

Received January 3, 2019, accepted January 20, 2019, date of publication February 7, 2019, date of current version March 5, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2896641

# A New Secure Authentication Protocol for Telecare Medicine Information System and Smart Campus

MASOUMEH SAFKHANI<sup>1</sup> AND ATHANASIOS VASILAKOS<sup>2</sup>, (Senior Member, IEEE)

<sup>1</sup>Department of Computer Engineering, Shahid Rajaei Teacher Training University, Tehran 1678815811, Iran

<sup>2</sup>Department of Computer Science, Electrical and Space Engineering, Lulea University of Technology, 93187 Skellefteå, Sweden

Corresponding author: Athanasios Vasilakos (athanasios.vasilakos@ltu.se)

**ABSTRACT** Telecare Medicine Information System (TMIS)'s security importance attracts a lot of attention these days. Whatever the security of TMIS improves, its application becomes wider. To address this requirement, recently, Li *et al.* proposed a new privacy-preserving RFID authentication protocol for TMIS. After that, Zhou *et al.* and also Benssalah *et al.* presented their scheme, which is not secure, and they presented their new authentication protocol and claim that their proposal can provide higher security for TMIS applications. In this stream, Zheng *et al.* proposed a novel authentication protocol with application in smart campus, including TMIS. In this paper, we present an efficient impersonation and replay attacks against Zheng *et al.* with the success probability of 1 and a desynchronization attack which is applicable against all of the rest three mentioned protocols with the success probability of  $1 - 2^{-n}$ , where  $n$  is the protocols parameters length. After that, we proposed a new protocol despite these protocols can resist the attacks presented in this paper and also other active and passive attacks. Our proposed protocol's security is also done both informally and formally through the Scyther tool.

**INDEX TERMS** Telecare medicine information system, security, authentication, residue theorem, hash function.

## I. INTRODUCTION

One of the applications of Radio Frequency Identification (RFID) is in e-health and medical care systems. RFID is a technology for the identification of objects using radio waves. Three main components of an RFID system are tags, readers and a database for the access and authentication management. There are three different types of tags mainly attending to the signal's distance and its price: active, semi-passive and passive. Active tags are the most expensive and are able to establish a connection within a reader by itself. Semi-passive tags can either, use the power of a reader's signal or communicate within a reader without the reader's signal; while passive tags are the cheapest and need a reader's signal to communicate with it. The connection between the tag and the reader is generally considered as wireless and insecure while the connection between the reader and the server/database could be a permanent connection based on for example fiber optic as the media or wireless and insecure. In the latter case,

The associate editor coordinating the review of this manuscript and approving it for publication was Sabah Mohammed.

the reader is called mobile. A mobile reader provides more flexibility and could be used in a wider variety of applications. A protocol that supports a mobile reader could be appropriate for Telecare Medicine Information System (TMIS) where for example a nurse should move from a patient to another patient and track their critical data. However, the security of such a protocol could be very vital due to the patient safety and the sensitivity of his/her personal data that should not be revealed to any unauthorized party.

## A. RELATED WORK AND MOTIVATION

In the field of health care, similar to other applications, the privacy of the users and the security of their information is a very critical issue. The user in this application could be a doctor, a nurse or a patient and their information could vary from their personal data to the history of their medical services and so on. New technologies, such as RFID and the Internet of things (IoT) can improve the quality and the speed of a medical service which is provided for a patient and also provide enough information for medical service providers

to provide better service for their patients. However, as it has been already mentioned, besides these benefits any new technology has its own risk and concerns. For RFID and IoT, illegal access to the system information is a potential risk which can cause many problems varies from compromising the user's privacy to given wrong medical service to a patient which lead to his/her death. Hence, during the last decade, many researchers tried to address these concerns by providing various protocols for access control, authentication and key management in such applications, e.g. [1]–[8], [8]–[11], [11]–[13]. However, the later analyzes performed in the field of security analysis of those protocols [14]–[19] show that the research community has not yet reached a comprehensive secure protocol. In this paper, we target the security analysis of some recent protocols in this category.

To address these concerns, Srivastava *et al.* [20] proposed a hash-based mutual RFID authentication protocol in Telecare Medicine Information System (TMIS) and claimed security against active and passive attacks such as forgery, traceability, replay, and desynchronization attack. However, later in [21], Li *et al.* pointed out the security vulnerabilities of Srivastava *et al.* [20] including having a weak login phase which increases the adversary's advantage to a successful login, its weakness against the stolen/lost reader and also having low efficiency. To remedy these flaws, they also presented a new protocol. Benssalah *et al.* [28] showed the security flaws of Li *et al.* protocol including its vulnerability against desynchronization and impersonation attacks and also not ensuring the protocol's transferred messages integrity and also data privacy and also proposed an improved version. In addition, Zhou *et al.* [22] showed weaknesses of Li *et al.* protocol and proposed a protocol, used residue theorem and also hash functions as building blocks of their proposed proposal and claimed to be suitable for TMIS applications. Mir and Nikooghadam [23] tried to employ biometric in their authentication with key agreement protocol for TMIS. Later, Abbasinezhad-Mood and Nikooghadam [24] tried to provide data protection in TMIS using elliptic curve based cryptography (ECC) to employ. However, it is yet may not be possible to implement such a cryptosystem in a passive RFID tag with very constrained environments. Very recently, Tan [25] presented a new secure delegation-based authentication protocol by using the identity-based cryptography for TMIS. In addition, Li *et al.* [26] proposed a chaotic map-based remote authentication scheme for TMIS. In order to address the need for the security protocols in smart campus including TMIS, Zheng *et al.* [27] also proposed a new authentication scheme.

## B. MAIN CONTRIBUTIONS

The main contribution of this paper is two folds. At first, we consider the security of some of those protocols that are proposed for mobile readers and show that they do not provide the desired security. More precisely, we show that Zheng *et al.* [27] protocol suffers from replay attacks. In addition, assuming that the adversary can control the time setting

TABLE 1. Notations used in the protocols' description.

Notations	Description
$\mathcal{T}$	An RFID tag
$\mathcal{R}$	An RFID Reader
$\mathcal{A}$	The Adversary
$ID_{k_j}$	The identifier of the $k^{th}$ tag in the $j^{th}$ session
$s_j, s_{j-1}$	The tag's current and old secret values respectively
$RID_k$	The identifier of the $k^{th}$ reader
$RID'_k$	The database's record from reader's identifier
$PRW_k$	The password of the $k^{th}$ reader
$x_j$	The reader's secret value
$x_{j-1}$	The reader's secret value in the previous session
$T_i$	A time stamp
$\Delta T$	The acceptable time delay
$\Delta T'$	The record of database for the time interval
$p, q, g, h$	The four prime numbers as private keys
$n = p.q, m = g.h$	The public keys which are stored in the reader and tags
$R_{ri}$	A random number which is generated by the reader
$R_t$	The random number which is generated by the tag
$R_s$	The random number which is generated by the server
$\oplus$	The exclusive or operation
$h(.)$	The one-way has function
$\parallel$	The concatenation operation

of the readers, then it is possible to apply desynchronization attack against Li *et al.* [21] protocol and its successors.

Secondly, we propose a secure authentication protocol for mobile readers that only uses a hash function and bitwise XOR operation, which is more realistic to be implemented in a passive tag than other proposals such as Elliptic Curve Cryptography (ECC) and other public key based solutions. Moreover, we analyze its security against various attacks in the context such as desynchronization, traceability, secret disclosure, and impersonation attacks and show its security against these attacks. We also verify the correctness of our protocol using a formal approach through Scyther tool.

## C. PAPER ORGANIZATION

The rest of the paper is structured as follows: In Section II, we briefly review Zheng *et al.* protocol, Li *et al.* protocol, Benssalah *et al.* protocol and Zhou *et al.* protocol respectively. Our proposed attacks against these protocols are explained in Section III. Section IV and Section V describe

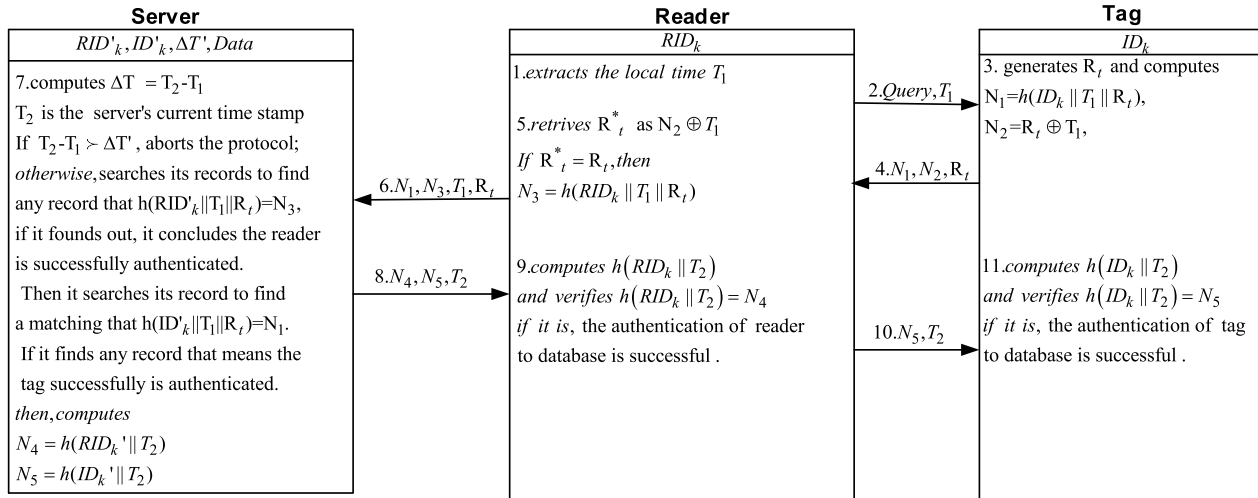


FIGURE 1. The Zheng *et al.*'s hash based mobile RFID mutual authentication protocol [27].

our new proposed protocol and its security proof respectively and finally this paper concludes in Section VI.

