IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 19, 2018, accepted December 26, 2018, date of publication January 1, 2019, date of current version January 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2890282

Secure RFID Authentication Schemes Based on Security Analysis and Improvements of the USI Protocol

LIJUN GAO[1], LU ZHANG[1], FENG LIN[1], AND MAODE MA[2], (Senior Member, IEEE)
[1]Department of Computer Science and Technology, Shenyang Aerospace University, Shenyang, China
[2]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798

Corresponding author: Lu Zhang (zhangluchn2016@163.com)

This work was supported by the Aerospace Science Foundation under Grant 20158054008 and Grant 20148001001.

ABSTRACT Wireless radio frequency identification (RFID) has been widely used as the core technology of the Internet of Things, but it also brings many potential security risks. In this paper, two typical RFID security authentication protocols are analyzed in depth. The results show that the two protocols have security risks, and two targeted attack methods are proposed. Next, two RFID security authentication protocols which can resist the DoS, replay, and tracing attacks are designed. The first protocol uses index grouping and dynamic renewal mechanism. Since the Rabin public key encryption algorithm verification only requires square and modulo operations, the second protocol introduces the public key encryption algorithm into the cost-sensitive RFID tag, and formally verifies the functionality of the proposed scheme by SPIN. Furthermore, the security of the protocol is proven by the BAN logic. The grouping index method can effectively improve the ability to locate tags in a large database and enhance the practicability of the protocol. The introduction of the Rabin public key algorithm improves the security of the protocol under the condition of cost control.

INDEX TERMS RFID, index grouping, dynamic renewal, Rabin public key encryption algorithm.

I. INTRODUCTION

As ''all-in-the-net'' age is coming, wireless radio frequency identification (RFID) has been utilized widely as the key technology of Internet of Things. The classical scenario applied of the RFID is that a central database stores the relevant information of tag, a reader reads and identifies tags without contact. The tag is the information carrier of identification object. In this scenario, the database server and the reader can adopt the most advanced encryption algorithm because of powerful computing capability, which is regarded as security channel. The channel between the tag and the reader is regarded as insecure channel due to the tag is sensitive to the cost of the high-security encryption algorithms, it is impossible to use the high-security encryption algorithms directly, such as RSA and AES, etc. Therefore, it often suffers the DoS, tracking, and replay attacks, etc. In recent years, with the wide spread of the RFID, the attacks aiming at the RFID system is diverse, and seriously hindered the promotion of the RFID technology. Then how to protect the privacy problems of the RFID is getting hotter.

The research shows that the RFID has two utility scenarios, Scenario I: the reader is related to the database directly. The study on the security protocol is divided into two types: (1) one type is the low-cost security privacy protection protocol with low-cost operations such as AND, OR, XOR, and CRC, the disadvantage is poor security. (2) The other type is the high-cost RFID protocol which introduces the high-cost public key and symmetric encryption algorithm to guarantee the security of the protocol. The cost is seriously exceeded, which limits the promotion of such protocols. Scenario II: the readers and the database are separate, that is off-line work mode, which can extend the work scope of the identification system effectively. In the off-line situation, the reader needs to store a large amount of raw data of tags. The off-line mode has poorer security due to lack of real-time interaction with the central database, and the reader is vulnerable to information loss, tracing and DoS attacks. In this paper, we have a deep study on two representative literatures [1], [2]. Analysis shows that there are security risks in them. The reason is as follows: (1) Scenario defects. Although the off-line solution increases the mobility of the RFID application range, it limits the real-time information renewal between the reader and the tag. (2) Protocol design defect. The protocol does not take into account the freshness of the variables, so that there will be

the replay, DoS and tracing attacks. Based on the above two reasons, two aiming attack against the above protocols were proposed. And then we design two security authentication protocols which can resist the DoS, replay and tracing attacks of the RFID system. The scenarios of the paper: with the rapid development of the 4G and 5G network, the QoS of remote wireless network has a great development. And it can be applied in the remote secure communication between the reader and the database. The wireless communication technology extends the mobile range of the RFID tag, and allows the real-time data exchange between the tag and the database. In this scenario, we proposed two RFID security authentication protocols: the first protocol used grouping index and dynamic renewal mechanism; Since the Rabin public key encryption algorithm signature only requires square and modulo operations, the second protocol introduces the public key encryption algorithm into the cost- sensitive RFID tag, and formally verify the functionality of the proposed scheme by SPIN (Simple Promela Interpreter). The grouping index method can effectively improve the ability to locate tags in the large data base and enhance the practicability of the protocol. The dynamic update mechanism and the one-way hash function are introduced to reduce the replay, tracing and DoS attacks. In addition, the protocol introduces the Rabin public key encryption algorithm to improve the security with controllable costs. Furthermore, the security of the protocol was proven by BAN logic.

II. RELATED WORK

A lightweight RFID authentication protocol (abbreviation as protocol HB) based on LPN (learning parity with noise) [3] has been introduced in [4], which achieves good results in terms of security and cost control. Juels and Weis modified the HB protocol with aim to get the HB+ protocol [5], but the HB+ protocol is insecure to the man-in-the-middle attack with disguising legal reader. Then Bringer proposed HB++ protocol with aim to the defect of the HB+ [6]. Three ultra-lightweight authentication protocols: M^2AP, EMAP, LMAP, which are regarded as ultra-lightweight two-way authentication protocols cluster has been presented in [7]–[9], which is a good exploration in terms of phases dealing and cost control. An ultra lightweight authentication protocol (abbreviation in SASI) in [10] only uses AND, Shift and XOR operations to simplify the cost to meet the design demand of the low-cost RFID system. However, the forward security feature of the protocol is poor. Two methods of DoS attacks for the SASI protocol have been proposed in [11]. Inspired by the SASI protocol, the Gossamer protocol has been proposed in [12], which introduced the MixBits() function. However, the design of the protocol is flawed and there is no rigorous theoretical proof. In 2010, Eslamamal *et al.* [13] gave a passive attack method for the Gossamer protocol. The UAPP protocol has been introduced in [14], which has achieved excellent results in cost control and security performance. However, the recently research results show that the UAPP protocol is easy to be attacked by DoS. A Dos attack can successfully implement within a certain probability range for the UAPP protocol with a bit-tampering asynchronous attack method has been reported in [15]. An ultralightweight mutual-authentication protocol UMAPSS based on secret sharing has been introduced in [16], which includes mechanisms for double verification, session control, mutual authentication, and dynamic update to enhance security and provide a robust privacy protection. In 2018, Rad *et al.* [17] gave an attack method for the distance-bounding protocols MP and KA, and proved that these protocols are vulnerable against terrorist fraud attack and force attack. Finally, they improved these two schemes. A new radio-frequency identification authentication protocol based on elliptic curve cryptography to eliminate these vulnerabilities has been introduced in [18]. The protocol achieves a set of security properties as mutual authentication, anonymity, confidentiality, forward security, location privacy, resistance of man-in-the-middle attack, resistance of replay attack and resistance of impersonation attack. An RFID authentication architecture for distributed IoT applications and the future smart city environments has been presented in [19], which provides forward secrecy, anonymity and untraceability of RFID-tag, and secure localization.

