Journal of Physics: Conference Series

PAPER • OPEN ACCESS

Lightweight hybrid signature scheme for Internet of Thing based on bilinear mapping

To cite this article: Gangpeng Duan 2022 *J. Phys.: Conf. Ser.* **2294** 012012

Lightweight hybrid signature scheme for Internet of Thing based on bilinear mapping

Gangpeng Duan[1]

[1] School of Computer and Information Science, Chongqing Normal University, Chong qing, 401331, China

Abstract. Most of the existing IoT communication encryption schemes have the following two problems: the sensor side needs to perform complex bilinear mapping calculations; most schemes separate key agreement and data encryption, which increases the user's computational burden and management difficulty. This paper proposes a lightweight IoT hybrid signature scheme based on bilinear mapping. In terms of transmission efficiency, the calculation process of the bilinear map is transferred to the initialization phase of the system. The sensor side only needs low-cost operations such as hash mapping and exponential operation, which reduces the overall computing cost of the solution. In terms of security, the mathematical difficulty caused by the bilinear mapping calculation in the initialization phase is used to ensure the security of data transmission. In solving the problem of key management, the scheme uses the semi-trusted key generation center (KGC) and sensor ID to generate user session keys and data keys, which solves the public key authentication and key escrow problems of massive sensors in the Internet of Things.

1. Introduction

In order to ensure the secure transmission of sensor data in public channels, researchers apply identity authentication technology and key agreement technology to data transmission in the IoT(Internet of Things). In 1976, Diffie-Hellman first introduced the concept of key agreement. On the basis of their study, the researchers proposed different key negotiation models [1]. However, most of the existing schemes have two problems: in order to achieve key agreement, most schemes require complex bilinear mapping calculations on the sensor side, which are not suitable for sensor processors with limited computing power; most schemes will Key agreement is handled separately from data encryption. When a user has a large number of sensors, a large number of key agreement management and data key management are required, which greatly increases the user's computational burden and management difficulty, and wastes computing and communication resources [2].

In order to further enhance the security of data transmission or improve the efficiency of computing and communication in the IoT, Li first proposed a certificateless mixed signature and encryption mechanism [3] combining the advantages of certificateless cryptography, signature and mixed encryption cryptography. Sun et al. proposed a new certificateless mixed signature, which is efficient but cannot satisfy semi-public authentication [4]. Yu et al. proposed a variety of certificateless mixed signatures, but failed to take into account the algorithm efficiency and verified security [5,6]. To meet various security requirements, Luo proposed a new certificate-free hybrid signature scheme for session-specific temporary information security, but used too much bilinear, resulting in no significant improvement in computing efficiency [7].

2. Preliminaries

Please follow these instructions as carefully as possible so all articles within a conference have the same style to the title page. This paragraph follows a section title so it should not be indented.

2.1. Bilinear Mapping

Let G_1 and G_2 be a cyclic groups of order p, u and v is generator of G_1. a bilinear pairing is a map with following properties: Bilinear. $e(u^a, u^b)=e(u, v)^{ab}$, a, $b \in Z_p$ (Z_p is an integer group); Nondegeneracy. $e(u, v) \neq 1$; Computability. $e(u, v)$ always can be calculated effectively.

2.2. Computing Diffie-Hellman (CDH) problems

Define $a,b \in Z_p$ and g is generator of G, The adversary A knows (g, g^a, g^b).The CDH problem is adversary A hard to compute g^{ab}.

2.3. Computing Diffie-Hellman (CBDH) problems

Define $a, b, c \in Z_p$ and $T \in G_2$, The adversary A knows (g, g^a, g^b, g^c, T).The CBDH problem is adversary A hard to determine wherher the equation $T = e(g, g)^{abc}$.

