On the security models for certificateless signature schemes achieving level 3 security

Yu-Chi Chen and Gwoboa Horng
Department of Computer Science and Engineering
National Chung Hsing University
Taichung, 402 Taiwan
Email: { s9756034, gbhorng}@cs.nchu.edu.tw

Public key cryptography has found many applications in our modern society. To guarantee the authenticity of public keys, we need a trusted third party (TTP). In 1991, Girault defined three trust levels for a TTP. The higher the trusted level of the TTP is, the higher the security level of the cryptographic scheme is. In 2007, Hu *et al.* proposed a generic construction of a certificateless signature scheme, together with a security model, achieving Girault's level 3 security. In 2011, Fan *et al.* presented a certificateless short signature scheme based on pairings. In this paper, we consider in depth the security requirements of certificateless signature schemes and show that previous models are inappropriate for achieving the desired level of security. We also present a new security model for a certificateless signature scheme to achieve level 3 security.

Keywords: certificateless public key cryptography, certificateless signature, Girault's trust level, level 3 security

1. Introduction

In traditional public key cryptosystems, the certificates of public keys, generated by a trusted certificate authority (CA), serve as the authentication of the public keys. Whereas, identity-based public key cryptosystems (ID-PKC) [7] and certificateless public key cryptosystems (CL-PKC) [1] do not need the extra trusted party to manage certificates. However, they still need a trusted key generation center (KGC) to generate private keys. Differing from ID-PKC, the key generation center (KGC) in CL-PKC is unable to derive the user's actual private key. That is, CL-PKC does not suffer from the key escrow problem. Both CA and KGC are third parties. The security of the corresponding public key schemes depends on the trustiness of these third parties. In 1991, Girault defined three trust levels for a trusted third party TTP [5]. The higher the trusted level of the TTP is, the higher the security level of the cryptographic scheme is.

- Level 1. The TTP knows the private key of any user and is able to impersonate any user without being detected.
- Level 2. The TTP does not know the private key of any user. But the TTP is able to generate a false private key to impersonate any user.

- Level 3. The TTP does not know the private key of any user. But if the TTP generates a false private key to impersonate a user then it is possible for that user (the victim) to prove that the TTP generated a false private key.

Schemes with trust level 1 or trust level 2 are not acceptable in many applications, such as providing non-repudiations. Reaching Trust Level 3 is generally the goal. In a traditional public key scheme, if the CA forges certificates, the CA's misbehave can be identified through the existence of two valid certificates for the same user. However, a false public key can be created by the KGC without being detected in the certificateless PKC, since new public keys can be created by both the legitimate user and the KGC. Therefore, the traditional public key schemes achieve trust level 3, whereas, the certificateless public key schemes reach only trust level 2.

Certificateless signature (CLS) is a new paradigm for providing non-repudiation. In this paper we study the security models for a CLS scheme to achieve level 3 security from application point of view. We show that previous models fail to guarantee level 3 security. We demonstrate this by pointing out a weakness in a recently proposed CLS. More precisely, a user can generate valid signatures without using his full private key. We also present a new security model for CLS schemes with level 3 security which is more appropriate than previous ones.

The rest of the paper is organized as follows: In Section 2, we describe a general construction of CLS schemes. Some security models for CLS are briefly reviewed in Section 3. A new security model for CLS with level 3 security is presented in Section 4. Finally, we conclude this paper in Section 5.

2. Certificateless Signature Schemes

A certificateless signature scheme consists of the following algorithms:

Setup: This algorithm, run by the KGC, takes a security parameter as input, then outputs **master-key** and system parameter **params**.

Partial-Private-Key-Extract: This algorithm, run by the KGC, takes **params**, **master-key** and a user's identity *ID* as inputs, then outputs a partial-private-key D_{ID} to the user.

Set-Secret-Value: This algorithm, run by a user, returns a secret value.