## II. PRELIMINARIES

The notations which are used in this paper are represented in Table 1.

### A. ZHENG *et al.* PROTOCOL FOR SMART CAMPUS

To address the need for the security protocols in smart campus including TMIS, recently, Zheng *et al.* [27] proposed a new authentication protocol represented in Fig. 1 and proceeds as below:

- 1) The reader starts a session and sends *Query* message along with its local timestamp  $T_1$  to the tag.
- 2) Once the tag receives the message, it generates a random number  $R_t$  and computes  $N_1 = h(ID_k || T_1 || R_t)$ ,  $N_2 = R_t \oplus T_1$  and sends them along  $R_t$  to the reader.
- 3) When the reader receives the message, it computes  $R_t^* = N_2 \oplus T_1$  and verifies whether  $R_t^* \stackrel{?}{=} R_t$  is or not. If it is not, aborts the protocol otherwise computes  $N_3 = h(RID_k || T_1 || R_t)$  and sends it along with  $N_1$ ,  $T_1$  and  $R_t$  to the database.
- 4) Upon receipt the message, the database retrieves its local time  $T_2$  and computes the time interval  $\Delta T$ , if  $\Delta T > \Delta T'$ , it concludes there is an attack and aborts the protocol, otherwise searches its records to find any record that  $h(RID'_k || T_1 || R_t) = N_3$ , if it finds out a matching record, it concludes the reader is successfully authenticated. Then it searches its record to find a matching that  $h(ID'_k || T_1 || R_t) = N_1$ . If it finds any record that means the tag is successfully authenticated. After that it computes  $N_4 = h(RID_k || T_2)$ ,  $N_5 = h(ID_k || T_2)$  and sends them along with  $T_2$  to the reader.
- 5) Once receipt the message, the reader using received  $T_2$ , computes  $h(RID_k || T_2)$  and verifies whether  $h(RID_k || T_2) \stackrel{?}{=} N_4$  is or not. If it is not, aborts the

protocol otherwise the reader's authentication for the database has successfully done and sends  $N_5$  along with  $T_2$  to the tag.

- 6) When receives the message, the tag using received  $T_2$ , computes  $h(ID_k || T_2)$  and verifies whether  $h(ID_k || T_2) \stackrel{?}{=} N_5$  is or not. If it is not, aborts the protocol otherwise the tag's authentication to the database has successfully done and so the protocol ends.

### B. LI *et al.* PROTOCOL FOR TMIS

Li *et al.* [21] pointed out the security vulnerabilities of Srivastava *et al.* [20] including having a weak login phase which increases the adversary's advantage to successful login, its weakness against the stolen/lost reader and also having low efficiency. To remedy these flaws, they presented a new protocol which as illustrated in Fig. 2, proceeds as below in two phases:

#### 1) BOOT READER PHASE

Since Li *et al.* designed their protocol to be employed in Telecare Medicine Information System, take in to account the boot reader phase. To boot the reader, the TMIS staff must:

- 1) input the reader identifier  $RID_k$  and the reader password  $RPW_k$ ;
- 2) The reader computes  $V'_k = W_K \oplus RID_k \oplus RPW_k$ , and checks whether  $V'_k \stackrel{?}{=} V_k$  is or not. If it holds, the reader successfully booted otherwise the reader stops the protocol process.

#### 2) AUTHENTICATION PHASE

- 1) The reader starts this phase by generating random number  $R_r$  and computing  $A = V'_k \oplus R_r$ ,  $B = h(V'_k \oplus T_1 \oplus R_r)$  and sending them along time stamp  $T_1$  to the tag;
- 2) Once the tag received the message, it:

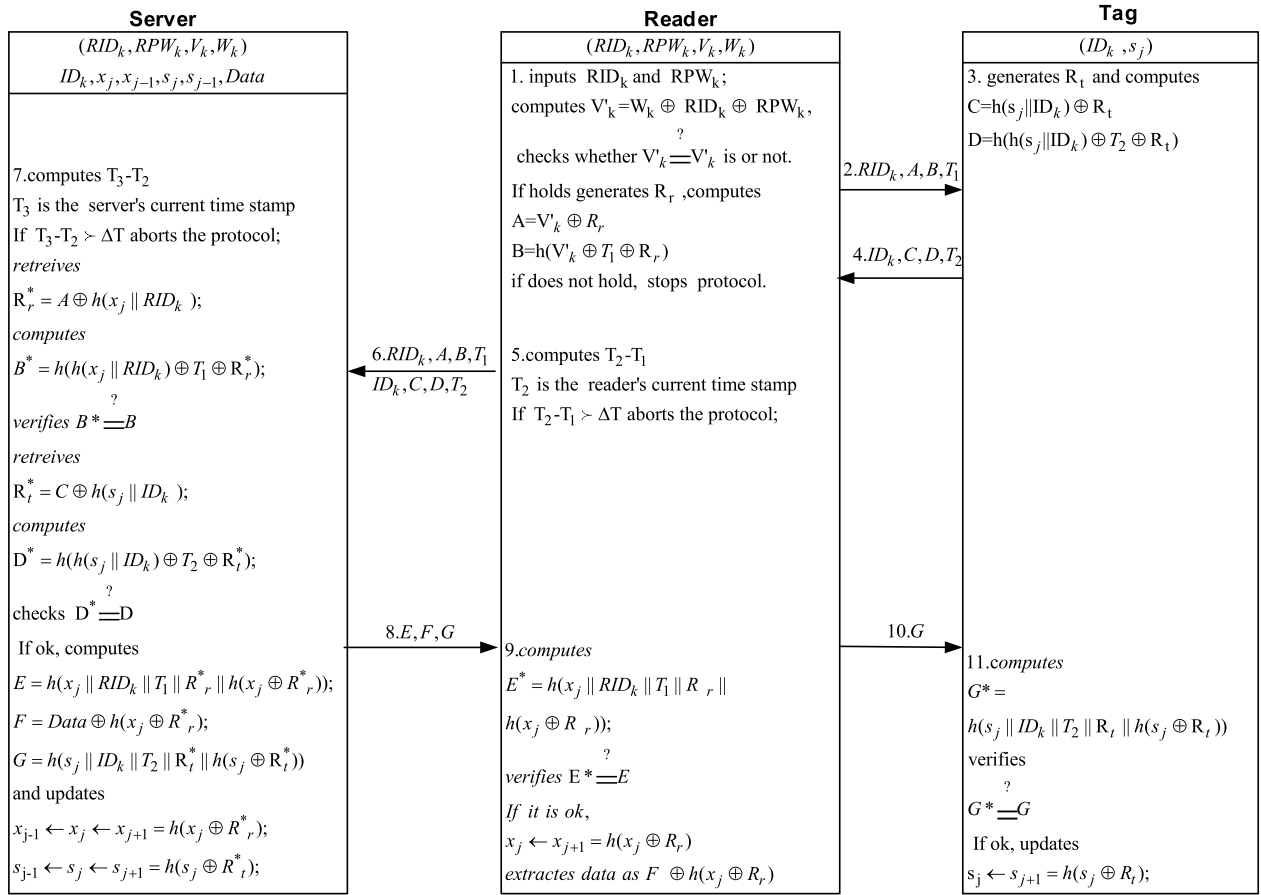


FIGURE 2. The Li et al.'s hash based mobile RFID mutual authentication protocol [21].

- generates another random number  $R_t$  and computes  $C = h(s_j || ID_k) \oplus R_t$ ,  $D = h(h(s_j || ID_k) \oplus T_2 \oplus R_t)$ ;
  - and sends  $ID_k$ ,  $C$ ,  $D$  and  $T_2$  to the reader.
- 3) Upon receipt the message, the reader:
- computes  $T_1 - T_2$  where  $T_1$  is the reader's current time stamp and checks whether  $T_1 - T_2 \leq \Delta T$  is or not. If it is not, it aborts the protocol process and otherwise sends  $RID_k$ ,  $A$ ,  $B$ ,  $T_1$ ,  $ID_k$ ,  $C$ ,  $D$  and  $T_2$  to the server.
- 4) When the server receives the message, computes  $T_3 - T_2$  where  $T_3$  is the server's current timestamp and checks whether  $T_3 - T_2 \leq \Delta T$  is or not. If it is not, the server aborts the protocol process and otherwise continues as below;
- retrieves  $R_r^*$  as  $R_r^* = A \oplus h(x_j || RID_k)$ ;
  - computes  $B^* = h(h(x_j || RID_k) \oplus T_1 \oplus R_r^*)$  and checks whether  $B^* \stackrel{?}{=} B$ , if it does not hold, stops the protocol otherwise continues as below;
  - retrieves  $R_t^* = C \oplus h(s_j || ID_k)$ ;
  - computes  $D^* = h(h(s_j || ID_k) \oplus T_2 \oplus R_t^*)$  and checks whether  $D^* \stackrel{?}{=} D$ , if it does not hold, stops the protocol otherwise continues as below.
- computes  $E = h(x_j || RID_k || T_1 || R_r^* || h(x_j \oplus R_r^*))$ ,  $F = Data \oplus h(x_j \oplus R_r^*)$ ,  $G = h(s_j || ID_k || T_2 || R_t^* || h(s_j \oplus R_t^*))$  and updates its records as follows;
  - updates  $x_j$  and  $x_{j-1}$  as  $x_{j+1} = h(x_j \oplus R_r^*)$  and  $x_j$  respectively;
  - updates  $s_j$  and  $s_{j-1}$  as  $s_{j+1} = h(s_j \oplus R_t^*)$  and  $s_j$  respectively;
  - and finally sends  $E$ ,  $F$  and  $G$  to the reader;
- 5) Once the reader receives the message, it:
- computes  $E^* = h(x_j || RID_k || T_1 || R_r || h(x_j \oplus R_r))$  and checks whether  $E^* \stackrel{?}{=} E$  is or not. If it is not stops the protocol's process otherwise the  $Data$ 's integrity successfully verified and it successfully authenticates the server;
  - obtains  $Data$  as  $F \oplus h(x_j \oplus R_r)$ ;
  - updates its related records i.e.  $x_j$  as  $x_{j+1} = h(x_j \oplus R_r)$ ;
  - and sends  $G$  to the tag.
- 6) Upon receipt the message, the tag computes  $G^* = h(s_j || ID_k || T_2 || R_t || h(s_j \oplus R_t))$  and checks whether  $G^* \stackrel{?}{=} G$ . If it is not, stops the protocol process otherwise successfully authenticates the reader and updated its  $s_j$  to  $s_{j+1} = h(s_j \oplus R_t)$ .

