On the Security of a Certificateless Public-Key Encryption

Zhenfeng Zhang, Dengguo Feng

State Key Laboratory of Information Security,
Institute of Software, Chinese Academy of Sciences, Beijing 100080, P.R.China
zfzhang@is.iscas.ac.cn

Abstract. Certificateless public-key cryptosystem is a recently proposed attractive paradigm using public key cryptosystem, which avoids the key escrow inherent in identity-based public-key cryptosystems, and does not need certificates to generate trust in public keys. Recently, Al-Riyami and Paterson proposed a new certificateless public-key encryption scheme [2, 3] and proved its security in the random oracle model. This paper shows that their scheme is vulnerable to adaptive chosen ciphertext attacks, and presents a countermeasure to overcome such a security flaw.

1 Introduction

In traditional certificate-based public key cryptosystems, an entity's public-key is generated from some random information that is unrelated to his identity, and hence need to be certified with a certificate issued by a certification authority. Any participant who wants to use a public-key must first verify the corresponding certificate to check the validity of the public-key. Certificate-based public key cryptosystems require a large amount of storage and computing time to verify and revoke certificates.

The notion of identity-based cryptography (ID-PKC) was introduced by Shamir [7], in which the public-key of a user can be derived from his unique identifier information. ID-PKC eliminates the certificates and greatly simplifies the key management. However, an inherent problem of ID-PKC is the key escrow, i.e., the private-key of each user is known to a private key generator, who can then decrypt any ciphertext and forge signature on any messages for any user. Moreover, ID-PKC requires a secure channel between users and PKG to deliver private keys. Because of these problems, it seems that ID-PKC should be considered to be suitable only for small private network with lower security requirements.

To alleviate the problems associated with the use of identity-based cryptosystems and certificate authorities in traditional public-key cryptosystems, Al-Riyami and Paterson [1] introduced the concept of certificateless public key cryptography (CL-PKC). Unlike ID-PKC, user's private-key of CL-PKC schemes is

not generated by a Key Generation Center (KGC) alone. Instead, it is a combination of KGC-produced partial-private-key and an additional user-chosen secret. In this way, they successfully eliminate the built-in escrow properties, since KGC could not control the user's private-key entirely. Meanwhile, CL-PKC is not identity-based any longer, and an additional public-key must be generated from user's randomly-chosen secret information. The complex structure of this scheme also means that a user who is encrypting a message can do it without having to verify the correctness of the public key via a public key certificate.

A certificateless scheme's security is assessed in terms of two different kinds of attackers. The first kind of attacker (or Type I attacker) is meant to represent a normal third party attack against the confidentiality of the system. Here, an entity in possession of all users' public keys attempts to break the IND-CCA2 security of the scheme. Due to the uncertified nature of the public-keys produced by the users, we must assume that an attacker is able to replace these entities' public keys at will. This represents the attackers' ability to fool a user into sending a confidential message using a public key that has been supplied by the attacker. The second kind of attacker represents a malicious key generation center, who is given the key generation center's long term secret, but may not replace entities' public keys.

In 2005, Al-Riyami and Paterson proposed a new certificateless public key encryption (CL-PKE) scheme [2, 3], whose security is proven to rest on the hardness of the Bilinear Diffie-Hellman Problem (BDHP) in the random oracle model. The new scheme is more efficient than the original scheme [1], and then is used to constructed an efficient certificate based encryption scheme [2]. In this paper, we analyze the security of their new CL-PKE scheme and show that it is vulnerable to adaptive chosen ciphertext attacks against the Type I attacker. A countermeasure is also presented to resist such an attack.

2 Certificateless Public-Key Encryption

A certificateless public-key encryption scheme is defined by seven probabilistic, polynomial-time algorithms [1, 6]:

- `Setup`: This algorithm takes as input a security parameter 1^k and returns the master private key SK and the master public key PK. The master public key defines a message space $\mathcal{M}$ and a ciphertext space $\mathcal{C}$. This algorithm is run by a KGC to initially set up a system.
- `Extract-Partial-Private-Key`: This algorithm takes as input the master public key PK, the master private key SK, and identifier $ID \in \{0,1\}^*$. It outputs a partial private key D_{ID}. This algorithm is run by a KGC once for each user, and the corresponding partial private key is distributed to that user in a suitably secure manner.

