IJICTR International Journal of Information & Communication Technology Research

Volume 4- Number 3- June 2012 (33- 43)

Forgery Attack is a Piece of Cake on a Class of Mutual Authentication Protocols

Nasour Bagheri
Electrical Engineering Department,
Shahid Rajaee Teacher Training University,
Tehran, Iran
NBagheri@srttu.edu

Masoumeh Safkhani
Electrical Engineering Department,
Iran University of Science & Technology,
Tehran, Iran
M_Safkhani @iust.ac.ir

Majid Naderi
Electrical Engineering Department,
Iran University of Science & Technology,
Tehran, Iran
M_Naderi@iust.ac.ir

Yiyuan Luo
Department of Computer Science & Engineering,
Shanghai Jiao Tong University,
China,
luoyiyuan@gmail.com

Qi Chai
Department of Electrical & Computer Engineering,
University of Waterloo,
Canada,
q3chai@uwaterloo.ca

Received: February 14, 2012- Accepted: April 23, 2012

Abstract—**A suitable mutual authentication protocol for an RFID system should provide mutual authentication along with user privacy. In addition, such protocol must be resistant to active and passive attacks, e.g. man-in-the-middle attack, replay attack, reader-/tag-impersonation attack, denial of service attack and traceability attack. Among them, tag-impersonation attack refers to a forgery attack in which the adversary fools the legitimate reader to authenticate it as a valid tag. In this paper we exam the security of three RFID mutual authentication protocols which have been recently proposed by Luo *et al.*, Shen *et al.* and Habibi and Gardeshi, under tag impersonation attack. We found that these three protocols share a same vulnerability – in each session, the tag and the reader generate a random value respectively and they use the exclusive-or (XOR) of those random values in the authentication process. We exploit this vulnerability to present effective and efficient tag impersonation attacks against these protocols, e.g., the success probabilities of our attacks are "1" and the complexity is at most two runs of each protocol. In addition, we exhibit the improved version of these protocols, which are immune from tag impersonation attacks.**

Keywords- RFID; Authentication; Tag Impersonation; WG-7; ARAP.

I. INTRODUCTION

Nowadays, radio frequency identification (RFID) is a favorite technology for automated identification in various applications, e.g., libraries, supply chain management, e-passports, human implants and toll payment. The tag, the reader and the back-end server are three basic components for an RFID system: (1) tags are connected to the objects that are supposed to be identified by the reader through radio frequency

signals; (2) the back-end server aids the reader by extra storage spaces and further computational capability. In addition, it is much more reliable to keep the valuable data of all tags in back-end server and transfer the necessary data of a particular tag, in case of request, to the reader which prevent the loss of all data in case of reader theft. In the design of RFID authentication protocols, an assumption is implicitly made that the channel between the reader and the back end server is secure. Hence, in this paper we do not distinguish reader and back-end server and just call them reader. Memory and computing power of low-cost tags (also called passive tags) is very limited. Therefore, to provide privacy and security for these tags, computationally intensive algorithms are not considered. In other words, the target protocol should be a composition of a few computational-efficient primitives to meet the low-cost manufacture requirements. In the literature, several such protocols, called lightweight mutual authentication protocols, e.g., [27, 29, 26, 10, 3, 7, 25, 1 ,2, 13, 12], have already been proposed. However, most of these protocols do not satisfy all the claimed security properties [11, 8, 20, 6, 21, 23, 14, 22, 18, 16, 17, 15, 19].

In this paper, we analyze the security of several recent mutual authentication protocols against tag impersonation. All protocols, that we focus on, randomize their authentication sessions to avoid various attacks, e.g. tag's location traceability and replay attack. In these protocols, in each session, the tag and the reader generate random values and use the exclusive-or of those random values in the authentication process. We show that this is a vital drawback for the authenticity, by exploiting which the adversary can launch effective and efficient tag impersonation. The first protocol which we analyze is a recently proposed protocol by Luo et al. [9], which uses a lightweight stream cipher called WG-7 to provide confidentiality and authenticity for RFID systems. The second protocol is ARAP protocol and proposed by Shen et al. [24]. This protocol uses one-way hash function to provide the desired security properties. Another protocol which we consider in this paper is one of the most recent EPC Class-1 Generation-2 standard [4, 5] compliant protocols, which is an improvement to the Yeh et al. 's protocol [30] proposed by Habibi and Gardeshi [6]. However, in this paper we show that they were not succeeding in their attempt and the proposed protocol is vulnerable to tag impersonation attack. At last, we show the slightly modified versions of these protocols, which are immune from our tag impersonation attacks.

The rest of the paper is organized as follows: We describe the Luo et al.'s protocol and our novel attack against this protocol in section II. In section III, we give an improved version of Luo et al.'s protocol and the corresponding security analysis. In section IV, we analyze the ARAP protocol and propose a similar tag impersonation attack and also give an improved version of ARAP protocol. In section V, we analyze a protocol recently proposed by Habibi and Gardeshi and propose our tag impersonation attack against the protocol and also present an improved version of it. Section VI concludes the paper.

