Journal of Ambient Intelligence and Humanized Computing (2019) 10:3603–3615
https://doi.org/10.1007/s12652-018-1088-5

ORIGINAL RESEARCH

A quadratic residue-based RFID authentication protocol with enhanced security for TMIS

Zhiping Zhou[1,2] · Ping Wang[1] · Zhicong Li[1]

Received: 24 April 2018 / Accepted: 2 October 2018 / Published online: 12 October 2018

Abstract
Telecare medicine information system (TMIS) is one of most important RFID applications in the healthcare field. Li et al. proposed a RFID tag authentication protocol with privacy preserving in TMIS. They claimed that the protocol can resist many existing attacks and possess the advantages of high efficiency. However, we demonstrate that this protocol still have replay attack, strong forward traceability attack, de-synchronization attack, unguaranteed data integrity and the problem of tag/reader anonymity. Aiming to efficiently improve the security of Li et al.'s protocol, we propose a more secure and effective authentication protocol based on quadratic residue theory, which is suitable for TMIS with the requirements of strong privacy protection. In order to resist replay attack, the timestamp generated by the reader is used to compute reader request message sent to the server and the message is encrypted by hash function and quadratic residue theory. The improved protocol does not transmit reader and tag identifier in plaintext to guarantee anonymity and the data integrity is ensured by means of encrypting tag data using hash function. To guarantee strong forward untraceability, random number is introduced in tag key update operation and is encrypted by quadratic residue theory. Using the feature of public key cryptography of quadratic residual theory can meet the purpose of constant time identification. Our security analysis and Performance comparisons proves that our scheme has higher security and better performance to be applicable to TMIS.

Keywords RFID authentication · Telecare medicine information system (TMIS) · Enhanced security · Quadratic residue theory · Constant time identification

1 Introduction

The electronic tags manufactured by RFID technology have storage and computing capabilities and can be quickly identified without line-of-sight. Applying RFID technology to the medical field can speed up medical procedures, reduce medical errors and improve medical management efficiency, etc. However, RFID technology transmits data in a wireless communication manner, and an attacker can eavesdrop and tamper with the channel data at will, which will unable to avoid security issues. In addition, the medical information is sensitive. The security and privacy issues of medical data are key factors that hinder the potential of RFID in the medical field (Pokala et al. 2016). Solving the shortcomings of RFID technology and applying it to the medical field is of great significance. Utilizing and integrating RFID into hospital information systems without the safety hazard is an open question.

The RFID tags can be attached to patients and medical related items to achieve automatic identification, tracking and monitoring. For example, the tracking treatment of special patients, the supervision of drugs, the management of medical devices, and the storage of pathological files. Telecare medicine information systems (TMIS) realized by RFID technology can monitor the real time patient's health data related to heartbeat, blood pressure, etc. It can help nurses to carry out remote treatment for patients. However, the attacker can eavesdrop and modify the physiological data transmitted over wireless channels. Hence, maintaining the privacy, integrity and confidentiality of the data are the critical requirements (Rahman et al. 2016a). The transmission of data after authentication is an important strategy to

✉ Zhiping Zhou
zzp@jiangnan.edu.cn

[1] School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China

[2] Engineering Research Center of Internet of Things Technology Applications Ministry of Education, Jiangnan University, Wuxi 214122, China

ensure the security of data. In the process of authentication, it is necessary to transfer unique attribute values to ensure the legality of identification. The leakage of these unique attribute values can lead to a series of security and privacy issues. The RFID authentication protocols have the following deficiencies: (1) most protocols cannot resist common attacks, such as desynchronize attack, replay attack and so on. (2) most protocols are not suitable for mobile RFID systems because assumption of secure channel between server and reader. (3) most protocols are not scalability because they generally require a linear search on database to identify a tag. In addition, an ideal RFID authentication protocol in telecare medicine information system is required to meet the following criterion (Li et al. 2015; Srivastava et al. 2015): (1) mutual authentication between communicating parties. (2) resistance to common attacks, such as replay, desynchronize and impersonation attacks. (3) protection of data security and user privacy. (4) provide forward security, such as strong forward untraceability and forward untraceability. (5) reduce the search cost and meet the needs of scalability. (6) minimize the tag cost. In the paper, we assumes the communication channel between the reader and the server is wireless to satisfy mobility and then puts forward a secure and scalability RFID tag authentication protocol for TMIS based on quadratic residue theory.

1.1 Related Work

Recently, researchers have focused on the application of RFID in the medical field. Malasinghe et al. (2017) indicated remote monitoring of patients has many advantages in a fast aging world population with increasing health complications. In Sareen et al. (2016), a novel architecture based on RFID and cloud computing infrastructure is proposed for the detection and monitoring of Ebola infected patients. Amiribesheli et al. (2015) use RFID technology to construct continuous state sensors to collect physiological signals as a health monitoring module for residents in smart homes. In a platform (Poncela et al. 2018) for eAssistance, RFID tags to identify objects, together with the construction of a Dynamic Bayesian Network to model the interaction between the human and the objects.

Since the insecure nature of wireless channel, RFID brings lots of privacy and security problems while bringing convenience to manufacture and life (Qing et al. 2016). These issues have aroused the concern of scholars. Su et al. (2017) pointed out that reader/tag mutual authentication is a major theme in RFID security and privacy research. Aiming at the problem of privacy infringement and tag forgery, Cho et al. (2015) proposed a Hash-based RFID authentication protocol. In the protocol, the random numbers are grouped to prevent them from being exposed. However, Dehkordi and Farzaneh (2014) pointed out that the weakness of random number encryption algorithm can lead to impersonation attack, traffic analysis attack and DOS attack. Based on this, Dehkordi and Farzaneh (2014) proposed an improved RFID authentication protocol that effectively solves the impersonation attack and DOS attack in Cho et al. (2015). Unfortunately, Alavi et al. (2015) found that the unreliability of the key update operation causes literature (Dehkordi and Farzaneh 2014) to suffer forward and backward traceability attack. Therefore, Alavi et al. proposed an improved key update algorithm to ensure untraceability. With the rapid development of mobile smart terminals in recent years, the demand for mobile RFID systems is becoming more and more urgent. Hoque et al. (2010) proposed a server-less RFID authentication protocol with enhancing privacy and security. The participants only include reader and tags, so it can be applied to mobile RFID systems. However, Deng et al. (2014) showed that their protocol is vulnerable to desynchronization attack and further proposed an improved protocol. Regrettably, Deng et al. (2014) fail to solve the problem of de-synchronization attack and location privacy. To meet the needs of mobile RFID, Sundaresan et al. (2015) proposed an RFID authentication protocol using mobile reader. The protocol only uses pseudo-random function operation in the tag side and achieves EPCC1G2 compliance, but it is vulnerable to replay and de-synchronization attacks, etc (Jannati and Bahrak 2016). In addition, the reader needs to download an access list of multiple tags authorized to search during the setup phase, so the requirements on the reader storage capacity are higher. With the rapid development of healthcare delivery services and non-contact identification technologies, RFID technology is widely used to medical field such as telecare medicine information system (Li et al. 2015). Meanwhile, many privacy preserving RFID authentication protocols for TMIS were proposed. In 2013, in order to enhance patient medication safety, Kaul and Awasthi (2013) design a dynamic ID based lightweight RFID authenticaiton protocol that can prevent the tag from traceability by updating the secret key and indentity after each successful authentication between the tag and the server. Srivastava et al. (2015) proposed an RFID tag authentication protocol for telecare medicine information system and claimed that the protocol is effective against a variety of active and passive attacks such as forged attacks, replay attacks, and so on. However, it cannot resist de-synchronization attack and impersonation attack, and cannot ensure the integrity of tag data. Li et al. (2015) proposed an improved RFID tag authentication protocol with privacy preserving in telecare medicine information system based on literature (Srivastava et al. 2015). The improved protocol improves the authentication efficiency by using the tag identifier as index. However, the tag identifier is fixed value and transmitted in plain text, which cause that the protocol cannot guarantee anonymity of tag and reader. Meanwhile, the protocol is unable to

provide resistance to de-synchronization, impersonation and other attacks. Wu et al. (2017) proposed a two-factor authentication scheme based on ECC for weak security such as internal attacks, offline password guessing and user fake attacks. Mohammedi et al. (2017) proposed a secure and remote patient authentication scheme for mobile healthcare environments. The proposed scheme translates the patient biometric data to ECC-based keys. Although, The scheme of literature (Wu et al. 2017; Mohammedi et al. 2017) achieve high security, the ECC encryption primitives are used multiple times to implement a public key encryption system, in a strict sense, these protocols cannot be used in an RFID system due to low cost restrictions of RFID.

