Cryptanalysis of pairing-free certificateless authenticated key agreement protocol

Zhian Zhu

China Ship Development and Design Center (CSDDC), Wuhan, China

Email: zhuzhian2012@gmail.com

Abstract*:* Recently, He et al. [D. He, J. Chen, J. Hu, A pairing-free certificateless authenticated key agreement protocol, *International Journal of Communication Systems*, 25(2), pp. 221-230, 2012] proposed a pairing-free certificateless authenticated key agreement protocol and demonstrated that their protocol is provable security in the random oracle model. However, in this paper, we show that t He et al. protocol is completely broken.

Key words*: certificateless cryptography; authenticated key agreement; provable security; elliptic curve*

1. Introduction

To simplify the complex certificate management in the traditional public key cryptography (PKC), Shamir [1] proposed the concept of identity-based public key cryptography (ID-PKC). In ID-PKC, there is no need of the certificate of a public key since the user's public key is his identity such as e-mail address, telephone number et al. However, ID-PKC inherently has the key escrow problem, i.e., the key generation center (KGC) knows the user's private key. To solve the problem, Al-Riyami et al. [2] introduced the concept of the certificateless public key cryptography (CLPKC). Since then, many certificateless key agreement protocols using bilinear pairings have been proposed.

However, the theoretical analysis [3] and experimental result [4] show that the relative computation cost of a pairing is approximately twenty times higher than that of the scalar multiplication over elliptic curve group. Therefore, the performance of all the above protocols is not satisfactory. To improve performance, He et al. [5] proposed a pairing-free certificateless key agreement protocol. They also demonstrated that their protocol is provably secure in the random oracle model. However, we will show He et al.'s protocol is insecure against both of the type I adversary and the type II adversary.

The organization of the paper is sketched as follows. The Section 2 gives a brief review of He et al.'s protocol. Cryptanalysis on He et al.'s protocol is shown in Section 3. Finally, we give some conclusions in Section 4.

2. He et al.'s protocol

In this section, we will review He et al.'s protocol. For convenience, some notations used in this paper are described as follows.

- p, n : two large prime numbers;
- F_p : a finite field;
- E / F_p : an elliptic curve defined on F_p ;
- G : the cyclic additive group composed of the points on E / F_p ;
- P : a generator of G ;
- $H_1(\cdot)$: a secure one-way hash function, where $H_1 : \{0,1\}^* \to Z_n^*$;
- $H_2(\cdot)$: a secure one-way hash function, where $H_2 : \{0,1\}^* \to Z_n^*$;
- ID_i : the identity of user i ;
- (x, P_{pub}) : the KGC's private/public key pair, where $P_{pub} = xP$;
- (x_i, P_i) : the user i 's secret value/public key pair, where $P_i = x_i \cdot P$;
- (r_i, R_i) : a random point generated by KGC, where $R_i = r_i \cdot P$;
- (s_i, R_i) : the user i 's partial private key, where $s_i = r_i + h_i x \bmod n$, $h_i = H_1(ID_i, R_i)$;
- t_i, T_i): the user i 's ephemeral private/public key pair, where $T_i = t_i \cdot P$;

In this section, we present a CTAKA protocol without pairing. Our protocol consists of the following six algorithms: *Setup*, *Partial-Private-Key-Extract*, *Set-Secret-Value*, *Set-Private-Key*, *Set-Public-Key* and *Key-Agreement*.

Setup: This algorithm takes a security parameter k as inputs and returns system parameters and a master key. Given k , KGC does the following.

1) KGC chooses the master private key $x \in Z_n^*$ and computes the master public key $P_{pub} = xP$.
2) KGC chooses two cryptographic secure hash functions $H_1 : \{0,1\}^* \to Z_n^*$ and $H_2 : \{0,1\}^* \to Z_n^*$.
3) KGC publishes $params = \{F_p, E / F_p, G, P, P_{pub}, H_1, H_2\}$ as system parameters and secretly keeps the master key x .

Partial-Private-Key-Extract: This algorithm takes master key, a user's identifier, system parameters as input and returns the user's ID-based private key. With this algorithm, for each user with identifier ID_i, KGC works as follows.