III. REVIEW OF USI SCHEME

A. REVIEW OF THE USI SCHEME

The USI scheme, an off-line RFID security protocol, has been proposed in [2], which enhanced the mobility of readers. Study shows that the protocol is vulnerable to tracing attacks. The USI scheme includes two sections: (1) authentication between the reader and the tag; (2) authentication between the reader and the database. We lay more emphasis on the authentication stage between the reader and the tag.

$$R_{is} \rightarrow T_j : n_i, \quad r_i$$
$$T_j \rightarrow R_i : n_j, h(f(r_{cd}, t_j))_m, h(f(r_i, t_j)||n_i||n_j) \oplus id_i$$

Check through L_i, get all entries with same $h(f(r_{cd}, t_j))_m$. Calculate $h(f(r_i, t_j)||n_i||n_j) \oplus id_i$ of all matching entries. Get the correct one and take its id as id_i.

The storage content of tag according to the scheme design is:

$$r_{cd} = 1000, \quad T_1 =< id_1 = 0001, \quad t_1 = 0010,$$
$$f(r_{cd}, t_1) = (1000, \quad 0001)_m = 0101010101 >,$$
$$T_2 = < id_2 = 0011, \quad t_2 = 0100,$$
$$f(r_{cd}, t_2) = (1000, 0100)_m = 1010100101 > .$$

The information stored in R_1 and R_2 is as follows:

$$L_1 = \{h(f(r_{cd}, \quad t_1))_m = 0101010101,$$
$$f(r_1, t_1) = 0010001010, id_1 = 0001;$$
$$h(f(r_{cd}, t_2))_m = 0101010101,$$
$$f(r_1, t_2) = 0111001110, \quad id_2 = 0011; \}$$
$$L_2 = \{h(f(r_{cd}, t_1))_m = 0101010101,$$
$$f(r_2, t_1) = 1100011010, id_1 = 0001;$$

$$h(f(r_{cd}, t_2))_m = 0101010101,$$
$$f(r_2, t_2) = 1101001010, id_2 = 0011; \}$$

The reader sends query signal which includes $< r_i, n_1 >=< 1101, 1000>$. The T_1 sends back $n_j = 2001$, $h(f(r_{cd}, t_1))_m = 0101010101$, and further computes $h(f(r_i, t_j)||n_i||n_j) \oplus id_i = h(f(1101, 0001)||1000||2001) \oplus 0001 = 0010103121$ and sends to the reader. Analysis shows that although the introduction of r_{cd} can reduce the probability of tracking attacks, it can still not completely overcome the tracking attack. Therefore, the scheme is not as secure as that the author claimed.

B. TRACING ATTACK METHOD AIMING AT THE SERVER INDEPENDENT AUTHENTICATION SCHEME

1) ATTACK METHOD I

With the development of electronic analysis technology, attackers can know the $f()$ and $h()$ functions by analyzing chip circuit. The method is based on the assumption that the attacker knows the $f()$ and $h()$ functions. In the second step, the tag returns $h(f(r_i, t_j)||n_i||n_j) \oplus id_i$ as the identification between the tag and the reader. The attack scenario is like this: in an important meeting, all participants will be identified by the RFID tag. There are two people attending the meeting and each person wears an RFID tag, in addition, there are two readers at the meeting place. We can quickly identify the owner of the tag and track them by the following method.

The storage of tags:

$$T_1 = < id_1 = 0001, \quad t_1 = 0010 >,$$
$$T_2 = < id_2 = 0011, \quad t_2 = 0100 >$$

The storage of readers:

$R_1 =< r_1 = 0101, L_1 >, R_2 =< r_2 = 0110, L_2 >$, where r_1, r_2 are identification tokens of readers, $r_{cd} = 1000$.

When R_1 sends $< r_1, n_{i1} >=< 0101, 1010>$, T_1 replies $n_{j1} = 1001$, $h(f(r_{cd}, t_1))_m = h(f(1000, 0010))_m = 0101010101$, and

$$\begin{aligned} V_1 &= h(f(r_i, t_j)||n_i||n_j) \oplus id_i \\ &= h(f(0101, 0010)||1010||1001) \oplus 0001 \\ &= h(1010100011||1010||1001) \oplus 0001 \end{aligned}$$

When R_1 sends $< r_1, n_{i2} >=< 0101, 2010>$, T_1 replies, $n_{j2} = 2001$, $h(f(r_{cd}, t_1))_m = h(f(1000, 0010))_m = 0101010101$, and

$$\begin{aligned} V_2 &= h(f(r_i, t_j)||n_i||n_j) \oplus id_i \\ &= h(f(0101, 0010)||2010||2001) \oplus 0001 \\ &= h(1010100011||2010||2001) \oplus 0001 \\ V_1 \oplus V_2 &= [h(f(r_i, t_j)||n_{i1}||n_{j1}) \oplus id_i] \\ &\quad \oplus [h(f(r_i, t_j)||n_{i2}||n_{j2}) \oplus id_i] \\ &= h(f(r_i, t_j)||n_{i1}||n_{j1}) \oplus h(f(r_i, t_j)||n_{i2}||n_{j2}) \end{aligned}$$

Suppose $x = h(f(r_i, t_j)$

$$\begin{aligned} V_1 \oplus V_2 &= h(f(r_i, t_j)||n_{i1}||n_{j1}) \oplus h(f(r_i, t_j)||n_{i2}||n_{j2}) \\ &= h(x||n_{i1}||n_{j1}) \oplus h(x||n_{i2}||n_{j2}) \end{aligned}$$

Because $h, f, n_{i1}, n_{i2}, n_{j1}, n_{j2}, V_1, V_2$ are known, we can crack the value of x by computing off-line (the time may not be too long.). The $f(r, t) = h(r||t)$, $h(r||t)$ is one-way hash function, length of which is fixed as l, and we can get $x = f(r_i, t_j) = 1010100011$ in the computation complexity of 2^l.

After getting the value of the $f(r_i, t_j)$ of the T_j, we can distinguish the tag quickly because the value of the $f(r_i, t_j)$ keeps the same.