It's necessary to guarantee security while ensuring scalability in RFID system (Avoine et al. 2013). To improve the authentication efficiency, tree-based authentication protocols (Li et al. 2012; Deng et al. 2013) were proposed successively. These schemes reduce the authentication complexity of the protocol from $o(n)$ to $o(\log_{\alpha} n)$ successfully, but this kind of scheme is suffered from privacy disclosure. Once a tag is compromised by the attacker, the privacy of other tags in the system will be leaked seriously. To improve the shortcomings of such schemes, Avoine et al. (2007) proposed a group-based RFID authentication protocol using symmetric encryption. This protocol has higher privacy performance than tree-based protocol, and simultaneously ensuring the authentication efficiency. Rahman et al. (2016b) proposed a new group-based scheme to improve privacy on the basis of Avoine et al. (2007). The protocol stores an identity set containing multiple identifiers in the tag side to provide unlinkability even if the adversary realizes the identifier used in previous response. However, the lack of key update mechanism leads to forward traceability attack. In addition, encrypting the tag response message with real identify is not conducive to ensuring the anonymity of the system. Although the group-based method reduces the possibility of privacy leakage, it cannot completely solve the problem of privacy disclosure. Akgün and Aglayan (2015) proposed a scalability RFID authentication protocol based on PUFs (Physically unclonable functions). This protocol uses one master key shared by all tags to meet constant-time identification and provides resistance against tag compromising attacks by using PUFs as a secure storage to keep secrets of the tag. A major drawback of PUF is that it can produce fluctuating results based on the operating conditions. Thus, the large-scale implementation of PUF is yet to be a reality and remains an open problem (Sundaresan et al. 2015). To address the problem of low authentication efficiency, Doss et al. (2013) proposed a privacy preserving mutual authentication protocol for RFID systems based on quadratic residue. In the protocol, the quadratic residue theory can ensure constant-time identification while solving privacy leakage problem. However, since $h(RID)$ is constant and y is generated by s and u in (Doss et al. 2013), the message $< x'', t'', y'', u'', s >$ replayed by the attacker can be authenticated by the server successfully. In addition, the result of the key update misses the unpredictability, so the attacker can deduce the key of each round by obtaining the key in a certain state. Wu et al. (2018) proposed a anonymous RFID tag authentication protocol for e-healthcare applications. This protocol only use Hash function to encrypt operation that can guarantee lightweight. However, the complexity of the server-side identification tags is linear, which results in the unscalability of the system.

1.2 Our contributions

The main contributions of the paper are listed below: (1) we analyze the security of the authentication protocol for TMIS proposed by Li et al. and indicate that the protocol is vulnerable to some common attacks. Based on this, we propose a protocol based on quadratic residue theory for TMIS and prove the security of the protocol by formal analysis. (2) We add the quadratic residual theory to the protocol flow and use its feature of public key encryption to realize the distribution of fresh secrets, which can effectively prevent strong forward privacy. In addition, the method using public key to encryption and private key to decryption reduces the search cost of the server.we solve the contradiction between fast search and tracking attacks by using the quadratic residual theory. (3) By using the privacy model enhanced the ability of the adversary to analyze the protocol, we proves that the protocol can achieve strong forward privacy and meet the needs of strong privacy protection of TMIS.

1.3 Structure of our paper

The rest of this paper is organized as follows. The weak security of Li et al's protocol is analyzed in Sect. 2. we describes the details of our scheme in Sect. 3. The security and performance of the proposed scheme is evaluated in Sect. 3. In addition, the GNY-Logic based formal analysis is presented in this section. Finally, we give a conclusion and expectation in Sect. 5.

2 Preliminaries and related analysis

2.1 Privacy Model

This paper adopts Vaudenay's privacy model in Poncela et al. (2018) and the definition of strong forward untraceability in Qing et al. (2016) to analyze the security of the protocol. In Vaudenay's model, the attackers are allowed to run the following oracles:

CreatTag(ID_T): Generates free tag with sole ID_T with *SetupTag*(ID_T).

DrawTag(*dist*) $\rightarrow$ ($vtag^1, \cdots, vtag^n$): Moves from the set of free tags to the set of drawn tags a tuple of tags at random following the probability distribution *dist*.

FreeTag(*vtag*): Moves the virtual tag back to the set of the free tags. This makes *vtag* unreachable.

Execute(T) $\rightarrow$ (π, *transpcript*): Executes the protocol instance π and return the session record *transpcript*.

Launch() $\rightarrow \pi$: Makes the reader launch a new protocol instance π.

SendReader(m, π) $\rightarrow r$: Sends a message m to a protocol instance π for the reader and receives the answer r.

SendTag(m, π) $\rightarrow r$: Sends a message m to a protocol instance for the tag and receives the answer r.

Result(π) $\rightarrow x$: When π is complete, returns 1 if Output $\neq \perp$ and 0 otherwise.

Corrupt(*vtag*) $\rightarrow S$: Returns the current state of the tag. If *vtag* is no longer used after this oracle call, we say that *vtag* is destroyed.

The Vaudenay model divided the attacker into 4 categories (Weak, Forward, Destructive, Strong) according to the regulation that whether the attacker can call the *Corrupt*(*vtag*) oracle. The Strong adversary can query *Corrupt*(*vtag*) oracle and Weak adversary cannot. Meanwhile, Vaudenay divided the attacker into 2 categories (Wide, Narrow) according to *Result*(π) oracle.

Definition 1 [(*Forward untraceability)* (Chen et al. 2016)] The narrow-strong adversary cannot trace the tag at the round $(j+2)$, even though the adversary corrupts the target tag in the *jth* session and misses the $(j+1)$*th* session.

Definition 2 [(*Strong Forward untraceability*) (Chen et al. 2016)] It is impossible for narrow-strong adversary to trace the tag in the $(j+1)$*th* session, even though the adversary corrupts the tag's keys in the *jth* session.

2.2 Attacker model and security assumptions

The participating entities in this paper include back-end server, readers and tags. To meet the needs of mobile RFID, we assume that the channels of reader-tag and reader-back-end server are wireless. An adversary can not only eavesdrop, but also intercept and modify the communication messages transferred through wireless transmission channel. The adversary even can initiate authentication session. This paper proves strong forward untraceability and forward untraceability under the narrow-strong attacker model. Other security is proved under the weak attacker model. Without loss of generality, we assume that the reader and backend server cannot be compromised.

2.3 Weakness of Li et al.'s protocol

Table 1 provides a brief description for the notations that are used in the protocol. The specific process of the protocol is described in Li et al. 2015. Li et al.'s scheme uses direct identifier index to ensure constant time authentication. This scheme transmits identifier plaintext of the tag and reader in the protocol execution process, which results in anonymity of the tag and reader cannot be achieved. Moreover, Li et al. (2015) protocol is based on the assumption of secure channel, which is not applicable to mobile RFID environments. To accommodate the mobile RFID environment, it is assumed that the communication channel between reader and backend server is wireless. Li et al.'s protocol will face

Table 1 Notations in the protocol

Notation	Description
ID_k	The identifier of the *k th* tag
RID_k	The identifier of the *k th* reader
RPW_k	The password of the *k th* reader
$T_1, T_2, T_3, \Delta T$	The timestamp T_1, T_2, T_3 and the expected legitimate time interval for transmission delay
R_r, R_t	The random number generated by reader and tag
s_j	The secret value used in the current *jth* session and is shared between server and tag
s_{j-1}	The secret value used in the previous $(j-1)$*th* session. Initially, $s_j = s_{j-1}$
x_j	The secret value used in the current *jth* session and is shared between server and reader
x_{j-1}	The secret value used in the previous $(j-1)$*th* session. Initially, $x_j = x_{j-1}$
$h(\cdot)$	The one-way hash function
Data	The information of the tagged object
$\|\|, \oplus$	The concatenation operation and bitwise XOR operation
n, m	The positive integers stored in the tag and reader
p, q, g, h	The large prime numbers, where $n = p \cdot q$, $m = g \cdot h$
mod	The modular multiplication operation

replay attack and data integrity problems in mobile RFID environment. Moreover, Li et al.'s protocol is vulnerable to de-synchronization attack and strong forward traceability attack. The specific analyses are as follows:

1. Tag and reader anonymity

Proof In the Li et al.'s protocol, the identifiers of tag and reader are transmitted in plaintext to ensure constant time authentication. The adversary can eavesdrop the identifier of the tag and reader through insecure channel, which causes the anonymity of the tag and reader cannot be guaranteed. In addition, the value of the transmitted identifier are fixed, so the attacker can trace the target tag/reader by comparing the value of ID_k/RID_k, which will result in location privacy leaks. □