II. LUO ET AL.'S RFID AUTHENTICATION PROTOCOL AND OUR TAG IMPERSONATION ATTACK

Recently, Luo et al. have proposed a mutual authentication protocol for RFID systems based on a lightweight stream cipher called WG-7 [9]. In sections III, IV and this section of the paper, we use the following notations which are depicted in Table I.

The protocol randomizes each authentication session by employing two random values R_r and R_t, generated by the reader and the tag respectively. Luo et al.'s protocol as depicted in Figure 1 works as follows:

1. The reader chooses an 80-bit random number R_r and sends Query and R_r to the tag.

2. As the tag receives the message it does the following:

(a) generates another 80-bit random number R_t,

(b) computes $M_1 = t_i \oplus R_t$,

(c) initializes the internal states of WG-7 by $(R_r \oplus R_t) \| k_i \| 1$,

(d) assigns the first 80-bit of WG-7 output key stream to M_2,

(e) sends M_1 and M_2 to the reader.

3. As the reader receives the message, for each (tj , kj) in the database, it behaves as follows:

(a) retrieves R'_t from $M_1 \oplus t_i$,

(b) initializes the internal states of WG-7 by $(R_r \oplus R'_t) \| k_i \| 1$,

(c) assigns the first 80-bit of WG-7 output key stream to M'_2

(d) verifies whether $M'_2 \stackrel{?}{=} M_2$ if yes:

– authenticates the tag,

– assigns the second 80-bit of WG-7 output key stream to M_3,

– sends M_3 to the tag.

4. As the tag receives M_3, it does as follows:

(a) assigns the second 80-bit of WG-7 output key stream to M'_3

TABLE I. LUO ET AL.'S AND ARAP PROTOCOLS NOTATIONS

R	RFID reader
Ti	RFID tag i
A	Adversary
T_i	Static identifier of T_i
R_r	Random number generated by the reader
R_t	Random number generated by the tag
h(.)	One-way hash function
K_i	Secret key of i^{th} tag
WG-7	A lightweight stream cipher with 161 bit internal state registers
PID	Tag's pseudonym
X||Y	Concatenation of strings X and Y

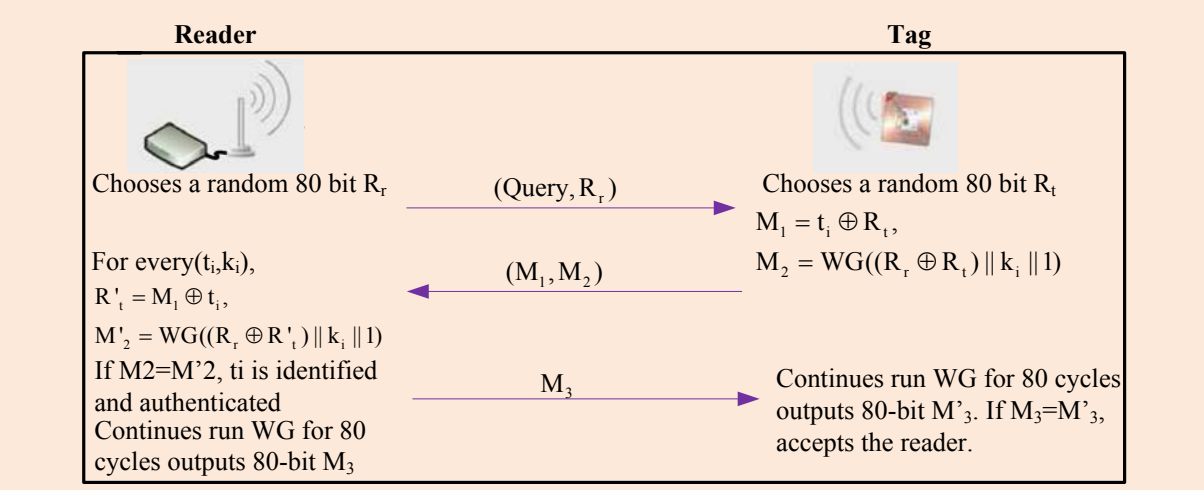

Figure 1. The Mutual Authentication Protocol proposed by Luo et al.

(b) verifies whether $M'_3 \stackrel{?}{=} M_3$ to authenticate the reader.

Luo et al. [1] claim that it would not be possible for the adversary to generate a tuple M_1 and M_2 such that the reader authenticates the adversary as a valid tag. More precisely, the authors state that to generate a valid M_1 and M_2 and impersonate the tag, the adversary requires to find the secret values t_i and k_i that are protected by the encryption function WG-7. However, we present a rather simple attack which can impersonate a legitimate tag without any knowledge of the secret values t_i and k_i. To impersonate the tag T_i, the adversary A follows the steps described as below:

1. A eavesdrops one execution of protocol between the reader R and T_i and stores all transferred values between R and T_i. Those values include R_r, $M_1 = t_i \oplus R_t$, M_2 and M_3 (M_2 and M_3 are the first and the second blocks of length 80-bit generated by the WG-7 stream cipher for which the internal states loaded by $(R_r \oplus R_t) \| k_i \| 1$.

2. On the next round of protocol, when R sends Query and R'_r to the tag, the adversary responses with the tuple M'_1 and M'_2 where $M'_1 = M_1 \oplus R_r \oplus R'_r$ and $M'_2 = M_2$.