If the R_1 sends two queries $< r_1, n_{i3} >=< 0101, 3010>$, $< r_1, n_{i4} >=< 0101, 4010>$, the T_x replies:

$$\begin{aligned} n_{j3} &= 3001 \\ h(f(r_{cd}, t_1))_m &= h(f(1000, 0010))_m = 0101010101 \\ V_3 &= h(f(r_i, t_j)||n_i||n_j) \oplus id_i \\ &= h(f(r_i, t_j)||3010||3001) \oplus 0001 \\ n_{j4} &= 4001 \\ h(f(r_{cd}, t_1))_m &= h(f(1000, 0010))_m = 0101010101 \\ V_4 &= h(f(r_i, t_j)||n_i||n_j) \oplus id_i \\ &= h(f(r_i, t_j)||4010||4001) \oplus 0001 \\ V_3 \oplus V_4 &= h(f(r_i, t_j)||n_{i3}||n_{j3}) \oplus h(f(r_i, t_j)||n_{i4}||n_{j4}) \\ &= h(x||n_{i3}||n_{j3}) \oplus h(x||n_{i4}||n_{j4}) \\ &= h(f(r_i, t_j)||3010||3001) \\ &\quad \oplus h(f(r_i, t_j)||4010||4001) \end{aligned}$$

If the amount of readers and tags is more than 2, we can establish a rainbow table K of T_j and $f(r_i, t_j)$, in which we can identify the value of tag token identification tuple off-line. Thus we can replace x with characteristic value of a special tuple, and complete the tracing identification instantly.

$$K_i = \begin{cases} f\left(r_i, t_j\right) : T_j \\ f\left(r_i, t_n\right) : T_n \end{cases}$$

2) ATTACK METHOD II

The $h(f(r_{cd}, t_j))_m$ is still can be seen as the only token of a tag because it contains the secret key of the tag. The $h(f(r_{cd}, t_j))_m$ is the first m bits of the access control list and has multiple matching results. Because the possibility of $h(f(r_{cd}, t_j))$ is 2^l, that of $h(f(r_{cd}, t_j))_m$ is 2^m, then the minimum conflict area is 2^{l-m}. Suppose B is the conflict probability, then $B > 2^{l-m}$. A malicious R sends two random numbers n_1, n_2 to T_1. If B is very small, the $h(f(r_{cd}, t_j))_m$ can locate a tag directly. If B is big, the attacker can send multiple queries to locate the tag. If a malicious R sends three query signals, the tracing probability will be decreased to B^2, four to B^3, and so on, until B^n is decreased to small enough to distinguish the specific tag, so tracing attack is possible.

IV. OUR PROTOCOL

The nature defects of Literature [2] are as follows: (1) the scheme lacks of dynamic renewal mechanism of the secret key; (2) the scheme is vulnerable to various attacks due to the public encryptions algorithm. However, the literature [1], [2]

TABLE 1. Notations.

notation	Meaning
$\oplus$	XOR
$hash$	One-way hash function
key_x^{old}	key_x^{old} *in database*, including key_h^{old}, key_m^{old}, key_l^{old}, used to store the last time successfully authenticated secret key.
key_x^{new}	key_x^{new} in database, including key_h^{new}, key_m^{new}, key_l^{new}, used to store this time successfully authenticated secret key.
g_i^{old}	used to store the last time successfully authenticated index grouping in database
g_i^{new}	used to store this time successfully authenticated index grouping
Key_t	Tag Key_t secret group, contains Key_h, Key_m, Key_l, used to store secret key of tag
g_t	Index grouping stored in tag
n_i	random number
mod	Modulo operation

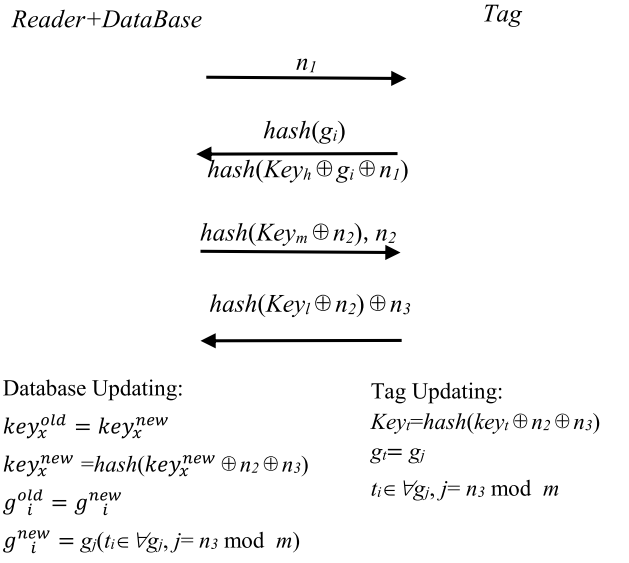

FIGURE 1. Scheme I.

gives a good inspiration in improving the search speed of the database. With the enlightenment of these two literature, two RFID security authentication schemes are presented in this paper, which introduce the dynamic renewal mechanism, index grouping and Rabin public key encryption algorithm to ensure the practicality and anti-attack ability.

A. SCHEME I -INTRODUCES HASH FUNCTION, DYNAMIC MECHANISM, INDEX GROUPING INTO RFID SECURITY AUTHENTICATION PROTOCOL

The practical scenario of the scheme is: the reader and the database use wireless remote communication technology, such as the 4G, and cooperates with the high cost encryption algorithms. Every tag has three secret keys, one index grouping, one hash function and one random number generator. The database also contains three current update keys and one group index number. Besides, in order to prevent the DoS attacks, the database contains three last successful authentication key groups and one last successful authentication index grouping number. If the RFID system is subjected to a DoS attack, the authentication can be completed with the last successful authentication key. Table 1 shows all notations used in the protocol.

Suppose tags set of the database is $T = \{t_{11}, t_{12}, t_{13}, \ldots, t_{1n}, t_{21} \ldots t_{2n}, \ldots, t_{m1}, \ldots, t_{mn}\}$, g_1 is the index of $\{t_{11}, t_{12}, t_{13}, \ldots, t_{1n}\}$, g_2 is the index of $\{t_{21}, t_{22}, t_{23}, \ldots, t_{2n}\}$, g_m is the index of $\{t_{m1}, t_{m2}, t_{m3}, \ldots, t_{mn}\}$. The total amount of tags in the database is $m \times n$, the value of m, n can be adjusted. When the m becomes smaller, the n will become larger. If the number of tags in each group increases, the positioning speed of the first round will increase, but the positioning speed of the second round will become slow. Therefore, if the values of m and n are properly adjusted, the positioning speed will be optimal, which is significantly better than the exhaustive search and enhances the practicability of the protocol.