- `Set-Secret-Value:` This algorithm takes as input the master public key PK and an entity's identifier ID as input, and outputs a secret value x_{ID} for that identity. This algorithm is run once by the user.
- `Set-Private-Key:` This algorithm takes as input the master public key PK, an entity's partial private key D_{ID} and an entity's secret value x_{ID}. It outputs the full private key sk_{ID} for that user. This algorithm is run by the user.
- `Set-Public-Key:` This algorithm takes as input the master public key PK and an entity's secret value x_{ID}. It output a public key pk_{ID} for that user. This algorithm is run once by the user and the resulting public key is widely and freely distributed.
- `Encrypt:` This algorithm takes as input the master public key PK, a user's identity ID and public key pk_{ID} and a message $m \in \mathcal{M}$. It outputs a ciphertext $C \in \mathcal{C}$.
- `Decrypt:` This algorithm takes as input the master public key PK, a user's private key sk_{ID} and a ciphertext $C \in \mathcal{C}$. It returns $m \in \mathcal{M}$ or the error symbol $\perp$.

The security of a certificateless encryption scheme is expressed by two (but very similar) games. In both cases, an attacker $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ is trying to break the IND-CCA2 security, the formal model describing confidentiality. The game runs as follows:

1. The challenger generates $(PK, SK) = \texttt{Setup}(1^k)$.

2. The attacker executes $\mathcal{A}_1$ on PK and (possibly) some extra information aux. During its execution $\mathcal{A}_1$ may have access to certain oracles (described subsequently). $\mathcal{A}_1$ terminates by outputting an identity ID^*, two messages of equal length (m_0, m_1), and some state information $state$.

3. The challenger computes a public key value pk_{ID^*} for ID^* (if one does not already exist) by running algorithms `Set-Secret-Value` and `Set-Public-Key`. Next it randomly chooses a bit $b \in \{0, 1\}$, computes and returns to the attack a ciphertext $C^* = \texttt{Encrypt}(PK, ID^*, pk_{ID^*}, m_b)$.

4. The attacker executes $\mathcal{A}_2$ on input $(C^*, state)$. During its execution $\mathcal{A}_2$ may have access to the following oracles. $\mathcal{A}_2$ terminates by outputting a guess b' for b.

The attacker wins the game if $b = b'$ and its advantage is defined to be:

$$|\Pr[b = b'] - 1/2|.$$

The oracles that the attacker may have access to are defined as following.

- Request Public Key: The attacker provides an identity ID and the challenger responds with the public key for ID. If the identity ID has no associated public key, then the challenger generates a public key for ID by running `Set-Public-Key` (after running `Set-Secret-Value` if necessary).

- Replace Public Key: The attacker supplies an identity ID and a public key value pk_{ID}, and the challenger replaces the current public key (if it exists) with pk_{ID}.
- Extract Partial Private Key: The attacker supplies an identity ID and the challenger responds with a partial private key D_{ID}. If the identity has no partial private key, the challenger generates a partial private key by running `Extract-Partial-Private-Key` on ID.
- Extract Private Key: The attacker supplies an identity ID and the challenger responds with the private key sk_{ID}. If the identity has no associated private key, the challenger generates a private key using `Set-Private-Value` (after running `Set-Secret -Value` and `Extract-Partial-Private-Key` if necessary). The attacker may never query this oracle on any identity for which it has replaced the public key.
- Decrypt: The attacker supplies an identity ID and a ciphertext C, and the challenger responds with the decryption of C under the private key sk_{ID}. Note that if the attacker has replaced the public key for ID, then this oracle should return the correct decryption of C using the private key that is associated with the public key pk_{ID}.

A certificateless scheme should resist attacks made by attackers with access to these oracles in the following ways.

