Cryptanalysis of a Certificateless Signature Scheme without Bilinear Pairings

Kuo-Hui Yeh[1], Kuo-Yu Tsai[2], Chuan-Yen Fan[3]

[1]Department of Information Management,
National Dong Hwa University, Hualien 97401, Taiwan, R.O.C.

[2]Department of Management Information Systems,
Hwa Hsia Institute of Technology, New Taipei City 235, Taiwan, R.O.C.

[3]Department of Information Management
National Taiwan University of Science and Technology, Taipei 10607, Taiwan, R.O.C.

[1]khyeh@mail.ndhu.edu.tw; [2]KuoYu.Nicklas.Tsai@gmail.com; [3]kynwu.tw@gmail.com

Abstract

During these years, the research field of certificateless signature (CLS) scheme without bilinear pairings is promptly investigated as the key escrow problem in identity-based cryptography can be solved via such concept. In this paper, we demonstrate that a certificateless signature scheme proposed by Gong and Li cannot fulfill its security claims. The authors argued that their proposed certificateless signature scheme is able to resist to the super adversary. However, this security argument can be improved. We present a series of attack processes to point out that Gong and Li's scheme is insecure against a super type I adversary.

Keywords: certificateless cryptography; digital signature; bilinear pairings; cryptanalysis

1. Introduction

In traditional public key cryptography, signature schemes allow a singer to sign a message with his/her private key to guarantee non-repudiation property (and more). However, each signature activity must accompany with corresponding certificates to complete. In order to solve the certificate management problem, Shamir [8] introduced a concept of identity-based cryptosystem. In such approach, every user does not have an explicit public key as before. The public key is replaced by his/her publicly available identity information, which can uniquely identify him/her and can be undeniably associated with him/her. The corresponding private key is computed from a one-way trapdoor function of some privileged information known only to the system authority, such as key generation center (KGC). Compared to certificate-based cryptosystem, identity-based cryptosystem does not require extra effort and information for users to validate the authenticity of public keys.

Based on the ideas of self-certified cryptosystem, Al-Riyami and Paterson [6] proposed an approach in 2003, namely certificateless public key cryptography (CL-PKC). In this approach, KGC generates partial private key, each user then generates his/her private key and public key using user's secret value and partial private key. This concept was to oppose to KGC having access to each user's private key in identity-based approach and was the absence of digital certificates and their important management overhead. However, CL-PKC approach is insecure against to type I adversary [9]. In 2004, Yum and Lee [13] proposed another CLS scheme. Nevertheless, Hu et al. [11] pointed out that Yum and Lee's CLS protocol cannot resist to type I adversary. Later, Li et al. [12] and Gorantla et al. [10] presented CLS schemes using bilinear pairings, respectively. Unfortunately, these schemes require heavy operation of bilinear pairing on signature verification. Therefore, the development of CLS scheme without bilinear pairings is promptly investigated in recent years.

In 2011, He et al. [3] demonstrated an efficient CLS scheme which does not adopt the technique of bilinear pairings. Without the heavy computation cost from bilinear pairings, the efficiency of He et al.'s CLS scheme is better than previous CLS protocols. Later, a variant of such CLS concept is adopted in the authors' another study involved with authenticated key agreement [4]. In 2012, however, Tian and Huang [5] and Tsai et al. [2] both presented that He et al.'s CLS scheme is vulnerable to a type II adversary who is able to access the master secret key of KGC. Recently, Gong and Li [1] proposed a CLS scheme without bilinear pairings. The authors claimed that their proposed scheme is secure against the super adversary. Nevertheless, the security claim is not true. In this paper, we will demonstrate that Gong and Li's CLS scheme cannot fulfill their claimed security robustness, i.e. resistance to the super adversary.

2. Preliminary

2.1 Elliptic Curve

Let the notation E/F_p denote an elliptic curve E over a prime finite field F_p, defined by an equation $y^2 = x^3 + ax + b$, where $a, b \in F_p$ are constants such that $\Delta = 4a^3 + 27b^2 \neq 0$. All points $P_i=(x_i, y_i)$ on E and the infinity point O forms a cyclic group G under the operation of point addition $R=P+Q$ defined according to a chord-and-tangent rule. In particular, we define $t \cdot P = P + P + \ldots + P$ (t times) as *scalar multiplication*. Note that P is a generator of G with order n.