Set-Private-Key: This algorithm, run by a user, takes the user's partial-private-key D_{ID} and secret value as inputs, and outputs the full private key.

Set-Public-Key: This algorithm, run by a user, takes **params** and the user's full private key as inputs, and outputs a public key pk_{ID} for the user.

CL-Sign: This algorithm, run by a signer, takes **params**, a message *m*, and the user's full private key as inputs, and outputs *S* as the signature for the message *m*.

CL-Verify: This algorithm, run by a verifier, takes **params**, a public key pk_{ID}, a message *m*, a user's identity *ID*, and a signature *S* as inputs. The verifier accepts a valid signature *S* if and only if *S* is the signature of the message *m* for the public key pk_{ID} of the user with identity *ID*.

For non-repudiation, the authenticity of public keys must be guaranteed. Based on Girault's trust hierarchy, we define three trust levels of the KGC in certificateless signature schemes:

- Level 1. The KGC knows the full private key of any user and is able to act as any user to forge signatures and these forged signatures cannot be repudiated by that user (the victim).
- Level 2. The KGC does not know the full private key of any user. But the KGC is able to generate a false private key for any user to forge signatures and these forged signatures cannot be repudiated by that user (the victim).
- Level 3. The KGC does not know the full private key of any user. But the KGC is able to generate a false private key for any user to forge signatures but that user (the victim) can repudiate these forged signatures.

For convenience we will say that a CLS scheme achieves level-*i* security if the KGC is of trust level *i* where $i = 1$, 2, or 3. And, in CL-PKC, we will call a signature to be *valid* if it can be verified by the CL-verify algorithm. To achieve level-3 security, a user must be able to repudiate forged signatures. From legal point of view, using a digital signature scheme with security level 1 or 2, a signer can always repudiate signatures by blaming the KGC. Therefore, only schemes with security level-3 offer strong non-repudiation. Usually, as in conventional PKC, this is done by showing that a user can only have a unique key pair and he cannot generate other valid signatures without using his key pair.

Most previous CLS schemes (for example, Du and Wen's scheme [3]) only achieve level 2 security. Let Bob, with identity ID, be a user in a certificateless signature scheme. The malicious KGC, says Alice, can launch the following attack:

1. Alice uses the master-key to compute Bob's partial-private-key D_{ID}.
2. Alice sets a new secret value corresponding to identity *ID*, and computes the public key pk'_{ID}.
3. Finally, Alice computes a valid signature S' which can be verified using Bob's identity ID and the public key pk'_{ID}.

We note that Bob cannot repudiate S' since he has no way to prove that pk'_{ID} is not his public key.

3. Security Models for CLS schemes

Traditionally, a certificate-based digital signature scheme is secure if it is existentially unforgeable against adaptive chosen message attacks. The adversaries do not include the signers themselves and the attack methods centered on querying signatures for adaptive chosen messages. For a CLS scheme, the situation is more complicated due to potentially many valid public keys of a user. Therefore, we need to consider legitimate users acting as adversaries. Furthermore, there are interactions between users and the KGC (when generating keys). Therefore, the attackers can do a lot more, for example, they can query partial private key of any user, than merely querying signatures.

An outsider, referred as a Type 1 adversary, can try to forge a valid signature. The KGC, referred as a Type 2 adversary, cannot perform the public key replacement attack since the victim can prove that the KGC has misbehavior (assuming that any user can has only a single key pair). The signer himself, referred as a Type 3 adversary, can try to perform the public key replacement attack to come up with a valid signature to frame the KGC. different types of adversaries are with different capabilities. A Type 1 adversary, A_1, does

not access to the master key, but it is able to replace the public key of any user. A Type 2 adversary, A_2, cannot replace the public key of any user, but it is able to access to the master key [1]. They will perform the chosen message attack to existentially forge signatures. These two types of adversarial models are described in two games, namely Game I and Game II.

Game I: A_1 interacts with Challenger C.