Definition 1 (Type I Attacker). *Any probabilistic polynomial-time attacker* $\mathcal{A}^{\mathrm{I}} = (\mathcal{A}_1^{\mathrm{I}}, \mathcal{A}_2^{\mathrm{I}})$ *should have negligible advantage in winning the IND-CCA2 game subject to the following constraints:*

- $\mathcal{A}^{\mathrm{I}}$ *cannot extract the private key for the challenge identity* ID^* *at any time,*
- $\mathcal{A}^{\mathrm{I}}$ *cannot extract the private key of any identity for which it has replaced the public key,*
- *If* $\mathcal{A}_1^{\mathrm{I}}$ *replaces the public key of* ID^**, then* $\mathcal{A}^{\mathrm{I}}$ *cannot extract the partial private key for* ID^* *at any time after the public key was replaced,*
- $\mathcal{A}_2^{\mathrm{I}}$ *cannot decrypt the challenge ciphertext* C^* *for the identity* ID^* *unless the public key* pk_{ID^*} *used to create the challenge ciphertext has been replaced.*

Note that an attacker is allowed to make decryption queries, even for public keys which it has replaced. This means that the challenger must be able to correctly answer decryption queries even for public keys for which it does not know the corresponding secret key. This is a very strong requirement and it is unclear how realistic this restriction is. Some authors [4, 6] have chosen to weaken this definition so that the challenger is not forced to decrypt ciphertexts for which the public key has been replaced. They presented a definition of Type I^- security [4], which adds an additional oracle constraint given below:

- $\mathcal{A}^{\mathrm{I}}$ *can only decrypt ciphertexts on identities for which it has replaced the public key with some value that is unequal to its original value if it also supplies the secret value corresponding to the new public key.*

The security definition against a Type II attacker states that the key generation center should not be able to break the confidentiality of the scheme. In this case, an attacker $\mathcal{A}^{\mathrm{II}}$ has access to the oracles as a Type I attacker, subject to the oracle constraints that $\mathcal{A}^{\mathrm{II}}$ cannot extract the private key for the challenge identity ID^* at any time, cannot replace public keys at any point in time, and cannot decrypt the challenge ciphertext C^* for the identity ID^*. We refer to [1-3] for detail.

3 Al-Riyami and Paterson's CL-PKE Scheme

In this section, we describe Al-Riyami and Paterson's new certificateless public-key encryption scheme [2, 3].

`Setup:` 1. On input a security parameter k, this algorithm output $\langle \mathcal{G}_1, \mathcal{G}_2, e \rangle$ first, where $(\mathcal{G}_1, +)$ and $(\mathcal{G}_2, \cdot)$ are groups of prime order q, $e : \mathcal{G}_1 \times \mathcal{G}_1 \to \mathcal{G}_2$ is a bilinear pairing [5].

2. Choose an arbitrary generator $P \in \mathcal{G}_1$.

3. Select a master-key $s \in \mathbf{Z}_q^*$ randomly and set $P_0 = sP$.

4. Choose cryptographic hash functions $H_1 : \{0,1\}^* \to \mathcal{G}_1^*$, $H_2 : \mathcal{G}_2 \to \{0,1\}^n$, $H_3 : \{0,1\}^n \times \{0,1\}^n \to \mathbf{Z}_q^*$, $H_4 : \{0,1\}^n \to \{0,1\}^n$ and $H_5 : \mathcal{G}_1 \to \{0,1\}^n$, where n is the bit-length of messages. The master public key is $\texttt{params} = \langle \mathcal{G}_1, \mathcal{G}_2, e, n, P, P_0, H_1, H_2, H_3, H_4, H_5 \rangle$.

`Partial-Private-Key-Extract:` This algorithm takes as input $ID_A \in \{0,1\}^*$, computes $Q_A = H_1(ID_A)$ and outputs $D_A = sQ_A$ as a partial private-key for entity A.

`Set-Secret-Value:` This algorithm takes as inputs `params` and an entity A's identifier ID_A as inputs. It selects $x_A \in \mathbf{Z}_q^*$ at random and outputs x_A as A's secret value.

`Set-Private-Key:` This algorithm takes as inputs `params`, an entity A's partial private-key D_A and A's secret value $x_A \in \mathbf{Z}_q^*$. The output of the algorithm is the pair $S_A = (D_A, x_A)$. So the private key for A is just the pair consisting of the partial private key and the secret value.

`Set-Public-Key:` This algorithm takes `params` and entity A's secret value $x_A \in \mathbf{Z}_q^*$ as inputs and constructs A's public-key as $P_A = x_A P$.