2.2 The Overview of Certificateless Signature Scheme

According to the study [6], two types of CLS scheme, denoted as CLS and CLS*, exist. A normal CLS scheme consists of seven phases, i.e. *Setup*, *Partial-Private-Key-Extract*, *Set-Secret-Value*, *Set-Private-Key*, *Set-Public-Key*, *Sign* and *Verify*. We briefly review each phase as follows.

- *Setup*: With the security parameter k, KGC generates a master secret key mk, a corresponding master public key P_{pub} and the public parameters *params*.
- *Partial-Private-Key-Extract*: With the master secret key mk, the public parameters *params* and the user i's identity ID_i, KGC generates a partial secret key D_i for the user i.
- *Set-Secret-Value*: The user i randomly selects a value $x_i \in Z_n^*$ as his/her secret.
- *Set-Private-Key*: With the public parameters *params*, the user i's partial private key D_i and his/her chosen secret value x_i, the user i generates a full private key. Note that in some studies, *Set-Private-Key* phase may be integrated with *Set-Secret-Value* phase.
- *Set-Public-Key*: With the public parameters *params* and the user i's secret value x_i,, the user i outputs his/her public key PK_i.
- *Sign*: With any target message m, this phase outputs a signature $\sigma_i = (R_i, T_i, \tau_i)$ on m.
- *Verify*: With the signature $\sigma_i = (R_i, T_i, \tau_i)$ of the message m, this phase returns 1 if $\sigma_i = (R_i, T_i, \tau_i)$ is valid. Otherwise, it returns 0.

Furthermore, the other kind of certificateless signature scheme CLS* also possesses seven phases: *Setup*, *Partial-Private-Key-Extract*, *Set-Secret-Value*, *Set-Private-Key*, *Set-Public-Key*, *Sign* and *Verify*. The main difference between CLS and CLS* is in the procedure of *Partial-Private-Key-Extract* phase which additionally requires the user i's public key as an input.

2.3 Adversaries against Certificateless Signature Scheme

In general, there exist two categories of adversaries against certificateless signature scheme, i.e. type I and type II Adversaries [6]. The type I adversary models an outside adversary who does not know the master secret key of KGC; however, the type I adversary is able to replace any entity's public key with specific values chosen by the adversary itself. The type II adversary models a malicious KGC who is allowed to access to the master secret key of KGC. Nevertheless, the type II adversary cannot replace the public keys of other entities. In addition, based on the security model defined by Huang et al. [7], type I and II adversaries against CLS schemes can further be classified into three categories: normal, strong and super levels. A normal-level type I (and II) adversary only has the ability to learn valid signatures. A strong-level type I (and II) adversary is able to replace a public key to forge a valid signature when the adversary possesses a corresponding secret value. A super-level type I (and II) adversary is able to learn valid signatures for a replaced public key without any submission.

Here, we only present the definition of the super-level type I adversary j which will mainly be involved with the cryptanalysis of Gong-Li's CLS scheme [1]. The game is performed between a challenger C and a super-level type I adversary j for a CLS scheme as follows.

Initialization: C runs *Setup* phase and generates a master secret key mk, public system parameters *params*. Next, C keeps mk and gives *params* to the adversary j.

Queries: The adversary j can adaptively issue the following oracle queries [1, 3], i.e. *ExtractPartialPrivateKey*(i), *ExtractSecretValue*(i), *RequestPublicKey*(i), *ReplacePublicKey*(i), and *Sign*(i, m), to C.

Output: Eventually, the adversary j outputs (ID_t, m_t, σ_t). The adversary j wins the game if

(1) *ExtractPartialPrivateKey* (t) and *Sign*(t, m_t) queries have never been queried.