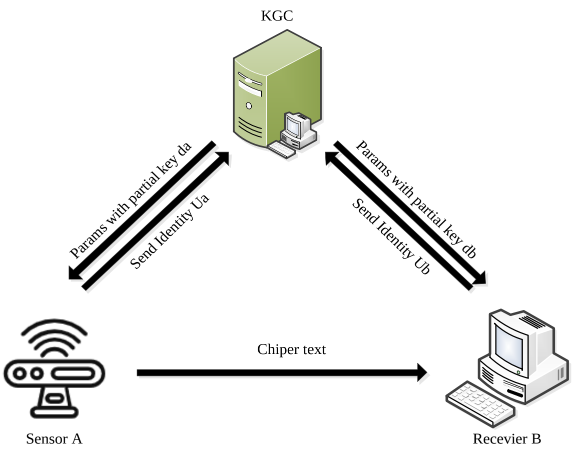

Figure 1. Bilinear based mixed signcryption model

Table 1. Symbols and description in scheme

Symbols	Description
params	System parameter
U_i	Identity of user *i*
X_i	User secret key of user *i*
d_i	Partial secret key of user *i*
P_i	Public key of user *i*
C_t	Ciphertext of signcryption algor-ithmm output
E,D	Encryption / decryption algorit-hm of symmetric encryption

3. Lightweight hybrid signature scheme for IoT

This part mainly introduces the algorithm and the main process of our scheme, The scheme model is shown in Figure 1, and the symbols in our scheme and their descriptions are presented in Table 1.

3.1. Setup

KGC performs system initialization by giving safety parameters. Select order p $(p > 2^Y)$ Additive cyclic group G_1, multiplicative cyclic group G_2 and integer group Z_p. The definition is a bilinear mapping $e: G_1 \times G_1 \to G_2$, which g is a random generator of G_1. Master key selection $x \in Z_p$. Define two hash functions $h_1: \{0,1\} \to Z_p$, h_2: $\{0,1\} \to \{0,1\}$. The the system parameters $params = \{G_1, G_2, Z_p, e, p, g, h_1, h_2, E, D\}$, Where E is the encryption algorithm of symmetric encryption algorithm and D is the decryption algorithm of symmetric encryption algorithm.

3.2. Generate-Partial-Private-Key

The user U_i send a registration application to KGC, KGC inputs the user identity U_i and output part of the private key $d_i = g^x \cdot h_1(U_i)$ of the user U_i. Return d_i to U_i by secure channel, U_i \`s identity add the system parameter ***params***.

3.3. Generate-User-Keys

User $\boldsymbol{U_i}$ choose private key $\boldsymbol{X_i \in Z_p}$, compute $\boldsymbol{U_i}$ `s public key $P_i = e(g,g)^{X_i}$ and join the system parameters.

3.4. Signcrypt

Base on the system parameters **{*params*, $\boldsymbol{U_a}$, $\boldsymbol{U_b}$, $\boldsymbol{P_a}$, $\boldsymbol{P_b}$}**, sender $\boldsymbol{U_a}$ signcrypt plaintext ***m*** to ciphertext ***Ct*** and send to receiver $\boldsymbol{U_b}$. ***Ct***`s generate process as follow:

- Compute session parameter $R = g^r$, $\boldsymbol{r \in Z_p}$.
- Embedding ***R*** and $\boldsymbol{U_a}$ `s private key $\boldsymbol{X_a}$ into $\boldsymbol{U_b}$ `s public key $\boldsymbol{P_b}$, compute $v = P_b^{(X_a \cdot h_1(R))}$.
- Embedding $\boldsymbol{U_b}$ `s identity and $\boldsymbol{U_a}$ `s partial private key $\boldsymbol{d_a}$, compute $t = h_1(Q_b \cdot d_a)$, where $Q_b = g^{h_1(U_b)}$.
- Compute symmetric encryption key $K = h_2(R \cdot t \cdot v)$, and encryption plaintext ***m*** to cihpertext $c = E(m,K)$.
- Generate the transport security parameter $f = h_1(U_a + U_b + R + P_a + P_b)$ and $n = \frac{r \cdot t \cdot f}{x_a}$.
- Sender $\boldsymbol{U_a}$ sends ciphertext set ***Ct* = (*c*, *R*, *n*)** to receiver $\boldsymbol{U_b}$.

