be implemented with any groups and bilinear mapping that satisfy the requirements in the setup algorithm, Boneh and Franklin give a concrete system using Weil Pairing on elliptic curves as the mathematical basis.

3 Security

The basic scheme proposed by Boneh and Franklin is secure against chosen plaintext attacks. In fact, the definition of this type of security requires a small modification in identity-based schemes [1]. The definitions of chosen plaintext security (IND-CPA) and chosen ciphertext security (IND-CCA) are extended to allow an adversary to perfom private key extractions on a bounded number of identities. Additionally, a new hardness assumption is needed to ensure the security of the encryption. In many cryptosystems, the security of the system is based on the Decisional Diffie-Hellman problem (DDH). The formal definition of DDH is as follows.

Definition 1. Given a multiplicative group $(\mathbb{G}, \cdot)$ and an element α of order n, the Decisional Diffie-Hellman problem is to determine whether

$$\log_\alpha \delta \equiv \log_\alpha \beta \times \log_\alpha \gamma \pmod{n}$$

Unfortunately, this problem is not hard enough in the group $\mathbb{G}_1$ due to the existence of the bilinear map $\hat{e}$. Instead, the security of this identity-based encryption scheme relies on the hardness of another version of this problem — the Bilinear Diffie-Hellman problem (BDH).

Definition 2. Given two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of order q, a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$, and a generator $P \in \mathbb{G}_1$, the BDH problem is to compute $\hat{e}(P, P)^{abc}$ given values for P, aP, bP, cP for some $a, b, c \in \mathbb{Z}_q^*$.

It has been proven that BDH is no harder than the Computational Diffie-Hellman problem (CDH), but it is still an open problem to prove that CDH is no harder than BDH. We assume that the groups $\mathbb{G}_1$ and $\mathbb{G}_2$ are chosen such that BDH is hard.

Given the hardness assumption of BDH, we can now outline security definitions for IBE schemes. As Boneh and Franklin point out [1], we need to expand the notion of IND-CPA and IND-CCA security to be compatable with identity-based schemes. For example, in a chosen plaintext security context, we must assume that the adversary can not only view as many encrypted plaintexts as she wishes, but may also perform an arbitrary number of private key extraction requests for unused identities. This notion of security is strictly stronger than IND-CPA as it gives the adversary additional information.

We must then extend these definition for IBE schemes and provide a means to prove that a system is IND-ID-CPA and IND-ID-CCA secure.

We first discuss a slight variation of chosen plaintext security known as one-way security. A One-Way cryptosystem means that if a random message is encrypted it is infeasible for the adversary to produce the corresponding plaintext from the given ciphertext in its entirety [6]. More precisely according to Boneh and Franklin, an IBE scheme is one-way if no polynomial adversary A has a non-negligible advantage against the challenger in the following game, also summarized in figure 1.

In this game, there are two players: a challenger and an adversary. The goal of the adversary will be to successfully guess the correct plaintext decryption of a random ciphertext challenge.

There are 5 steps to this game. In the setup step, the IBE scheme is initialized. Then, the adversary is given the opportunity to extract private keys from identities of her choice in two different phases. After the first, phase, a challenge identity is established. At the end of the second phase, the adversary must then guess a plaintext encrypted with the challenge identity.

1. To setup the game, the challenger chooses a security parameter k and initializes the IBE scheme with the setup algorithm discussed in section 2. The adversary then receives the system parameters. Note that only the challenger (acting as the PKG) has access to the master key.

2. In the first extraction query phase, the adversary is able to issue an arbitrary number of extraction queries. She does this by iteratively sending m identities $\mathrm{ID}_1, \mathrm{ID}_2, \ldots, \mathrm{ID}_m$ to the challenger and receives the corresponding private keys $d_{\mathrm{ID}_1}, \ldots, d_{\mathrm{ID}_m}$.

   These queries may be asked adaptively, meaning the adversary may choose to request the extraction of a particular identity based on previously received extractions.