(2) $1 \leftarrow Verify(params, m_t, PK_t, P_{pub}, \sigma_t)$. Note that PK_t and P_{pub} may be replaced by the adversary j.

Definition: A CLS scheme is existentially unforgeable against a super-level type I adversary, if for any polynomially bounded super-level Type I adversary j, $Succ_j$ is negligible, where $Succ_j$ is the success probability that j wins in the above game.

3. Cryptanalysis of Gong-Li's CLS scheme

In this section, we briefly review Gong et al.'s CLS schemes [1]. Then, the cryptanalysis of Gong-Li's CLS scheme is demonstrated.

3.1 Review of Gong-Li scheme

Gong-Li's CLS scheme, short for Gong-Li scheme, consists of six steps, i.e. *Setup*, *PartialPrivateKeyExtract*, *SetSecretValue*, *SetPublicKey*, *Sign* and *Verify*. The detail of these steps is described as follows.

Setup: Given *k*, KGC generates the system parameters and the master key via the following computations.

(1) KGC generates a group G of elliptic curve points with prime order n and determines a generator P of G.

(2) KGC chooses the master key $mk = s \in Z_n^*$, and three secure hash functions H_1, H_2 and H_3, where $H_1 : \{0,1\}^* \times G \to Z_q^*$, $H_2 : \{0,1\}^* \times G \times G \times G \to Z_q^*$, $H_3 : \{0,1\}^* \times \{0,1\}^* \times G \times G \times G \to Z_q^*$. Next, KGC creates the master public key $P_{pub} = s \cdot P$.

(3) KGC publishes $params = \{G, P, P_{pub}, H_1, H_2, H_3\}$ as system parameters, and secretly keeps the master key *mk*.

PartialPrivateKeyExtract: Given *params*, *mk*, and user *i* 's identity ID_i, KGC generates a random number $r_i \in Z_n^*$, and calculates $R_i = r_i \cdot P$, $h_i = H_1(ID_i, R_i)$ and $s_i = r_i + h_i s \bmod n$. After that, KGC returns the partial private key $D_i = (s_i, R_i)$ to the user. The validity of D_i can be realized via the examination of the equation $s_i \cdot P = R_i + h_i \cdot P_{pub}$.

SetSecretValue: Given *params*, the user *i* with identity ID_i picks a random number $x_i \in Z_n^*$ as his/her secret value.

SetPublicKey: Given *params* and x_i, the user *i* computes $PK_i = x_i \cdot P$ as his/her public key.

Sign: Given *params*, D_i, x_i, and a message *m*, the user *i* generates a signature of *m* through the following steps.

(1) Compute $T_i = t_i \cdot P$ with a newly generated random number $t_i \in Z_n^*$.

(2) Compute $k_i = H_2(ID_i, PK_i, R_i, P_{pub})$, $l_i = H_3(m, T_i, ID_i, PK_i, R_i, P_{pub})$ and $\tau_i = t_i + l_i(k_i x_i + s_i) \bmod n$. Note that in Gong-Li's paper, the original equation of l_i is $l_i = H_3(m, T_i, ID_i, P_i, R_i, P_{pub})$; obviously, there exists a typo on the value P_i (actually, it should be PK_i) within the equation l_i.

(3) Return $\sigma_i = (R_i, T_i, \tau_i)$ as the signature of the message *m*.

Verify: Given *params*, ID_i, PK_i, *m* and $\sigma_i = (R_i, T_i, \tau_i)$, the verifier exploits the following steps to verify the validity of σ_i.

(1) Compute $h_i = H_1(ID_i, R_i)$, $k_i = H_2(ID_i, PK_i, R_i, P_{pub})$ and $l_i = H_3(m, T_i, ID_i, PK_i, R_i, P_{pub})$.