B. AUTHENTICATION PROCESS OF SCHEME I

The scheme divides into 3 parts:

1) The first stage: grouping location stage

$R \rightarrow T$: R sends query $< n_1 >$

$T \rightarrow R$: T receives query, then reply, $hash(g_i)$, $hash(Key_H \oplus g_i) \oplus n_1$

$R \rightarrow T$: The $hash(g_i) \in \{hash(g_1), hash(g_2), \ldots, hash(g_i)\}$, if we can find the group index equal to the $hash(g_i)$ in the database, we can initially determine the response T in this group.

2) The second stage: authentication stage

$R \rightarrow T$: Furthermore, the second comparison will be performed in $\{t_{i1}, t_{i2}, t_{i3}, \ldots, t_{in}\}$. If the $hash(Key_h \oplus g_i \oplus n_1)$ is equal to the response value, the authentication is successful, otherwise the authentication fails. We perform a second query in the conflict tag set to solve this problem, but it is a detailed issue and will not be discussed here. If the authentication from the T to the R passes, the R generates random number n_2, computes $hash(Key_m \oplus n_2)$, n_2 and sends to the tag.

$T \rightarrow R$: If the value of the hash$(Key_m \oplus n_1)$ transmitted by the reader is equal to the value calculated by the tag, the authentication from the T to the R succeeds. And then the tag generates n_3, computes $hash(Key_l \oplus n_2) \oplus n_3$, and sends them to the reader. The reader extracts n_3 to update the secret key, calculates the value of $hash(Key_l \oplus n_2) \oplus n_3$ with the information stored in the database. If it is equal to the value passed by the tag, the authentication from the T to the R succeeds.

3) The third stage: index grouping and secret key renewal stage

After the authentication from the R to the T succeeds, the value of t_i is updated to $key_x = hash(key_x \oplus n_2 \oplus n_3)$, $t_i \in \{g_i\}$, $t_i \in \forall g_j, j = n_3 \bmod m$. Similarly, after the authentication from the T to the R succeeds, the value of index grouping is updated to $key_x^{old} = key_x^{new}$, $key_x^{new} = hash(key_x^{new} \oplus n_2 \oplus n_3)$ and $g_i^{old} = g_i^{new}$, $g_i^{new} = g_j$ ($t_i \in \forall g_j, j = n_3 \bmod m$). The actual authenticating process is shown in the Fig. 1.

C. DoS ATTACK ANALYSIS

The DoS attacks divide into interception DoS attack, three times cheat DoS attack and bit-tampering DoS attack. The intercepting DoS attack is that the attacker forcibly blocks the communication process between the tag and the reader to make the authentication two sides cannot keep the key update make the reader update the secret key, but the tag does not update the secret key. Although the reader uses a dual secret key backup mechanism, the attacker can skillfully destroy the consistency of the secret keys of the communication parties by using the three times cheat asynchronous attack to achieve the purpose of hindering subsequent authentication. The more detailed attack steps are described in [10]. Bit-tampering DoS attack, which uses the coupling between transfer variables to change one or more bits of a particular variable to infer the affected bit in another variable. If we can't determine the location affected, we can use the guessing technique to make the illegal information pass the authentication. After the authentication, the tag and the reader will update the secret key with the wrong information, so that the next authentication will be performed abnormally. The more detailed attack steps are described in [15].

Pointing at the intercepting DoS attack, the scheme applies double secret key and double index copy mechanism in the database. If communication is intercepted, the database renewal secret key will be $key_x^{old} = key_x^{new}$, $key_x^{new} = hash(key_x^{new} \oplus n_2 \oplus n_3)$ and $g_i^{old} = g_i^{new}$, $g_i^{new} = g_j$ ($t_i \in \forall g_j, j = n_3 \bmod m$), while tag secret key still is $key_x = hash(key_x \oplus n_2 \oplus n_3)$, g_i, it can use the last time $\{key_x^{old}, g_i^{old}\} \in R$, $\{key_x, g_i\} \in T$ to complete the authentication successfully. Thus, the scheme can block the DoS attacks. Pointing at the three times cheat asynchronous attack, the analysis shows that the protocol being attacked generally has retransmission defects. The reason for the retransmission defect is that the freshness of the variable is poor in the communication process, so the attacker can easily intercept and replay the communication information. In the proposed protocol each communication step of the protocol introduces a random number, and encrypts with the hash one-way function to ensure the freshness of the information. The secret key and the group index are also updated after each successful authentication. So the scheme can prevent the three times cheat asynchronous attack efficiently. Aiming at bit-tampering asynchronous attack, analysis shows that the reason for this type attack is that the coupling between communication variables is too strong. For example, we transmit $A = n_2$ and $B = ROR(key_m \oplus n_2)$ to complete the authentication from the reader to the tag, during the communication process, in which the ROR is a recycling shift bit function. If we flip a bit in n_2, a bit of B will be changed correspondingly. Although we don't know which one is changed, if B is k-bit, we can use guessing technology to pass the authentication within $O(2^k)$ time complexity. After that, the tag will update the secret key with the wrong n_2, which will cause the communication authentication failure in the next time. The protocol proposed in this paper introduces a one-way hash function and uses the random number to maintain freshness, which can effectively guarantee the integrity of information. Because the characteristic of the hash function with small change in the input parameters leading to a huge change in the output variables, which can greatly reduce the coupling between the variables, so that the attacker cannot implement the bit-tampering asynchronous attack.

D. TRACING ATTACK ANALYSIS

In order to analyze tracing attack carefully, the P is used to indicate the protocol to be executed, superscript run is used to indicate the running process of the protocol, and $\#\theta$ is used to indicate the unique identifier of each query information, $time_{run\#\theta}$ is used to represent the value of the variable θ during the specific execution, $time_i^{run}$ is the ith tracking, and $Link(time_i^{run}, time_j^{run})$ is the relationship between two authentications for the same tag. If the same tag $Link(time_i^{run}, time_j^{run}) \neq \phi$ in different running process, the tag is traceable. $\forall t \in Tracing(P)$, $\forall_{i \neq j}\ Link(time_i^{run}, time_j^{run}) \neq \phi \Rightarrow \exists t' \in Tracing(P)$. If the same tag exists in two different runs, $Link(time_i^{run}, time_j^{run}) = \phi$, there is no tracking attack. $\forall t \in Tracing(P)$, $\forall_{i \neq j} Link(time_i^{run}, time_j^{run}) = \phi \Rightarrow \neg(\exists t' \in Tracing(P))$. In this scheme, the important exchange information is $hash(g_i)$, $hash(Key \oplus g_i) \oplus n_1$ from the tag to the reader. Since the hash function is unidirectional and the protocol uses dynamic key update technology, g_i and Key_x will be renewed after every successful authentication. Besides random numbers have been inserted into communication process, it also improves anti-attack ability. The anti-tracking capability analysis process is as follows. The attacker sends the query information twice, the first time is $\#\theta 1$, and the second time is $\#\theta 2$.