3. After making a satisfactory number of queries, the adversary chooses an identity ID and sends it to the challenger. The challenger will then encrypt a random message $M \in \mathcal{M}$ and encrypt it with this challenge identity. The corresponding ciphertext, C, is sent to the adversary.

4. The adversary is then given the chance to issue a second wave of extraction queries once she receives the ciphertext C. These, again, consist of an arbitrary number of identities $\mathrm{ID}_{m+1}, \ldots, \mathrm{ID}_n$ with the constrant that none of the queries include the challenge identity ID.

   The queries may, again, adaptively chosen based on the received private keys $d_{m+1}, \ldots, d_n$.

5. The last step in the game requires the adversary to guess at the plaintext corresponding to the ciphertext C. If the guess, M', matches the original plaintext M, the adversary wins.

Definition 3. An IBE scheme is OWE-ID-CPA secure if the adversary's advantage in the previously described OWE-ID-CPA game is significiant. The advantage is calculated as

$$Adv_{\mathcal{A}}(k) = \Pr[M = M']$$

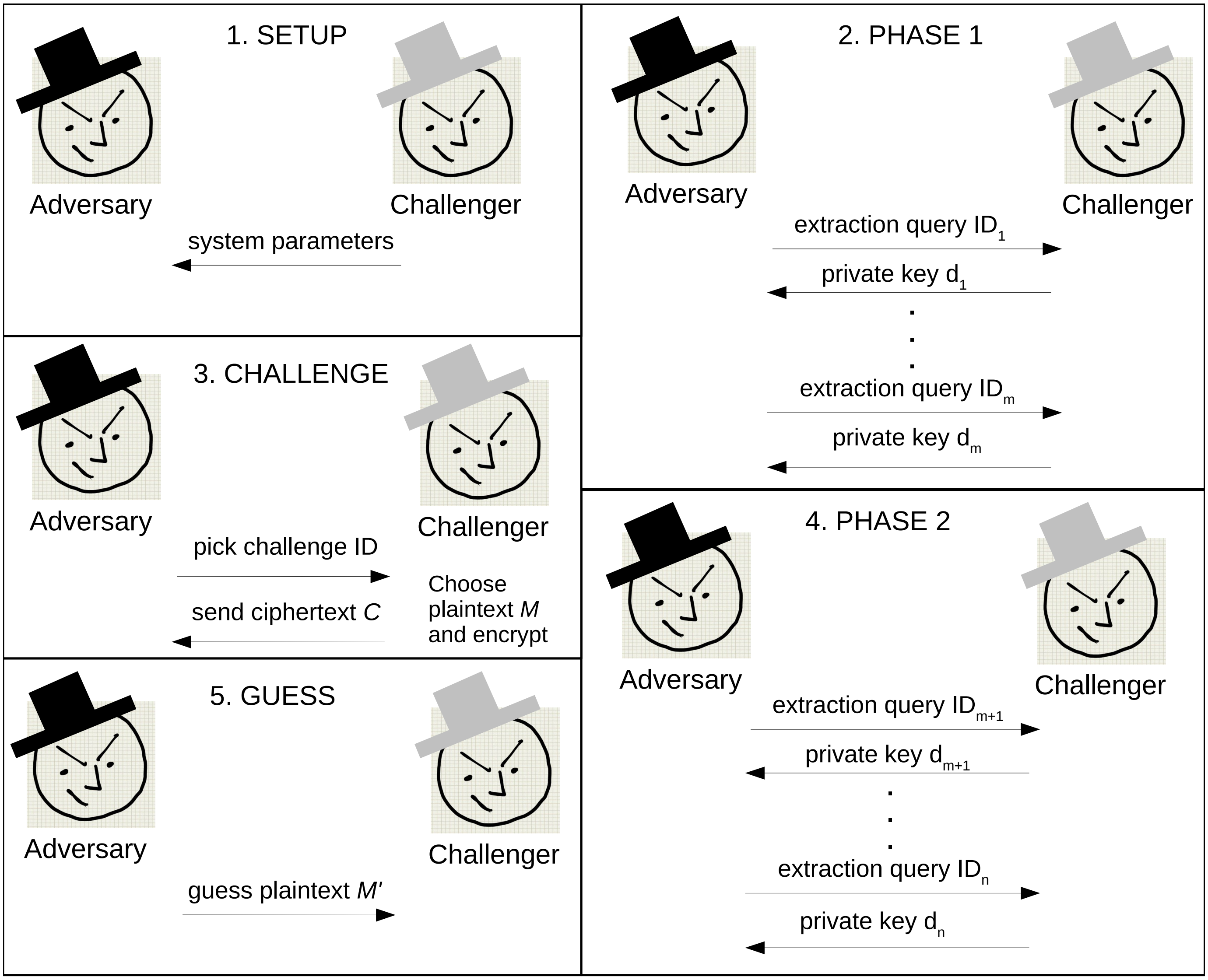

Figure 1: Summary of the challenger-adversary game for OWE-ID-CPA security

with an adversary $\mathcal{A}$ and a security parameter k.

Boneh and Franklin also extended the definition of chosen ciphertext security to Identity based cryptosystems. A cryptosystem is said to be IND-ID-CCA secure if any polynomial time adversary has negligible advantage in the following game (summarized in figure 2.

As in the previous game, this game involves both an adversary and a challenger. The end goal of the adversary is to guess at which plaintext corresponds to a ciphertext issued by the challenger. Unlike the previous game, this game allows a challenger to issue decryption requests on ciphertext messages. Therefore, the IND-ID-CCA notion of security defined here is stronger than the previous OWE-ID-CPA notion.