2. Strong forward traceability attack

Proof Initialization phase:

1 $CreateTag(s_0), CreateTag(s_1)$.
Learning phase:
2 $DrawTag(s_e) \rightarrow vtag^e, e \in (0,1)$.
3 $Corrupt(vtag^e) \rightarrow (ID^j_{ke}, s^j_e)$.
4 $SendReader-Tag(Init, s^j_e, \pi^j) \rightarrow (RID^j_k, A^j, B^j, T^j_1)$.
5 $SendTag((RID^j_k, A^j, B^j, T^j_1), vtag^e) \rightarrow (ID^j_k, C^j_e, D^j_e, T^j_2)$.
6 $SendReader((ID^j_k, C^j_e, D^j, T^j_2), \pi^j) \rightarrow (G^j)$.
7 $FreeTag(vtag^e)$.
Challenge phase:
8 $DrawTag(s_c) \rightarrow vtag^c, c \in (0,1)$.
9 $Launch() \rightarrow \pi^{j+1}$.
10 $SendReader-Tag(Init, s_c, \pi^{j+1}) \rightarrow (RID^{j+1}_k, A^{j+1}, B^{j+1}, T^{j+1}_1)$.
11 $SendTag((RID^{j+1}_k, A^{j+1}, B^{j+1}, T^{j+1}_1), vtag^c) \rightarrow (ID^{j+1}_k, C^{j+1}_c, D^{j+1}_c, T^{j+1}_2)$.
12 $SendReader((ID^{j+1}_k, C^{j+1}_c, D^{j+1}, T^{j+1}_2), \pi^{j+2}) \rightarrow (G^{j+1})$.
Guessing phase:
13 Compute $(C^j_e, C^{j+1}_c, s^{j+1}_e) \rightarrow D^{j+1}_e$.
14 Compare (D^{j+1}_e, D^{j+1}_c), if $D^{j+1}_e = D^{j+1}_c$, then $c = e$ else $c = |1-e|$.
15 $Adv^{Forward-Untrace}_A(k) = 1$.

□

Since the random number used in key update operation can be extracted from the eavesdropped messages, the attacker can compute tag's key of the $(j+1)th$ session via the corrupted tag's key in the jth session.Using this drawback, the attackers can implement strong forward traceability attack. The adversary randomly sends $DrawTag(s_e)$ chosen from s_0 and $s_1(e \in (0,1))$to launch the scheme. The adversary chooses another time interval $j+1$ to monitor the transmitted messages and compute the updated keys of tag^e such as $R^j_t = C^j_e \oplus h(s^j_e || ID^j_{ke}), s^{j+1}_e = h(s^j_e \oplus R^j_t)$. Therefore, using the above tag's key s^{j+1}_e and ID^j_{ke}, R^{j+1}_t can be extracted from the eavesdropped values C^{j+1}_c. Given $vtag^e$, the adversary calculates the values D^{j+1}_e using the values R^{j+1}_t, s^{j+1}_e and T^{j+1}_2. The calculation process is as follows: $D^{j+1}_e = h(h(s^{j+1}_e || ID^j_{ke}) \oplus T^{j+1}_2 \oplus R^{j+1}_t)$. At last, if the monitored message $D^{j+1}_c = D^{j+1}_e$, the given tag tag_c is differentiated from the set by using the tag's responds.

3. Replay attack

Proof

(a) As the channels of reader-tag and reader-server are insecure, an attacker can eavesdrop the jth session messages $\langle RID_k, A, B, T_1\rangle$, $\langle ID_k, C, D, T_2\rangle$, $\langle RID_k, A, B, T_1, ID_k, C, D, T_2\rangle$, $\langle E, F, G\rangle$ and $\langle G\rangle$.
(b) In the $(j+1)th$ session, the attacker replay the jth session message $< RID_k, A, B, T_1 >$ to the tag. After receiving the messages, the tag calculate $C' = h(s_{j+1} || ID_k) \oplus R'_t, D' = h(h(s_{j+1} || ID_k) \oplus T'_2 \oplus R'_t)$ and send message $\langle ID_k, C', D', T'_2\rangle$ to the attacker. Then, the attacker can replay the composite messages $\langle RID_k, A, B, T_1, ID_k, C', D', T'_2\rangle$ to the server.
(c) The server verify $T'_3 - T'_2 < \Delta T$ successfully. Due to the server stores two round keys x_j and x_{j+1}, the random number R^*_r can be extracted by $R^*_r = A \oplus h(x_j || RID_k)$. The server further calculates$B^* = h(h(x_j || RID_k) \oplus T_1 \oplus R^*_r)$ and verifies $B^* = B$ successfully. Similarly, the server extracts R^*_t by $R^*_t = C' \oplus h(s_{j+1} || ID_k)$ and calculates $D^* = h(h(s_{j+1} || ID_k) \oplus T'_2 \oplus R^*_t)$. So, the tag and the attacker is authenticated by the server successfully. The server continue to calculates and sends message $\langle E', F', G'\rangle$ to the attacker and updates the secret. To sum up, replay attack is successful. □

4. De-synchronization attack

Proof The reader only updates the secret x_j and the value of W_k and V_k are not updated normally, which will causes de-synchronization attack.

(a) In the *jth* session, the reader updates x_j with $x_{j+1} = h(x_j \oplus R_r)$ and the value of W_k, V_k are not updated. Meanwhile, the server updates x_{j-1} and x_j with x_j and $x_{j+1} = h(x_j \oplus R_r^*)$.

(b) In the $(j+1)th$ session, the staff input RID_k and RPW_k. Then, the reader computes $V_k' = W_k \oplus RID_k \oplus RPW_k = V_k$, and is successfully booted. The reader generates random number R_r, computes $A = V_k' \oplus R_r$, $B = h(V_k' \oplus T_1 \oplus R_r)$ and sends $< RID_k, A, B, T_1 >$ to the tag.

(c) The tag generates random number R_t and calculates $C = h(s_{j+1}||ID_k) \oplus R_t$, $D = h(h(s_{j+1}||ID_k) \oplus T_2 \oplus R_t)$. Then, it sends $< ID_k, C, D, T_2 >$ to the reader.

(d) The reader checks $(T_2 - T_1) < \Delta T$ and transmits $< RID_k, A, B, T_1 >$, $< ID_k, C, D, T_2 >$ to the backend server.

(e) The server checks $(T_3 - T_2) < \Delta T$ and calculates $R_r^* = A \oplus h(x_j||RID_k)$ (The server stores two rounds keys x_j and x_{j+1}). Then, it calculates $B^* = h((x_j||RID_k) \oplus T_1 \oplus R_r^*) = B$. Similarly, it computes $R_t^* = C \oplus h(s_{j+1}||ID_k)$ and $D^* = h(h(s_{j+1}||ID_k) \oplus T_2 \oplus R_t^*) = D$. Finally, the server computes $E{=}h(x_j||RID_k||T_1||R_r^*||h(x_j \oplus R_r^*))$, $F = Data \oplus h(x_j \oplus R_r^*)$, $G = h(s_{j+1}||ID_k||T_2||R_t^*||h(s_{j+1} \oplus R_t^*))$ and sends $< E, F, G >$ to the reader.

(f) Upon receiving the message, the reader computes $E^* = h(x_{j+1}||RID_k||T_1||R_r||h(x_{j+1} \oplus R_r))$ using the updated key x_{j+1}. Therefore, E^* is not equal to E, which causes the server cannot be authenticated by the reader successfully. To sum up, The protocol has de-synchronization attack. □

5. Data integrity vulnerability

Proof

(a) The reader sends messages $\langle RID_k, A, B, T_1 \rangle$ to the tag and receives response messages $\langle ID_k, C, D, T_2 \rangle$. Then, it transmits messages $\langle RID_k, A, B, T_1, ID_k, C, D, T_2 \rangle$ to the server. After the server validates the reader and the tag, the message $\langle E, F, G \rangle$ is calculated and sent to the reader.

