On the Security of Certificateless Signcryption Schemes

S. Sharmila Deva Selvi, S. Sree Vivek*, C. Pandu Rangan*

{sharmila,svivek,prangan}@cse.iitm.ac.in,
Indian Institute of Technology Madras
Theoretical Computer Science Laboratory
Department of Computer Science and Engineering
Chennai, India

Abstract. Signcryption is a cryptographic primitive which offers authentication and confidentiality simultaneously with a very low cost when compared to signing and encryption a message independently. Certificateless cryptography (CLC) is a relatively new filed where the public key of the user is not certified by a central authority, which overcomes the cumbersome certificate verification which is an ill fate in public key infrastructure (PKI). Certificateless systems provide a natural way to reduce the key escrow in identity based cryptosystems (IBC). In the literature there are four certificateless signcryption schemes and in this paper, we show that three out of them are insecure.

Keywords: Certificateless Signcryption, Cryptanalysis, Provable Security, Bilinear Pairing, Pairing-free Certificateless Signcryption.

1 Introduction

Signcryption is a cryptographic primitive, proposed by Zheng [7] that provides both authenticity and confidentiality with a very low computational cost when compared to signing and encrypting a message independently. The conventional public key cryptography employs a central authority that issues certificates for the public key of a user and manages a public key infrastructure. The PKI requires significant processing and storage capabilities inorder to maintain the certificates. An improvement proposed by Shamir [5] to reduce the overhead on PKI is identity based cryptosystem. In an IBC, the public key of a user in extracted from user identity, which is an unique string that identifies a user in the system. This eliminates the certificates required to link the public key with a user. In IBC, the private key corresponding to a user's public key is derived by a trusted authority called the private key generator (PKG), which leads to the key escrow problem in IBC. Certificateless cryptosystem was introduced by Al-Riyami and Paterson [1] inorder to reduce the trust on the PKG. In CLC, the private key of a user is a combination of a partial private key generated by the trusted authority, namely the key generation center (KGC) and a user secret value chosen by the user. Thus the KGC has access to the partial private key alone, which leaves it with a partial knowledge of the private key of any user in the system.

There are almost four certificateless signcryption (CLSC) schemes in the literature [3], [2], [6] and [4]. Three of them are totally pairing based [3], [2], [6] and one among them is a pairing-free system [4]. The scheme in [4] uses pairing only for the verification of public keys and may be considered as pairing-free. In this paper we show that the certificateless signcryption schemes in [3], [2] and [6] are insecure.

2 Preliminaries

In this section, we introduce the preliminary concepts used the the papers.

* Work supported by Project No. CSE/05-06/076/DITX/CPAN on Protocols for Secure Communication and Computation sponsored by Department of Information Technology, Government of India

2.1 Bilinear Pairing

Let $\mathbb{G}_1$ be an additive cyclic group generated by P, with prime order q, and $\mathbb{G}_2$ be a multiplicative cyclic group of the same order q. A bilinear pairing is a map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ with the following properties.

- ***Bilinearity.*** For all $P, Q, R \in \mathbb{G}_1$,
  - $\hat{e}(P + Q, R) = \hat{e}(P, R)\hat{e}(Q, R)$
  - $\hat{e}(P, Q + R) = \hat{e}(P, Q)\hat{e}(P, R)$
  - $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$
- ***Non-Degeneracy.*** There exist $P, Q \in \mathbb{G}_1$ such that $\hat{e}(P, Q) \neq I_{\mathbb{G}_2}$, where $I_{\mathbb{G}_2}$ is the identity element of $\mathbb{G}_2$.
- ***Computability.*** There exists an efficient algorithm to compute $\hat{e}(P, Q)$ for all $P, Q \in \mathbb{G}_1$.