1) KGC chooses at random $r_i \in Z_n^*$, computes $R_i = r_i \cdot P$ and $h_i = H_1(ID_i, R_i)$.
2) KGC computes $s_i = r_i + h_i x \bmod n$.

The user's s partial private key is the tuple $D_i = (s_i, R_i)$ and he can validate her private key by checking whether the equation $s_i \cdot P = R_i + h_i \cdot P_{pub}$ holds. The private key is valid if the equation holds and vice versa.

Set-Secret-Value: The user with identity ID_i picks randomly $x_i \in Z_n^*$ sets x_i as his secret value.

Set-Private-Key: The user with identity ID_i takes the pair $S_i = (x_i, D_i)$ as its private key, where $D_i = (s_i, R_i)$.

Set-Public-Key: The user with identity ID_i takes *params* and its secret value x_i as inputs, and generates its public key $P_i = x_i \cdot P$.

Key-Agreement: Assume that an entity A with identity ID_A has private key $S_A = (x_A, D_A)$ and public key $P_A = x_A \cdot P$, an entity B with identity ID_B has private key $S_B = (x_B, D_B)$ and public key $P_B = x_B \cdot P$. As shown in Fig. 1, A and B run the protocol as follows.

1) A send $M_1 = (ID_A, R_A, P_A)$ to B.

2）After receiving M_1, B chooses at random the ephemeral key $b \in Z_n^*$ and computes $T_B = b \cdot (P_A + R_A + H_1(ID_A, R_A) P_{pub})$, then B send $M_2 = (ID_B, R_B, P_B, T_B)$ to A.

3) After receiving M_2, A chooses at random the ephemeral key $a \in Z_n^*$ and computes $T_A = a \cdot (P_B + R_B + H_1(ID_B, R_B) P_{pub})$, then A send $M_3 = (T_A)$ to B.

Then both A and B can compute the shared secrets as follows.

A computes

$$K_{AB}^1 = (x_A + s_A)^{-1} \cdot T_B + a \cdot P \quad \text{and} \quad K_{AB}^2 = a \cdot (x_A + s_A)^{-1} \cdot T_B \qquad (1)$$

B computes

$$K_{BA}^{1} = (x_B + s_B)^{-1} \cdot T_A + b \cdot P \quad \text{and} \quad K_{AB}^{2} = b \cdot (x_B + s_B)^{-1} \cdot T_A \qquad (2)$$

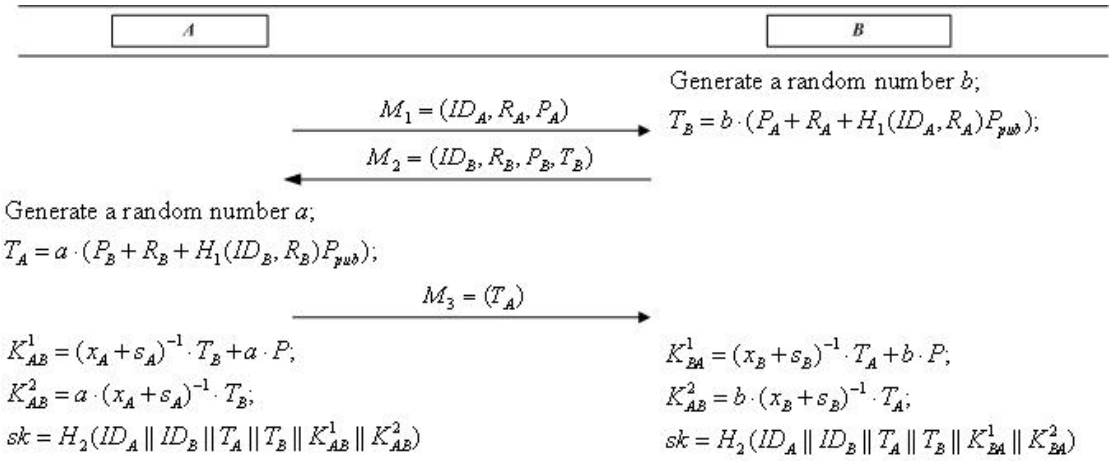

Fig. 1. Key agreement of He et al.'s protocol

3. Cryptanalysis of He et al.'s protocol

In certificateless signature protocol, as defined in [2, 5], there are two types of adversaries with different capabilities, we assume type I adversary, $\mathcal{A}1$ acts as a dishonest user, while type II adversary, $\mathcal{A}2$ acts as a malicious KGC.