1. The game setup involves the challenger initializing an IBE scheme with a security parameter k. The output of the setup algorithm consists of both the system paramters and a master key. The challenger keeps the master key and then gives the remaining system parameters to the adversary.

2. The first querying phase again allows the adversary to submit an arbitrary number of extraction queries. Additionally, she may also submit decryption queries on an arbitrary number of ciphertext messages. These messages may be decrypted with the private key corresponding to any identity of the adversary's choosing.

   Each query and response is then of one of the following forms.

   - Extraction query: given an identity ID_i, the challenger returns the corresponding private key d_{ID_i}.
   - Decryption query: given an identity ID_i and a ciphertext C_i, the challenger returns both the private key d_{ID_i} and the plaintext $d_{\text{ID}_i}(C_i)$.

   These queries may be asked adaptively, i.e. chosen based on the response of previous queries.

3. After the adversary is content with phase 1 queries, she chooses two plaintexts $M_0,\ M_1 \in \mathcal{M}$ and an identity ID. Note: ID must not be included in the queries of the previous step.

   The challenger then chooses $b \in \{0, 1\}$ at random, thereby choosing randomly among the plaintexts M_0 and M_1. He encrypts this plaintext with identity ID and sends the ciphertext $C = e_{\text{ID}}(M_b)$ to the adversary.

4. In the second querying phase, the adversary is able to issue additional extraction and decryption requests after having received the ciphertext C. These requests proceed in the same way as phase 1 with the additional constraint that the extraction query may not be made on identity ID and the decryption query may not be made on the pair (ID, C). All other combinations are valid and may again be queried adaptively.

5. Once the second query phase is over, the adversary must guess at which plaintext $M_{b'}$ the ciphertext represents. If she guesses correctly, she wins the game.