3. R uses the tuple (t_i, k_i) of T_i to extract R'_t as follows:

$$R'_t = M'_1 \oplus t_i = M_1 \oplus R_r \oplus R'_r \oplus t_i$$
$$= R_t \oplus t_i \oplus R_r \oplus R'_r \oplus t_i = R_t \oplus R_r \oplus R'_r$$

4. R verifies whether $M'_2 \stackrel{?}{=} M_2$ by:

(a) to generate M_2, the internal states of WG-7 has been initialized by $(R_r \oplus R_t) \| k_i \| 1$

(b) to generate M'_2, the internal states of WG-7 has been initialized by $(R'_r \oplus R'_t) \| k_i \| 1 = (R'_r \oplus R_t \oplus R'_r \oplus R_r) \| k_i \| 1 = (R_r \oplus R_t) \| k_i \| 1$,

5. With the probability of "1", R authenticates the adversary as T_i.

Hence, following the above attack the reader authenticates the adversary as a legitimate tag. The success probability of above attack is "1" and the complexity is two runs of protocol.

III. IMPROVED ON LUO ET AL.'S PROTOCOL

In this section we show that Luo et al.'s protocol can be modified slightly to against our attack. The revised protocol works as follows and also shown in Figure 2:

1. The reader chooses an 80-bit random number R_r and sends Query and R_r to the tag.

2. As the tag receives the message it does the following:

(a) generates another 80-bit random number R_t,

(b) computes $M_1 = t_i \oplus R_t$,

(c) initializes the internal states of WG-7 by $R_r \| (R_t \oplus k_i) \| 1$,

(d) assigns the first 80-bit of WG-7 output key stream to M_2,

(e) sends M_1 and M_2 to the reader.

3. As the reader receives the message, for each (t_j, k_j) in the database, it behaves as follows:

(a) retrieves R'_t from $M_1 \oplus t_i$,

(b) initializes the internal states of WG-7 by $R_r \| (R'_t \oplus k_i) \| 1$,

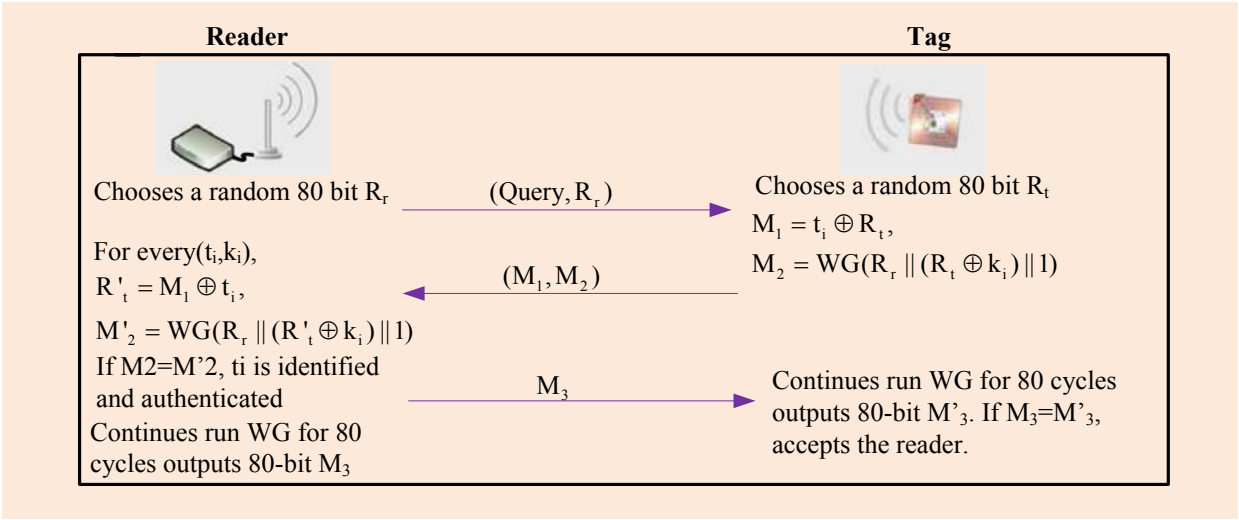

Figure 2. The revised protocol of Luo et al.'s

(c) assigns the first 80-bit of WG-7 output key stream to M'_2

(d) verifies whether $M'_2 \stackrel{?}{=} M_2$ if yes:

– authenticates the tag,

– assigns the second 80-bit of WG-7 output key stream to M_3,

– sends M_3 to the tag.

4. As the tag receives M_3, it does as follows:

(a) assigns the second 80-bit of WG-7 output key stream to M'_3

(b) verifies whether $M'_3 \stackrel{?}{=} M_3$ to authenticate the reader.

After the simplified modification, the above protocol can prevent our attack. This is because if the attacker changes any bits of R_t or R_r, the output of WG-7 will be different and unexpected, and thus it can prevent the proposed impersonation attack. For more details see Table II.