3 Certificateless Signcryption (CLSC) Scheme of Barbosa et al.

3.1 Review of the scheme

The certificateless signcryption scheme uses an Encrypt-then-Sign approach. The randomness is shared between signature and encryption schemes. This scheme uses a symmetric bilinear group description Γ is defined as follows. The four cryptographic hash functions used are : $H_1 : \{0, 1\}^* \rightarrow G_1$, $H_2 : \{0, 1\}^* \rightarrow \{0, 1\}^\kappa$, $H_3 : \{0, 1\}^* \rightarrow G_1$, $H_4 : \{0, 1\}^* \rightarrow G_1$.
The master secret key Msk is selected uniformly at random from Z_p, and the master public key $Mpk = Msk.P$. The public parameters of the system are $= (\Gamma, Mpk)$. The partial private key extraction algorithm on input (ID, Msk) returns $D = Msk.H_1(ID)$. The user key generation algorithm returns a random element $x \in Z_p$ as the secret value, and $PK = x.P$ as the public key of user with identity ID. The full private key of user with identity ID is $S = (x, D)$. Message, ciphertext and randomness spaces are $\{0, 1\}^\kappa$, $G_1 \times \{0, 1\}^\kappa \times G_1$ and Z_p respectively
Signcrypt$(m, S_S = (x_S, D_S), ID_S, PK_S, ID_R, PK_R, Mpk)$

- Choose $r \in Z_p$.
- Compute $U = rP$ and $T = \hat{e}(Mpk, Q_R)^r$
- Compute $h = H_2(U, T, rPK_R, ID_R, PK_R)$
- Compute $V = m \oplus h$
- Compute $H = H_3(U, V, ID_S, PK_S)$ and $H' = H_4(U, V, ID_S, PK_S)$
- Compute $W = D_S + rH + x_S H'$
- Set $c = (U, V, W)$
- Return the signcryption c of message m from ID_S to ID_R.

Unsigncrypt$(c, S_R = (x_R, D_R), ID_R, PK_R, ID_S, PK_S, Mpk)$

- The ciphertext c is of the form (U, V, W).
- $H = H_3(U, V, ID_S, PK_S)$ and $H' = H_4(U, V, ID_S, PK_S)$
- Check if $e(Mpk, Q_S)e(U, H)e(PK_S, H') \overset{?}{=} e(P, W)$, else return $\perp$.
- Compute $T = \hat{e}(D_R, U)$
- Compute $h = H_2(U, T, x_R U, ID_R, PK_R)$
- Retrieve $m = V \oplus h$
- Return the message m

3.2 Analysis of the CLSC Scheme by Barbosa et al.

The scheme proposed by Barbosa et al. in [3] is liable to existential forgery. The scheme uses the Encrypt-then-Sign approach with public verifiability of ciphertext. The intuition behind the attack: for any signcryption scheme following the Encrypt-then-Sign approach, the identity of the sender should be bound to the encryption and the identity of the receiver should be bound to the signature. In [3], the authors have achieved this binding by using a common randomness for encryption and signature independently but they failed to bind the receiver to the signature. This led to the attack on existential unforgeability of [3]. The attack is shown below.

- During the unforgeability game(both type-1 and type-2), the forger requests a signcryption on a message m from ID_S^* to a arbitrary user with identity ID_A.
- Let the signcryption of m from ID_S^* to ID_A be $c = (U, V, W)$.
- Now, the forger submits $c^* = (U, V, W)$ as a signcryption from user ID_S^* to ID_R^* where ID_S^* and ID_R^* are the target identities for which the forger is restricted from knowing the private keys (partial private key for type-1 and private key for type-2). Note that c^* is a valid signcryption of some random message $m^* = m \oplus h \oplus h^*$ where $h^* = H_2(U, T^*, x_R^* U, ID_R^*, PK_R^*)$ and $T^* = \hat{e}(D_R^*, U)$. Here $h = H_2(U, T, x_A U, ID_A, PK_A)$ is the key used for encrypting the message m from ID_S^* to ID_A during signcryption.
- The signature W will pass the verification because none of the components of the H and H' are altered. The correctness of the signcryption is straight forward as follows.

  $$e(Mpk, Q_S)e(U, H)e(PK_S, H') = e(P, W)$$

  where $H = H_3(U, V, ID_S, PK_S)$ and $H' = H_4(U, V, ID_S, PK_S)$

  So the challenger will accept c^* as a valid forgery on message $m^* = m \oplus h \oplus h^*$.

4 Certificateless Signcryption (CLSC) Scheme of Diego et al.

In this section we give the review and attack of the certificateless signcryption scheme by Diego et al. given in [2].