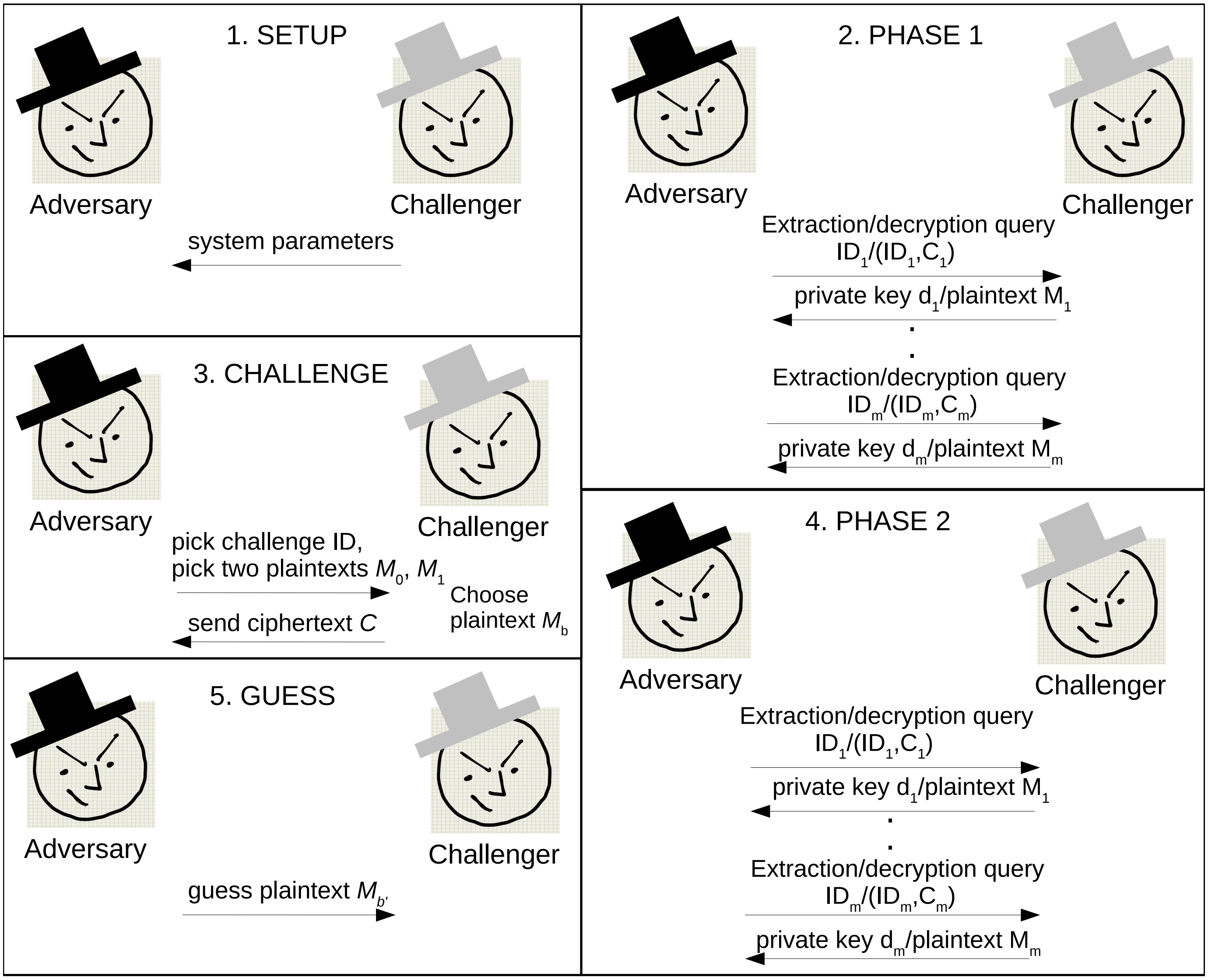

Figure 2: Summary of the challenger-adversary game for IND-ID-CCA security

Definition 4. An IBE scheme is IND-ID-CCA secure if the adversary's advantage in the previously described game is significant. The advantage is calculated as

$$Adv_{\mathcal{A}}(k) = |Pr[b = b'] - 1/2|$$

where k is the security paramter and $\mathcal{A}$ is the adversary.

With these two definitions of security in IBE schemes, we can continue our discussion of the implementation of Boneh and Franklin's IBE scheme. The scheme discussed in section 2 is proven to be OWE-ID-CPA secure [1]. To produce a full IND-ID-CCA secure system, they employed the Fujisaki-Okamoto transformation [6].