(b) The attacker eavesdrops and intercepts the messages $\langle E, F, G \rangle$. Then, it tampers the message F into $F \oplus \Delta$ and transmits the messages $\langle E, F \oplus \Delta, G \rangle$ to the reader. Because the message E has not been tampered by the attacker and the reader lacks the authentication of the message F, the reader will not be able to detect whether is being tampered by the attacker, which results in the reader receiving the wrong tag data information $Data \oplus \Delta$. □

3 The improved protocol

This section mainly elaborates the improved scheme. The description of the notation used is also shown in Table 1. The proposed scheme consists of three phases: the pre-phase, the boot reader phase and the authentication phase. Fig. 1 shows the entire flowchart of our improved protocol. The specific process is as follows:

3.1 Pre-phase

1. The backend server generates and stores four large primes p, q, g, h as the private key. Then, it calculates the corresponding public keys $n = p \cdot q$ and $m = g \cdot h$.
2. In tag side, the tag and the back-end server share tag's identifier ID_k, one-way hash function $h(\cdot)$, and secret value of tag s_j. In addition, the tag stores the positive integers n.
3. In reader side, the back-end server computes $V_k = h(x_j||RID_k)$and$W_k = h(x_j||RID_k) \oplus RID_k \oplus RPW_k$. Then, it stores V_k, W_k, one-way hash function $h(\cdot)$, reader's identifier RID_k and secret value of reader x_j in the

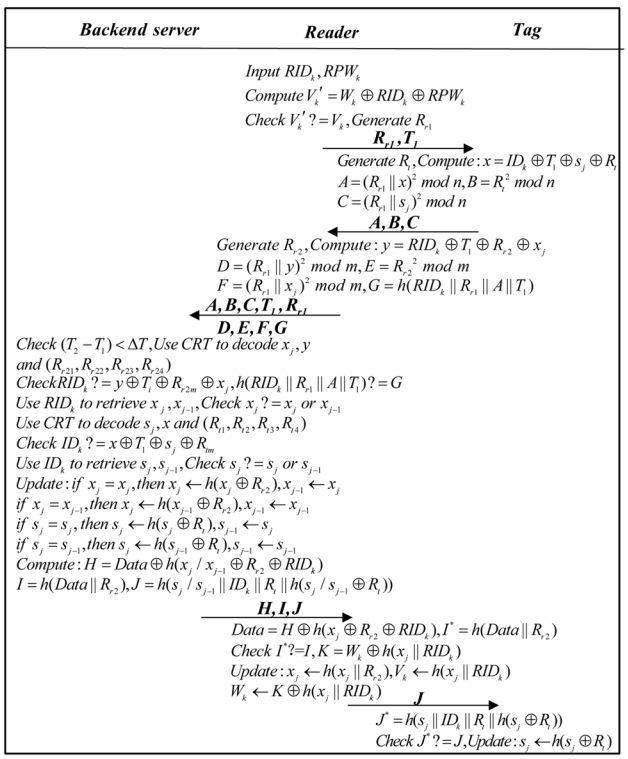

Fig. 1 The Authentication Process of Improved Protocol

memory of it. In addition, the reader stores the positive integers m.

4. The tag and reader have its own random number generator.
5. The back-end server saves the information ID_k, s_j and s_{j-1} for each tag. Initially, the value $s_j = s_{j-1}$.
6. The back-end server saves the information RID_k, x_j and x_{j-1} for each reader. Initially, the value $x_j = x_{j-1}$.

3.2 Boot reader phase

Before the telecare staff can use the reader to provide telecare services, the telecare staff must be successful boot reader. The staff inputs correct identifier RID_k and password RPW_k. Then the reader computes $V'_k = W_k \oplus RID_k \oplus RPW_k$ and checks if $V'_k = V_k$ holds or not. If it does not hold, the reader halts the process. Otherwise, the reader is successfully booted.

3.3 Authentication phase

Step 1 Reader's request

(a) The reader generates a random number R_{r1} and records the timestamp T_1 at this time.
(b) The reader sends a request message $\langle R_{r1}, T_1 \rangle$ to the tag.

Step 2 Tag's response message

(a) Upon receiving the request message from the reader, the tag generates a random number R_t.
(b) The tag computes $x = ID_k \oplus T_1 \oplus s_j \oplus R_t$, $A = (R_{r1}||x)^2 \ mod \ n$, $B = (R_t)^2 \ mod \ n$, $C = (R_{r1}||s_j)^2 \ mod \ n$.
(c) The tag sends response message $\langle A, B, C \rangle$ to the reader.

Step 3 Reader's response message

(a) After receiving the response message from the tag, the reader generates a random number R_{r2}.
(b) The reader calculates the messages as follows: $y = RID_k \oplus T_1 \oplus R_{r2} \oplus x_j$, $D = (R_{r1}||y)^2 \ mod \ m$, $E = R_{r2}^2 \ mod \ m$, $F = (R_{r1}||x_j)^2 \ mod \ m$, $G = h(RID_k||R_{r1}||A||T_1)$.
(c) The reader forwards the message $\langle A, B, C, T_1, D, E, F, R_{r1}, G \rangle$ to the backend server.

Step 4 Reader authentication and tag authentication

(a) Once receiving the response message from reader, the back-end server checks if $T_2 - T_1 > \Delta T$ or not, where T_2 is the current timestamp of the back-end server. if it holds, the server rejects this request
(b) If $T_2 - T_1 < \Delta T$, the backend server uses Chinese Residue Theorem with g and h to decode x_j, y and $(R_{r21}, R_{r22}, R_{r23}, R_{r24})$ from $F = (R_{r1}||x_j)^2 \ mod \ m$, $D = (R_{r1}||y)^2 \ mod \ m$ and $E = R_{r2}^2 \ mod \ m$, where x_j, y are uniquely determined by R_{r1} from four solutions.
(c) The server computes $RID_k = y \oplus T_1 \oplus R_{r2m} \oplus x_j$ and quickly search the reader's record in the database by using RID_k index. In the worst case, this process only takes 4 times. If there exists the reader's record, the server continues to check $h(RID_k||R_{r1}||A||T_1)? = G$. When the above formula holds, the server uses RID_k to quickly retrieve x_j, x_{j-1} and check if $x_j = x_j or \ x_{j-1}$ or not. The reader will be authenticated by the server successfully if the above steps are all passed. If not, it is judged to be an abnormal authentication message and the session is terminated.
(d) If above step holds, the backend server uses Chinese Residue Theorem with p and q to obtain s_j, x and $(R_{t1}, R_{t2}, R_{t3}, R_{t4})$ from $C = (R_{r1}||s_j)^2 \ mod \ n$, $A = (R_{r1}||x)^2 \ mod \ n$, $B = (R_t)^2 \ mod \ n$, where s_j and x are uniquely determined by R_{r1} from four solutions.
(e) The backend server calculates $ID_k = x \oplus T_1 \oplus s_j \oplus R_{tm}$ and quickly search the tag's record in the database by using ID_k index. Similarly, this process only takes 4 times in the worst case. If there has the tag record, the backend server can retrieve s_j, s_{j-1} by ID_k quickly and check if $s_j? = s_j \ or \ s_{j-1}$. The tag is authenticated by the server successfully if the above steps hold. If not, it is judged to be an abnormal authentication message and the session is terminated.
(f) After verifying the reader and the tag, the backend server updates original x_j, x_{j-1} and s_j, s_{j-1} as follows: if $x_j = x_j$, then $x_j \leftarrow h(x_j \oplus R_{r2})$, $x_{j-1} \leftarrow x_j$, else if $x_j = x_{j-1}$, then $x_j \leftarrow h(x_{j-1} \oplus R_{r2})$, $x_{j-1} \leftarrow x_{j-1}$. If $s_j = s_j$, then $s_j \leftarrow h(s_j \oplus R_t)$, $s_{j-1} \leftarrow s_j$, else if $s_j = s_{j-1}$, then $s_j \leftarrow h(s_{j-1} \oplus R_t)$, $s_{j-1} \leftarrow s_{j-1}$.
(g) The backend server informs the relevant data of the tag to the reader and computes $H = Data \oplus h(x_j/x_{j-1} \oplus R_{r2} \oplus RID_k)$ (if $x_j = x_j$, then $H = Data \oplus h(x_j \oplus R_{r2} \oplus RID_k)$,else if $x_j = x_{j-1}$, then $H = Data \oplus h(x_{j-1} \oplus R_{r2} \oplus RID_k)$), where $Data$ is the tag information which needs to be transmitted to the reader, $I = h(Data||R_{r2})$, $J = h(s_j/s_{j-1}||ID_k||R_t||h(s_j/s_{j-1} \oplus R_t))$ (if $s_j = s_j$, then $J = h(s_j||ID_k||R_t||h(s_j \oplus R_t))$, else if $s_j = s_{j-1}$, then $J = h(s_{j-1}||ID_k||R_t||H(s_{j-1} \oplus R_t))$).
(h) The backend server forms a new message $\langle H, I, J \rangle$ to the reader.

Step 5 Reader receives the relevant data of the tag

(a) After receiving the message from backend server, the reader extracts $Data$ from H as

$Data = H \oplus h(x_j \oplus R_{r2} \oplus RID_k)$ and computes $I^* = h(Data||R_{r2})$. The backend server checks if I^* is equal to the received I. if $I^* = I$, the integrity of $Data$ is guaranteed and the server is authenticated by the reader successfully.