Setup: C performs Setup by inputing a security parameter to obtain the system parameter, *params*. C sends *params* to A_1.

Attack: A_1 can adaptively perform the following polynomially bounded queries.
Partial-Private-Key query: A_1 can query the partial private key of any user with identity ID_i. C will return the partial private key D_i to A_1.
Public-Key query: A_1 can query the public key of any user with identity ID_i. C will return the public key pk_i of the user.
Secret-Value query: A_1 can query the secret value of any user with identity ID_i. C will return the secret value r_i of the user to A_1.
Public-Key-Replacement: For any user with identity ID and public key pk, A_1 can set a new secret value r' and the corresponding public key pk', then replace r,pk with r',pk'.
Sign query: A_1 can query the signature generated by a user with identity ID_i for a message m_i. C will generate a signature S_i corresponding to ID_i,m_i and public key pk_i, and return S_i to A_1.

Forgery: A_1 outputs a signature S^* for a message m^* corresponding to identity ID^* and public key pk^*.

A_1 wins the game if and only if the following conditions hold.
- The forged signature S^* is valid.
- The private key (secret value and partial private key) of ID^* and the signature S^* on (m^*, ID^*, pk^*) have never been queried.
- The public key pk^* has never been replaced.

Game II: A_2 interacts with Challenger C.

Setup: C performs Setup by inputing a security parameter to obtain the master-key and the system parameter, *params*. C sends *params* and the master-key to A_2.

Attack: A_2 can adaptively perform the following polynomially bounded queries.
Public-Key query: A_2 can query the public key of any user with identity ID_i. C will return the public key pk_i of the user.
Secret-Value query: A_2 can query the secret value of any user with identity ID_i. C will return the secret value r_i of the user to A_2.
Sign query: A_2 can query the signature generated by a user with identity ID_i for a message m_i. C will generate a signature S_i corresponding to ID_i, m_i and public key pk_i, and return S_i to A_2.

Forgery: A_2 outputs a signature S^* for a message m^* corresponding to identity ID^* and public key pk^*.

A_2 wins the game if and only if the following conditions hold.

- The forged signature S^* is valid.
- The secret value of ID^* and the signature S^* on (m^*, ID^*, pk^*)$ have never been queried.

Most schemes that claimed to achieve level 3 security are based on a security model that guarantee a user can only have a unique key pair. A user in a CLS scheme can try to generate more than one key pair by attacking the Partial-Private-Key-Extract algorithm. The following game is used to model the unforgeability of the partial-private-key under the chosen (ID, pk_{ID}) attack.

Game III: An adversary A_3 interacts with a challenger C. A_3 acts as a legitimate user but it wants to obtain another key pair under a single identity.

Setup: The challenger C runs **Setup** to generate the system parameters and sends them to A.

Attack: A_3 can query the public key pk_{ID} of any user with identity ID and the partial-private-key of any user with identity ID and public key pk_{ID}. C will return the partial-private-key D_{ID} or the public key pk_{ID} to A_3.

Key-Forgery: A_3 outputs a key pair (pk'_{ID*}, D_{ID*}) corresponding to identity ID^*.

A_3 wins this game if and only if the following conditions hold.

- The partial-private-key, secret value, and public key with ID^* have been queried.
- The outputted user's key pair with identity ID^* is different from the original public key created by previous queries.

Thus, if A_3 wins Game III, then A_3 can generate another key pair without formally interacting with the KGC.