4 Strengthening Security with FO-transformation

The Fujisaki-Okamoto transformation can be applied to any OWE-CPA public key cryptosystem to make it IND-CCA secure. This idea of applying a transformation to existing schemes to increase their security or their efficiency can also be seen with hybrid encryption systems. Basically, elements are combined or added into encryption schemes to strengthen favourable features. Consider first the task of transforming an inefficient public-key encryption scheme to an efficient hybrid scheme.

Hybrid cyrptosystems combine the advantages of both public and private key systems. The result is a system that is both efficient and secure. Public key cryptosystems offer better key distribution whereas symmetric key cryptosystems offer better performance. The concept of hybrid cryptosystems was realized in mid 90's. In a hybrid cryptosystem, the message is encrypted using a symmetric key k. Additionally, the symmetric key is encrypted using a public key K_{pub} and hence the ciphertext contains two encryptions. In order to decrypt, K_{priv} of the public key cryptosystem is used to find the symmetric key k which is in turn used to decrypt the message.
Encryption: To encrypt a plaintext message M, Alice first chooses a symmetric key k and then computes:

$$(c1; c2) = (E(K_{pub}, k); E(k, m))$$

and sends the ciphertexts $(c1; c2)$ to Bob.
Decryption: Bob first decrypts the encrypted symmetric key k

$$k = D(K_{priv}, c1)$$

and then uses this to decrypt the ciphertext message to retrieve m.

$$m = D(k, c2)$$

We now explore how these basic ideas are employed to the IBE scheme discussed earlier. Let $\{S, X, E, D\}$ represent the setup, extraction, encryption, and decryption algorithms in the basic scheme. We can now construct a new cryptosystem $\{S', X', E', D'\}$ by making the following changes.

Setup

The setup algorithm remains mostly the same, with the addition of two more hash functions included in the output parameters.

$$G_1 : \{0,1\}^n \times 0, 1^n \rightarrow \mathbb{Z}_q^*$$

$$G_2 : \{0,1\}^n \rightarrow 0, 1^n$$

where $\{0,1\}^n$ is the plaintext message space $\mathcal{M}$.

Extract

The extraction algorithm in the modified scheme will behave exactly as the basic scheme described in section 2.

Encrypt

The new computed ciphertext, given an identity ID and a plaintext message M, is

$$C = \langle rP,\ \sigma \oplus H_2(\hat{e}(Q_{\text{ID}},\ P_{pub})^r), M \oplus G_2(\sigma)\rangle$$

where $\sigma \in \{0,1\}^n$ is chosen randomly and $r = G_1(\sigma, M)$.

Decrypt

Given a ciphertext $C = \langle U, V, W\rangle$, the plaintext M can be computed with the private key d_{ID} as follows.

First compute σ as

$$\sigma = V \oplus H_2(\hat{e}(d_{\text{ID}}, U))$$

Note that if we expand this out, we have

$$\begin{aligned} V \oplus H_2(\hat{e}(d_{\text{ID}}, U)) &= \sigma \oplus H_2(\hat{e}(Q_{\text{ID}},\ P_{pub})^r) \oplus H_2(\hat{e}(sQ_{\text{ID}}, rP)) \\ &= \sigma \oplus H_2(\hat{e}(Q_{\text{ID}},\ P)^{rs}) \oplus H_2(\hat{e}(Q_{\text{ID}}, P)^{rs}) \\ &= \sigma \end{aligned}$$

The plaintext is now easily computed as

$$M = W \oplus G_2(\sigma)$$

Additionally, the following test should be performed:

$$U = (G_1(\sigma, M))P$$

If the equality does not hold, the ciphertext should be rejected.

5 Applications

Identity-based cryptography can be used in several applications. While the original motivation was to eliminate the need of a certificate authority and facilitate the deployment of a public-key scheme[5], additional applications include key expiration and credential management. These are accomplished by concatenating the date or clearance level to a user's email address to form the public key.