3.5. Unsigncrypt

Base on the system parameters **{*params*, $\boldsymbol{U_a}$, $\boldsymbol{U_b}$, $\boldsymbol{P_a}$, $\boldsymbol{P_b}$}** and ciphertext set ***Ct* = (*c*, *R*, *n*)** from $\boldsymbol{U_a}$, receiver $\boldsymbol{U_b}$ unsigncrypt ***Ct*** process as follow:

- Compute $v = P_a^{(X_b . h_1(R))}$.
- Compute $t = h_1(Q_a \cdot d_b)$, where $Q_a = g^{h_1(U_a)}$.
- Compute symmetric encryption key $K = h_2(R \cdot t \cdot v)$, and try to decryption cihpertext ***c*** to plaintext $m = E(c,K)$.
- if decryption success, verify the ***Ct*** set is modified or not,compute $f = h_1(U_a + U_b + R + P_a + P_b)$ and checks $P_a^n = e(t \cdot g, f \cdot R)$ hold or not. If they hold, $\boldsymbol{U_b}$ get the message ***m***, otherwise give up.

4. Proof of security

4.1. Correctness analysis

We assume that each role executes the protocol base on the preset scheme in our scheme, and it can be verified as follow equation (1) whether the file received by the receiver $\boldsymbol{U_b}$ is the ciphertext transmitted by the sender $\boldsymbol{U_a}$ to $\boldsymbol{U_b}$.

$$\begin{aligned} P_a^n &= e(g,g)^{X_a \cdot n} \\ &= e(g,g)^{X_a \cdot \frac{r \cdot t \cdot f}{X_a}} \\ &= e(t \cdot g, f \cdot r \cdot g) \\ &= e(t \cdot g, f \cdot R) \end{aligned} \tag{1}$$

The parameter **t** is constructed by the receiver $\boldsymbol{U_b}$ through the identity of the sender $\boldsymbol{U_a}$ and the private key $\boldsymbol{X_b}$ of the receiver $\boldsymbol{U_b}$, and the parameter ***n*** transmitted by the sender $\boldsymbol{U_a}$ contains the private key X_a of the sender U_a. Therefore, it means that the ciphertext received by U_b has not been tampered with verifying whether equation (1) holds.

4.2. Security analysis

Theorem. In ***ROM*** (Random Oracle Model), the proposed scheme can satisfy ***IND-CCA2*** (Indistinguishability under Adaptive Chosen Ciphertext Attack) security.

Proof. Assuming that there is an adversary A that can defeat the proposed scheme with a non-negligible probability ε_1, this paper can construct an algorithm to solve the **CDH** in polynomial time. Challenger C gets a set of ground-truth values (g, g^a, g^b) and the goal is to find g^{ab}. Suppose that adversary A can achieve a non-negligible advantage in ***IND-CCA2*** with probability ε_1. In the next proof process, A will act as a subroutine of C, and C will act as a challenger to A.

In order to solve the ***CDH*** problem, the adversary A and the challenger C need to perform an initialization procedure, the process is as follows. Challenger C runs the initialization algorithm and generates system parameters ***params*** and system master key x, and maintains lookup tables L_1, L_2, and L_k. L_1 is the meta-combination set that h_1 participates in, L_2 is the meta-combination set that h_2 participates in, and L_k is the meta-combination set of public-private key pairs. Challenger C randomly selects a challenge identity U_j ($j \in [1, p]$).

Phase 1

Signature encrypted query. when C receives a signed encrypted query with sender Ua, receiver Ub and message m to be transmitted, C will check whether U_j equals ui in L_k. If U_a is not equal to U_j, C performs the normal signature encryption process, if Ua is equal to U_j, C obtains the tuple (X_b, d_b, P_a) and generates the ciphertext collection Ct according to the following process.