IV. Tag Impersonation Attack on ARAP Protocol and Improving it

In this section, we analyze the security of another mutual authentication protocol for RFID systems, which has been recently proposed by Shen et al. [24] called ARAP and present the improved version of it. The ARAP protocol employs a one-way hash function to provide the desired security. Similar to the Luo et al. [9], it randomizes each authentication session using two random values R_r and R_t, respectively generated by the reader and the tag. In addition, the tag changes its pseudonym PID after each successful run of protocol. The new value of PID is selected from a collision-free set, which is known to the reader (back-end server) and each value of PID is used in one session. The ARAP protocol as depicted in Figure 3 works as follows:

1. The reader chooses a k-bit random number R_r and sends Query and R_r to the tag.

2. As the tag receives the message, it does as follows:

(a) generates another k-bit random number R_t,

(b) computes $S = h(PID \oplus k_i)$ and $M = h(R_t \oplus R_r \oplus PID) \oplus S$

(c) sends R_t, PID and M to the reader.

3. As the reader receives the message (R_t, PID, M) it does as follows:

(a) searches its storage to locate the pseudonym PID and find out k_i,

(b) computes $S' = h(PID \oplus k_i)$ and $M' = h(R_t \oplus R_r \oplus PID) \oplus S'$

(c) verifies whether $M' \stackrel{?}{=} M$ if yes:

– authenticates the tag,

– computes $N' = h(M' \oplus S')$,

– sends N' to the tag.

4. As the tag receives N' does as follows:

(a) computes $N = h(M \oplus S)$,

(b) verifies whether $N' \stackrel{?}{=} N$ to authenticate the reader,

(c) After successful reader authentication, the tag updates PID and S.

Now, we present a similar attack which can impersonate a legitimate tag without any knowledge of the secret values k_i. To impersonate the tag T_i, the adversary A follows the steps described as below:

1. A supplants R and sends Query and R_r to the target T_i and receives its response which is (R_t, PID, $M = h(R_t \oplus R_r \oplus PID) \oplus S$).

TABLE II. PERFORMANCE AND COMPLEXITY COMPARISON BETWEEN THE LUO ET AL. ORIGINAL AND THE IMPROVED PROTOCOLS. IN THIS TABLE EACH ENTRY DENOTES NUMBER OF BITS AND L DENOTES THE BIT LENGTH OF PARAMETERS. IN ADDITION T, R AND T.I.A. DENOTE TAG, READER AND TAG IMPERSONATION ATTACK RESPECTIVELY.

Protocol	# ⊕ in R	# ⊕ in R	# ‖ in T	# ‖ in R	# E in T
Luo et al.	2L	2L	2	2	6L
Improved Luo et al.	2L	2L	2	2	6L

Protocol	# E in T	T storage	R storage	# Transfer	T.I.A.
Luo et al.	6L	2L	2L	5L	Yes
Improved Luo et al.	6L	2L	2L	5L	No

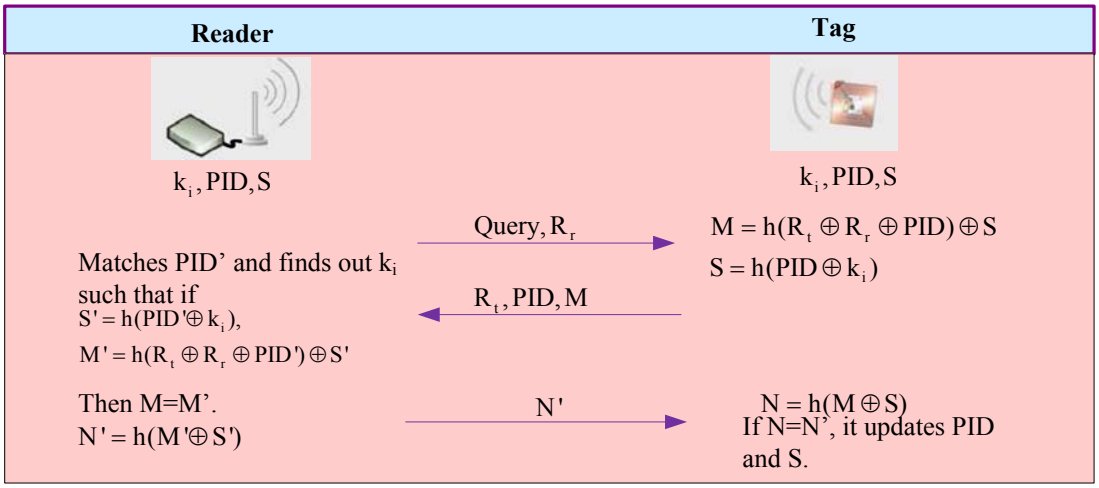

Figure 3. ARAP Protocol

2. In the next session, when R sends Query and R'_r, A impersonates T_i and sends $R'_t = R_r \oplus R'_r \oplus R_t$, PID and $M = h(R_t \oplus R_r \oplus PID) \oplus S$ to the reader.