(2) Verify whether the equation $\tau_i \cdot P = T_i + l_i(k_i \cdot PK_i + R_i + h_i \cdot P_{pub})$ holds.

$$\begin{aligned}\tau_i \cdot P &= [t_i + l_i(k_i x_i + s_i)] \cdot P \\ &= t_i \cdot P + l_i(k_i x_i \cdot P + r_i \cdot P + h_i s \cdot P) \\ &= T_i + l_i(k_i \cdot PK_i + R_i + h_i \cdot P_{pub})\end{aligned}$$

3.2 Cryptanalysis of Gong-Li scheme

The Gong-Li scheme is vulnerable to a type I adversary with the following attack procedures. Suppose there exists a malicious type I adversary *j* which intends to forge a valid signature $\sigma_i' = (R_i', T_i', \tau_i')$ on the message *m*' chosen by the adversary *j*.

(1) The adversary *j* eavesdrops a valid signature $\sigma_i = (R_i, T_i, \tau_i)$ with message *m* issued by the user *i* from any previous session, where $T_i = t_i \cdot P$, $R_i = r_i \cdot P$ and $\tau_i = t_i + l_i(k_i x_i + r_i + h_i s)$.

(2) The adversary *j* performs the following computations to forge a valid signature on a chosen message *m'*. Since the adversary *j* is a Type I adversary, *j* can replace any entity's public key including KGC's public key.

a. Known values retrieved from previous session: $R_i = r_i \cdot P$, $T_i = t_i \cdot P$, $PK_i = x_i \cdot P$, $P_{pub} = s \cdot P$, $h_i = H_1(ID_i, R_i)$, $k_i = H_2(ID_i, PK_i, R_i, P_{pub})$ and $l_i = H_3(m, T_i, ID_i, PK_i, R_i, P_{pub})$.

b. The adversary *j* chooses a random number $t_i' \in Z_n^*$, and derives $T_i' = t_i' \cdot P$, $R_i' = (l_i)^{-1} T_i + R_i$, $h_i' = H_1(ID_i, R_i')$, $P_{pub}' = (h_i')^{-1} h_i P_{pub}$, $k_i' = H_2(ID_i, PK_i, R_i', P_{pub}')$ and $l_i' = H_3(m', T_i', ID_i, PK_i, R_i', P_{pub}')$.

c. Now, the adversary *j* can forge a valid signature $\sigma_i' = (R_i', T_i', \tau_i')$ on the chosen message *m'*. Note that the secret x_i can be retrieved via *ExtractSecretValue*(*i*) query.

i. Compute $\tau_i - l_i k_i x_i = t_i + l_i(r_i + h_i s)$.

ii. $t_i + l_i(r_i + h_i s)$ multiplies by $(l_i')(l_i)^{-1}$, i.e. $(l_i') \cdot [(l_i)^{-1} \cdot t_i + r_i + h_i s]$.

iii. Add t_i' and $l_i' k_i' x_i$ on the result from (ii).

$$t_i' + (l_i') \cdot [(l_i)^{-1} \cdot t_i + r_i + h_i s] + l_i' k_i' x_i$$
$$= t_i' + (l_i') \cdot \{k_i' x_i + [(l_i)^{-1} \cdot t_i + r_i] + h_i s\}$$

iv. Let τ_i' be the result from (iii)

$$\tau_i' = t_i' + (l_i') \cdot \{k_i' x_i + [(l_i)^{-1} \cdot t_i + r_i] + h_i s\}$$

d. With the following equation, it is obvious that the forge signature $\sigma_i' = (R_i', T_i', \tau_i')$ for the chosen message m' is valid, where $R_i' = (l_i)^{-1} T_i + R_i$, $T_i' = t_i' \cdot P$ and $P_{pub}' = (h_i')^{-1} h_i P_{pub}$.

$$\tau_i' \cdot P$$
$$= t_i' \cdot P + (l_i') \cdot \{k_i' x_i + [(l_i)^{-1} \cdot t_i + r_i] + h_i s\} \cdot P$$
$$= t_i' \cdot P + (l_i') \cdot \{k_i' x_i \cdot P + [(l_i)^{-1} \cdot t_i \cdot P + r_i \cdot P] + h_i s \cdot P\}$$
$$= T_i' + (l_i') \cdot \{k_i' \cdot PK_i + [(l_i)^{-1} \cdot T_i + R_i] + h_i \cdot P_{pub}\}$$
$$= T_i' + (l_i') \cdot \{k_i' \cdot PK_i + [(l_i)^{-1} \cdot T_i + R_i] + h_i' (h_i')^{-1} h_i \cdot P_{pub}\}$$
$$= T_i' + l_i' \cdot (k_i' \cdot PK_i + R_i' + h_i' \cdot P_{pub}')$$