In this section, we will show that He et al. protocol is vulnerable to the impersonation attack against of $\mathcal{A}1$ and $\mathcal{A}2$. The detailed steps are described as follows.

3.1. Impersonation attack of type I adversary

Let $\mathcal{A}1$ be a type I adversary. Then, $\mathcal{A}1$ does not have access to the master key, but $\mathcal{A}1$ can replace the public keys of any entity with a value of his choice, since there is no certificate involved in certificateless signature protocol.

- **Impersonation attack against initiator**

1) $\mathcal{A}1$ generates a random number t and replaces A's public key $P_A = x_A \cdot P$ with $P_A' = tP - R_A - H_1(ID_A, R_A)P_{pub}$.

2) $\mathcal{A}1$ impersonate A to sends the message $M_1 = (ID_A, R_A, P_A')$ to B.

3) After receiving M_1, B chooses at random the ephemeral key $b \in Z_n^*$ and computes $T_B = b \cdot (P_A' + R_A + H_1(ID_A, R_A)P_{pub})$, then B send $M_2 = (ID_B, R_B, P_B, T_B)$ to $\mathcal{A}1$.

4) After receiving M_2, $\mathscr{A}1$ chooses at random the ephemeral key $a \in Z_n^*$, computes $T_A = a \cdot (P_B + R_B + H_1(ID_B, R_B)P_{pub})$ and send $M_3 = (T_A)$ to B.

5) After receiving M_3, B will computes $K_{BA}^1 = (x_B + s_B)^{-1} \cdot T_A + b \cdot P = aP + bP$, $K_{AB}^2 = b \cdot (x_B + s_B)^{-1} \cdot T_A = abP$ and the session key $sk_{BA} = H_2(ID_A \| ID_B \| T_A \| T_B \| K_{BA}^1 \| K_{BA}^2)$.

Since $P_A' = tP - R_A - H_1(ID_A, R_A)P_{pub}$ and $T_B = b \cdot (P_A' + R_A + H_1(ID_A, R_A)P_{pub})$, then $\mathscr{A}1$ computes

$$\begin{aligned} K_{AB}^1 &= t^{-1} \cdot T_B + a \cdot P = t^{-1} \cdot b \cdot (P_A' + R_A + H_1(ID_A, R_A)P_{pub}) + a \cdot P \\ &= t^{-1} \cdot b \cdot (tP - R_A - H_1(ID_A, R_A)P_{pub} + R_A + H_1(ID_A, R_A)P_{pub}) + a \cdot P \\ &= t^{-1} \cdot b \cdot tP + aP = bP + aP = K_{BA}^1 \end{aligned} \quad (3)$$

and

$$\begin{aligned} K_{AB}^2 &= a \cdot t^{-1} \cdot T_B = a \cdot t^{-1} \cdot b \cdot (P_A' + R_A + H_1(ID_A, R_A)P_{pub}) \\ &= a \cdot t^{-1} \cdot b \cdot (tP - R_A - H_1(ID_A, R_A)P_{pub} + R_A + H_1(ID_A, R_A)P_{pub}) \\ &= a \cdot t^{-1} \cdot b \cdot tP = abP = K_{BA}^2 \end{aligned} \quad (4)$$

Thus $\mathscr{A}1$ could compute $sk_{AB} = H_2(ID_A \| ID_B \| T_A \| T_B \| K_{AB}^1 \| K_{AB}^2)$ as the session key. It is easy to say that sk_{AB} and sk_{BA} are equal since $K_{AB}^1 = K_{BA}^1$ and $K_{AB}^2 = K_{BA}^2$. Then $\mathscr{A}1$ impersonate the initiator A successfully.

- **Impersonation attack against responder**

1) $\mathscr{A}1$ generates a random number t and replaces B's public key $P_B = x_B \cdot P$ with $P_B' = tP - R_B - H_1(ID_B, R_B)P_{pub}$.