The first exchange information of query includes:

$$\{(hash(g_i\#\theta_1)), (hash(Key_h\#\theta_1 \oplus g_i\#\theta_1) \oplus n_1\#\theta_1)\},$$
$$\{hash(Key_m\#\theta_1 \oplus n_2\#\theta_1), n_2\#\theta_1\},$$
$$\{hash(Key_l\#\theta_1 \oplus n_2\#\theta_1) \oplus n_3\#\theta_1\}$$

The second exchange information of query includes:

$$\{(hash(g_i\#\theta_2)), (hash(Key_h\#\theta_2 \oplus g_i\#\theta_2) \oplus n_1\#\theta_2)\},$$
$$\{hash(Key_m\#\theta_2 \oplus n_2\#\theta_2), n_2\#\theta_2\},$$
$$\{hash(Key_l\#\theta_2 \oplus n_2\#\theta_2) \oplus n_3\#\theta_2\}$$

The variables marked with $\#\theta$ are updated after each authentication, therefore we can judge:

$$Link(hash(g_i\#\theta_1), hash(g_i\#\theta_2)\} = \phi$$
$$Link((hash(Key_h\#\theta_1 \oplus g_i\#\theta_1) \oplus n_1\#\theta_1),$$

$hash(Key_h\#\theta_2 \oplus g_i\#\theta_2) \oplus n_1\#\theta_2)\} = \phi$
$Link(hash(Key_m\#\theta_1 \oplus n_2\#\theta_1), hash(Key_m\#\theta_2 \oplus n_2\#\theta_2)\} = \phi$
$Link(hash(Key_l\#\theta_1 \oplus n_2\#\theta_1) \oplus n_3\#\theta_1,$
$hash(Key_l\#\theta_2 \oplus n_2\#\theta_2) \oplus n_3\#\theta 2)\} = \phi$

So $\forall_{i\neq j}\ Link(time_i^{run}, time_j^{run}) = \phi \Rightarrow \neg\ (\exists t' \in Tracing(P))$, the scheme can resist tracing attack effectively.

E. SHEME II–RFID MUTUAL AUTHENTICATION SECURITY PROTOCOL BASED ON RABIN PUBLIC KEY CRYPTOGRAPHY

1) RABIN ALGORITHM SPECIFICATION

The Rabin is a symmetric encryption algorithm, which is as secure as RSA. The theoretical basis of both is the difficulty of decomposition of a large number. However, the Rabin encryption algorithm has been shown to be able to reduce to large number decomposition, but whether RSA can be reduced to large number decomposition has not been proven. The principle of the Rabin encryption algorithm is as follows [20]:

Theorem I: $n = p.q$, p and q are different odd prime numbers, if $(x, n) = l$, then $x^{(p-1)(q-2)/2} \equiv 1 \bmod n$.

Suppose n is a multiplication group of integer $Z_n^* = \{k \in Z_n | (k, n)\}$ modulus n. Suppose $a \in Z_n^*$, if $x \in Z_n^*$ makes $x^2 \equiv a \bmod n$, then call $a$is the quadratic remainder of modulus n, otherwise non-quadratic remainder. We use Q_n as quadratic remainder set of modulus n, and $\overline{Q}_n$ is the quadratic non-remainder set modulus n.

Thermo II: Suppose $x \in Q_n$, $n = p.q$ is a Blum integer, then $x^{(n-p-q+5)/8} \bmod n$ is the square root of $x \bmod n$.

Thermo III: $J_n = \{a \in Z_n^* | (\frac{n}{a}) = 1\}$, $\tilde{Q}_n = J_n$ -Q_n(denotes modulus n is square roots set), $x \in J_n$, $n = p.q$, is a Blum integer, then $x^{2d} \equiv \begin{cases} x & \text{if } x \in Q_n \\ n - x & \text{if } x \in \tilde{Q}_n \end{cases}$, where $d = (n - p - q + 5)/8$.

According to the China Remainder Theorem, every formula gets four solutions. But the solution from the encrypted text is not the sole. If we want to get a unique solution, we can get a unique solution by adding additional information after the plaintext, such as the identification of sender, identification of receiver and time stamp. Furthermore, we can find out the sole solution from many solutions according to appendix information.

Solution process is as follows:

We choose two large prime numbers p and q randomly, satisfying $p \equiv q \equiv 3 \bmod 4$, that is, the two prime can be expressed as $4k + 3$, compute $n = p \times q$. Setting n *as* public key K_{pub}, p, q as the secret keys K_{Pri}. When encrypting, $f \equiv x^2 \bmod n$, where m is the plain text group, and c is the cipher text group corresponding to m. When decrypting, the equation $x^2 \equiv f \bmod n$ is solved, which is equal to solve the following equation set.

$$\begin{cases} x^2 \equiv f \ mod\ q \\ x^2 \equiv f \ mod\ p \end{cases}$$

If both p and q choose prime number of 3 (mod 4), then the square roots of $f \bmod n$ are as follows:

Using the extended Euclidean algorithm, it is found that the integers a, and b satisfy $a \cdot p + b \cdot q = 1$. The values of a and b can be calculated during the key generation phase and used repeatedly in the in the phase of the key generation.

- Compute $r \equiv f^{(p+1)/4} (\bmod p)$
- Compute $s \equiv f^{(q+1)/4} (\bmod q)$
- Compute $x \equiv (a \cdot p \cdot s + b \cdot q \cdot r) (\bmod n)$
- Compute $y \equiv (a \cdot p \cdot s - b \cdot q \cdot r) (\bmod n)$
- Four square roots of $f \bmod n$ are $\pm x (\bmod n), \pm y (\bmod n)$

Research shows that the Rabin public key encryption algorithm has a very significant feature, the computational complexity of the encryption and decryption process is asymmetric. The encryption process only requires square and modulo operations, while the decryption process requires complex exponential calculations. The asymmetry of encryption and decryption process is very similar to the structure of the RFID system. In the RFID system, the reader's computational capability is strong enough to undertake large computational tasks, while the RFID tag's computational capability is weak, so it can only perform limited calculations.

Based on this, we propose a two-way RFID security authentication protocol that introduces the public key Rabin encryption algorithm. The encryption part of the protocol, which is completed by the RFID tag, requires only one square and modulo operations. The decryption part is done by the reader, which effectively solves the problem of computing capability weak of the RFID tag. And it raises the grade of RFID security authentication to the public key level.