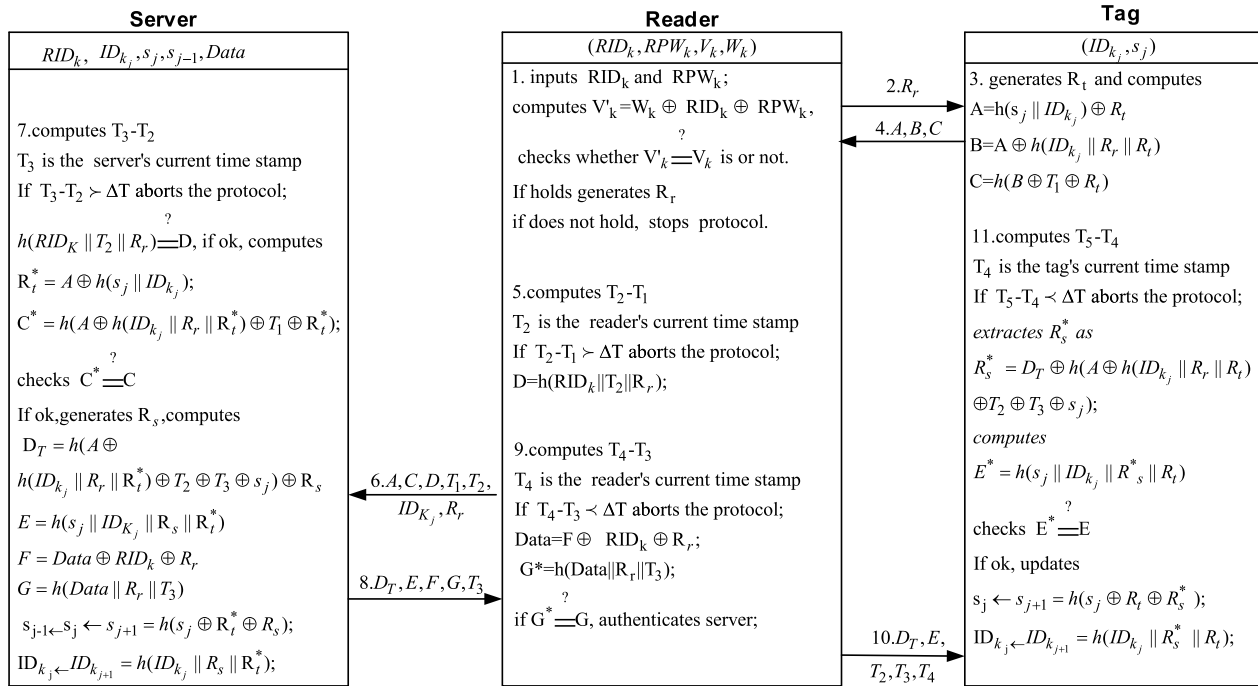


FIGURE 3. The Benssalah *et al.*'s hash based mobile RFID mutual authentication protocol [28].

### C. BENSSALAH *et al.* PROTOCOL FOR TMIS

Benssalah *et al.* [28] showed the security flaws of Li *et al.* protocol including its vulnerability against desynchronization and impersonation attacks and also not ensuring the protocol's transferred messages integrity and also data privacy. All of these weaknesses lead to not employing this protocol in TMIS systems. They also tried to propose a new improved version of the Li *et al.*'s protocol and claim their protocol have a suitable level of security. As illustrated in Fig. 3, their protocol works as follows in two phases:

#### 1) BOOT READER PHASE

Since Benssalah *et al.* designed their protocol to be employed in Telecare Medicine Information System, take in to account the boot reader phase. To boot the reader, same as Li *et al.* protocol, the TMIS staff must:

- input the reader identifier  $RID_k$  and reader password  $RPW_k$ ;
- the reader computes  $V'_k = W_k \oplus RID_k \oplus RPW_k$ , and checks whether  $V'_k \stackrel{?}{=} V_k$  is or not. If it holds the reader successfully booted otherwise the reader stops the protocol process.

#### 2) AUTHENTICATION PHASE

- 1) The reader starts this phase by generating random number  $R_r$  and sending it to the tag;
- 2) Once the tag received the message, it:
  - generates another random number  $R_t$  and computes  $A = h(s_j || ID_{k_j}) \oplus R_t$ ,  $B = A \oplus h(ID_{k_j} || R_r || R_t)$ ,  $C = h(B \oplus T_1 \oplus R_t)$ ;
  - and sends  $A$ ,  $B$  and  $C$  to the reader.

- 3) Upon receipt the message, the reader:

- computes  $T_2 - T_1$  where  $T_2$  is the reader's current time stamp and checks whether  $T_2 - T_1 \leq \Delta T$  is or not. If it is not, the reader aborts the protocol process and otherwise computes  $D = h(RID_k || T_2 || R_r)$ ;
- and sends  $A$ ,  $C$ ,  $D$ ,  $T_1$ ,  $T_2$ ,  $ID_{k_j}$ ,  $R_r$  to the server.

- 4) When the server receives the message, computes  $T_3 - T_2$  where  $T_3$  is the server's current time stamp and checks whether  $T_3 - T_2 \leq \Delta T$  is or not. If it is not, the server aborts the protocol process and otherwise continues as below;

- computes  $D^* = h(RID_k || T_2 || R_r)$  and checks  $D^* \stackrel{?}{=} D$ , if it does not hold, stops the protocol otherwise continues as below;
- retrieves  $R_s^*$  as  $R_s^* = A \oplus h(s_j || ID_{k_j})$  and computes  $C^* = h(A \oplus h(ID_{k_j} || R_r || R_t^*) \oplus T_1 \oplus R_t^*)$ ;
- checks whether  $C^* \stackrel{?}{=} C$ , if it does not hold, stops the protocol otherwise continues as below;
- generates  $R_s$  and computes  $D_T = h(A \oplus h(ID_{k_j} || R_r || R_s^*) \oplus T_2 \oplus T_3 \oplus s_j) \oplus R_s$ ;  $E = h(s_j || ID_{k_j} || R_s || R_t^*)$ ;  $F = Data \oplus RID_k \oplus R_r$ ;  $G = h(Data || R_r || T_3)$ ;
- After successful tag and reader authentication, the server updates its related records from the reader and tags as follows:
  - updates  $s_j$  and  $s_{j-1}$  as  $s_{j+1} = h(s_j \oplus R_t^* \oplus R_s)$  and  $s_j$  respectively;
  - updates  $ID_{k_j}$  as  $ID_{k_{j+1}} = h(ID_{k_j} || R_s || R_t^*)$ ;
- and sends  $D_T$ ,  $E$ ,  $F$ ,  $G$  and  $T_3$  to the reader;



5) Once the reader receives the message, it:

- computes  $T_4 - T_3$  where  $T_4$  is the reader's current time stamp and checks whether  $T_4 - T_3 \leq \Delta T$  is or not. If it is not, the server aborts the protocol process and otherwise continues as below;
- obtains  $Data$  as  $F \oplus RID_k \oplus R_r$ ;
- computes  $G^* = h(Data \| R_r \| T_3)$  and checks whether  $G^* \stackrel{?}{=} G$  is or not. If it is not, stops the protocol's process otherwise the  $Data$ 's integrity successfully verified and it successfully authenticates the server;
- and sends  $D_T, E, T_2, T_3$  and  $T_4$  to the tag.

6) Upon receipt the message, the tag:

- computes  $T_5 - T_4$  where  $T_5$  is the tag's current time stamp and checks whether  $T_5 - T_4 \leq \Delta T$  is or not. If it is not, the server aborts the protocol process and otherwise continues as below;
- retrieves  $R_s^*$  as  $D_T \oplus h(A \oplus h(ID_{k_j} \| R_r \| R_t) \oplus T_2 \oplus T_3 \oplus s_j)$ ;
- computes  $E^* = h(s_j \| ID_{k_j} \| R_s^* \| R_t)$  and checks whether  $E^* \stackrel{?}{=} E$ . If it is not, stops the protocol process otherwise successfully authenticates the reader and updates its  $s_j$  to  $s_{j+1} = h(s_j \oplus R_t \oplus R_s^*)$  and its  $ID_{k_j}$  as  $ID_{k_{j+1}} = h(ID_{k_j} \| R_s^* \| R_t)$ .

#### D. ZHOU *et al.* PROTOCOL FOR TMIS

Zhou *et al.* [22] due to address the weaknesses of Li *et al.* protocol and also provide the strong privacy preservation for their proposed scheme which makes it suitable for TMIS applications, used residue theorem and also hash functions as building blocks of their proposed proposal. They also designed their protocol in such a way despite the Li *et al.* and also Benssalah *et al.* the reader identifier  $RID_k$  and also the tag's identifier  $ID_k$  have not been transferred in plain text form. Their proposed protocol as depicted in Fig. 4 proceeds in three phases of pre-phase, boot reader phase, and authentication phase.