3. As R receives the message, it does as follows:

(a) uses PID to find the tuple (PID, k_i) of T_i,

(b) computes $S' = h(PID \oplus k_i)$,

(c) computes M′ as follows:

$$M' = h(R'_t \oplus R'_r \oplus PID) \oplus h(PID \oplus k_i)$$
$$= h(R_t \oplus R_r \oplus R'_r \oplus R'_r \oplus PID) \oplus h(PID \oplus k_i) =$$
$$h(R_t \oplus R_r \oplus PID) \oplus S = M$$

(d) verifies whether $M' \stackrel{?}{=} M$

(e) with the probability of "1", R authenticates the adversary as T_i.

The whole idea of the above attack is similar to what we used to impersonate the tag for Luo et al. protocol. However, in this protocol the tag uses each PID value only once, where the values of PID are selected from a finite pre-shared set between tag and reader. The success probability of our tag impersonation attack against ARAP is "1" and the complexity of attack is two runs of protocol. In addition, a similar improvement can be applied to ARAP protocol as well – by changing the XOR operation of $R_t \oplus R_r \oplus PID$ to the concatenation operation $R_t \| R_r \| PID$ (See Figure 4. We omit the details here. After that, our attack can be thwarted since each small modification to R_t or R_r will results in significant change of the output of the hash value. For more details see Table III.

V. TAG IMPERSONATION ATTACK ON HABIBI AND GARDESHI PROTOCOL AND IMPROVING IT

In this section, we analyze the security of another mutual authentication protocol for RFID systems, which has been recently proposed by Habibi and Gardeshi [6]. This protocol, which is an EPC Class-1 Generation-2 standard [4, 5] compliant protocol, has been proposed as an improvement to its predecessor which has been analyzed by them [6] and Yoon [31]. In this section, we use the following notations which are depicted in Table IV.

Similar to the Luo et al. [9] and ARAP [24] protocols, Habibi and Gardeshi protocol randomizes each authentication session using two random values R_r and R_t, respectively generated by the reader and the tag. In addition, the tag and reader update their secret keys after each successful run of protocol.

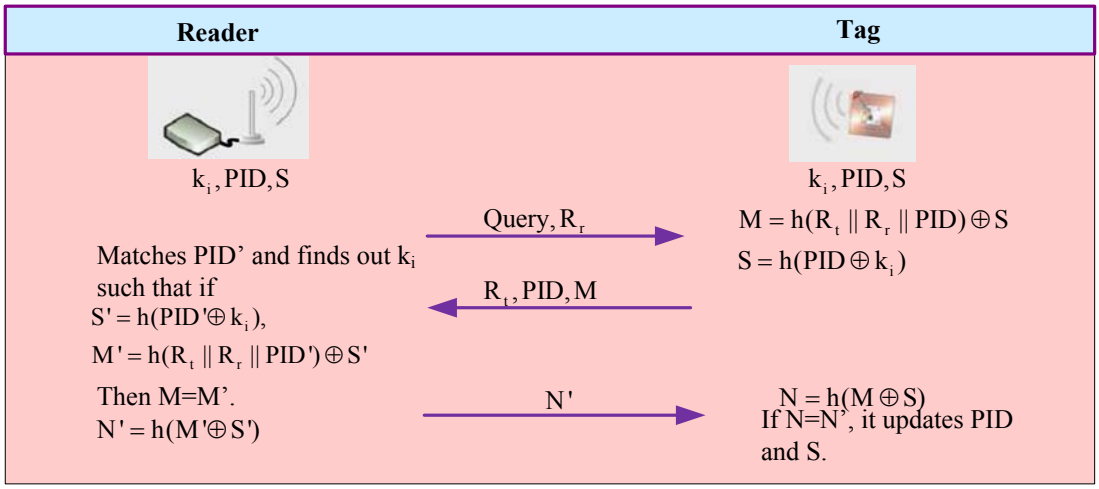

Figure 4. The revised protocol of ARAP.

TABLE III. PERFORMANCE AND COMPLEXITY COMPARISON BETWEEN THE ORIGINAL ARAP AND THE IMPROVED PROTOCOLS. IN THIS TABLE EACH ENTRY DENOTES NUMBER OF BITS AND L DENOTES THE BIT LENGTH OF PARAMETERS AND T, R AND T.I.A. DENOTE TAG, READER AND TAG IMPERSONATION ATTACK RESPECTIVELY.

Protocol	# ⊕ in T	# ⊕ in R	# || in T	# || in R	# Hash in T
ARAP	5L	5L	0	0	3L
Improved ARAP	3L	3L	2	2	5L

Protocol	# Hash in R	T storage	R storage	# Transfer	T.I.A.
ARAP	3L	3L	3L	6L	Yes
Improved ARAP	5L	3L	3L	6L	No

The Habibi and Gardeshi protocol which is depicted in Figure 5 works as follows:

TABLE IV. HABIBI AND GARDESHI PROTOCOL NOTATIONS

EPC_s	The EPC code is divided into six 16-bit words and they are XORed to form EPCs.
DATA	The corresponding information for the tag kept in the back-end database
K_i	The authentication key stored in the tag to be used at the $(i+1)^{th}$ session
P_i	The access key stored in the tag to authenticate the back-end database at the $(i+1)^{th}$ session.
K_{old} and K_{new}	The old and new authentication key stored in the back-end database respectively
P_{old} and P_{new}	The old and new access key stored in the back-end database respectively
C_i	The index of the record of the i^{th} tag's information in back-end database stored in the tag
C_{old} and C_{new}	The old and new back-end database index for the i^{th} tag, respectively
⊕	Exclusive-or operation
RID	The reader identification number

A. *Initialization Phase*

In this phase, the manufacture generates random values for K0 and P0 and respectively and sets the values of the record in the tag, i. e. $K_i = K0$, $P_i = P0$, $C_i = 0$, and the corresponding record in the back-end database $K_{old} = K_{new} = K0$, $P_{old} = P_{new} = P0$, $C_{old} = C_{new} = 0$.