2) PROTOCOL DESCRIPTION

The public key of the tag is K_{Pub}, the private key is K_{Prip}. The public key of the server is K_{Ser_Pub}. The specific description of the protocol is as follows:

- $R_i \rightarrow T_j$: *request*
- $T_i \rightarrow R_j$: The tag receives query, and generates a random number n_1; computes $m_1 = n_1 \oplus Key_h$, and encrypts n_1 and $(IDS \oplus n_1)$ with the server's public key K_{Ser_Pub}, the ciphertext is $A = n_1^2 \bmod K_{Ser\_Pub}$, $B = (IDS \oplus n_1)^2 \bmod K_{Ser\_Pub}$; encrypts the m_1 with the tag's public key K_{Pub}, the ciphertext is $C = m_1^2 \bmod K_{Pub}$; sends A, B and C to the reader.
- The reader receives A, B, C and sends them to the server. The server decrypts A and B with its own private key K_{Ser_Pub} to obtain the cipher text n_1 and $(IDS \oplus n_1)$; computes $IDS = n_1 \oplus (IDS \oplus n_1)$ to obtain the IDS and retrieves the database to obtain the tag's private key and shared secret keys; decrypts C with tag's private key to obtain the cipher text m_1; compares m_1 and $(n_1 \oplus Key_m)$, if $m_1 = n_1 \oplus Key_h$, then the authentication is successful, otherwise failed; computes $D = \{(n_1 \oplus Key_m)^2 mod\ K_{Pub}\} \oplus Key_l$, and then the D is sent to the tag. (Illustration: When decrypting, there will be four solutions, and the unique solution can

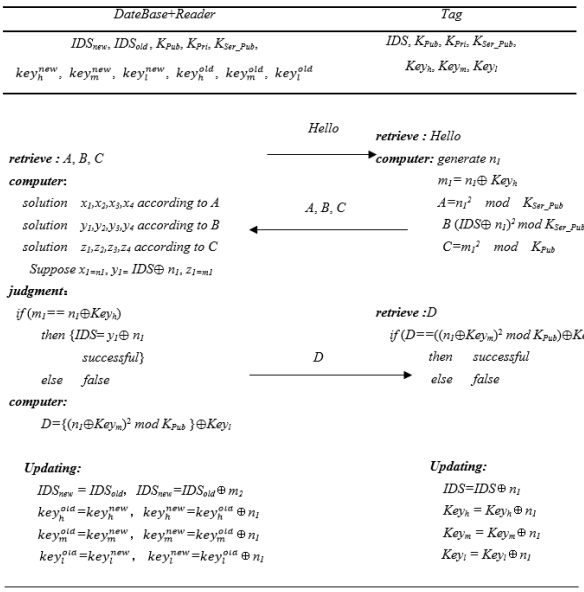

FIGURE 2. RFID mutual authentication protocol based on the Rabin cryptography algorithm.

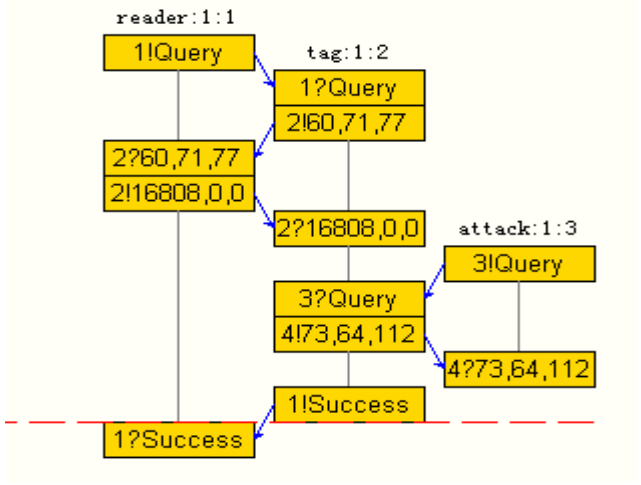

FIGURE 3. SPIN simulation model of the sheme II.

be obtained by appending information after the cipher text. Suppose the solutions of A, B and C are x_1, y_1*and* z_1 respectively.)

- The tag receives D, and computes $\{(n_1 \oplus Key_m)^2 mod K_{Pub}\} \oplus Key_l$ with its own variables n_1, Key_m, K_{Pub}, Key_l. If $D = \{(n_1 \oplus Key_m)^2 \ mod \ K_{Pub}\} \oplus Key_l$, then the authentication is successful, otherwise failed.
- The secret renewal process is the same as the first protocol. The Protocol 2 can effectively prevent the DoS, replay and tracking attacks with the dynamic update mechanism. The detailed analysis process will be given later. In addition, if we want to improve the search speed of the *IDS*, the group indexing mechanism can be introduced. The authentication process of the protocol is shown in the fig. 2.

3) SIMULATION EXPERIMENT

The protocol SPIN simulation model is shown in Fig. 3. In the simulation model, we set IDS=1323, $K_{ser\_pri} = \{p = 7, q = 11\}$ (private key of the database), $K_{ser\_pub} = 77$ (public key of the database), $key_h = 12321$, $key_m = 13139$, $key_l = 16651$, $a = 8$, $b = -5$, $K_{pri} = \{p_1 = 19, q_1 = 23\}$ (private key of the tag), $K_{pub} = 437$ (public key of the tag), $n_1 = 37$ ($A = n_1^2 \bmod K_{ser\_pub}$). According to the Rabin algorithm we can deduce $a \cdot p + b \cdot q = 1$, $r \equiv f^{(p+1)/4} \bmod p$, $s \equiv f^{(q+1)/4} \bmod q$, $x \equiv (a \cdot p \cdot s + b \cdot q \cdot r) \ (\bmod n)$, y$\equiv(a \cdot p \cdot s - b \cdot q \cdot r) \ (\bmod n)$. For example, the decryption process of n_1 is as follows: $A = n_1^2 \bmod K_{ser\_pub} = 60$, $n_1 = 37 = 100101 = 1001||\ 01$ (the 1st and 2nd bits are the same as the 3rd and 4th bits to confirm the unique solution). Then $x = 37$, $y = 24$ are calculated with the formula, the four roots are 37, -37, 26 and -26. The n_1 is a positive integer and the last two bits are overlapping, so $x = 37$ is the only solution. The decryption process of the B and C is the same ($B = 71$, $C = 77$). In the simulation, the attacker tries to query tag, then the tag replies encrypted A, B and C to it. Due to no secret key, the attacker cannot complete the decryption process, the attack is failed. The result of the SPIN simulation shows that the authentication process of the tag only performs a square multiplication and mod n operations. The analysis shows that the proposed protocol can complete high security authentication based on the public key encryption algorithm with low cost.