##### 1) PRE-PHASE

In this phase, the protocol parties generate and store their related secrets. Precisely:

- The server generates and stores four prime numbers  $p, q, g$  and  $h$  as its private keys and  $n = p \cdot q$  and  $m = g \cdot h$  as its public keys.
- The tag generates and stores its identifier  $ID_k$  and its secret value  $s_j$ . Beside on this, the tag saves the  $n$  and shares  $(ID_k, s_j, s_{j-1})$  and the hash function  $h(\cdot)$  with the server.
- The reader stores its identifier  $RID_k$  and its secret value  $x_j$  and also  $V_k = h(x_j \| RID_k)$ ,  $W_k = h(x_j \| RID_k) \oplus RID_k \oplus RPW_k = V_k \oplus RID_k \oplus RPW_k$ ,  $m$  and  $h(\cdot)$  as one way hash function. The reader shares  $RID_k, x_j$  and  $x_{j-1}$  with the server.

##### 2) BOOT READER PHASE

Since Zhou *et al.* designed their protocol to be employed in Telecare Medicine Information System, take in to account the boot reader phase. To boot the reader, the TMIS staff must:

- input the reader identifier  $RID_k$  and the reader password  $RPW_k$ ;
- the reader computes  $V'_k = W_k \oplus RID_k \oplus RPW_k$ , and checks whether  $V'_k \stackrel{?}{=} V_k$  is or not. If it holds, the reader successfully booted otherwise the reader stops the protocol process.

##### 3) AUTHENTICATION PHASE

- 1) The reader starts this phase by generating a random number  $R_{r1}$  and sending it along time stamp  $T_1$  to the tag;
- 2) Once the tag received the message, it:
  - generates another random number  $R_t$  and computes  $x = ID_k \oplus T_1 \oplus s_j \oplus R_t$ ,  $A = (R_{r1} \| x)^2 \bmod n$ ,  $B = R_t^2 \bmod n$  and  $C = (R_{r1} \| s_j)^2 \bmod n$ ;
  - and sends  $A, B$  and  $C$  to the reader.
- 3) Upon receipt the message, the reader:
  - generates another random number  $R_{r2}$  and computes  $y = RID_k \oplus T_1 \oplus R_{r2} \oplus x_j$ ,  $D = (R_{r1} \| y)^2 \bmod m$ ,  $E = R_{r2}^2 \bmod m$ ,  $F = (R_{r1} \| x_j)^2 \bmod m$  and  $G = h(RID_k \| R_{r1} \| A \| T_1)$ ;
  - and sends  $A, B, C, T_1, D, E, F, R_{r1}$  and  $G$  to the server.
- 4) When the server receives the message, computes  $T_2 - T_1$  where  $T_2$  is the server's current timestamp and checks whether  $T_2 - T_1 \leq \Delta T$  is or not. If it is not, the server aborts the protocol process and otherwise continues as below:
  - by using Chinese Remainder Theorem (CRT) and  $g$  and  $h$  solves the equations  $F = (R_{r1} \| x_j)^2 \bmod m$ ,  $D = (R_{r1} \| y)^2 \bmod m$  and  $E = (R_{r2})^2 \bmod m$  and obtains  $x_j$  and  $y$  and  $(R_{r21}, R_{r22}, R_{r23}, R_{r24})$  where  $x_j$  and  $y$  are uniquely retrieved by using  $R_{r1}$  from four equations.
  - computes  $RID_k = y \oplus T_1 \oplus R_{r2m} \oplus x_j$  and searches the tags related records by using  $RID_k$  as index which in the worst case it only occurs four times.
  - If any related record is founded, checks whether  $h(RID_k \| R_{r1} \| A \| T_1) \stackrel{?}{=} G$ , if it holds the sever can obtain  $RID_k, x_j$  and  $x_{j-1}$  and checks whether  $x_j \stackrel{?}{=} x_j$  or  $x_{j-1}$ . If it holds, the server successfully authenticates the reader and otherwise it stops the authentication process.
  - by using Chinese Remainder Theorem (CRT), solves the equations  $C = (R_{r1} \| s_j)^2 \bmod n$ ,  $A = (R_{r1} \| x)^2 \bmod n$  and  $B = (R_t)^2 \bmod n$  and obtains  $x$  and  $s_j$  and  $(R_{t1}, R_{t2}, R_{t3}, R_{t4})$ ;

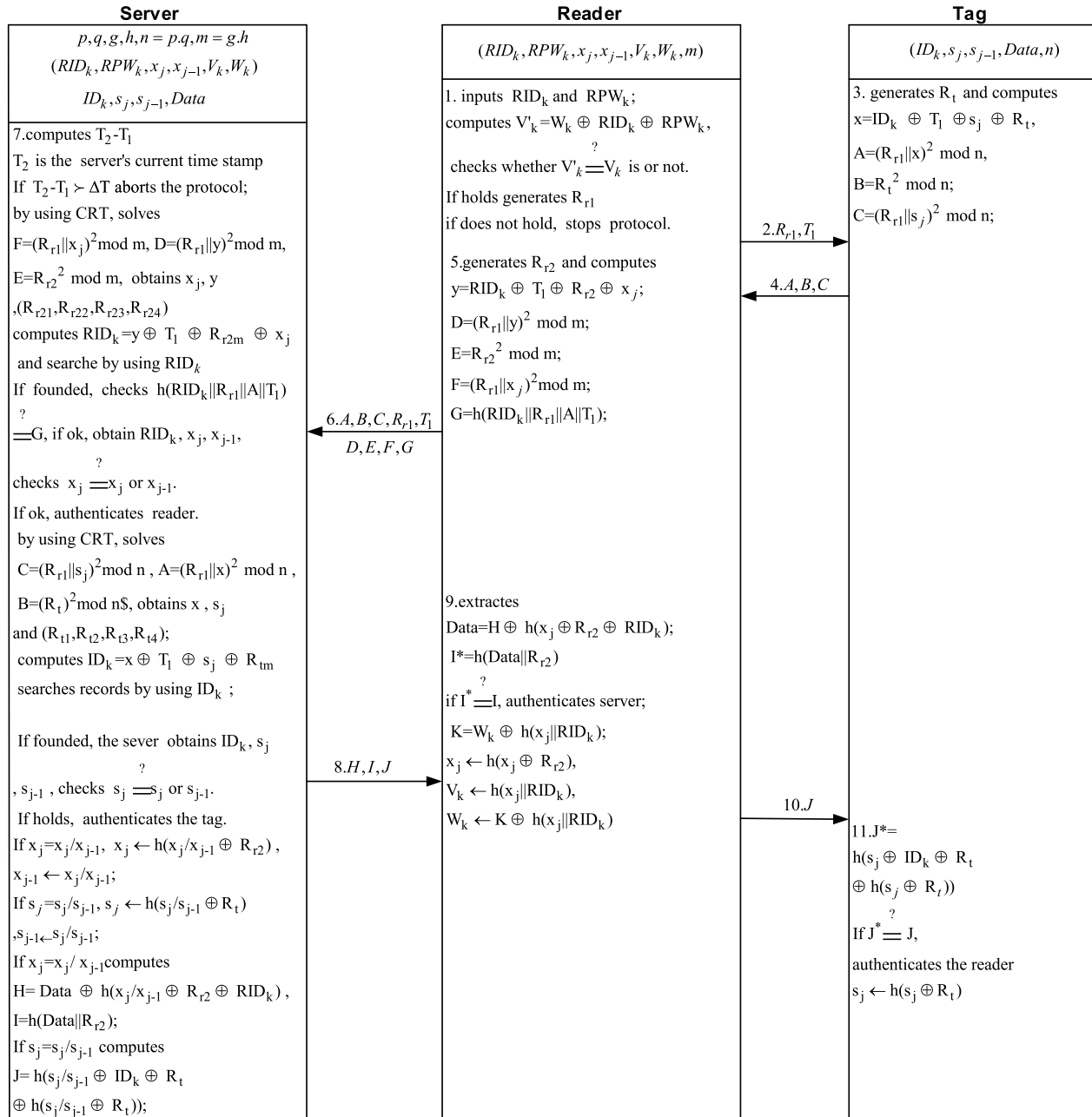


FIGURE 4. The Zhou et al.'s residue based TMIS authentication protocol [22].