4.1 Overview of the Scheme

Diego et al.'s CLSC scheme [2] consists of five algorithms namely: *Setup*, *Extract*, *Keygen*, *Signcrypt* and *Unsigncrypt*, which we describe below.

- ***Setup.*** Let κ be the security parameter. The KGC performs the following to set up the system.
  - The KGC selects cyclic groups $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ of same order q with generators $P \in_R \mathbb{G}_1$ and $Q \in_R \mathbb{G}_2$.
  - Selects the master secret key $s \in_R \mathbb{Z}_q^*$ and the master public key is set to be $P_{pub} = sP$.
  - Selects an admissible pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$.
  - Computes $g = \hat{e}(P, Q)$.
  - Selects three hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{Z}_q^*, H_2 : \mathbb{G}_T \rightarrow \{0,1\}^n, H_3 : \{0,1\}^n \times \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{Z}_q^*$, Here n is the length of the message.
  - The public parameters of the scheme are set to be *params*=(q, $\mathbb{G}_1$, $\mathbb{G}_2$, $\mathbb{G}_T$, $\hat{e}$, g, P, Q, P_{pub}, H_1, H_2, H_3).
- ***Extract.*** Here, ID_A is the identity of the user U_A, the KGC computes the partial private key of user U_A as follows.
  - Computes the hash value $y_A = H_1(ID_i)$ and the partial private key $D_A = (y_A + s)^{-1}Q \in \mathbb{G}_2$.
  - The KGC sends D_A to the user U_i via a secure authenticated channel.
- ***Keygen.*** User U_A computes the full private key by performing the following steps:
  - U_A chooses $x_A \in_R \mathbb{Z}_q^*$ as the secret value.
  - Computes the full private key $S_A = x_A^{-1} D_A \in \mathbb{G}_2$.

  - Computes the public key as $P_A = x_A(y_A P + P_{pub}) \in \mathbb{G}_{\mathbb{K}}$.
  - It is to be noted that $\hat{e}(P_A, S_A) = g$.

- ***Signcrypt.*** Inorder to signcrypt the message m to the receiver U_B, the sender U_A does the following:
  - Chooses $r \in_R \mathbb{Z}_q^*$, computes $u = r^{-1}$ and $U = g^u$.
  - Computes $c = m \oplus H_2(U)$, $R = rP_A$ and $S = uP_B$.
  - Computes $h = H_3(c, R, S)$ and $T = (r + h)^{-1} S_A$.

  Finally, the sender outputs the signcryption on message m as $\sigma = (c, R, S, T)$.

- ***Unsigncrypt.*** Inorder to unsigncrypt a ciphertext σ, the receiver U_B does the following:
  - Computes $h' = H_3(c, R, S)$.
  - Computes $U' = \hat{e}(S, S_B)$.
  - Recovers the message as $m' = c \oplus H_2(U')$.
  - Checks whether $\hat{e}(R + h'P_A, T) \overset{?}{=} g$.

  If the check holds, then accepts m' as the message, otherwise outputs *Invalid*.

4.2 Analysis of the CLSC Scheme by Diego et al.

Type-I Forgeability: The Type-I adversary who is capable of replacing the public keys of all users and is restricted from knowing the master private key can forge a valid signcryption on any message m, from any legitimate user U_A to U_B by performing the following:

- Let ID_A be the identity of user U_A.
- The adversary chooses $r \in_R \mathbb{Z}_q^*$, computes $u = r^{(-1)}$.
- Computes $U = g^u$ and sets $c = m \oplus H_2(U)$.
- Set $T = r^{-1}Q$, $R = rP - P$ and $S = uP_B$.
- Compute $h = H_3(c, R, S)$.
- Set $P_A = h^{-1}P$.

Finally, the forger outputs the signcryption on message m as $\sigma = (c, R, S, T)$ which is a valid signcryption on m from U_A to U_B.

Correctness: The correctness of the scheme with respect to the verification test is given below,

$$\begin{aligned}
\hat{e}(R + hP_A, T) &= \hat{e}(rP - P + hh^{-1}P, r^{-1}Q) \\
&= \hat{e}(rP, r^{-1}Q)\hat{e}(-P + P, r^{-1}Q) \\
&= \hat{e}(P, Q)\hat{e}(-P, r^{-1}Q)\hat{e}(P, r^{-1}Q) \\
&= \hat{e}(P, Q) \\
&= g
\end{aligned}$$

This proves that the forgery generated is valid.