F. SECURITY ANALYSIS

1) SECURITY FORMAL VERIFICATION WITH BAN LOGIC

The security formal verification with BAN Logic is as follow:

Result1: Server $\xleftrightarrow{K_{Pub}K_{Pri}K_{Ser\_Pub}Key_hKey_mKey_l}$ Tag

Result2: $fresh(n_1)$

Result3: Tag$| \equiv \#(n_1) | -$ Tag$| \equiv \#(n_1 \oplus Key_h) | -$ Tag$| \equiv \#(m_1)$

Result4: Server $\lhd \{n_1\}K_{Ser\_Pub}, \{IDS \oplus n_1\}K_{Ser\_Pub}$

Result5: $\xrightarrow{K^{-1}_{Ser\_Pub}}$ Server, Tag $| \equiv \xrightarrow{K_{Ser\_Pub}}$ Server, Server $\lhd \{n_1\}K_{Ser\_Pub}, \{IDS \oplus n_1\}K_{Ser\_Pub} | -$ Server $\lhd \ n_1, IDS \oplus n_1$

Result6: Server $\lhd n_1, IDS \oplus n_1 | -$ Server $\lhd IDS$

At this point, the server can query the database according to the *IDS* to obtain the relevant information of the tag. In the process of the certification, the *IDS* is encrypted and blended with random number, which can effectively avoid tracking attacks.

Result7: Tag $\xleftrightarrow{K_{Pub}K_{Pri}Key_h}$ Server, Server $\lhd \{m_1\}K_{Pub} | -$ Server $\lhd \ m_1$

Result8: Server$| \equiv$ Tag $\xleftrightarrow{Key_h}$ Server, Server $\lhd \ m_1 | -$ Server$| \equiv$ Tag$| \sim m_1$

Result7 and result8 complete the authentication process from tag to server. The server believes that the variable m_1 is sent by the tag.

Result9: Server $\xleftarrow{K_{Ser\_Pub}Key_mKey_l}$ Tag, Tag $\lhd$ D=$\{n_1 \oplus Key_m\}K_{Pub} \oplus Key_l| -$ Tag $\lhd$ $n_1 \oplus Key_m$

Result10: Tag $| \equiv$ Server $\xleftarrow{Key_h}$ Tag, Tag $\lhd$ $n_1 \oplus Key_m| -$ Tag$| \equiv$ Server$| \sim$ D.

Result9 and result10 complete the authentication process from the server to the tag. The tag believes that the variable D is sent by the server. Then the server and the tag update secret keys. Random numbers and symmetric encryption algorithms are introduced into the protocol to effectively prevent the tracking attack. Only the square and modulo operations are used in the tag to increase the communication security to the public key encryption level within a controllable cost range.

2) RESISTS TO DoS ATTACKS

Before analyzing the DoS attack, we analyze the replay attack first, because replay attack is the basis of the DoS attack. In the replay attack, the attacker intercepts the information in the unsecured channel, and retransmits the historical information after analysis to pass the authentication. In the protocol, the changed variables include A, B, C, D.

Because these variables renewal process all include random numbers. After the authentication is successful, all the secret key information is updated with the random number, so it can effectively resist the replay attack. The essence of resisting the DoS attacks is that the protocol has good ability to resist the replay attacks and reduce the correlation of communication variables. The protocol is good at resisting replay attack which has been analyzed before. About the coupling relationship between variables, our protocol uses the public key signature twice during the verification process. The public key cryptography algorithm is irreversible without knowing the secret key, and its security can be reduced to the NP-Hard problem of the large integer decomposition. Hence the protocol can resist the DoS attack efficiently.

V. COST ANALYSIS

The focus of this paper is not to minimize cost. Instead, the public key encryption algorithm is introduced to improve the security to public key encryption hierarchy. Therefore, the cost of our protocol is slightly higher than the low-cost RFID protocol that only employs AND, OR, XOR, and SHIFT operations. The security cost of RFID tags is within 5000-10000 logic gates. The second scheme introduces the Rabin encryption algorithm whose verification process only requires square and modulo operations. The protocol uses a pipelined idea and parallel computing technology to save cost. Moreover, the shift register is employed to implement the data shift function.

One 1024 bit large number multiplying unit is designed as follows: An 8×1024 bit multiplication unit is designed with shift addition, which can perform parallel calculations by four 8×1024 bit multiplication units to complete 1024 bit large number multiplication operation. The multiplier uses four 8×1024 bit multiplier units, three 8-bit adders and one 8-bit adder. Each $8 \times 1\ 024$ bit multiplier unit consists of an 8-bit multiplier and a 16-bit addition. Analysis shows that an 8-bit multiplier requires 528 logic gates and a 16-bit adder requires 165 logic gates, so an $8\times1\ 024$ bit multiplier unit requires 693 logic gates. The analysis results show that the large multiplier requires a total of 3080 logic gates, which applies to the resource-limited RFID tag. As for the mod operation, the main cost is a 32×32 bit multiplication unit and a 32×512 bit multiplication unit, which can be completed by the previous 1024 bit large number multiplying unit without additional logic gates.

TABLE 2. The cost of encryption algorithms.

Encryption Algorithm	Cost(gate)
MD5	16000
SHA	23000
AES	20000
RSA	more than 10000
ECC	more than 10000
Rabin	more than 10000
Improved Rabin	3080
Hash	1700

The cost of common hash functions, symmetric encryption algorithms, asymmetric encryption algorithms and the improved Rabin encryption algorithm are shown in table 2. The improved Rabin encryption algorithm requires only 3080 logic gates. So we can conclude that the simple hash and the improved Rabin algorithm are suitable for the low-cost RFID tags. The main contribution of this paper is also to use the features of Rabin encryption to improve the security of RFID to the level of public key encryption.

VI. CONCLUSION

The paper analyzed the advantages and shortages of the recent classical RFID security authentication protocols, gave two tracing attack ways aiming at literature [2], and designed two RFID security authentication protocols based on the above analysis. The protocol scenario is that the reader uses the 4G, 5G technology to remotely connect with the database to enhance the mobility of the RFID system. In addition, the database and tag information is updated in real time to resist the DoS, replay, and tracking attacks. Furthermore, the second protocol introduces the Rabin encryption algorithm, whose verification process only requires square and modulo operations to raise the security grade of the RFID protocol to the public key level. Finally, we performed a simulation on the protocol with the SPIN. The research shows that the protocol can effectively resist the DoS, replay, and tracking attacks.