`Encrypt:` To encrypt $M \in \mathcal{M}$ for entity A with identifier ID_A and public-key P_A, perform the following steps:

1. Check that P_A is in $\mathcal{G}_1^*$, if not output $\perp$ and abort.
2. Compute $Q_A = H_1(ID_A) \in \mathcal{G}_1^*$.
3. Choose a random value $\sigma \in \{0,1\}^n$.
3. Set $r = H_3(\sigma, M)$.

4. Compute and output the ciphertext:

$$C = \langle rP, \sigma \oplus H_2(e(Q_A, P_0)^r) \oplus H_5(rP_A), M \oplus H_4(\sigma) \rangle.$$

`Decrypt:` Suppose $C = \langle U, V, W \rangle \in \mathcal{C}$. To decrypt this ciphertext using private-key $S_A = (D_A, x_A)$:

1. Compute $V \oplus H_2(e(D_A, U)) \oplus H_5(x_A U) = \sigma'$.
2. Compute $W \oplus H_4(\sigma') = M'$.
3. Set $r' = H_3(\sigma', M')$ and test if $U = r'P$. If not, output $\perp$ and reject the ciphertext. Otherwise, output M'.

Al-Riyami and Paterson have shown that the proposed CL-PKE scheme is provable secure in the random oracle model.

Theorem 1 ([2, 3]). *Let $H_i (1 \leq i \leq 5)$ be random oracles. Suppose that there is no polynomially bounded algorithm can solve the bilinear Diffie-Hellman problem with non-negligible advantage. Then the* CL-PKE *scheme is IND-CCA2 secure.*

4 A Type I Attacker's CCA2 Attack

In this section we consider the security model against a Type I attacker and show that the CL-PKE scheme is insecure against a Type I attacker under adaptive chosen ciphertext attacks. A Type I attacker $\mathcal{A}^{\mathrm{I}} = (\mathcal{A}_1^{\mathrm{I}}, \mathcal{A}_2^{\mathrm{I}})$ can break the IND-CCA2 security of their CL-PKE scheme in the following manner.

The challenger first executes $\mathtt{Setup}(1^k)$ to generate a master private key s, $P_0 = sP$, and other parameters `params`.

The attacker executes $\mathcal{A}_1^{\mathrm{I}}$ on `params`. During its execution $\mathcal{A}_1^{\mathrm{I}}$ chooses $t \in \mathbf{Z}_q^*$ at random, and then has access to the oracles to replace the public key of an entity with identity ID^* with $P_{ID^*} = rP$. Then $\mathcal{A}_1^{\mathrm{I}}$ terminates by outputting the identity ID^*, and two messages (m_0, m_1) of equal length.

The challenger randomly chooses $b \in \{0, 1\}$, and computes

$$C^* = \mathtt{Encrypt}(\mathtt{params}, ID^*, P_{ID^*}, m_b)$$

as following: Compute $Q_{ID^*} = H_1(ID^*) \in \mathcal{G}_1^*$, choose a random value $\sigma \in \{0, 1\}^n$, set $r = H_3(\sigma, M)$, and compute the ciphertext

$$\begin{aligned} C^* &= (U, V, W) \\ &= \big(rP, \sigma \oplus H_2(e(Q_{ID^*}, P_0)^r) \oplus H_5(rP_{ID^*}), m_b \oplus H_4(\sigma)\big). \end{aligned}$$

and returns C^* to the attacker.

Upon receipt of C^*, the attacker executes $\mathcal{A}_2^{\mathrm{I}}$ to determine the value of b. During its execution $\mathcal{A}_2^{\mathrm{I}}$ may have access to the oracles under the constraints described in Definition 1. Particularly, $\mathcal{A}_2^{\mathrm{I}}$ accesses to the Replace Public Key

oracle, and replace the public P_{ID^*} with $P'_{ID^*} = x'P_{ID^*}$, where $x' \in \mathbf{Z}_q^*$ is randomly chosen by $\mathcal{A}_2^{\mathrm{I}}$.

Then $\mathcal{A}_2^{\mathrm{I}}$ compute $V' = V \oplus H_5(tU) \oplus H_5(x'tU)$ and set $C^{**} = (U, V', W)$. Now $\mathcal{A}_2^{\mathrm{I}}$ can access the Decrypt oracle and request decrypting C^{**} for (ID^*, P'_{ID^*}).