Type-I and Type-II Attacks on Confidentiality:

- Let $\sigma^* = (c^*, R^*, S^*, T^*)$ be the challenge signcryption on message m_b, $b \in \{0, 1\}$ with ID_A as the sender and ID_B as the receiver.
- The adversary is capable of generating a new signcryption σ' on the message m_b (The message is same as in σ^*) with ID_C as sender and ID_B as receiver (Note that the adversary knows the private key of ID_C).
- σ' is computed by performing the following:
  - Sets $c' = c^*$.
  - Computes $R' = r'P_C$, where $r' \in_R \mathbb{Z}_q^*$.
  - Set $S' = S^*$.
  - Computes $h' = H_3(c', R', S')$
  - Set $T' = (r' + h')^{-1} S_C$
  - The signcryption corresponding to this change is $\sigma' = (c', R', S', T')$.

- Now, the adversary can query the unsigncryption oracle for the unsigncryption of σ' (Note that this query is valid because σ' is different from the challenge signcryption σ^*).
- The unsigncryption oracle will give back the message m_b since the key used in both σ^* and σ' are the same i.e., $U' = \hat{e}(S', S_B) = \hat{e}(S^*, S_B) = U^*$ and note that $S' = S^*$. Hence , $c' \oplus H_2(U') = c^* \oplus H_2(U^*) = m_b$.
- Therefore, designcryption of σ' outputs the message m_b, which is used for generating the challenge ciphertext σ^*. Thus the adversary can completely determine whether m_b=m_0 or m_1. Hence, breaking the indistinguishability.

5 Certificateless Signcryption (CLSC) Scheme of Chen-Huang et al.

In this section we give the review and attack of the certificateless signcryption scheme by Chen-Huang et al. given in [6].

5.1 Overview of the Scheme

The CLSC scheme of Chen-Huang et al. [6] consists of the following four algorithms.

- ***Setup.*** Given κ as the security parameter, the KGC does the following to setup the system parameters.
  - The KGC selects $\mathbb{G}_1$, $\mathbb{G}_2$ of same order q with a generator $P \in_R \mathbb{G}_1$.
  - Selects the master secret key $s \in_R \mathbb{Z}_q^*$ and the master public key is set to be $P_{pub} = sP$.
  - Selects an admissible pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$.
  - Selects three cryptographic hash functions $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1, H_2 : \{0,1\}^* \rightarrow \mathbb{Z}_q^*, H_3 : \{0,1\}^* \rightarrow \{0,1\}^n$.
  - Computes $T = \hat{e}(P, P)$.
  - The public parameters of the scheme are set to be *params*=(q, $\mathbb{G}_1$, $\mathbb{G}_2$, $\hat{e}$, n, P, P_{pub}, T, H_1, H_2, H_3).
- ***Keygen.*** Let, ID_A is the identity of the user U_A. The KGC computes the partial private key of user U_A as follows.
  - Computes $Q_A = H_1(ID_A)$ and the partial private key $D_A = sQ_A \in \mathbb{G}_2$.
  - The KGC sends D_A to the user U_i via a secure authenticated channel.

  On receiving the partial private key D_A, user U_A computes his full private key by performing the following steps:
  - U_A chooses $x_A \in_R \mathbb{Z}_q^*$ as the secret value.
  - Sets the full private key $S_A = \langle x_A, D_A \rangle$.
  - The corresponding public key is $P_A = T^{x_A} \in \mathbb{G}_{\not 2}$.
- ***Signcrypt.*** Inorder to signcrypt the message m of length n to the receiver U_B, the sender U_A does the following:
  - Chooses $r, r_1, r_2 \in_R \mathbb{Z}_q^*$, computes $R_1 = T^{r_1}$ and $R_2 = T^{r_2}$.
  - Computes $h = H_2(m \| R_1 \| R_2 \| P_A \| P_B)$.
  - Computes $U = r_1 P - hS_A$ and $u = r_2 - x_A h$.
  - Computes $K = \hat{e}(S_A, Q_B)^r T_B^{x_A}$ and $W = rQ_A$.
  - Sets $c = H_3(K) \oplus m$