(b) If above step holds, the reader successfully authenticates the back-end server. Then, the reader computes $K = W_k \oplus h(x_j||RID_k)$ and updates x_j, V_k, W_k as follows: $x_j \leftarrow h(x_j \oplus R_{r2})$, $V_k \leftarrow h(x_j||RID_k)$ (the x_j used here is updated), $W_k \leftarrow K \oplus h(x_j||RID_k)$ (the x_j used here is updated). If not, it is judged to be an abnormal authentication message and the session is terminated.

(c) The reader sends the remaining message J to the tag for further communication.

Step 6 Tag authenticates the backend server and updates the secret.

(a) After receiving J from reader, the tag calculates $J^* = h(s_j||ID_k||R_t||h(s_j \oplus R_t))$ and checks if J^* is equals to the received message J.

(b) If above step holds, the tag successfully authenticates the back-end server and updates original s_j with $s_j \leftarrow h(s_j \oplus R_t)$. If not, it is judged to be an abnormal authentication message and the session is terminated.

4 The security and performance analysis of improved protocol

4.1 GNY logic correctness analysis

GNY logic is proposed by L.Gong et al, which is optimized and derived from BAN logic. GNY mechanism enables systematic way of understanding the working of cryptographic protocols. In the semantic and axiom, GNY logic is more detailed than BAN logic. The GNY model enables the expression of different trust levels and implicit conditions behind protocol steps. We now verify the correctness of our protocol by using this method. S, R and T respectively represents the server, the reader and the tag in the course of proof. Protocol messages and the goals of analysis are shown in Table 2. Assumptions used in the analysis are shown in Table 3. The security correctness proof is shown in Table 4. The proof of goal $G1$ is shown by the verification step $V5$(which is derived using $V1$, $V2$, $V3$, $V4$); proof of goal $G2$ is shown by the verification step $V8$(which is derived using $V6$, $V7$); proof of goal $G3$ is shown by the verification step $V13$ (which is derived using $V9$, $V10$, $V11$, $V12$) and proof of goal $G4$ is shown by the verification step $V17$ (which is derived using $V14$, $V15$, $V16$).

Table 2 Protocol messages and security correctness goals

Protocol formal	Goal notation
$M1$: $T \lhd R_{r1}, T_1$	$G1$: $S \mid\equiv R \mid\sim \#(h(RID_k\|\|R_{r1}\|\|A\|\|T_1))$
$M2$: $R \lhd (A, B, C)$	$G2$: $S \mid\equiv T \mid\sim \#((R_{r1}\|\|x)^2 \bmod n))$
$M3$: $S \lhd (A,B,C,T_1,R_{r1},D,E,F,G)$	$G3$: $R \mid\equiv S \mid\sim \#(h(Data\|\|R_{r2}))$
$M4$: $R \lhd (H, I, J)$	$G4$: $T \mid\equiv S \mid\sim \#(h(s_j\|\|ID_k\|\|R_t\|\|h(s_j \oplus R_t))$
$M5$: $T \lhd (J)$	

Table 3 Assumptions used in the analysis

$A1$: $T(ID_k, s_j, n, R_t)$	$A6$: $R \mid\equiv R \overset{RID_k, x_j}{\longleftrightarrow} S$
$A2$: $R(V_k, W_k, RID_k, x_j, m, R_{r1}, R_{r2})$	$A7$: $S \mid\equiv S \overset{RID_k, x_j}{\longleftrightarrow} R$
$A3$: $S(ID_k, s_j, RID_k, x_j, p, q, g, h)$	$A8$: $R \mid\equiv \#(R_{r1}, R_{r2}, T_1)$
$A4$: $T\mid\equiv T \overset{ID_k, s_j}{\longleftrightarrow} S$	$A9$: $T \mid\equiv \#R_t$
$A5$: $S \mid\equiv S \overset{ID_k, s_j}{\longleftrightarrow} T$	$A10$: $S\mid\equiv \#T_2$

Table 4 Security correctness proof

No.	Proof notation	Postulate
$V1$	$S \ni (R_{r1}, T_1, A)$	$M3$, $P1$, $A3$, $P6$
$V2$	$S \ni (RID_k\|\|R_{r1}\|\|A\|\|T_1)$	A3, $P2$, $V1$
$V3$	$S \mid\equiv \#(RID_k\|\|R_{r1}\|\|A\|\|T_1)$	A10, $F1$
$V4$	$S \mid\equiv \#h(RID_k\|\|R_{r1}\|\|A\|\|T_1)$	$F10$, $V2$, $V3$
$V5$	$S \mid\equiv R \mid\sim \#h(RID_k\|\|R_{r1}\|\|A\|\|T_1)$	$M3$, $V4$, $A7$, $I3$
$V6$	$S \ni (R_{r1}\|\|x)^2 \bmod n$	$M3$, $T1$, $P1$
$V7$	$S \mid\equiv \#((R_{r1}\|\|x)^2 \bmod n)$	A10, $F1$
$V8$	$S \mid\equiv T \mid\sim \#((R_{r1}\|\|x)^2 \bmod n))$	$M3$, $A3$, $A5$, $V6$, $V7$, $I1$, $P6$
$V9$	$R \ni Data$	$M4$, $P1$, $A2$, $P6$
$V10$	$R \ni (Data\|\|R_{r2})$	$V9$, $P2$
$V11$	$R \mid\equiv \#(Data\|\|R_{r2})$	$A8$, $F1$
$V12$	$R \mid\equiv \#h(Data\|\|R_{r2})$	$F10$, $V10$, $V11$
$V13$	$R \mid\equiv S \mid\sim \#h(Data\|\|R_{r2})$	$M4$, $V13$, $A6$, $I3$
$V14$	$T \ni (s_j\|\|ID_k\|\|R_t\|\|h(s_j \oplus R_t))$	$A1$, $P2$
$V15$	$T \mid\equiv \#(s_j\|\|ID_k\|\|R_t\|\|h(s_j \oplus R_t))$	A9, $F1$
$V16$	$T \mid\equiv \#h(s_j\|\|ID_k\|\|R_t\|\|h(s_j \oplus R_t))$	$F10$, $V14$, $V15$
$V17$	$T \mid\equiv S \mid\sim \#h(s_j\|\|ID_k\|\|R_t\|\|h(s_j \oplus R_t)$	$M5$, $V17$, $A4$, $I3$

4.2 Security analysis

Theorem 1 *The improved protocol satisfies strong forward untraceability and forward untraceability under the narrow-strong attacker model.*

Proof Initialization phase:

1 $CreateTag(s_0), CreateTag(s_1)$.
Learning phase:
2 $DrawTag(s_e) \rightarrow vtag^e, e \in (0,1)$.
3 $Corrupt(vtag^e) \rightarrow (ID_{ke}^j, s_e^j, n)$.
4 $SendReader - Tag(Init, s_e^j, \pi^j) \rightarrow (R_{r1}^j, T_1^j)$.
5 $SendTag((R_{r1}^j, T_1^j), vtag^e) \rightarrow (A^j, B^j, C_e^j)$.
6 $SendReader((A^j, B^j, C_e^j), \pi^j) \rightarrow (J^j)$.
7 $FreeTag(vtag^e)$.
Challenge phase:
8 $DrawTag(s_c) \rightarrow vtag^c, c \in (0,1)$.
9 $Launch() \rightarrow \pi^{j+1}$.
10 $SendReader - Tag(Init, s_c, \pi^{j+1}) \rightarrow (R_{r1}^{j+1}, T_1^{j+1})$.
11 $SendTag((R_{r1}^{j+1}, T_1^{j+1}), vtag^c) \rightarrow (A^{j+1}, B^{j+1}, C_c^{j+1})$.
12 $SendReader((A^{j+1}, B^{j+1}, C_c^{j+1}), \pi^{j+2}) \rightarrow (J^{j+1})$.
Guessing phase:
13 Compute $(C_e^j, R_{r1}^{j+1}, s_e^{j+1}) \rightarrow C_e^{j+1}$.
14 Compare (C_e^{j+1}, C_c^{j+1}), if $C_e^{j+1} = C_c^{j+1}$, then $c = e$, else $c = |1 - e|$.
15 $Adv_A^{Forward-Untrace}(k) = 0 \leq \varepsilon$

□

On the one hand, the attacker computes the tag's keys s_j in the $(j+1)th$ session using the known cryptographic structures and the jth keys. Specifically, the common key-update parameter of the $(j+1)th$ key is R_t^j. If the attacker can compute R_t^j using the monitored jth and $(j+1)th$ messages, then she/he can calculate the tag output C_e^{j+1} and the $(j+1)$th keys. If the attacker cannot compute $(j+1)th$ outputs and keys, she/he cannot compare the computed value C_e^{j+1} with the monitored C_c^{j+1}. For example, due to the difficulty about the factor decomposed of great number n, the attacker cannot solve R_t^j using B^j and n. In addition, due to the one-way property of the hash function, the attacker cannot compute R_t^j. Therefore, the attacker cannot compute $(j+1)th$ key s_e^{j+1}. At last, the attacker cannot distinguish the target tag $vtag^e$ from $vtag^c$ in the $(j+1)th$ session. According to literature Qing et al. (2016), if the improved protocol achieves strong forward untracability, then it meets forward untracability definition.