- Randomly select $r \in Z_p$, compute $R=rP$, compute $h_1(R)$, add or update list L_1 and compute $v = P_a^{(X_b \cdot h_1(R))}$.
- Obtain Q_a from the list L_1, calculate $t = h_1(Q_a \cdot d_b)$ and update the list L_1.
- Calculate $K = h_2(R \cdot t \cdot v)$ and update or add L_2.
- Calculate $c = E(m, K)$.
- Calculate $f = h_1(U_a + U_b + R + P_a + P_b)$ and update $(U_a, U_b, R, P_a, P_b, f)$ in L_2.
- Randomly select $n \in Z_p$ and return the ciphertext set $Ct = (c, R, n)$.

Signature decryption query. when C receives a signed decryption query with sender U_a, receiver U_b and ciphertext Ct, C will check whether there is U_j equal to U_a in Lk. If U_a is not equal to U_j, C performs normal unsigned encryption as our scheme. Otherwise, C decrypts the ciphertext set Ct according to the following procedure.

- Obtain the tuple (X_a, d_a, P_a, P_b) in L_k and Q_b of U_b in L_1, and calculate $t = h_1(Q_b \cdot d_a)$.
- Traverse the tuple (v, t, R, K) and check whether the equation $P_b^{X_a \cdot h_1(R)} = v$ holds. If it holds, C gets K in (v,t,R,K).
- Calculate $m = D(c, K)$.
- Obtain f by asking L_2, and check whether it holds. If the equation $P_a^n = e(t \cdot g, f \cdot R)$ holds, C returns message m to A. Otherwise, C returns null.

Phase 2

A performs a polynomially bounded adaptive query like **Phase 1**, but cannot query the receiver's d_i and X_i. Furthermore, it cannot perform a signature decryption query on the ciphertext Ct^* when U_a sends it to U_b.

Guess. At the end of the challenge, One of those stored in L_2 is the solution to the ***CDH*** problem. Therefore, C sequentially selects the tuple (v^*, t^*, R^*, K^*) from L_2, and outputs v^* as the solution of ***CDH***.

$$\begin{aligned} v^* &= P_{\mathrm{a}}^{X_B \cdot h_1(R)} \\ &= P_{\mathrm{b}}^{X_A \cdot h_1(R)} \\ &= e(g,g)^{X_A \cdot X_B \cdot h_1(R)} \end{aligned}$$

Analysis. Because adversary **A** cannot obtain part of the private key transmitted in the secure channel, adversary **A** cannot distinguish the query results in the polynomial time in the signature encrypted/ decryption query . Therefore, the adversary **A** advantage is a negligible probability under adaptive ciphertext attack. In the case that h_1 and h_2 are safe one-way hash functions, and ***CDH*** difficulty hypothesis holds, the hybrid signcryption scheme proposed in this paper is ***IND-CCA2*** safe.

5. Efficiency analysis

5.1. Theoretical analysis

The time complexity estimation of the scheme in this paper is mainly for the calculation overhead and communication cost. The efficiency comparison between the proposed scheme and the literature [10] is shown in Table 2, where **p** represents bilinear mapping, **m** represents multiplication, and **e** repress-ents exponential operation. As can be seen from Table 2, the scheme in this paper only needs to perfo-rm one bilinear mapping calculation for one communication, and only needs to perform exponential and multiplication operations on the sensor side, which can be better applied to environments with different computing power.

5.2. Experimental simulation

The simulation experiment based on JPBC (Java Pairing Based Cryptography library), the experimental environment is Windows 10 system, the hardware configuration is AMD Ryzen 5 2600X CPU, 16GB RAM. The simulation results of this scheme and the traditional attribute-based searchable encryption scheme [7] are shown in Figure 2 and Figure 3. Figure 2 shows the overall computational cost of this scheme versus the number of sensors. As can be seen from Figure 2, when using one sensor, the computation time of the reference scheme [7] already exceeds the scheme in this paper. Figure 3 shows the relationship between the computational cost on the sensor side and the number of sensors. Therefore, our scheme is suitable for multi-sensor.