4. A new security model for proving level 3 security

In 2011, Fan *et al.* based on Du and Wen's scheme and proposed an improved CLS scheme [4]. They claimed their scheme can achieve level-3 security based on the security model of [6], assuming that Boneh and Boyen's short signature scheme is secure. The scheme consisting of the following algorithms:

Setup: Given a security parameter k, the KGC determines two cyclic additive groups, G_1 and G_2, of prime order q respectively with generators P_1 and P_2, a cyclic multiplicative group G_3 of the same order, a bilinear pairing $e : G_1 \times G_2 \rightarrow G_3$, and two

hash functions $H_1 : \{0, 1\}^* \rightarrow Z_q^*$, $H_2 : \{0, 1\}^* \times G_2 \rightarrow Z_q^*$. The KGC then randomly chooses an $s \in Z_q^*$ as the **master-key** and sets $P_{pub} = sP_2$. Finally, the KGC publish **params** ={ G_1, G_2, G_3, e, q, P_2, g, P_{pub}, H_1, H_2 } where $g = e(P_1, P_2)$.

Set-Secret-Value: A user sets a secret value $r_{ID} \in Z_q^*$.

Set-Public-Key: A user takes **params** and the user's secret value r_{ID}, then constructs his public key pk_{ID} =[$pk1_{ID}$, $pk2_{ID}$] where $pk1_{ID} = r_{ID}P_2$ and $pk2_{ID}$ = $r_{ID}(P_{pub} + H_1(ID)P_2)$. It worthwhile to note that the public key consists of two parts.

Partial-Private-Key-Extract: The KGC takes **params**, **master-key**, the user's identity *ID*, and the first part of his public key $pk1_{ID}$, then returns the partial-private-key D_{ID} to the user via a secure channel where

$$D_{ID} = \frac{1}{s + H_1(ID) + H_1(ID \,||\, pk1_{ID})} P_1 .$$

Set-Private-Key: The user sets the user's partial-private-key D_{ID} and secret value r_{ID} as his full private key $sk_{ID} = (D_{ID}, r_{ID})$.

CL-Sign: Taking **params**, a message m, and full private key sk_{ID}, the user (signer) generates S as the signature for the message m where

$$S = \frac{1}{r_{ID} + H_2(m, pk1_{ID})} D_{ID} .$$

CL-Verify: Given **params**, the public key pk_{ID}, the message m, the user's identity *ID*, and the signature S as inputs, the verifier computes $h = H_2(m, pk1_{ID})$ and accepts the signature if and only if

$$e(S, pk2_{ID} + H_1(\mathrm{ID}||pk1_{ID})pk1_{ID} + h(P_{pub}+H_1(ID)P_2+H_1(ID||pk1_{ID})P_2)) = g.$$

Fan *et al*. showed that if Boneh-Boyen's short signature scheme is existentially unforegable against the chosen message attack, the CLS scheme achieves Girault's level-3 security based on the above models. However, in the following we show that in fact their scheme fails to achieve level-3 security by presenting an attack which allows the signer to forge signatures. Assume Alice is an adversary who is also a user with identity ID_A.

Step 1. Alice sets her secret value r_A and two public key components $pk1_A$ and $pk2_A$ as in Section 2.

Step 2. Alice obtains her partial-private-key D_A from the KGC, then she sets her full private key sk_A= (D_A, r_A).

Step 3. Alice randomly chooses $r^* \in Z_q^*$, and then replaces her second part of public key, $pk2_A$, with $pk2_A{}^*$ where $pk2_A{}^* = r_A(P_{pub} + H_1(ID_A)P_2) + r^*(P_{pub} + H_1(ID_A)P_2) + r^*H_1(ID_A || pk1_A)P_2$.

Step 4. For any message m, Alice computes $h = H_2(m, pk1_A)$, and then she generates a signature S by computing $S = \frac{1}{r^* + r_A + h} D_A$.