Theorem 2 *The improved protocol can guarantee tag and reader anonymity under the weak attacker model.*

Proof This protocol prevents a tag's/reader's real identifier information from disclosure. In the improved protocol, the tag's identifier is transmitted in ciphertext. That is, $x = ID_k \oplus T_1 \oplus s_j \oplus R_t$, $A = (R_{r1}||x)^2 \ mod \ n$. Similarly, the reader's identifier is transferred by $D = (R_{r1}||y)^2 \ mod \ m$, where $y = RID_k \oplus T_1 \oplus R_{r2} \oplus x_j$. For the attacker, it is difficult to solve the tag's and reader's identifier from message A, D due to the difficulty about the factor decomposed of great number. To sum up, the improved protocol can guarantee tag and reader anonymity effectively. □

Theorem 3 *Under the weak attacker model, the improved protocol can resist de-synchronization attack.*

Proof An attacker can implement de-synchronization attack between tag and server through following three ways: (1) in order to cause de-synchronization attack, the server updates keys using the wrong parameters R_t by tampering with the value of B. In the first way, if the attacker tampers with B, it will cause the equation of $ID_k? = x \oplus T_1 \oplus s_j \oplus R_{tm}$ does not hold. Therefore, the server will terminate the protocol. (2) The attacker blocks the messages $\langle J \rangle$ from reader to tag, so the tag cannot update the key. However, due to that the server stores updated and previous keys s_j and s_{j-1}, the tag can still be authenticated by the backend server with key s_{j-1}. (3) The attacker prompts the server to update the tag's keys twice by the tag impersonation attack. In our protocol, it is quite difficult for an attacker to forge the validation message $\langle A, B, C \rangle$ without the knowledge of ID_k, s_j and n. It means that the attacker cannot easily produce a set of fake information from the tag that can be verified by the reader. Similarly, an attacker cannot implement de-synchronization attack between reader and server. When the attacker tamper with E to make the server updates key x_j by using the wrong parameters R_{r2}, the server will find that the equation of $RID_k = y \oplus T_1 \oplus R_{r2m} \oplus x_j$ does not hold and terminate the protocol. Because the server stores the updated and previous keys x_j and x_{j-1}, even if the attacker blocks the messages $\langle H, I, J \rangle$ from server to reader to prevent the reader from updating the key, the reader can still be authenticated by the backend server with key x_{j-1}. In addition, the attacker cannot prompts the server to update the reader's keys twice by the reader impersonation attack, because it is quite difficult for an attacker to forge the validation message $\langle D, E, F, G \rangle$ without the knowledge of RID_k, x_j and m. In summary, the improved protocol can resist de-synchronization attack between server and reader as well as server and tag. □

Theorem 4 *The improved protocol can resist replay attack under the weak attacker model.*

Proof In order to resist replay attack, the reader adds timestamp T_1 to its response message $y = RID_k \oplus T_1 \oplus R_{r2} \oplus x_j$, $G = h(RID_k||R_{r1}||A||T_1)$. After receiving the reader's message, the server first checks whether transmission delay is within the setting threshold. If the delay exceeds this threshold, the protocol will be terminated. Assuming that the attacker eavesdrops the messages $\langle R_{r1}, T_1 \rangle$, $\langle A, B, C \rangle$, $\langle A, B, C, T_1, D, E, F, G, R_{r1} \rangle$, $\langle H, I, J \rangle$ and $\langle J \rangle$ in

the *jth* session. In $(j+1)th$ session, the attacker replays messages $\langle R_{r1}, T_1 \rangle$ to the tag. After receiving the message, the tag computes and transmits $\langle A', B', C' \rangle$ to the attacker. Then, the attacker transfers composite messages $\langle A', B', C', T_1, D, E, F, G, R_{r1} \rangle$ to the backend server. Once receiving the message from the reader, the server first check if transmission delay is within the given threshold, that is $T_2' - T_1 < \Delta T$. Because T_1 is the timestamp of previous session, the protocol will be terminated due to the delay is greater than the given threshold. Assuming that the attacker updates the timestamp so that the delay is less than the threshold, the attacker still cannot calculate correct message D, G without the knowledge of reader identifier RID_k and secret x_j. Therefore, the server fails to authenticate $RID_k? = y \oplus T_1' \oplus R_{r2m} \oplus x_j$ and $h(RID_k||R_{r1}||A||T_1')? = G$. Similarly, if the attacker replays the message $\langle A, B, C, T_1, D, E, F, G, R_{r1} \rangle$ to the server, the protocol will be terminated due to $T_2' - T_1 > \Delta T$. Finally, if the attacker replays message $\langle A, B, C \rangle$ to the reader after receiving the message $\langle R_{r1}', T_1' \rangle$ in the $(j+1)th$ session. The reader will compute and send message $\langle A, B, C, T_1', D', E', F', G', R_{r1}' \rangle$ to the backend server. Although this formula of $T_2' - T_1' < \Delta T$ is true, the server cannot authenticate the tag successfully. Because R_{r1} used in A, C is generated in the jth session, the server cannot decode correct x, s_j from A, C using R_{r1}',which causes $ID_k \neq x \oplus T_1' \oplus s_j \oplus R_{tm}$. In summary, the improved protocol can resist the replay attack. □

Theorem 5 *The improved protocol can ensure data integrity under the weak attacker model*

Proof In the improved protocol, the server sends the tag data *Data* to the reader via $H = Data \oplus h(x_j/x_{j-1} \oplus R_{r2} \oplus RID_k)$ and guarantees the integrity of *Data* through $I = h(Data||R_{r2})$. We assumes that an attacker tampered the message H into $H' = H \oplus \Delta$. The reader calculates $Data' = H' \oplus h(x_j \oplus R_{r2} \oplus RID_k) = Data \oplus \Delta$ when it receives the messages $\langle H', I \rangle$. At this point, $I^* = H(Data'||R_{r2})$, so $I^* \neq I$. The reader will be aware of that the message has been tampered with and terminate the protocol. In addition, it is impossible for the attacker to calculate correct I' corresponding to H' without the value of R_{r2}. The above analysis shows that the improved protocol can guarantee data integrity. □

Theorem 6 *The improved protocol can resist reader stolen/lost attack under the weak attacker model.*

Proof Without loss of generality, it can be assumed that the authenticated reader has losted. An adversary obtained reader and uses it to collect sensitive data of any tagged object. However, the adversary cannot successfully boot the reader due to the adversary has no knowledge of RID_k, RPW_k. It is computationally infeasible for the adversary to derive RID_k, RPW_k and x_j at the same time in polynomial time, so the attacker cannot boot the reader successfully. In addition, online boot reader testing can be defeated by limiting the number of failed boot requests. To sum up, even if the attacker owns the authenticated reader, it is still impossible to get the data of tagged object. Therefore, the improved protocol can resist reader stolen/lost attack. □

Theorem 7 *The improved protocol can resist impersonation attack under the weak attacker model*

Proof If the attacker wants to successfully impersonate the tag, the correct $\langle A, B, C \rangle$ must be calculated. However, the attacker cannot calculate the legitimate message to impersonate the tag without the knowledge of the key s_j and the identifier ID_k. Similarly, the adversary cannot compute correct $\langle D, E, F, G \rangle$ without known the identifier RID_k and the key x_j. Thus, the adversary cannot impersonate the reader successfully. In addition, through **Theorem** 4 we can know that the adversary cannot impersonate the tag and the reader by replaying the eavesdropped message. To sum up, the improved protocol is able to resist impersonation attack.