B. *Authentication Phase*

The authentication phase of the improved Yeh et al. 's protocol at its $(i+1)^{th}$ run is as follows:

1. The reader generates a random number R_r and sends it to the tag.

2. The tag receives R_r, generates a random number R_t, computes M_1, D, E as below and sends M_1, D, C_i, and E to the reader:

$$M_1 = PRNG(EPC_S \oplus R_r \oplus R_t) \oplus K_i$$
$$D = R_t \oplus K_i$$
$$E = R_t \oplus PRNG(C_i \oplus K_i)$$

3. Once the reader receipts the message, it computes $V = h(RID \oplus R_r)$ and forwards M_1, D, C_i, E, R_r, V to the back-end database.

4. The back-end database receives M_1, D, C_i, E, R_r and V. After the receiving these values, it proceeds as follows:

– For each stored RID in the database, computes $h(RID \oplus R_r)$ and compares it with the received V . In the case of equality, the back-end database authenticates the reader.

– If $C_i = 0$, which means that it is the first access to the tag, it proceeds as follows iteratively:

• picks up an entry $(K_{old}, P_{old}, C_{old}, K_{new}, P_{new}, C_{new}, RID, EPC_S, DATA)$ stored in database,

• verifies whether $M_1 \oplus K_{old} \stackrel{?}{=} PRNG(EPC_S \oplus R_r \oplus D \oplus K_{old})$. If "Yes" marks X as old.

• verifies whether $M_1 \oplus K_{new} \stackrel{?}{=} PRNG(EPC_S \oplus R_r \oplus D \oplus K_{new})$ .If "Yes" marks X as new.

– else, uses C_i as an index to find the corresponding record in the database.

• If the record is found in its records for the field C_{old}, mark X as old and if it is in its records for the field C_{new} mark X as new.

• verifies whether– $PRNG(EPC_S \oplus R_r \oplus D \oplus K_X) \oplus K_X \stackrel{?}{=} M_1$. If "No" the protocol aborts.

– verifies whether $R_t \oplus PRNG(C_X \oplus K_X) \stackrel{?}{=} E$ .If "No" the protocol aborts.

– computes M_2 M2 and Info as follows and forwards them to the reader:

$M_2 = PRNG(EPC_S \oplus R_t) \oplus P_X$

$Info = DATA \oplus RID$

– If X = new, updates the database as follows:

$K_{old} = K_X$

$K_{new} = PRNG(K_X)$

$P_{old} = P_X$

$P_{new} = PRNG(P_X)$

$C_{new} = PRNG(R_t \oplus R_r)$

– else, updates the database as follows:

$C_{new} = PRNG(R_t \oplus R_r)$

5. Once the reader receipts the message, it forwards M_2 to the tag.

6. Once the tag receipts the message, it proceeds as follows:

– verifies whether $PRNG(EPC_S \oplus R_t) \stackrel{?}{=} M_2 \oplus P_i$. If "No" the protocol aborts.

– authenticates the back-end database.

– updates the contents kept inside as follows:

$K_{i+1} = PRNG(K_i)$

$P_{i+1} = PRNG(P_i)$

$C_{i+1} = PRNG(R_t \oplus R_r)$

Now, we present an attack almost similar to the attacks presented in sections 2 and 4 which can impersonate a legitimate tag without any knowledge of the secret values K_i and P_i. To impersonate the tag T_i, the adversary A follows the steps described as below:

1. A supplants R and sends Query and R_r to the target T_i and receives its response which is M_1, D, C_i and E, where $M_1 = PRNG(EPC_S \oplus R_r \oplus R_t) \oplus K_i$, $D = R_t \oplus K_i$ and $E = R_t \oplus PRNG(C_i \oplus K_i)$.

2. In the next session, when R sends Query and R'_r, A impersonates T_i and sends M'_1, D′, C'_i and E′, where $M'_1 = M_1$, $C'_i = C_i$, $D' = D \oplus R_r \oplus R'_r$ and $E' = E \oplus R_r \oplus R'_r$ to the reader.

3. Once the reader receipts the message, it computes $V' = h(RID \oplus R'_r)$ and forwards M'_1, D′, C_i, E′, R'_r, V′ to the back-end database.

4. The back-end database receives M'_1, D′, C'_i, E′, R'_r and V′. After the receiving these values, it proceeds as follows:

– For each stored RID in the database, computes $h(RID \oplus R'_r)$ and compares it with the received V′. In the case of equality, the back-end database authenticates the reader.