The signature S is valid if we use the public key ($pk1_A$, $pk2_A{}^*$) to verify it. Let

$\alpha = s + H_1(ID_A) + H_1(ID_A \| pk1_A)$. Then

$$
\begin{aligned}
& e(S, pk2_A^* + H_1(ID_A \| pk1_A) pk1_A + h(P_{pub} + H_1(ID_A)P_2 + H_1(ID_A \| pk1_A)P_2)) \\
&= e(S, pk2_A^* + H_1(ID_A \| pk1_A) pk1_A + h\alpha P_2) \\
&= e(S, r_A(P_{pub} + H_1(ID_A)P) + r^*\alpha P_2 + H_1(ID_A \| pk1_A) pk1_A + h\alpha P_2) \\
&= e(S, r_A\alpha P_2 + r^*\alpha P_2 + h\alpha P_2) \\
&= e(\frac{1}{(r_A + r^* + h)\alpha} P_1, (r_A\alpha + r^*\alpha + h\alpha)P_2) \\
&= e(P_1, P_2) = g
\end{aligned}
$$

Due to the attack, the adversary can generate many $pk2_A^*$ corresponding to $pk1_A$ and generate signatures that are valid when verifying them with the forged public key ($pk1_A$, $pk2_A^*$). Hence, this scheme does not achieve level-3 security despite of the fact that the scheme is shown to be secure under the security models Game I, II, and III. Indeed, a cryptographic scheme can be provably secure under a security model, but it may still suffer from other attacks if the security model is incomplete to include attacks in real application environments. Observing Game III, the goal of the adversary is to forge another key pair. However, in our attack, the adversary aims to generate a valid signature only.

In the following, we use the above analysis to present a new security game, Game IV, which simulates an attacker (a signer) A_4 to forge a signature and a corresponding public key. Game IV consists of the following phases.

Setup: *C* sets **params** and **master-key**.

Attack: A_4 can query partial-private-keys, secret values, public keys, and signatures with any identity. A_4 also can replace any key pair.

Forgery: A_4 forges a signature S^* for a message m^* and outputs a corresponding public key of the user with identity ID^*.

A_4 wins this game if and only if the following conditions hold.

- The forged signature S^* is valid, which has never been queried before.
- The partial-private-key, secret value, and public key with ID^* have been queried.
- The outputted user's public key with identity ID^* is different from the original public key created by previous queries.

Definition A certificateless signature scheme achieves level 3 security if no probabilistic polynomial-time adversary has non-negligible probability to win Game I~IV.

5. Conclusions

In this paper, we have discussed the security models for CLS schemes to achieve level 3 security. Based on cryptanalyzing Fan *et al.*'s scheme, we have shown that previous models, which ensure that a user can have only one key pair, are not enough. More precisely, ensuring that a user can only have a unique key pair is a necessary condition for a CLS to achieve level 3 security, however this condition alone is not

sufficient. Finally, we present a new security model (Game IV) for proving CLS schemes that achieve level 3 security.

References

[1] S. Al-Riyami and K. G. Paterson, “Certificateless public key cryptography,” in Proceedings of ASIACRYPT, LNCS 2894, 2003, pp. 452-473. Full paper available at: http://eprint.iacr.org/2003/126.

[2] D. Boneh and X. Boyen, “Short signatures without random oracles and the SDH assumption in bilinear groups,” Journal of Cryptology, Vol. 21, 2008, pp. 149-177.

[3] H. Du and Q. Wen, “Efficient and provably-secure certificateless short signature scheme from bilinear pairings,” Computer Standards and Interfaces, Vol. 31, 2009, pp. 390-394.

[4] C. I. Fan, R. H. Hsu, and P. H. Ho, “Truly non-repudiation certificateless short signature scheme from bilinear pairings,” Journal of Information Science and Engineering, Vol. 27, 2011, pp. 969-982.

[5] M. Girault, “Self-certified public keys,” in Proceedings of Eurocrypto, LNCS 547, 1991, pp. 490-497.

[6] B. C. Hu, D. S. Wong, Z. Zhang, and X. Deng, “Certificateless signature: A new security model and an improved generic construction,” Designs, Codes and Cryptography, Vol. 42, 2007, pp. 109-126.

[7] A. Shamir, “Identity-based cryptosystems and signature schemes,” in Proceedings of CRYPTO, LNCS 196, 1985, pp. 47-53.