□

Theorem 8 *The improved protocol can guarantee mutual authentication under the weak attacker model.*

Proof In the improved protocol, the back-end server can verify the validity of the reader by checking whether computed $y \oplus T_1 \oplus R_{r2m} \oplus x_j$ is equals to RID_k in the database and further checking if computed $h(RID_k||R_{r1}||A||T_1)$ is equals to received G or not. In addition, the back-end server can verify the validity of the tag by checking whether there is the tag's record in the database that the tag's identifier ID_k equals to $x \oplus T_1 \oplus s_j \oplus R_{tm}$. If there is the tag's record, the backend server will use ID_k to retrieve the key s_j, s_{j-1} quickly and check if the solved s_j is equal to retrieved s_j or s_{j-1} or not. Thus, the backend server can authenticate the tag and the reader. In Step 5 of the improved protocol authentication phase, the back-end server replies the message $\langle H, I, J \rangle$ to the reader. Because only legitimate backend sever knows the reader's identifier RID_k and key x_j to compute correct message H, after receiving the message, the reader could verify the backend server by checking if $I^* = h(Data||R_{r2})? = I$ holds or not. In addition, Only the legitimate backend server possess the tag's identifier ID_k and key s_j, no one can retrieve ID_k, s_j, R_t and embed them into the message $\langle J \rangle$. Therefore, in Step 6 of the improved protocol, after receiving the reader's reply message $\langle J \rangle$, the tag could verify the backend server by checking if $J^* = h(s_j||ID_k||R_t||h(s_j \oplus R_t))? = J$ holds or not. Thus, if $I^* = I$ and $J^* = J$ hold, the reader and the tag convinced that the back-end server is a legal server.

Table 5 Security comparison

Performance	Ref. [5]	Ref. [7]	Ref. [10]	Ref. [12]	Ref. [18]	Ref. [19]	Our protocol
Strong forward untraceability	×	×	×	×	√	√	√
Replay attack	×	√	×	√	√	×	√
Data integrity	×	–	×	×	–	–	√
De-synchronization attack	√	×	×	×	√	√	√
Impersonation attack	×	×	√	×	√	√	√
Tag/reader anonymity	√	√	×	√	√	√	√
Reader stolen/lost attack	×	×	√	×	×	×	√
Mutual authentication	×	√	√	×	√	√	√

Table 6 Performance comparison

Protocol	Tag	Reader	Service	Complexity
Ref. [5]	$5P + 2R$	$1H + 1R$	$7P + 1H$	$O(n)$
Ref. [7]	$1P + 2H$	$2P + 1R + 2H$	–	$O(n)$
Ref. [10]	$1R + 5H$	$6H + 1R$	$6H$	$O(1)$
Ref. [12]	$2H + 1R$	$1R$	$3H$	$O(n)$
Ref. [18]	$1R + 4H + 2PUF$	$1R + 3H$	–	$O(1)$
Ref. [19]	$3MS + 3P$	$2P + 3MS + 1H$	$6SR + 1P$	$O(1)$
Our	$3MS + 1R + 2H$	$3MS + 1R + 5H$	$6SR + 6H$	$O(1)$

To sum up, the improved protocol could provide mutual authentication between backend server and reader as well as between backend server and tag. □

This paper compares the security of the improved protocol with the existing typical protocols (Alavi et al. 2015; Deng et al. 2014; Li et al. 2015; Srivastava et al. 2015; Akgün and Aglayan 2015; Doss et al. 2013), as shown in Table 5, Where "√" indicates safe, "×" means unsafe, "–" means not involved. Here, we suppose that the channels of reader-tag and reader-backend server are all insecure.

4.3 Performance analysis

To compare the computational costs, we defne some notations as follows: (1) H indicates a hash operation. (2) R indicates a random number generation operation. (3) P indicates Pseudo-random number generation operation. (4) PUF indicates a physical unclonable function. (5) MS indicates the modulo squaring operation. (6) SR indicates squaring root solving operation. (7) "–" means that the protocol does not involve the server. A comparison of the performance of related protocol (Alavi et al. 2015; Deng et al. 2014; Li et al. 2015; Srivastava et al. 2015; Akgün and Aglayan 2015; Doss et al. 2013) is shown in Table 6, which includes the computational cost of each participating entity and the complexity of identification tag. In the performance analysis, we ignore

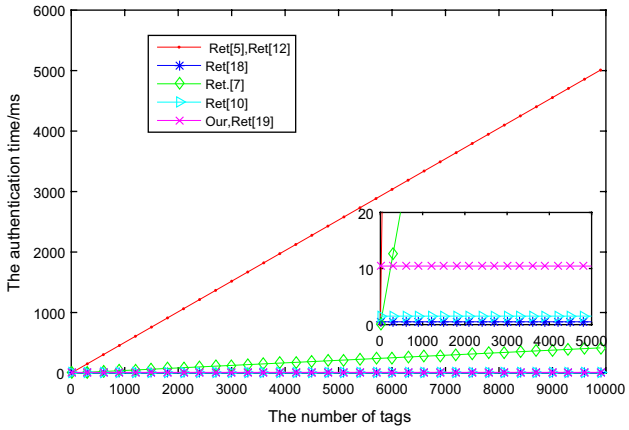

Fig. 2 The authentication process of improved protocol

XOR, concatenation and other light operations. The experimental simulation environment is Intel core i5-2.30 GHz, RAM-4 GB, and the programming language is Java. Because each experiment has subtle deviation, we test 30 times to take average value. The average time of executing hash function is 0.253 ms, pseudo-random function is 0.021 ms, PUF function is 0.053ms square root solving operation is 3.481 ms and the modulo squaring operation takes 1.896 ms.

This section compares the improved protocol with the existing protocol (Alavi et al. 2015; Deng et al. 2014; Li et al. 2015; Srivastava et al. 2015; Akgün and Aglayan 2015; Doss et al. 2013) in terms of identification tag efficiency. This paper compares the identification time of each protocol in the worst case. Fig. 2 shows that the variation trend of time cost each protocol spends on authentication tag as the number of database tags increases from 0 to 10000. From Fig. 2, we can observe that the authentication time of (Alavi et al. 2015), Alavi et al. (2015) and Deng et al. (2014) protocol increase linearly as the number of tags growth while literature (Li et al. 2015; Akgün and Aglayan 2015) and our protocol have constant-time identification. The protocol of Alavi et al. (2015) and Srivastava et al. (2015) has the fastest growth rate as a result of hash traversal method, and Alavi

et al. (2015) protocol uses pseudo-random function traversal so the growth rate is relatively slow. Akgün and Aglayan (2015) protocol has minimum authentication time because it use only one master key shared by all tags. The reader just need to perform hash function operation twice to complete the authentication of the tag. The main shortage of the protocol using one master key to keep constant-time identification is that the privacy/security will not be guaranteed as soon as one tag is compromised. So, Akgün et al. use PUFs to resistance against tag compromising attack. It is stated that the PUF is that it can produce fluctuating results based on the operating conditions. Thus, the large-scale implementation of PUF is yet to be a reality and remains an open problem (Sundaresan et al. 2015). Compared with (Akgün and Aglayan 2015) protocol, Li et al.'s protocol requires four more hash operations when authenticate the tag. The authentication time is 1.518 ms. However, it uses identifier index to ensure the constant-time identification and the tag/reader identifier are transmitted in plaintext in the protocol which result in the reader/tag anonymity cannot be guaranteed. This is insecure as an attacker may be able to clone a valid tag/reader. In our protocol, the server need perform three square root solving operations that is 10.443 ms to authenticate the tag. Although the authentication time of our protocol is slightly increased compared with Li et al. (2015) protocol, the authentication time of our protocol is still in constant level and we have greatly improved the security of the Li et al. (2015) protocol. In addition, Doss et al. (2013) protocol spend almost the same authentication time as our protocol, but Doss et al.'s protocol can't prevent replay attacks.

Here, we compare the total time spent of executing each protocol in the worst case when the number of tag is 10000. We ignore transmission time in communication time comparison and mainly consider the computation overheads on the sides of tag, reader and the back-end server. The time spent in Alavi et al. (2015) and Srivastava et al. (2015) protocol exceed five seconds that about are 5061.771 ms and 5062.024ms respectively. In the large scale RFID system with one million tags, it will take more than eight minutes to authenticate the tag, which is very inefficient. Although Deng et al. (2014) protocol also requires a linear search, it mainly uses pseudo-random function in the tag authentication process. The time spent of pseudo-random function is about one-tenth of the hash function, the time consuming is 421.075 ms and much less than Alavi et al.'s and Srivastava et al.'s protocol. However, in the large scale RFID system with one million tags, the communication time will up to 42 s. Our protocol, Akgün and Aglayan (2015) protocol and Li et al. (2015) protocol have the constant-time identification, so the time spent to authenticate a tag will not increase with the number of tags increased. Akgün et al.'s protocol spend least time that is 2.024 ms, but the large-scale implementation of PUF is yet to be a reality and remains an open problem. Li et al. (2015) protocol increased 1.9 times compared with Akgün et al.'s protocol (5.819 ms). Our protocol spend time is 5.132 times higher than Li et al.'s protocol. But our protocol overcomes the threats such as strong forward privacy, replay attack etc and our protocol has the constant-time identification which is suitable for large scale RFID system. Further, the modulo squaring is within the capabilities of EPC Class-1 Gen-2 tags and only a few hundred gates are required for implementing modular squaring operations (Doss et al. 2013), so the tag cost of improved protocol did not increase too much.