2) After intercept the message $M_1 = (ID_A, R_A, P_A)$ sent by A, $\mathscr{A}1$ chooses at random the ephemeral key $b \in Z_n^*$, computes $T_B = b \cdot (P_A + R_A + H_1(ID_A, R_A)P_{pub})$. Then $\mathscr{A}1$ impersonate B to send back $M_2 = (ID_B, R_B, P_B', T_B)$ to $\mathscr{A}1$.

3) After receiving M_2, A chooses at random the ephemeral key $a \in Z_n^*$, computes $T_A = a \cdot (P_B' + R_B + H_1(ID_B, R_B)P_{pub})$, $K_{AB}^2 = a \cdot (x_A + s_A)^{-1} \cdot T_B = abP$, $K_{AB}^1 = (x_A + s_A)^{-1} \cdot T_B + a \cdot P = aP + bP$, $sk_{AB} = H_2(ID_A \| ID_B \| T_A \| T_B \| K_{AB}^1 \| K_{AB}^2)$ and send $M_3 = (T_A)$ to $\mathscr{A}1$.

4) After receiving M_3, $\mathscr{A}1$ will computes $K_{BA}^1 = t^{-1} \cdot a \cdot T_A + b \cdot P$, $K_{AB}^2 = b \cdot t^{-1} \cdot T_A$ and the session key $sk_{BA} = H_2(ID_A \| ID_B \| T_A \| T_B \| K_{BA}^1 \| K_{BA}^2)$.

Since $P_B' = tP - R_B - H_1(ID_B, R_B)P_{pub}$ and $T_A = a \cdot (P_B' + R_B + H_1(ID_B, R_B)P_{pub})$, then $\mathscr{A}1$ computes

$$\begin{aligned} K_{BA}^1 &= t^{-1} \cdot a \cdot T_A + b \cdot P = t^{-1} \cdot (P_B' + R_B + H_1(ID_B, R_B)P_{pub}) + b \cdot P \\ &= t^{-1} \cdot a \cdot (tP - R_B - H_1(ID_B, R_B)P_{pub} + R_B + H_1(ID_B, R_B)P_{pub}) + b \cdot P \\ &= t^{-1} \cdot a \cdot tP + bP = aP + bP = K_{AB}^1 \end{aligned} \quad (5)$$

and

$$\begin{aligned} K_{AB}^2 &= b \cdot t^{-1} \cdot T_A = b \cdot t^{-1} \cdot a \cdot (P_B' + R_B + H_1(ID_B, R_B)P_{pub}) \\ &= b \cdot t^{-1} \cdot a \cdot (tP - R_B - H_1(ID_B, R_B)P_{pub} + R_B + H_1(ID_B, R_B)P_{pub}) \\ &= b \cdot t^{-1} \cdot a \cdot tP = abP = K_{BA}^2 \end{aligned} \quad (6)$$

Thus $\mathscr{A}1$ could compute $sk_{AB} = H_2(ID_A \| ID_B \| T_A \| T_B \| K_{AB}^1 \| K_{AB}^2)$ as the session key. It is easy to say that sk_{AB} and sk_{BA} are equal since $K_{AB}^1 = K_{BA}^1$ and $K_{AB}^2 = K_{BA}^2$. Then $\mathscr{A}1$ impersonate the responder B successfully.

3.2 Impersonation attack of type II adversary

Let $\mathscr{A}1$ be a type I adversary. Then, $\mathscr{A}2$ has access to the master key x, but cannot replace any user's public key.

- **Impersonation attack against initiator**

1) $\mathscr{A}2$ generates a random number t, computes $R_A' = tP - P_A$ and impersonate A to send the message $M_1 = (ID_A, R_A', P_A)$ to B.

2)After receiving M_1, B chooses at random the ephemeral key $b \in Z_n^*$ and computes $T_B = b \cdot (P_A + R_A' + H_1(ID_A, R_A')P_{pub})$, then B send $M_2 = (ID_B, R_B, P_B, T_B)$ to $\mathscr{A}2$.

3) After receiving M_2, $\mathscr{A}2$ chooses at random the ephemeral key $a \in Z_n^*$, computes $T_A = a \cdot (P_B + R_B + H_1(ID_B, R_B)P_{pub})$ and send $M_3 = (T_A)$ to B.