REFERENCES

[1] C. C. Tan, B. Sheng, and Q. Li, "Secure and serverless RFID authentication and search protocols," *IEEE Trans. Wireless Commun.*, vol. 7, no. 4, pp. 1400–1407, Mar. 2008.

[2] B. Wang and M. Ma, "A server independent authentication scheme for RFID systems," *IEEE Trans. Ind. Informat.*, vol. 8, no. 3, pp. 689–696, Aug. 2012.

[3] A. Blum, M. Furst, M. Kearns, and R. J. Lipton, *Cryptographic Primitives Based on Hard Learning Problems* (Lecture Notes in Computer Science), vol. 773. 1994, pp. 278–291.

[4] N. J. Hopper and M. Blum, "Secure human identification protocols," in *Proc. Int. Conf. Theory Appl. Cryptol. Inf. Secur.*, Dec. 2001, pp. 52–66.
[5] H. Gilbert, M. Robshaw, and H. Sibert, "An active attack against HB^+: A provably secure lightweight protocol," *Electron. Lett.*, vol. 41, no. 21, pp. 1169–1173, Oct. 2005.
[6] J. Bringer, H. Chabanne, and E. Dottax, "HB^{++}: A lightweight authentication protocol secure against some attacks," in *Proc. IEEE Int. Workshop Secur., Privacy Trust Pervasive Ubiquitous Comput.*, Jun. 2006, pp. 1–6.
[7] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda, "M^2AP: A minimalist mutual-authentication protocol for low-cost RFID tags," in *Proc. Int. Conf. Ubiquitous Intell. Comput.*, Sep. 2006, pp. 1–12.
[8] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda, "EMAP: An efficient mutual authentication protocol for low-cost RFID tags," in *Proc. OTM Workshops*, Oct. 2006, pp. 352–361.
[9] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. E. Tapiador, and A. Ribagorda, "LMAP: A real lightweight mutual authentication protocol for low-cost RFID tags," in *Proc. 2nd Workshop RFID Secur.*, Jul. 2006, pp. 1–12.
[10] H. Y. Chien, "SASI: A new ultralightweight RFID authentication protocol providing strong authentication and strong integrity," *IEEE Trans. Dependable Secure Computing*, vol. 4, no. 4, pp. 337–340, Oct. 2007.
[11] H.-M. Sun, W.-C. Ting, and K.-H. Wang, "On the security of Chien's ultralightweight RFID authentication protocol," *IEEE Trans. Dependable Secure Comput.*, vol. 8, no. 2, pp. 315–317, Mar./Apr. 2011.
[12] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. E. Tapiado, and A. Ribagorda, "Advances in ultralightweight cryptography for low-cost RFID tags: Gossamer protocol," in *Proc. Inf. Secur. Appl.*, vol. 5379, Sep. 2009, pp. 56–68.
[13] G. A. Eslamamal, S. Eman, and H. Mohamed, "Lightweight mutual authentication protocol for low cost RFID tags," *Int. J. Netw. Secur. Appl.*, vol. 2, pp. 27–37, May 2010.
[14] Y. Tian, G. Chen, and J. Li, "A new ultralightweight RFID authentication protocol with permutation," *IEEE Commun. Lett.*, vol. 16, no. 5, pp. 702–705, May 2012.
[15] L. Gao, M. Ma, Y. Shu, and Y. Wei, "An ultralightweight RFID authentication protocol with CRC and permutation," *J. Netw. Comput. Appl.*, vol. 41, pp. 37–46, May 2014.
[16] Y. Liu, M. F. Ezerman, and H. Wang, "Double verification protocol via secret sharing for low-cost RFID tags," *Future Gener. Comput. Syst.*, vol. 90, pp. 118–128, Jan. 2019.
[17] A. I. Rad, M. R. Alagheband, and S. B. Far, "Performing and mitigating force and terrorist fraud attacks against two RFID distance-bounding protocols," *J. Inf. Secur. Appl.*, vol. 42, pp. 87–94, Oct. 2018.
[18] A. A. Alamr, F. Kausar, J. Kim, and C. Seo, "A secure ECC-based RFID mutual authentication protocol for Internet of Things," *J. Supercomput.*, vol. 74, pp. 4281–4294, Sep. 2108.
[19] P. Gope, R. Amin, S. K. H. Islam, N. Kumar, and V. K. Bhalla, "Lightweight and privacy-preserving RFID authentication scheme for distributed IoT infrastructure with secure localization services for smart city environment," *Future Gener. Comput. Syst.*, vol. 83, pp. 629–637, Jun. 2018.
[20] W. Qin, K. Chen, and Y. Bai, "A new Rabin signature scheme," *J. Softw.*, vol. 11, pp. 1333–1337, Nov. 2000.

LIJUN GAO received the B.Sc. and M.Sc. degrees from the Department of Computer Science and Technology, Shenyang Aerospace University, in 2000 and 2007, respectively. He has been an Associate Professor with Shenyang Aerospace University, since 2005. He has extensive research interests including computing network and information security.

LU ZHANG received the B.Sc. degree from the Department of Computer Science and Technology, Shenyang Aerospace University, in 2006. She has been a Lecturer with Shenyang Aerospace University, since 2003. She has extensive research interests including wireless networking and wireless network security.

FENG LIN received the B.E. degree from Northeastern University, in 1985, the M.E. degree from Shenyang University of Technology, in 1987, and the Ph.D. degree from the Shenyang University of Technology, in 2003. He is currently a Professor with Shenyang Aerospace University. He has extensive research interests including computing network and information security.

MAODE MA received the B.E. degree from Tsinghua University, in 1982, the M.E. degree from Tianjin University, in 1991, and the Ph.D. degree in computer science from The Hong Kong University of Science and Technology, in 1999. He is currently an Associate Professor with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He has more than 200 international academic publications. He has extensive research interests including wireless networking and network security. He has led and/or participated in around 20 research projects funded by government, industry, military, and universities in various countries. He has been a member of the technical program committees for more than 120 international conferences. He is a Senior Member of the IEEE Communication Society and the IEEE Education Society. He has been the General Chair, the Technical Symposium Chair, the Tutorial Chair, the Publication Chair, the Publicity Chair, and the Session Chair for more than 50 international conferences. He was an Associate Editor of the IEEE Communications Letters, from 2003 to 2011. He currently serves as the Editor-in-Chief for the *International Journal of Electronic Transport*. He also serves as a Senior Editor for the IEEE Communications Surveys and Tutorials and as an Associate Editor for the *International Journal of Network and Computer Applications*, the *International Journal of Security and Communication Networks*, the *International Journal of Wireless Communications and Mobile Computing*, and the *International Journal of Communication Systems*.