- computes  $ID_k = x \oplus T_1 \oplus s_j \oplus R_{tm}$  and searches the tags related records by using  $ID_k$  as index which in the worst case it only occurs four times.
- If any related record is founded, the server can obtain  $ID_k, s_j$  and  $s_{j-1}$  and checks whether  $s_j \stackrel{?}{=} s_j$  or  $s_{j-1}$ . If it holds, the server successfully authenticates the tag and otherwise, it stops the authentication process.
- After successful tag and reader authentication, the server updates its related records from the reader and tags as follows:
  - If  $x_j = x_j$ , updates  $x_j$  and  $x_{j-1}$  as  $h(x_j \oplus R_{r2})$  and  $x_j$  respectively;
  - If  $x_j = x_{j-1}$ , updates  $x_j$  and  $x_{j-1}$  as  $h(x_{j-1} \oplus R_{r2})$  and  $x_{j-1}$  respectively;
  - If  $s_j = s_j$ , updates  $s_j$  and  $s_{j-1}$  as  $h(s_j \oplus R_t)$  and  $s_j$  respectively;
  - If  $s_j = s_{j-1}$ , updates  $s_j$  and  $s_{j-1}$  as  $h(s_{j-1} \oplus R_t)$  and  $s_{j-1}$  respectively;
- informs the related records of tag and the data of tag  $Data$  to the reader as follows:
  - if  $x_j = x_j$  computes  $H = Data \oplus h(x_j \oplus R_{r2} \oplus RID_k)$  and  $I = h(Data || R_{r2})$ ;

- if  $x_j = x_{j-1}$  computes  $H = Data \oplus h(x_{j-1} \oplus R_{r2} \oplus RID_k)$  and  $I = h(Data \parallel R_{r2})$ ;
  - if  $s_j = s_j$  computes  $J = h(s_j \oplus ID_k \oplus R_t \oplus h(s_j \oplus R_t))$ ;
  - if  $s_j = s_{j-1}$  computes  $J = h(s_{j-1} \oplus ID_k \oplus R_t \oplus h(s_{j-1} \oplus R_t))$ ;
  - and sends  $H$ ,  $I$  and  $J$  to the reader;
- 5) Once the reader receives the message, it:
- obtains  $Data$  as  $H \oplus h(x_j \oplus R_{r2} \oplus RID_k)$ ;
  - computes  $I^* = h(Data \parallel R_{r2})$  and checks whether  $I^* \stackrel{?}{=} I$  is or not. If it is not, the reader stops the protocol's process otherwise the  $Data$ 's integrity successfully verified and it successfully authenticates the server;
  - computes  $K = W_k \oplus h(x_j \parallel RID_k)$ ;
  - updates its related records i.e.  $x_j$ ,  $V_k$  and  $W_k$  as  $x_j = h(x_j \oplus R_{r2})$ ,  $V_k = h(x_j \parallel RID_k)$  and  $W_k = K \oplus h(x_j \parallel RID_k)$  respectively where  $x_j$  used them is updated.
  - and sends  $J$  to the tag.
- 6) Upon receipt the message, the tag computes  $J^* = h(s_j \oplus ID_k \oplus R_t \oplus h(s_j \oplus R_t))$  and checks whether  $J^* \stackrel{?}{=} J$ . If it is not, stops the protocol process otherwise successfully authenticates the reader and updates its  $s_j$  to  $h(s_j \oplus R_t)$ .

### III. SECURITY ANALYSIS OF THE PROTOCOLS

#### A. ADVERSARY MODEL

We consider an active adversary who can stand between the tag and the reader or the reader and the back-end server, given that the Zhou *et al.*'s protocol targets mobile readers and assumed that both the channels of reader-tag and reader-back-end server are all insecure [22, Sec. 4.2]. In addition, if it is required, we assume that the adversary can control the time stamp of the mobile reader, which is not an unrealistic assumption and it is possible to change the time setting of the mobile reader.

#### B. SECURITY ANALYSIS OF ZHENG *et al.* PROTOCOL

In the case of Zheng *et al.*'s protocol [27], the protocol's parties do not update shared values. Hence we cannot desynchronize them and it is better to try to impersonate a protocol party. An option is to impersonate the server and the reader toward the tag. The reason comes from the fact the tag has no local time and any time suggested from the reader is acceptable for the tag. In this protocol, any adversary can impersonate the reader to the tag as it is explained here. To end this, the adversary can eavesdrop transferred messages in a legitimate session between the tag, the reader and the server and stores tuple  $T_1, T_2, N_5 = h(ID_k' \parallel T_2)$ . Now, at any desired time, to impersonate the reader, the adversary sends a query with  $T_1$  to the tag. The tag sends its response as  $(N_1', N_2', R_t')$ , where  $N_1' = h(ID_k \parallel T_1 \parallel R_t')$ ,  $N_2' = R_t' \oplus T_1$  and  $R_t'$  is a fresh random value generated by the tag for this session. Now the adversary returns the stored  $N_5 = h(ID_k' \parallel T_2)$  and  $T_2$  to the

tag and will be definitely authenticated by the adversary. The flaw comes from the fact the server does not randomize its answer by the randomness contributed by the tag and the reader. This attack can be considered as tag traceability attack also which compromises the tag holder anonymity which shows that the protocol does not preserve the user privacy.

The above-mentioned impersonation attack which is also replay attack is fair enough to rule out any application of the Zheng *et al.*'s protocol [27]. However, if the adversary can force the reader time to a specific time, following the adversary model explained in Section III-A, then it can impersonate the reader and the tag set toward the server. It should be noted such an attack is of practical interest and in literature, a class of protocols called distance bounding protocols, e.g. see [29]–[32] for this concept, aim to withstand such an attack. Hence, it is important to defend an attack in which the adversary eavesdrops the generated messages by the tag and the reader in a location/time and use them in another location/time. The target of the attack procedure which is explained here against Zheng *et al.*'s protocol is such an attack. Assuming that the adversary is aiming to be authenticated by the server at a specified time  $T_1$ , the attack could be as follows:

- 1) Similarly, we assume that the adversary changes the reader's time, i.e forces it to be a  $T_1$  which is the target time for passing the server.
- 2) We assume that there will be a session between the reader and the tag next, where the reader starts a session and sends *Query* message along with its local timestamp  $T_1$  to the tag.
- 3) Once the tag receives the message, it generates a random number  $R_t$  and computes  $N_1 = h(ID_k \parallel T_1 \parallel R_t)$ ,  $N_2 = R_t \oplus T_1$  and send them along  $R_t$  to the reader.
- 4) When the reader receives the message, it computes  $R_t^* = N_2 \oplus T_1$  and verifies whether  $R_t^* \stackrel{?}{=} R_t$  is or not. If it is not, aborts the protocol otherwise computes  $N_3 = h(RID_k \parallel T_1 \parallel R_t)$  and send it along with  $N_1, T_1$  and  $R_t$  to the database.
- 5) Similarly, the adversary, who stands in the channel between the reader and the server, blocks the transferred message ( $N_3, N_1, T_1, R_t$ ) but stores it for the latter purposes.
- 6) The adversary reset/allow-to-be-reset the reader's time to the correct time which is synchronized with the server's time.
- 7) At appropriate time  $T_1$ , the adversary sends the stored messages from Step 5 to the server. The *messages/tag/reader* are authenticated by the server.

Moreover, it is easy to impersonate the server also. To impersonate the server, it would be enough to eavesdrop the transferred messages from the server to the tag and the reader in a legitimate session which is the tuple  $(N_4 = h(RID_k' \parallel T_2), N_5 = h(ID_k' \parallel T_2), T_2)$ , then returns it to the reader in a later session. More precisely, based on the Step 5 of this protocol, to authenticate the server, the reader using



received  $T_2$ , computes  $h(RID_k \| T_2)$  and verifies whether  $h(RID_k \| T_2) \stackrel{?}{=} N_4$ . Given  $RID_k$  is a static value, the reader authenticates the database (impersonated by the adversary) and sends  $N_5$  along with  $T_2$  to the tag. Now, based on the Step 6 of the protocol, the tag also using received  $T_2$ , computes  $h(ID_k \| T_2)$  and verifies whether  $h(ID_k \| T_2) \stackrel{?}{=} N_5$  and authenticates the server and the reader. One may argue that the attack can be fixed if the reader also evaluates the received  $T_2$ , to be in a reasonable duration from  $T_1$ . However, in this case, also we can apply the attack in the given adversary model. More precisely, the adversary at the first forces the reader time to a time close to  $T_2$  and then applies the attack.

The structure of the transferred messages has other weaknesses also but we omit them because we think the presented replay attack against the protocol is fair enough to rule out any possible application of this protocol.

### C. SECURITY ANALYSIS OF ZHOU *et al.* AND RELATED PROTOCOLS

Following the given adversary model in Section III-A, we propose a desynchronization attack against the Zhou *et al.*'s protocol. The attack procedure is as follows:

- 1) The adversary changes the reader's time, e.g. forces it to be a day ahead of the current time of the server;
- 2) We assume that there will be a session between the reader and the tag next, where the reader starts this phase by generating random number  $R_{r1}$  and sending it along its current timestamp  $T_1$  to the tag;
- 3) Once the tag received the message, it generates a random number  $R_t$  to compute  $A$ ,  $B$  and  $C$  and sends them to the reader, where  $x = ID_k \oplus T_1 \oplus s_j \oplus R_t$ ,  $A = (R_{r1} \| x)^2 \bmod n$ ,  $B = R_t^2 \bmod n$  and  $C = (R_{r1} \| s_j)^2 \bmod n$ .
- 4) Upon receipt  $A$ ,  $B$  and  $C$ , the reader generates another random number  $R_{r2}$ , computes  $y$ ,  $D$ ,  $H$  and sends  $A, B, C, T_1, D, E, F, R_{r1}$  and  $G$  to the server where:  $y = RID_k \oplus T_1 \oplus R_{r2} \oplus x_j$ ,  $D = (R_{r1} \| y)^2 \bmod m$ ,  $E = R_{r2}^2 \bmod m$ ,  $F = (R_{r1} \| x_j)^2 \bmod m$  and  $G = h(RID_k \| R_{r1} \| A \| T_1)$ .
- 5) The adversary, who stands in the channel between the reader and the server, blocks the transferred message but stores it for the later purposes.
- 6) The adversary reset/allow-to-be-reset the reader's time to the correct time which is synchronized with the server's time.
- 7) We again assume that there will be a session between the reader and the tag next, where the reader starts the session by generating random number  $R'_{r1}$  and sending it along its current timestamp  $T'_1$  to the tag;
- 8) Once the tag received the message, it generates a random number  $R'_t$  to compute  $A$ ,  $B$  and  $C$  and sends them to the reader, where  $x' = ID_k \oplus T'_1 \oplus s_j \oplus R'_t$ ,  $A' = (R'_{r1} \| x')^2 \bmod n$ ,  $B' = (R'_t)^2 \bmod n$  and  $C' = (R'_{r1} \| s_j)^2 \bmod n$ .
- 9) Upon receipt  $A'$ ,  $B'$  and  $C'$ , the reader generates another random number  $R'_{r2}$ , computes  $y'$ ,  $D'$ ,  $H'$  and sends  $A', B', C', T'_1, D', E', F', R'_{r1}$  and  $G'$  to the server where:  $y' = RID_k \oplus T'_1 \oplus R'_{r2} \oplus x_j$ ,  $D' = (R'_{r1} \| y')^2 \bmod m$ ,  $E' = (R'_{r2})^2 \bmod m$ ,  $F' = (R'_{r1} \| x_j)^2 \bmod m$  and  $G' = h(RID_k \| R'_{r1} \| A' \| T'_1)$ .
- 10) The adversary does not intercept the rest of the process. Hence, after authenticating the tag and the reader successfully, the server updates its related records from the reader and tags as follows:
  - If  $x_j = x_j$ , which without loss of generality we assume that it is the case, updates  $x_j$  and  $x_{j-1}$  as  $h(x_j \oplus R'_{r2})$  and  $x_j$  respectively;
  - If  $s_j = s_j$ , assuming it is, updates  $s_j$  and  $s_{j-1}$  as  $h(s_j \oplus R'_t)$  and  $s_j$  respectively;
- 11) The reader also updates its related records i.e.  $x_j$ ,  $V_k$  and  $W_k$  as  $x_j = h(x_j \oplus R'_{r2})$ ,  $V_k = h(x_j \| RID_k)$  and  $W_k = K \oplus h(x_j \| RID_k)$  respectively where  $x_j$  used them is updated.
- 12) The tag updates  $s_j$  to  $h(s_j \oplus R'_t)$ .
- 13) From now on, up to the time predefined time  $T_1$ , the adversary blocks any message from the reader and equivalently from the tag, to server. Hence, there will not be any authentication between the server and the reader/tag and their shared records remain fixed.
- 14) At appropriate time  $T_1$ , the adversary sends the stored messages from Step 5 to the server. The messages/tag/reader are authenticated based on the old records of the server.
- 15) Given that  $x'_j = x_j$  and  $s'_j = s_j$ , where  $x'_j$  and  $s'_j$  are the secret values used in the sent messages by the adversary, the server updates  $x_j$  and  $x_{j-1}$  as  $h(x_j \oplus R_{r2})$  and  $x_j$  respectively; it also updates  $s_j$  and  $s_{j-1}$  as  $h(s_j \oplus R_t)$  and  $s_j$  respectively.

Given that  $Pr(R_{r2} = R'_{r2}) = 2^{-n}$  and  $Pr(R_t = R'_t) = 2^{-n}$  then  $Pr(h(x_j \oplus R_{r2}) = h(x_j \oplus R'_{r2})) = 2^{-n}$  and  $Pr(h(s_j \oplus R_t) = h(s_j \oplus R'_t)) = 2^{-n}$ , where we considered the length of all parameters to be  $n$  bits. These show that the reader and the tags record of  $x_j$  and  $s_j$  respectively do not match any of their records in the server and the server will not authenticate them anymore, which means they have been desynchronized. The success probability of the given attack is  $1 - 2^{-n}$ .

It should be noted this attack is applicable to Li *et al.*'s protocol [21] almost as it is. In addition, beside Zhou *et al.* [22], Benssalah *et al.* [28] also analyzed the Li *et al.*'s protocol [21] and proposed an improved version for it, also targeting a secure authentication over a telecare medicine information system. Given that in a telecare medicine information system the reader should be mobile, it is possible to apply almost an identical attack against their protocol also, i.e. desynchronize the reader and the tag from the server. In addition, in Benssalah *et al.* protocol, each tag has an identifier  $ID_k$  which is updated after each successful run of the protocol. Given that we can desynchronize the reader from the server, the tag will never update its identifier hereafter. Hence, it is also possible to compromise the tag/tag-holder anonymity in that protocol.

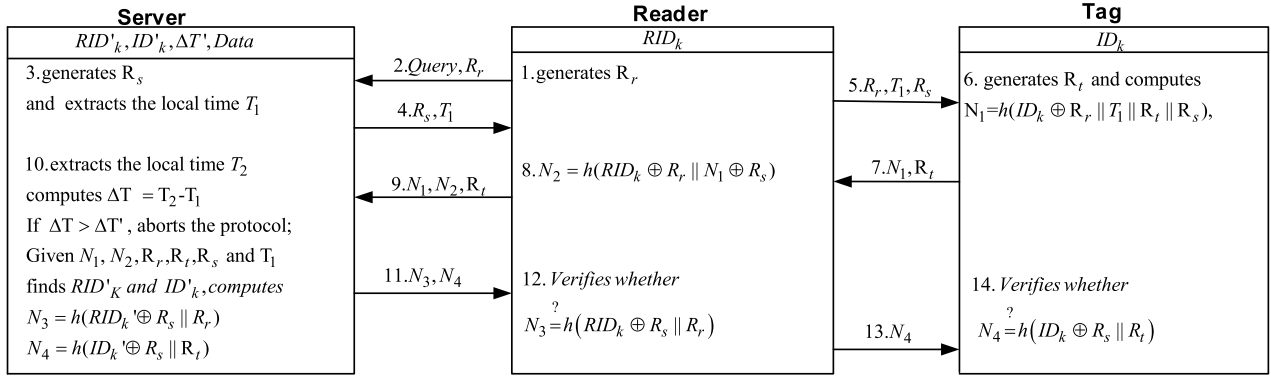


FIGURE 5. The proposed hash based mobile RFID mutual authentication protocol.

#### IV. PROPOSED PROTOCOL

In order to address the weaknesses of the Zheng *et al.* [27] protocol, in this section we propose an improved version of this protocol. The main idea is involving all protocol parties in the randomizing of the transferred messages. In addition, the only reliable source for the time could be the server. Hence, the timestamp is introduced by the server in the protocol and it will also verify the round trip time validity. The proposed protocol's representation is depicted in Fig. 5 and proceeds as below:

- 1) To start a session, the reader generates a fresh random number  $R_r$  and sends *Query* message along  $R_r$  to the server.
- 2) The server generates a fresh random number  $R_s$  and sends its current timestamps  $T_1$  along  $R_s$  to the reader.
- 3) The reader sends  $T_1$  along  $R_r$  and  $R_s$  to the tag.
- 4) Once the tag receives the message, it generates a random number  $R_t$  and computes  $N_1 = h(ID_k \oplus R_r || T_1 || R_t || R_s)$  and sends it along  $R_t$  to the reader.
- 5) When the reader receives the message, it computes  $N_2 = h(RID_k \oplus R_r || N_1 \oplus R_s)$  and sends it along with  $N_1$  and  $R_t$  to the server.
- 6) Upon receipt the message, at the first the server compares the current timestamps  $T_2$  with the sent  $T_1$  to ensure that the reader and tag have returned their responses in appropriate time. If it is so, then given the tuple  $(T_1, R_r, R_s, R_t, N_1, N_2)$  the server can search its records to find the match  $RID'_k$  and  $ID'_k$  that satisfy  $N_1$  and  $N_2$ . It is clear if it cannot find any match it means that the reader/tag is not legitimate and the messages intentionally/accidentally violated during the transfer process. If the server finds a matched  $RID'_k$  and  $ID'_k$  it authenticates the tag and computes  $N_3 = h(RID'_k \oplus R_s || R_r)$ ,  $N_4 = h(ID'_k \oplus R_s || R_t)$  and sends them to the reader.
- 7) Once receipt the message, the reader verifies whether  $h(RID_k \oplus R_s || R_r) \stackrel{?}{=} N_3$ . If it is so, the server is authenticated. It also means that the tag has been authenticated by the server. Hence the tag is also authenticated. On successful authentication, the reader forwards the received  $N_4$  to the tag.

- 8) Once receipt the message, the tag verifies whether  $h(ID_k \oplus R_s || R_t) \stackrel{?}{=} N_4$ . If it is so, the server is authenticated. It also means that the reader has been authenticated by the server. Hence the reader is also authenticated.

#### V. SECURITY PROOF OF THE PROPOSED PROTOCOL

In this section, first in informalized manner we show that the security correctness of proposed protocol and then based on Scyther tool [33] which is an automatic formal prover, prove the improved protocol has perfect security against all known active and passive attacks.

##### A. INFORMAL PROOF OF SECURITY

###### 1) RESISTANCE AGAINST DESYNCHRONIZATION ATTACK

Given that in the proposed protocol, similar to its predecessor protocol by Zheng *et al.* [27], protocol parties do not update their shared values, it is not possible to desynchronize them. Hence, the proposed protocol does not suffer from the desynchronization attack.

###### 2) Resistance against secret disclosure attack

The secret parameters in the proposed protocol are  $ID_k$  and  $RID_k$ . Given that any message which is transferred over a public channel is a function of these values are generated by the hash function and it is not feasible to invert a hash function in polynomial time, the adversary cannot extract secret parameters  $ID_k$  and  $RID_k$  in polynomial time. Hence, the proposed protocol does not suffer from a secret disclosure attack.

###### 3) RESISTANCE AGAINST ALL KINDS OF IMPERSONATION ATTACK AND REPLAY ATTACKS

In the proposed protocol, any message which is sent to a receiver has been randomized at least by a nonce suggested by the receiver. Hence, to impersonate  $X \in \{tag, reader, server\}$  toward  $Y \in \{tag, reader, server\}$ ,  $X$  should answer a challenge which is suggested by  $Y$  and the challenge is refreshed in any session. More precisely:

- To impersonate the tag, assuming that the reader and the server are legitimate, given a fresh  $R_r$  and a timestamp

$T_1$ , the adversary should generate a valid  $N_1 = h(ID_k \oplus R_r \| T_1 \| R_t \| R_s)$ , without the knowledge of  $ID_k$  which is not feasible. However, when received  $R_r$  and  $T_1$ , any adversary can return a random pair as  $N_1$  and  $R_t$ . In the server side, the severer searches whole its records to find a match for  $N_1$  based on  $R_t$ , assuming that the  $|N_1| = n$ , then any comparison is satisfied with the probability of  $2^{-n}$ . Assuming that the server keeps the records of  $t$  tags, the success probability of the adversary on each try will be almost  $t2^{-n}$  and after  $q$  queries, it will be almost  $qt2^{-n}$ .