– If $C'_i = 0$, which means that it is the first access to the tag, it proceeds as follows iteratively:

• picks up an entry $(K_{old}, P_{old}, C_{old}, K_{new}, P_{new}, C_{new}, RID, EPC_S, DATA)$ stored in database,

•verifies whether $M_1 \oplus K_{old} \stackrel{?}{=} PRNG(EPC_S \oplus R_r \oplus D \oplus K_{old})$. If "Yes" marks X as old.

•verifies whether $M_1 \oplus K_{new} \stackrel{?}{=} PRNG(EPC_S \oplus R_r \oplus D \oplus K_{new})$ .If "Yes" marks X as new.

– else, uses C'_i as an index to find the corresponding record in the database.

• If the record is found in its records for the field C_{old}, mark X as old and if it is in its records for the field C_{new} mark X as new.

•verifies whether $PRNG(EPC_S \oplus R'_r \oplus D' \oplus K_X) \oplus K_X \stackrel{?}{=} M'_1$ which it is.

– verifies whether $R'_t \oplus PRNG(C_X \oplus K_X) \stackrel{?}{=} E'$ which it is.

– computes M'_2 and Info as follows and forwards them to the reader:

$M'_2 = PRNG(EPC_S \oplus R_t) \oplus P_X$

$Info = DATA \oplus RID$

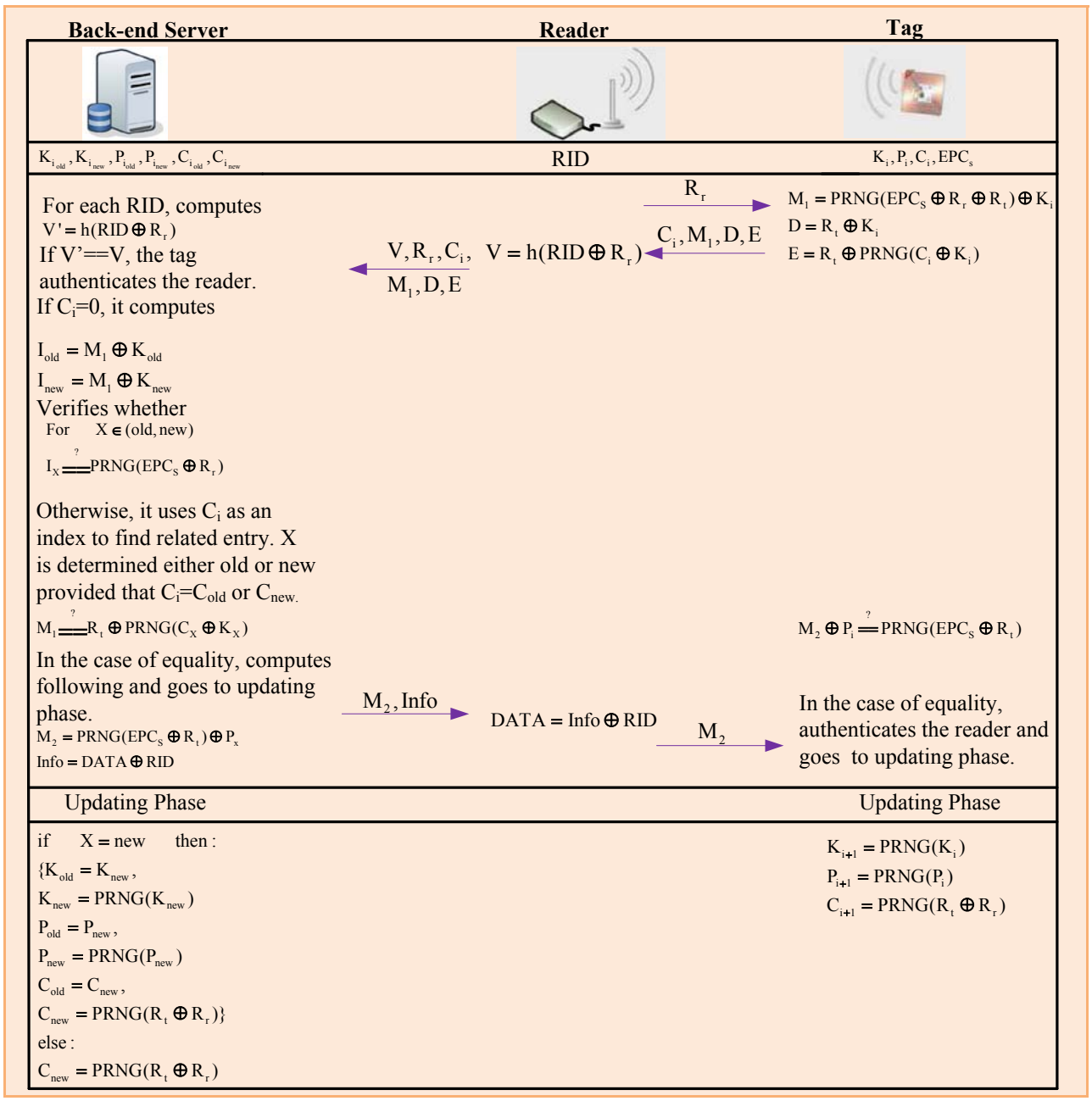

Figure 5. The Mutual Authentication Protocol proposed by Habibi and Gardeshi.