4) After receiving M_3, B will computes $K_{BA}^1 = (x_B + s_B)^{-1} \cdot T_A + b \cdot P = aP + bP$, $K_{AB}^2 = b \cdot (x_B + s_B)^{-1} \cdot T_A = abP$ and the session key $sk_{BA} = H_2(ID_A \| ID_B \| T_A \| T_B \| K_{BA}^1 \| K_{BA}^2)$.

Since $T_B = b \cdot (P_A + R'_A + H_1(ID_A, R'_A)P_{pub})$, $R'_A = tP - P_A$ and $\mathscr{A}2$ knows the mast key x. Then $\mathscr{A}2$ computes

$$\begin{aligned} K_{AB}^1 &= (t + xH_1(ID_A, R'_A))^{-1} \cdot T_B + a \cdot P \\ &= (t + xH_1(ID_A, R'_A))^{-1} \cdot b \cdot (P_A + R'_A + H_1(ID_A, R'_A)P_{pub}) + a \cdot P \\ &= (t + xH_1(ID_A, R'_A))^{-1} \cdot b \cdot (P_A + tP - P_A + H_1(ID_A, R'_A)P_{pub}) + a \cdot P \\ &= (t + xH_1(ID_A, R'_A))^{-1} \cdot b \cdot (tP + H_1(ID_A, R'_A)xP) + a \cdot P \\ &= (t + xH_1(ID_A, R'_A))^{-1} \cdot b \cdot (t + H_1(ID_A, R'_A)x)P + a \cdot P \\ &= bP + aP = K_{BA}^1 \end{aligned} \tag{7}$$

and

$$\begin{aligned} K_{AB}^2 &= a \cdot (t + xH_1(ID_A, R'_A))^{-1} \cdot T_B \\ &= a \cdot (t + xH_1(ID_A, R'_A))^{-1} \cdot b \cdot (P_A + R'_A + H_1(ID_A, R'_A)P_{pub}) \\ &= a \cdot (t + xH_1(ID_A, R'_A))^{-1} \cdot b \cdot (P_A + tP - P_A + H_1(ID_A, R'_A)P_{pub}) \\ &= a \cdot (t + xH_1(ID_A, R'_A))^{-1} \cdot b \cdot (tP + H_1(ID_A, R'_A)xP) \\ &= a \cdot (t + xH_1(ID_A, R'_A))^{-1} \cdot b \cdot (t + H_1(ID_A, R'_A)x)P \\ &= abP = K_{BA}^2 \end{aligned} \tag{8}$$

Thus $\mathscr{A}2$ could compute $sk_{AB} = H_2(ID_A \| ID_B \| T_A \| T_B \| K_{AB}^1 \| K_{AB}^2)$ as the session key. It is easy to say that sk_{AB} and sk_{BA} are equal since $K_{AB}^1 = K_{BA}^1$ and $K_{AB}^2 = K_{BA}^2$. Then $\mathscr{A}2$ impersonate the initiator A successfully.

- **Impersonation attack against responder**

1) $\mathscr{A}2$ generates a random number t and computes $R'_B = tP - P_B$.

2) After intercept the message $M_1 = (ID_A, R_A, P_A)$ sent by A, $\mathscr{A}2$ chooses at random the ephemeral key $b \in Z_n^*$, computes $T_B = b \cdot (P_A + R_A + H_1(ID_A, R_A)P_{pub})$. Then $\mathscr{A}2$ impersonate B to send back $M_2 = (ID_B, R'_B, R_B, T_B)$ to $\mathscr{A}2$.

3) After receiving M_2, A chooses at random the ephemeral key $a \in Z_n^*$, computes $T_A = a \cdot (P_B + R'_B + H_1(ID_B, R'_B)P_{pub})$, $K_{AB}^2 = a \cdot (x_A + s_A)^{-1} \cdot T_B = abP$, $K_{AB}^1 = (x_A + s_A)^{-1} \cdot T_B + a \cdot P = aP + bP$, $sk_{AB} = H_2(ID_A \| ID_B \| T_A \| T_B \| K_{AB}^1 \| K_{AB}^2)$ and send $M_3 = (T_A)$ to $\mathscr{A}2$.