There are several very practical applications of IBE. These schemes can play a role in the timed release of private information. It involves maintaining the confidentiality of digital documents to a preset time-based disclosure constraint. When encrypting information for this purpose, the IBE public encryption keys contain the date and time constraint of the confidential document. For example, the string 201312161200GMT specifies that its disclosure date is on December 16th, 2013 at 12:00 noon local time as per GMT, and it is sufficient to encrypt a confidential document. The IBE decryption key is generated after the creation of the encryption key, as per the IBE system. A trusted authority can then generate this decryption key at exactly the time of the release of the document.

IBE can also be used for increasing accountability of organizations when handling users personal information and providing them with greater control [4]. Identity or profile information is encrypted with certain policies. These policies are used as the encryption keys and are assigned to references of names of identity and profile attributes, disclosure details, etc. To obtain a valid IBE decryption key, the receiver must send these policies as well as authentication credentials and other related information to the trusted authorities, as long as it doesnt violate disclosure agreements. A TA will then issue a decryption key if it permits the user with its compliance to the disclosure policies.

More recently, Geambasu et al. [2] used identiy-based encryption to develop an auditing file system. In this system, a modified IBE scheme is used to detect invalid file access and prohibit future access of files once the device they are stored on has been compromised.

6 Extensions

Identity-based encryption provides a useful solution to many challenges and drawbacks present in previous public key encryption schemes. However, there are still drawbacks to the IBE scheme we described.

Consider the task of private key extraction. In this scheme, extraction is done solely by a trusted PKG. However, in many real life scenarios, placing this much trust in a single entity is undesireable. A malicious PKG would be able to keep all extracted private keys after sending them to users and have the ability to decrypt any messages meant for any user in the system. Even worse, the PKG can do all of this without being detected.

Under this current scheme, PKGs must also be robust as well as non-malicious. The PKG represents a single point of failure. In order to bring down the system during the extraction phase, an attacker only needs to somehow disable the PKG and the whole system will cease to function.

A natural solution to this problem would be to split the master key up amongst several generators so that no one person has the ability to extract private keys alone. Private key extraction would then be done in a distributed manner with private computations such that a user with identity ID is the only one who ever sees their private key d_{ID}.

7 Conclusion

In this paper, we discussed the implementation and security of Identity-based encryption as well as explored recent developments in the form of extensions and applications of IBE schemes.

IBE is convenient to use since the sender requires prior knowledge only of a receiver's ID to send a message. Its reliance on a single trusted key generator creates a weakness and a target for attackers; hence the possibility of distributed key generation makes the scheme more practical for real-world use. We also looked at practical applications that could result from further work on IBE.

References

[1] D. Boneh and M. Franklin. Identity-based encryption from the weil pairing. In J. Kilian, editor, *Advances in Cryptology CRYPTO 2001*, volume 2139 of *Lecture Notes in Computer Science*, pages 213–229. Springer Berlin Heidelberg, 2001.

[2] R. Geambasu, J. P. John, S. D. Gribble, T. Kohno, and H. M. Levy. Keypad: An auditing file system for theft-prone devices. In *Proceedings of the Sixth Conference on Computer Systems*, EuroSys '11, pages 1–16, New York, NY, USA, 2011. ACM.

[3] A. Kate and I. Goldberg. Distributed private-key generators for identity-based cryptography. In J. Garay and R. Prisco, editors, *Security and Cryptography for Networks*, volume 6280 of *Lecture Notes in Computer Science*, pages 436–453. Springer Berlin Heidelberg, 2010.

[4] M. C. Mont, M. C. Mont, P. Bramhall, and P. Bramhall. Ibe applied to privacy and identity management trusted. *HP Labs*, 2003, 2003.

[5] A. Shamir. Identity-based cryptosystems and signature schemes. In G. Blakley and D. Chaum, editors, *Advances in Cryptology*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer Berlin Heidelberg, 1985.

[6] P. Yang, T. Kitagawa, G. Hanaoka, R. Zhang, K. Matsuura, and H. Imai. Applying fujisaki-okamoto to identity-based encryption. In M. Fossorier, H. Imai, S. Lin, and A. Poli, editors, *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*, volume 3857 of *Lecture Notes in Computer Science*, pages 183–192. Springer Berlin Heidelberg, 2006.