4. Conclusions

In this paper, we have demonstrated that Gong and Li's CLS scheme is vulnerable to a malicious attack launched by a super type I adversary. This security vulnerability results from the weak connection among T_i, $l_i k_i PK_i$, R_i and $h_i P_{pub}$ within the signature $\sigma_i = (R_i, T_i, \tau_i)$. For this reason, Gong and Li's CLS scheme cannot fulfill the argued security claim, i.e. resistance to the super adversary.

5. Acknowledgment

The authors gratefully acknowledge the support from Taiwan Information Security Center (TWISC) and National Science Council, Taiwan, under the Grants Numbers NSC 102-2218-E-259-004, NSC 102-2218-E-146-002 and NSC 102-2218-E-011-012.

6. References

[1] P. Gong and P. Li, Further improvement of a certificateless signature scheme without pairing, *International Journal of Communication Systems*, DOI: 10.1002/dac.2457, Article first published online: 22 October 2012.

[2] J.-L. Tsai, N.-W. Lo and T.-C. Wu, Weaknesses and improvements of an efficient certificateless signature scheme without using bilinear pairings, *International Journal of Communication Systems*, DOI: 10.1002/dac.2388, Article first published online: 27 June 2012.

[3] D. He, J. Chen and R. Zhang, An efficient and provably-secure certificateless signature scheme without bilinear pairings, *International Journal of Communication Systems*, Vol.25, pp.1432-1442, 2012.

[4] D. He, J. Chen and J. Hu, A pairing-free certificateless authenticated key agreement protocol, *International Journal of Communication Systems*, Vol.25, pp.221-230, 2012.

[5] M. Tian and L. Huang, Cryptanalysis of a certificateless signature scheme without pairings, *International Journal of Communication Systems*, DOI: 10.1002/dac.2310, Article first published online: 20 Feb 2012.

[6] S. Al-Riyami and K. Paterson, Certificateless public key cryptography. In *Proceedings of ASIACRYPT 2003*, Lecture Notes in Computer Science, Vol. 2894, pp. 452-473, 2003.

[7] X. Huang, Y. Mu, W. Susilo, D.S.Wong and W. Wu, Certificateless signature revisited. In *Proceedings of ACISP 2007*, Lecture Notes in Computer Science, Vol. 4586, pp. 308-322, 2007.

[8] A. Shamir, "Identity-based cryptosystems and signature schemes," In *Proceedings of CRYPTO'84*, Lecture Notes in Computer Science, Vol. 196, pp. 47-53, 1985.

[9] X. Huang, W. Susilo, Y. Mu and F. Zhang "On the security of certificateless signature schemes from asiacrypt 2003," In *Proceedings of CANS 2005*, Lecture Notes in Computer Science, Vol. 3810, pp. 13-25, 2005.

[10] M. Gorantla and A. Saxena, "An efficient certificateless signature scheme," In *Proceedings of 2005 International Conference on Computational Intelligence and Security*, pp. 110-116, 2005.

[11] B.C. Hu, D.S. Wong, Z. Zhang and X. Deng, "Key replacement attack against a generic construction of certificateless signature," In *Proceedings of ACISP 2006*, Lecture Notes in Computer Science, Vol. 4058, pp. 235-46, 2006.

[12] X. Li, K. Chen and L. Sun, "Certificateless Signature and Proxy Signature Schemes from Bilinear Pairings," *Lithuanian Mathematical Journal*, Vol. 45, pp. 76-83, 2005.

[13] D. Yum and P. Lee, "Generic construction of certificateless signature," In *Proceeding of the 9th Australasian Conference on Information Security and Privacy*, pp. 200-211, 2004.