  Finally, the sender outputs the signcryption on message m as $\sigma = (c, u, h, U, W)$.
- ***Unsigncrypt.*** Inorder to unsigncrypt a ciphertext σ, the receiver U_B does the following:
  - Computes $K' = \hat{e}(S_B, W) T_A^{x_B}$.
  - Retrieves the message as $m' = c \oplus H_3(K')$.
  - Checks whether $h \overset{?}{=} H_2(m' \| \hat{e}(U, P) \hat{e}(Q_A, P_{pub})^h) \| T^u P_A^h \| P_A \| P_B)$.

  If the check holds, then accepts m' as the message, otherwise outputs *Invalid.*

5.2 Analysis of the CLSC Scheme by Chen-Huang et al.

In this section we show that the certificateless signcryption scheme by Chen-Huang et al. does not provide confidentiality as well as unforgeability with respect to both Type-I and Type-II attacks.

Attack on Type-I and Type-II Confidentiality: On getting the challenge signcryption $\sigma^* = \langle c^*, u^*, h^*,$ U^*, $W^*\rangle$,(σ^* is the encryption of either message m_0 or m_1 from user U_A to U_B) the adversary (Type-I and Type-II) is capable of generating a new ciphertext $\sigma' = \langle c', u', h', U', W'\rangle$ (signcryption of m_0 from user U_C to U_B) as follows:

- Replace the public key of user U_C with the public key of user U_A.
- Sets $c' = c^*$ and $W' = W^*$.
- Chooses $r_1, r_2 \in_R \mathbb{Z}_q^*$, computes $R_1 = T^{r_1}$ and $R_2 = T^{r_2}$.
- Computes $h' = H_2(m_0 \| R_1 \| R_2 \| P_C \| P_B)$.
- Computes $U' = r_1 P - h' S_C$ and $u = r_2 - x_C h'$.
- Gets the unsigncryption of σ'.
- If $Unsigncrypt(\sigma') = m_0$ then the adversary outputs that σ^* is the signcryption of m_0.
- If $Unsigncrypt(\sigma') = Invalid$ then the adversary outputs m_1.

Note that this attack can be done by both Type-I and Type-II adversaries.

6 Conclusion

In this paper, we have shown the weaknesses in three existing certificateless signcryption schemes, all the three are pairing based schemes.

References

1. Sattam S. Al-Riyami and Kenneth G. Paterson. Certificateless public key cryptography. In *Advances in Cryptology - ASIACRYPT 2003*, volume 2894 of *Lecture Notes in Computer Science*, pages 452–473. Springer, 2003.
2. Diego Aranha, Rafael Castro, Julio Lopez, and Ricardo Dahab. Efficient certificateless signcryption. http://sbseg2008.inf.ufrgs.br/proceedings/data/pdf/st03_01_resumo.pdf.
3. Manuel Barbosa and Pooya Farshim. Certificateless signcryption. In *ACM Symposium on Information, Computer and Communications Security - ASIACCS 2008*, pages 369–372. ACM, 2008.
4. Paulo S. L. M. Barreto, Alexandre Machado Deusajute, Eduardo de Souza Cruz, Geovandro C. F. Pereira, and Rodrigo Rodrigues da Silva. Toward efficient certificateless signcryption from (and without) bilinear pairings. http://sbseg2008.inf.ufrgs.br/proceedings/data/pdf/st03_03_artigo.pdf.
5. Adi Shamir. Identity-based cryptosystems and signature schemes. In *Advances in Cryptology, CRYPTO - 1984*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer, 1984.
6. Chenhuang Wu and Zhixiong Chen. A new efficient certificateless signcryption scheme. *International Symposium on Information Science and Engieering, 2008. ISISE '08.*, Vol: 1:661–664, 2008.
7. Yuliang Zheng. Digital signcryption or how to achieve cost(signature & encryption) < < cost(signature) + cost(encryption). In *Advances in Cryptology, CRYPTO - 1997*, volume 1294 of *Lecture Notes in Computer Science*, pages 165–179. Springer, 1997.