4) After receiving M_3, $\mathscr{A}2$ will computes $K_{BA}^1 = (t + xH_1(ID_B, R'_B))^{-1} \cdot a \cdot T_A + b \cdot P$, $K_{AB}^2 = b \cdot (t + xH_1(ID_A, R'_A))^{-1} \cdot T_A$ and the session key $sk_{BA} = H_2(ID_A \| ID_B \| T_A \| T_B \| K_{BA}^1 \| K_{BA}^2)$.

Since $R'_B = tP - P_B$ and $T_A = a \cdot (P_B + R'_B + H_1(ID_B, R'_B)P_{pub})$, then $\mathscr{A}2$ computes

$$\begin{aligned} K_{BA}^1 &= (t + xH_1(ID_B, R'_B))^{-1} \cdot T_A + b \cdot P \\ &= (t + xH_1(ID_B, R'_B))^{-1} \cdot a \cdot (P_B + R'_B + H_1(ID_B, R'_B)P_{pub}) + b \cdot P \\ &= (t + xH_1(ID_B, R'_B))^{-1} \cdot a \cdot (P_B + tP - P_B + H_1(ID_B, R'_B)P_{pub}) + b \cdot P \\ &= (t + xH_1(ID_B, R'_B))^{-1} \cdot a \cdot (tP + H_1(ID_B, R'_B)xP) + b \cdot P \\ &= (t + xH_1(ID_B, R'_B))^{-1} \cdot a \cdot (t + xH_1(ID_B, R'_B))P + b \cdot P \\ &= aP + bP = K_{AB}^1 \end{aligned} \tag{5}$$

and

$$\begin{aligned} K_{AB}^2 &= b \cdot (t + xH_1(ID_B, R'_B))^{-1} \cdot T_A \\ &= b \cdot (t + xH_1(ID_B, R'_B))^{-1} \cdot a \cdot (P_B + R'_B + H_1(ID_B, R'_B)P_{pub}) \\ &= b \cdot (t + xH_1(ID_B, R'_B))^{-1} \cdot a \cdot (P_B + tP - P_B + H_1(ID_B, R'_B)P_{pub}) \\ &= b \cdot (t + xH_1(ID_B, R'_B))^{-1} \cdot a \cdot (tP + H_1(ID_B, R'_B)xP) \\ &= b \cdot (t + xH_1(ID_B, R'_B))^{-1} \cdot a \cdot (t + xH_1(ID_B, R'_B))P \\ &= abP = K_{BA}^2 \end{aligned} \tag{6}$$

Thus $\mathscr{A}2$ could compute $sk_{AB} = H_2(ID_A \| ID_B \| T_A \| T_B \| K_{AB}^1 \| K_{AB}^2)$ as the session key. It is easy to say that sk_{AB} and sk_{BA} are equal since $K_{AB}^1 = K_{BA}^1$ and $K_{AB}^2 = K_{BA}^2$. Then $\mathscr{A}2$ impersonate the responder B successfully.

4. Conclusion

Recently, He et al. proposed a certificateless authenticated key agreement protocol without bilinear pairings and demonstrated its immunity against various attacks. However, after review of their protocol and analysis of its security, we show their protocol is insecure against both of the type I adversary and the type II adversary. The analyses show that the protocol is insecure for practical application.

Reference

[1]. Shamir A. Identity-based cryptosystems and signature protocols. In Proceedings of CRYPTO'84. Lecture Notes in Computer Science, Vol. 196. Springer: Berlin, 1985; 47–53.

[2]. Al-Riyami S, Paterson K. Certificateless public key cryptography. In Proceedings of ASIACRYPT 2003. Lecture Notes in Computer Science, Vol. 2894. Springer: Berlin, 2003; 452–473.

[3]. Chen L, Cheng Z, Smart N, Identity-based key agreement protocols from pairings, International Journal of Information Security, 6, pp. 213–241, 2007.

[4]. He D, Chen J, Zhang R. An efficient and provably-secure certificateless signature protocol without bilinear pairings. International Journal of Communication Systems, doi: 10.1002/dac.1330, 2011.

[5]. He D, Chen J, Hu J., A pairing-free certificateless authenticated key agreement protocol, International Journal of Communication Systems, 25(2), pp. 221-230, 2012.