- To impersonate the reader, assuming that the tag and the server are legitimate, given a fresh  $R_s$  and a timestamp  $T_1$ , the adversary should generate a valid  $N_2 = h(RID_k \oplus R_r \| N_1 \oplus R_s)$ , without the knowledge of  $RID_k$  which is not feasible. However, when received  $R_s$  and  $T_1$ , any adversary can return a random value  $N_2$ . In the server side, the severer searches whole its records to find a match for  $N_2$  based on  $R_r$ ,  $R_s$  and  $N_1$ . In this case, also any comparison is satisfied with the probability of  $2^{-n}$ . Assuming that the server keeps the records of  $r$  readers, the success probability of the adversary on each try will be almost  $r2^{-n}$  and after  $q$  queries, it will be almost  $qr2^{-n}$ .
- Impersonating the reader and the tag toward the server will not be easier than impersonating any of them.
- To impersonate the server, assuming that the tag and the reader are legitimate, given a fresh  $R_r$  and  $R_t$  and any desired timestamps  $T_1$  and value of  $R_s$ , the adversary should generate a valid  $N_3 = h(RID_k \oplus R_s \| R_r)$  and  $N_4 = h(ID_k \oplus R_s \| R_t)$ , without the knowledge of  $RID_k$  and  $ID_k$  which is not feasible. However, when received the reader response, any adversary can return random values as  $N_3$  and  $N_4$ . However, such random values are accepted by the tag and the reader each one by the probability of  $2^{-n}$ . Hence, the adversary's advantage to impersonate the server is upper bounded by  $2^{-n}$  on each query and  $q2^{-n}$  after  $q$  queries.

All in all, the proposed protocol is secure against impersonation and replay attacks.

#### 4) RESISTANCE AGAINST TRACEABILITY ATTACKS

To apply traceability attack, the adversary should find a connection between the transferred messages over the public channel and the identity of a protocol party or transferred messages in previous sessions. However, in the proposed protocol, any message which is transferred over a public channel is generated by the hash function and has at least one fresh random number as its input. Hence, from a session to another session the adversary can see the fresh output of the hash function that cannot be connected to each other. Hence, the proposed protocol does not suffer from traceability attack.

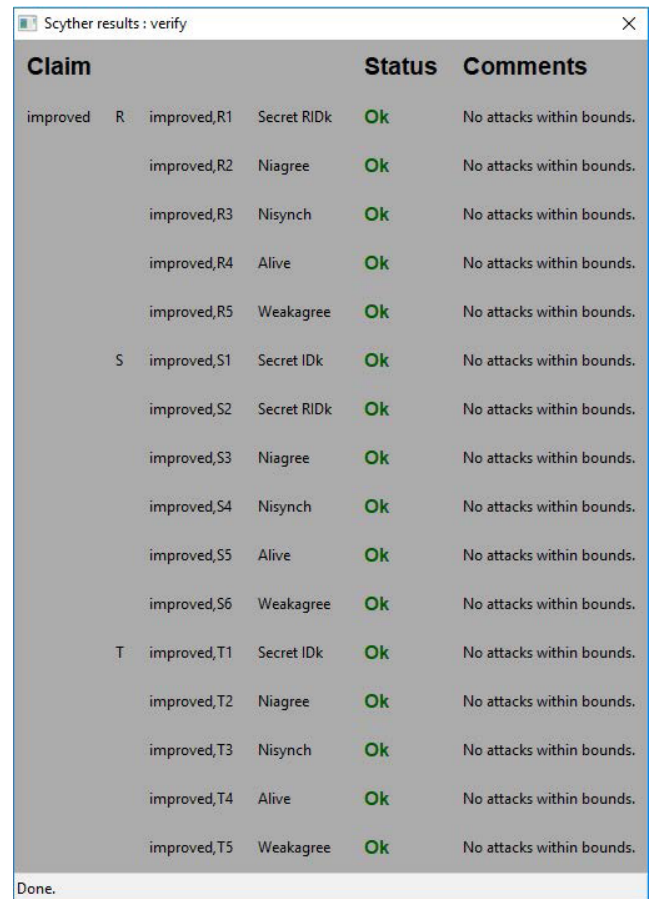
#### 5) RESISTANCE AGAINST AN ADVERSARY THAT CAN CONTROL THE TIME

In this paper, we showed that several protocols suffer from various attacks if the adversary can control the time of

a protocol's party. However, in the proposed protocol, the only source of time is the server which is more reliable than the tag and the reader, and even if an adversary controls the time, given that it has no control over the fresh nonce introduced by the protocol's parties, it cannot apply any of the attacks suggested against those protocols. It worth to mention that attack was possible because the server contribution to the protocol randomization was only the timestamps, while in the proposed protocol, the server also randomizes the transferred messages by  $R_s$ . Hence, the proposed protocol does not suffer from an adversary that can control the time and desynchronization attack.

#### B. FORMAL PROOF OF SECURITY

Scyther [33] is a utility for checking the security properties of a protocol which is written in Python. In this tool, the protocol roles are written in Security Protocol Description Language (spdl), and then the security claims of confidentiality and authenticity are evaluated.



Claim				Status	Comments
improved	R	improved,R1	Secret RIDk	Ok	No attacks within bounds.
		improved,R2	Niagree	Ok	No attacks within bounds.
		improved,R3	Nisynch	Ok	No attacks within bounds.
		improved,R4	Alive	Ok	No attacks within bounds.
		improved,R5	Weakagree	Ok	No attacks within bounds.
S		improved,S1	Secret IDk	Ok	No attacks within bounds.
		improved,S2	Secret RIDk	Ok	No attacks within bounds.
		improved,S3	Niagree	Ok	No attacks within bounds.
		improved,S4	Nisynch	Ok	No attacks within bounds.
		improved,S5	Alive	Ok	No attacks within bounds.
		improved,S6	Weakagree	Ok	No attacks within bounds.
T		improved,T1	Secret IDk	Ok	No attacks within bounds.
		improved,T2	Niagree	Ok	No attacks within bounds.
		improved,T3	Nisynch	Ok	No attacks within bounds.
		improved,T4	Alive	Ok	No attacks within bounds.
		improved,T5	Weakagree	Ok	No attacks within bounds.

Done.

**FIGURE 6.** The proposed hash based mobile RFID mutual authentication protocol.

The role description of improved protocol in spdl language is depicted in Appendix VI. As illustrated in Fig. 6, the Scyther tool cannot find any attacks for our proposed protocol. In addition, as shown in Table 2, the protocol designed

**TABLE 2.** The security comparison of the improved protocol with the other protocols.

Protocols	A1	A2	A3	A4	A5
Zheng <i>et al.</i> [27]	✓	✓	×	×	×
Li <i>et al.</i> [21]	×	×	×	×	×
Benssalah <i>et al.</i> [33]	×	×	✓	×	×
Zhou <i>et al.</i> [22]	×	✓	✓	✓	×
Proposed	✓	✓	✓	✓	✓
A1: Desynchronization attack; A2: Secret Disclosure attack; A3: Impersonation and Replay attack; A4: Traceability attack and Anonymity; A5: The attacks that the adversary can control the time; ✓: Resistant ×: Vulnerable					

in this paper, unlike its predecessors, provides desired security against the attacks in the context.

## VI. CONCLUSIONS

In this paper, we investigate the security of four authentication protocols which with the aim of providing security and privacy preservation in telecare medicine information system and smart campus have been proposed. Precisely, we presented reader and tag impersonation to server attack and also server impersonation to the reader attacks against Zheng *et al.* protocol.

Moreover, we present a new desynchronization attack against Li *et al.*, Zhou *et al.* and Benssalah *et al.* protocols which exploits this weakness in these protocols where timestamps are generated by the mobile readers.

We also designed a protocol that, in addition to resolving all the weaknesses of the four protocols, is also safe against other known attacks. We also demonstrated in an informal and formal way that the designed protocol is safe and suitable for use in telecare medication information system and smart campus.

## APPENDIX

### THE PROPOSED PROTOCOL DESCRIPTION IN SPDL

```

const con :Function;
const xor :Function;
hashfunction h;
secret RIDk;
secret IDk;
macro N1=h(con(xor(IDk,Rr),T1,Rt,Rs));
macro N2=h(con(xor(RIDk,Rr),
               xor(N1,Rs)));
macro N3=h(con(xor(RIDk,Rs),Rr));
macro N4=h(con(xor(IDk,Rs),Rt));

protocol improved(R, S, T) {
role R {
secret RIDk;
secret IDk;
fresh Rr;
var Rt;
var Rs;

```

```

var T1;
send_1(R, S, Rr);
recv_2(S, R, Rs, T1);
send_3(R, T, Rr, T1, Rs);
recv_4(T, R, N1, Rt);
send_5(R, S, N1, N2, Rt);
recv_6(S, R, N3, N4);
send_7(R, T, N4);
claim(R, Secret, RIDk);
claim(R, Niagree);
claim(R, Nisynch);
claim(R, Alive);
claim(R, Weakagree);
}
role S {
fresh T1;
fresh Rs;
secret RIDk;
secret IDk;
var Rr;
var Rt;
recv_1(R, S, Rr);
send_2(S, R, Rs, T1);
recv_5(R, S, N1, N2, Rt);
send_6(S, R, N3, N4);
claim(S, Secret, IDk);
claim(S, Secret, RIDk);
claim(S, Niagree);
claim(S, Nisynch);
claim(S, Alive);
claim(S, Weakagree);
}
role T{
fresh Rt;
var Rr;
var Rs;
secret IDk;
var T1;
recv_3(R, T, Rr, T1, Rs);
send_4(T, R, N1, Rt);
recv_7(R, T, N4);
claim(T, Secret, IDk);
claim(T, Niagree);
claim(T, Nisynch);
claim(T, Alive);
claim(T, Weakagree);
}
}

```

## REFERENCES

- [1] P. R. Sun, B. H. Wang, and F. Wu, "A new method to guard inpatient medication safety by the implementation of RFID," *J. Med. Syst.*, vol. 32, no. 4, pp. 327–332, Aug. 2008.
- [2] M. Wazid, A. K. Das, N. Kumar, M. Conti, and A. V. Vasilakos, "A novel authentication and key agreement scheme for implantable medical devices deployment," *IEEE J. Biomed. Health Informat.*, vol. 22, no. 4, pp. 1299–1309, Jul. 2018.