Note that $rP_{ID^*} = r \cdot tP = t \cdot U$, $P'_{ID^*} = x'P_{ID^*} = x'tP$ and then $x'tU = x't \cdot rP = rP'_{ID^*}$. Thus we have

$$\begin{aligned} V' &= \sigma \oplus H_2(e(Q_{ID^*}, P_0)^r) \oplus H_5(rP_{ID^*}) \oplus H_5(tU) \oplus H_5(x'tU) \\ &= \sigma \oplus H_2(e(Q_{ID^*}, P_0)^r) \oplus H_5(rP'_{ID^*}). \end{aligned}$$

That is, $C^{**} = (U, V', W)$ is a valid ciphertext of m_b for the entity with identity ID^* and public key P'_{ID^*}.

So the challenger will return m_b to $\mathcal{A}_2^{\mathrm{I}}$ as the answer, from which the attacker can determine a correct value for b, and thus break the IND-CCA2 security of CL-PKE scheme.

Note that, the above attack also works for the Type I^- attacker, as $\mathcal{A}_2^{\mathrm{I}}$ can supply the secret value $x't$ corresponding to the public key P'_{ID^*} to the challenger for decrypting C^{**}.

The reason that the above attack works is that $\mathcal{A}^{\mathrm{I}}$ can generate a valid ciphertext of m_b after receiving C^*, with only the secret value corresponding to P_{ID^*}. As a countermeasure to overcome such a flaw, we can let $H_2 : \mathcal{G}_2 \times \mathcal{G}_1 \to \{0,1\}^n$, and encrypt a message M by randomly choosing $\sigma \in \{0,1\}^n$, setting $r = H_3(\sigma, m)$ and then computing

$$C = \langle rP, \sigma \oplus H_2(e(Q_A, P_0)^r, rP_A), M \oplus H_4(\sigma) \rangle.$$

The decryption can be done in a similar way.

Now one needs both the partial private-key $D_A = sQ_A$ and the secret value corresponding to P_A to compute the masking factor $H_2(e(Q_A, P_0)^r, rP_A)$ from a given ciphertext C. Therefore the above attack can be thwarted.

5 Conclusion

Certificateless public-key encryption has recently been proposed as an attractive alternative to certificate-based and identity-based encryption schemes. Al-Riyami and Paterson proposed a new certificateless public key encryption scheme [2, 3], which is more efficient than their original one [1]. This paper shows that their new CL-PKE scheme is vulnerable to adaptive chosen ciphertext attacks against a Type I attacker. A countermeasure is also presented to resist such attacks.

References

1. S. Al-Riyami and K. Paterson, "Certificateless public key cryptography", *Advances in Cryptology-Asiacrypt'03*, Lecture Notes in Computer Science, vol. 2894, pp.452-473, Springer-Verlag, 2003.
2. S. Al-Riyami and K. Paterson, "CBE from CL-PKE: A generic construction and efficient schemes", *Public Key Cryptography-PKC'05*, Lecture Notes in Computer Science, vol. 3386, pp.398-415, Springer-Verlag, 2005.
3. S. Al-Riyami, *Cryptographic schemes based on elliptic curve pairings*. PhD thesis, Royal Holloway, University of London, 2004.
4. K. Bentahar, P. Farshim, J. Malone-Lee, and N. P. Smart. "Generic constructions of identity-based and certificateless KEMs". *Cryptology ePrint Archive*: Report 2005/058, Available from http://eprint. iacr.org/2005/058, 2005.
5. D. Boneh and F. Franklin, "Identity-based encryption from the Weil pairing", *SIAM Journal on Computing*, 32, 586-615, 2003.
6. Alexander W. Dent and Caroline Kudla, "On Proofs of Security for Certificateless Cryptosystems". *Cryptology ePrint Archive*: Report 2005/348, Available from http://eprint.iacr.org/2005/348, 2005.
7. A.Shamir, "Identity based cryptosystems and signature schemes", *Advances in Cryptology-Crypto'84*, Lecture Notes in Computer Science, vol. 196, pp. 47-53, Springer-Verlag, 1984.