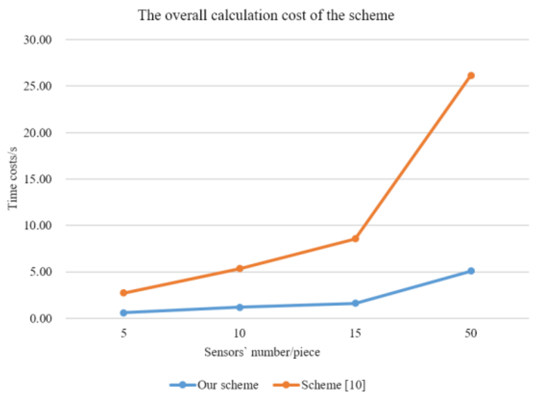

Figure 2. The overall calculation cost of the scheme

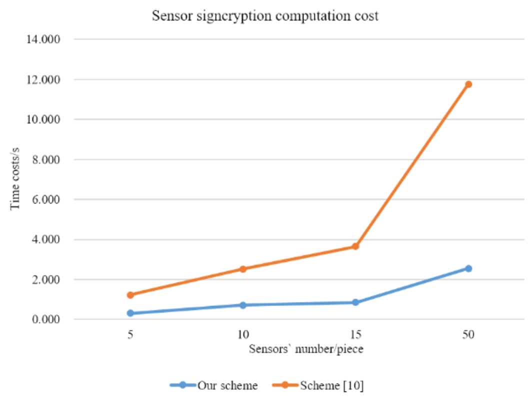

Figure 3. Sensor signcryption computation cost

Table 2. Scheme performance comparison

Scheme	Signcrypt encrypted			Signcrypt decryption		
	p	**m**	**e**	**p**	**m**	**e**
[10]	1	5	0	4	4	0
Our scheme	0	1	1	1	2	2

6. Conclusion

This paper proposes a lightweight IoT hybrid signcryption scheme based on a certificateless hybrid signcryption model. The certificateless signcryption is realized between the user and multiple sensors, which reduces the computational cost on the sensor side, and reduces the communication cost and cost; the pre-shared public key method is used, which makes the proposed scheme scalable and more in line with The practical application environment of the IoT. The comparison of simulation results shows that the proposed scheme in this paper is more efficient in terms of computational and communication costs compared with the scheme [7]. However, the scheme in this paper is currently only applicable to a one-to-many environment of users and sensors. In the future, the many-to-many negotiation between the client and the sensor can further reduce the computing and communication costs of the sensor.

Acknowledgments

The work was supported by the Innovation and Entrepreneurship Training Program for College Students(202110637008) and Chongqing Normal University Graduate Research Innovation Project (YKC21056).

References

[1] Diffie W, Hellman M. (1976) New directions in cryptography. *IEEE Transactions on Information Theory*, **22**(6): 644–654.
[2] Al-Turjman F, Ever Y K, Ever E. (2017) Seamless key agreement framework for mobile-sink in IoT based cloud-centric secured public safety sensor networks. *IEEE Access*, **5**: 24617–31.
[3] Sun Y X, Li H. (2011) Efficient certificateless hybrid signcryption. *J. Softw.*, **22**(7): 1690–8.
[4] Li F, Shirase M, Takagi T. 2009. Certificateless hybrid signcryption. In: International Conference on Information Security Practice and Experience, Springer. 112–123.
[5] Yu H, Yang H. (2015) Provably secure certificateless hybrid signcryption. *Chinese J. Comput.*, **38**(4): 804–813.
[6] Yu H, Yang B. (2017) Low-computation certificateless hybrid signcryption scheme. *Front. Inf. Technol. Electron. Eng.*, **18**(7): 928-40.
[7] Luo M, Wan Y, Huang D. (2017) Certificateless hybrid signcryption scheme with known session-specific temporary information security. *IJ Netw. Secur.*, **19**(6): 966–972.