- [3] M. Wazid, A. K. Das, M. K. Khan, A. Al-Dhawalie-Al-Ghaiheb, N. Kumar, and A. V. Vasilakos, "Secure authentication scheme for medicine anti-counterfeiting system in IoT environment," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1634–1646, Oct. 2017.
- [4] M. Benssalah, M. Djeddou, and K. Drouiche, "Dual cooperative RFID-telecare medicine information system authentication protocol for healthcare environments," *Secur. Commun. Netw.*, vol. 9, no. 18, pp. 4924–4948, 2016.
- [5] S. Chatterjee *et al.*, "On the design of fine grained access control with user authentication scheme for telecare medicine information systems," *IEEE Access*, vol. 5, pp. 7012–7030, 2017.
- [6] L. Zhang, S. Zhu, and S. Tang, "Privacy protection for telecare medicine information systems using a chaotic map-based three-factor authenticated key agreement scheme," *IEEE J. Biomed. Health Informat.*, vol. 21, no. 2, pp. 465–475, Mar. 2017.
- [7] V. Kumar, M. Ahmad, and A. Kumari, "A secure elliptic curve cryptography based mutual authentication protocol for cloud-assisted TMIS," *Telematics Inform.*, to be published.
- [8] M. Qi, J. Chen, and Y. Chen, "A secure biometrics-based authentication key exchange protocol for multi-server TMIS using ECC," *Comput. Methods Programs Biomed.*, vol. 164, pp. 101–109, Oct. 2018.
- [9] R. Amin, S. H. Islam, P. Gope, K.-K. R. Choo, and N. Tapas, "Anonymity preserving and lightweight multi-medical server authentication protocol for telecare medical information system," *IEEE J. Biomed. Health Inform.*, to be published.
- [10] W. Li, S. Zhang, Q. Su, Q. Wen, and Y. Chen, "An anonymous authentication protocol based on cloud for telemedical systems," *Wireless Commun. Mobile Comput.*, vol. 2018, Sep. 2018, Art. no. 8131367.
- [11] P. Chandrakar and H. Om, "An extended ECC-based anonymity-preserving 3-factor remote authentication scheme usable in TMIS," *Int. J. Commun. Syst.*, vol. 31, no. 8, p. e3540, 2018.
- [12] C.-T. Li, D.-H. Shih, and C.-C. Wang, "Cloud-assisted mutual authentication and privacy preservation protocol for telecare medical information systems," *Comput. Methods Programs Biomed.*, vol. 157, pp. 191–203, Apr. 2018.
- [13] S. Qiu, G. Xu, H. Ahmad, and L. Wang, "A robust mutual authentication scheme based on elliptic curve cryptography for telecare medical information systems," *IEEE Access*, vol. 6, pp. 7452–7463, 2018.
- [14] Q. Jiang, Z. Chen, B. Li, J. Shen, L. Yang, and J. Ma, "Security analysis and improvement of bio-hashing based three-factor authentication scheme for telecare medical information systems," *J. Ambient Intell. Humanized Comput.*, vol. 9, no. 4, pp. 1061–1073, 2018.
- [15] S. Likhitha and R. Saravanan, "Cryptanalysis of a multifactor authentication protocol," in *Recent Findings in Intelligent Computing Techniques*. Singapore: Springer, 2019, pp. 35–42.
- [16] J. Wei, W. Liu, and X. Hu, "On the security and improvement of privacy-preserving 3-factor authentication scheme for TMIS," *Int. J. Commun. Syst.*, vol. 31, no. 15, p. e3767, 2018.
- [17] P. Picazo-Sanchez, N. Bagheri, P. Peris-Lopez, and J. E. Tapiador, "Two RFID standard-based security protocols for healthcare environments," *J. Med. Syst.*, vol. 37, no. 5, p. 9962, 2013.
- [18] M. Saffkhani, N. Bagheri, and M. Naderi, "On the designing of a tamper resistant prescription RFID access control system," *J. Med. Syst.*, vol. 36, no. 6, pp. 3995–4004, 2012.
- [19] M. Saffkhani, N. Bagheri, and M. Naderi, "A note on the security of IS-RFID, an inpatient medication safety," *I. J. Med. Informat.*, vol. 83, no. 1, pp. 82–85, 2014.
- [20] K. Srivastava, A. K. Awasthi, S. D. Kaul, and R. C. Mittal, "A hash based mutual RFID tag authentication protocol in telecare medicine information system," *J. Med. Syst.*, vol. 39, no. 1, p. 153, 2015.
- [21] C.-T. Li, C.-Y. Weng, and C.-C. Lee, "A secure RFID tag authentication protocol with privacy preserving in telecare medicine information system," *J. Med. Syst.*, vol. 39, no. 8, p. 77, 2015.
- [22] Z. Zhou, P. Wang, and Z. Li, "A quadratic residue-based RFID authentication protocol with enhanced security for TMIS," *J. Ambient Intell. Humanized Comput.*, pp. 1–13, Oct. 2018. doi: [10.1007/s12652-018-1088-5](https://doi.org/10.1007/s12652-018-1088-5).
- [23] O. Mir and M. Nikooghadam, "A secure biometrics based authentication with key agreement scheme in telemedicine networks for e-health services," *Wireless Pers. Commun.*, vol. 83, no. 4, pp. 2439–2461, 2015.
- [24] D. Abbasinezhad-Mood and M. Nikooghadam, "Efficient design of a novel ECC-based public key scheme for medical data protection by utilization of NanoPi fire," *IEEE Trans. Rel.*, vol. 67, no. 3, pp. 1328–1339, Sep. 2018.
- [25] Z. Tan, "Secure delegation-based authentication for telecare medicine information systems," *IEEE Access*, vol. 6, pp. 26091–26110, 2018.
- [26] X. Li, F. Wu, M. K. Khan, L. Xu, J. Shen, and M. Jo, "A secure chaotic map-based remote authentication scheme for telecare medicine information systems," *Future Gener. Comput. Syst.*, vol. 84, pp. 149–159, Jul. 2018.
- [27] L. Zheng *et al.*, "A new mutual authentication protocol in mobile RFID for smart campus," *IEEE Access*, vol. 6, pp. 60996–61005, 2018.
- [28] M. Benssalah, M. Djeddou, and K. Drouiche, "Security analysis and enhancement of the most recent RFID authentication protocol for telecare medicine information system," *Wireless Pers. Commun.*, vol. 96, no. 4, pp. 6221–6238, 2017.
- [29] A. Yang, E. Pagnin, A. Mitrokovska, G. P. Hancke, and D. S. Wong, "Two-hop distance-bounding protocols: Keep your friends close," *IEEE Trans. Mobile Comput.*, vol. 17, no. 7, pp. 1723–1736, Jul. 2018.
- [30] 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.
- [31] C. Molina-Martínez, P. Galdames, and C. Duran-Faundez, "A distance bounding protocol for location-cloaked applications," *Sensors*, vol. 18, no. 5, p. 1337, 2018.
- [32] S. Mauw, Z. Smith, J. Toro-Pozo, and R. Trujillo-Rasua, "Distance-bounding protocols: Verification without time and location," in *Proc. IEEE Symp. Secur. Privacy (SP)*, San Francisco, CA, USA May 2018, pp. 549–566.
- [33] C. J. F. Cremers, "The scyther tool: Verification, falsification, and analysis of security protocols," in *Computer Aided Verification*, A. Gupta and S. Malik, Eds. Berlin, Germany: Springer, 2008, pp. 414–418.



**MASOUMEH SAKKHANI** received the Ph.D. degree in electrical engineering from the Iran University of Science and Technology, in 2012, with the security analysis of RFID protocols as her major field. She is currently an Assistant Professor with the Computer Engineering Department, Shahid Rajaei Teacher Training University, Tehran, Iran. Her current research interests include the security analysis of lightweight and ultra-lightweight protocols, targeting constrained environments, such as RFID, the IoT, VANET, and WSN. She is the author/co-author of over 50 technical articles in information security and cryptography in major international journals and conferences.



**ATHANASIOS VASILAKOS** is currently a Distinguished Professor with the Computer Science Department, Lulea University of Technology, Sweden. He has authored or co-authored over 600 technical papers in major international journals and conferences, and is the author/co-author of five books and 20 book chapters. His main research interests include cybersecurity, power systems cybersecurity, networking, the IoTs and smart cities, cloud computing, big data analytics, and machine learning. His papers received citations of more than 28400, with h-index= 92. He is also the ISI Highly Cited Researcher (the Highest Scientific Distinction).

Prof. Vasilakos has served as the General Chair and Technical Program Committee Chair for many international conferences. He is also serving/served as an Editor for many leading journals. He is a frequent keynote, panel, and tutorial speaker. Moreover, he is a Consultant to the European Commission.

...