5 Conclusion future work

In this paper, we analyzes the security risks of Li et al.'s protocol and proposes a quadratic residue-based RFID authentication protocol for TMIS. This improved protocol sets the timestamp reasonably to prevent replay attacks, and verifies data integrity through the hash mapping result of data. This protocol does not transmit reader and tag identifier in plaintext to ensure anonymity. In addition, based on quadratic residue theory, the protocol guarantees strong forward untraceability of the improved protocol while at the same time meeting the purpose of constant time identification. Compared with Li et al.'s protocol, although our protocol has slightly increased in the time spent of executing the protocol, it strengthens the security and satisfies the requirements of strong privacy in TMIS.

In fact, we exploits the characteristics of public key encryption of quadratic residue theory to achieve strong forward privacy and fast authentication on the server side. Compared with symmetric encryption systems, public key encryption systems have higher security, but high computational cost limits the application of public key encryption primitives in RFID systems. As we all know, the sensitivity of medical data requires the medical system to achieve strong privacy protection. In the future work, we will consider using a public key encryption mechanism to design a protocol and comprehensively improve the security of TMIS while meeting the resource limitations of RFID computing costs. In addition, we hope to implementation of the proposed scheme to design TMIS.

Compliance with ethical standards

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

Akgün M, Aglayan MU (2015) Providing destructive privacy and scalability in rfid systems using pufs. Ad Hoc Netw 32(C):32–42. https://doi.org/10.1016/j.adhoc.2015.02.001

Alavi SM, Baghery K, Abdolmaleki B, Aref MR (2015) Traceability analysis of recent rfid authentication protocols. Wirel Pers Commun 83(3):1663–1682. https://doi.org/10.1007/s11277-015-2469-0

Amiribesheli M, Benmansour A, Bouchachia A (2015) A review of smart homes in healthcare. J Ambient Intell Hum Comput 6(4):495–517. https://doi.org/10.1007/s12652-015-0270-2

Avoine G, Bingol MA, Carpent X, Yalcin SBO (2013) Privacy-friendly authentication in rfid systems: On sublinear protocols based on symmetric-key cryptography. IEEE Trans Mobile Comput 12(10):2037–2049. https://doi.org/10.1109/TMC.2012.174

Avoine G, Buttyant L, Holczer T, Vajda I (2007) Group-based private authentication. In: IEEE International symposium on a world of wireless, mobile and multimedia networks, pp 1–6, https://doi.org/10.1109/WOWMOM.2007.4351808

Chen X, Doss R, Zhai J (2016) Rfid ownership transfer protocol based on cloud. Comput Netw 105(C):47–59. https://doi.org/10.1016/j.comnet.2016.05.017

Cho JS, Jeong YS, Sang OP (2015) Consideration on the brute-force attack cost and retrieval cost: A hash-based radio-frequency identification (rfid) tag mutual authentication protocol. Comput Math Appl 69(1):58–65. https://doi.org/10.1016/j.camwa.2012.02.025

Dehkordi MH, Farzaneh Y (2014) Improvement of the hash-based rfid mutual authentication protocol. Wirel Pers Commun 75(1):219–232. https://doi.org/10.1007/s11277-013-1358-7

Deng G, Zhang Y, Wang J (2013) Tree-lshb: an lpn-based lightweight mutual authentication rfid protocol. Wirel Pers Commun 72(1):159–174. https://doi.org/10.1007/s11277-013-1006-2

Deng M, Yang W, Zhu W (2014) Weakness in a serverless authentication protocol for radio frequency identification. Springer International Publishing, New York. https://doi.org/10.1007/978-3-319-01273-5_119

Doss R, Sundaresan S, Zhou W (2013) A practical quadratic residues based scheme for authentication and privacy in mobile rfid systems. Ad Hoc Netw 11(1):383–396. https://doi.org/10.1016/j.adhoc.2012.06.015

Hoque ME, Rahman F, Ahamed SI, Park JH (2010) Enhancing privacy and security of rfid system with serverless authentication and search protocols in pervasive environments. Wirel Pers Commun 55(1):65–79. https://doi.org/10.1007/s11277-009-9786-0

Jannati H, Bahrak B (2016) Security analysis of an rfid tag search protocol. Inform Process Lett 116(10):618–622. https://doi.org/10.1016/j.ipl.2016.05.001

Kaul SD, Awasthi AK (2013) RFID authentication protocol to enhance patient medication safety. Plenum Press, New York. https://doi.org/10.1007/s10916-013-9979-7

Li CT, Weng CY, Lee CC (2015) A secure rfid tag authentication protocol with privacy preserving in telecare medicine information system. J Med Syst 39(8):1–8. https://doi.org/10.1007/s10916-015-0260-0

Li T, Luo W, Mo Z, Chen S (2012) Privacy-preserving rfid authentication based on cryptographical encoding. In: IEEE INFOCOM, pp 2174–2182. https://doi.org/10.1109/INFCOM.2012.6195601

Malasinghe LP, Ramzan N, Dahal K (2017) Remote patient monitoring: a comprehensive study. J Ambient Intell Hum Comput 10(4):1–20. https://doi.org/10.1007/s12652-017-0598-x

Mohammedi M, Omar M, Bouabdallah A (2017) Secure and lightweight remote patient authentication scheme with biometric inputs for mobile healthcare environments. J Ambient Intell Hum Comput 80(10):1–13. https://doi.org/10.1007/s12652-017-0574-5

Pokala JP, Reddy CM, Abdul JS, Bapana S, Vorugunti CS (2016) A secure rfid protocol for telecare medicine information systems using ecc. In: International conference on wireless communications, signal processing and networking, pp 2295–2300. https://doi.org/10.1109/WiSPNET.2016.7566552

Poncela A, Coslado F, Garca B, Fernndez M, Ariza J, Peinado G, Demetrio C, Sandoval F (2018) Smart care home system: a platform for eassistance. J Ambient Intell Hum Comput. https://doi.org/10.1007/s12652-018-0979-9

Qing MA, Guo Y, Zeng Q, Duo XU (2016) A new ultra-lightweight RFID mutual authentication protocol. Netinfo Secur 16(5):44–50. https://doi.org/10.3969/j.issn.1671-1122.2016.05.007

Rahman F, Bhuiyan MZA, Ahamed SI (2016a) A privacy preserving framework for rfid based healthcare systems. Future Gener Comput Syst. https://doi.org/10.1016/j.future.2016.06.001

Rahman F, Hoque ME, Ahamed SI (2016b) Anonpri: A secure anonymous private authentication protocol for rfid systems. Inform Sci. **379**(10), https://doi.org/10.1016/j.ins.2016.07.038

Sareen S, Sood SK, Gupta SK (2016) Iot-based cloud framework to control ebola virus outbreak. J Ambient Intell Hum Comput 9(12):1–18. https://doi.org/10.1007/s12652-016-0427-7

Srivastava K, Awasthi AK, Kaul SD, Mittal RC (2015) A hash based mutual rfid tag authentication protocol in telecare medicine information system. J Med Syst 39(1):153. https://doi.org/10.1007/s10916-014-0153-7

Su C, Santoso B, Li Y, Deng R, Huang X (2017) Universally composable rfid mutual authentication. IEEE Trans Dependable Secure Comput 14(1):83–94. https://doi.org/10.1109/TDSC.2015.2434376

Sundaresan S, Doss R, Piramuthu S, Zhou W (2015) Secure tag search in rfid systems using mobile readers. IEEE Trans Dependable Secure Comput 12(2):230–242. https://doi.org/10.1109/TDSC.2014.2302305

Wu F, Xu L, Kumari S, Li X (2017) A privacy-preserving and provable user authentication scheme for wireless sensor networks based on internet of things security. J Ambient Intell Hum Comput 8(1):101–116. https://doi.org/10.1007/s12652-016-0345-8

Wu F, Xu L, Kumari S, Li X, Das AK, Shen J (2018) A lightweight and anonymous rfid tag authentication protocol with cloud assistance for e-healthcare applications. J Ambient Intell Hum Comput 9(4):919–930. https://doi.org/10.1007/s12652-017-0485-5