TABLE V. PERFORMANCE AND COMPLEXITY COMPARISON BETWEEN THE HABIBI AND GARDESHI PROTOCOL (HG) AND THE IMPROVED PROTOCOL (IHG). IN THIS TABLE EACH ENTRY DENOTES NUMBER OF BITS AND *L* DENOTES THE BIT LENGTH OF PARAMETERS AND *T, R, S* AND *T.I.A.* DENOTE TAG, READER, SERVER AND TAG IMPERSONATION ATTACK RESPECTIVELY.

Protocol	# ⊕ in T	# ⊕ in R	# ⊕ in S	# Hash in T	# Hash in R	# Hash in S	# PRNG in T
HG	9L	2L	9L	0	L	L	6L
IHG	10L	2L	9L	0	L	L	6L

Protocol	# PRNG in R	# PRNG in S	T storage	R storage	S storage	# Transfer	T.I.A.
HG	0	6L	4L	L	6L	14L	Yes
IHG	0	6L	4L	L	6L	14L	No

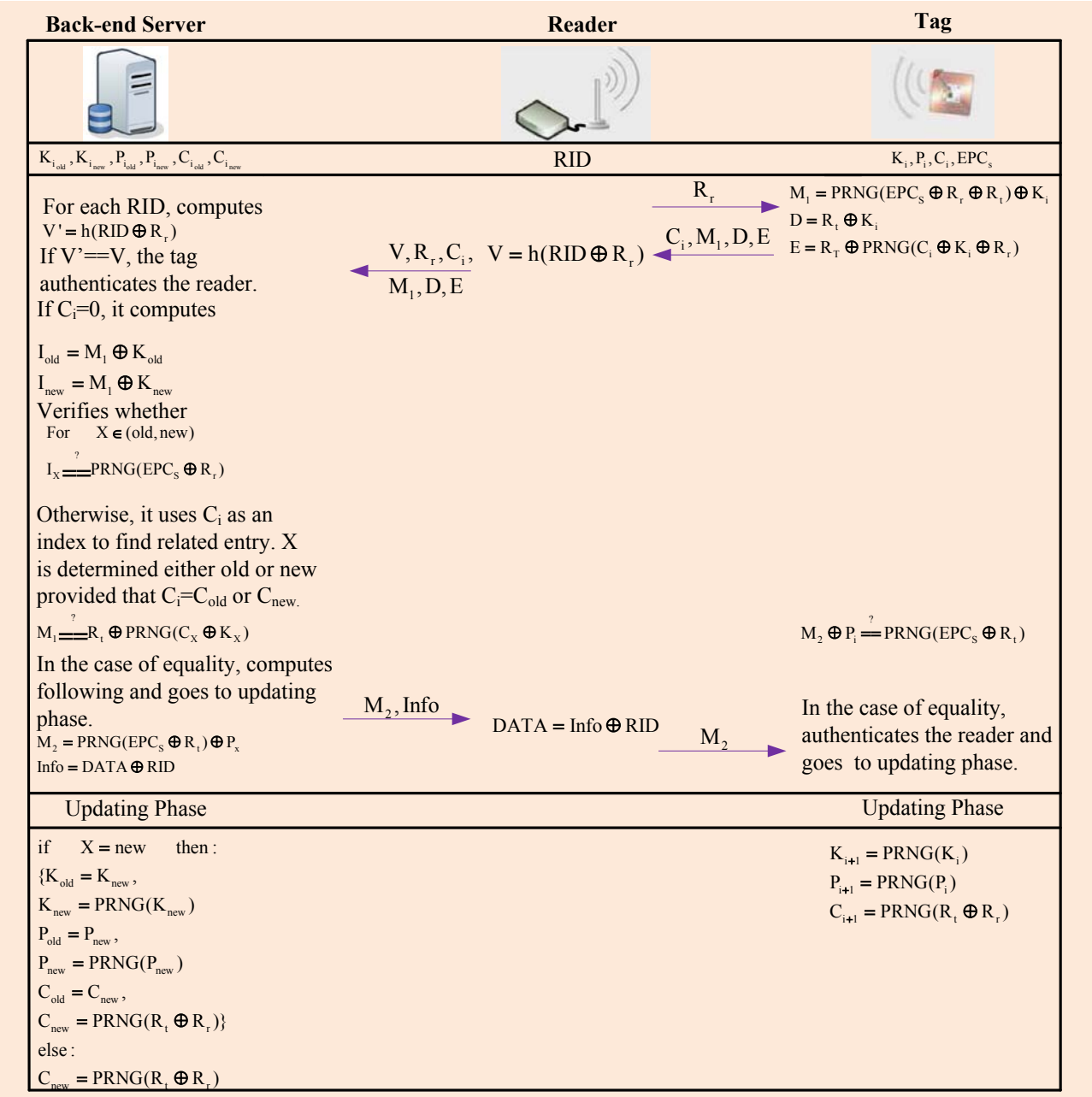

Figure 6. The revised protocol of Habibi and Gardeshi.

–If X = new, updates the database as follows:

$K_{old} = K_X$

$K_{new} = PRNG(K_X)$

$P_{old} = P_X$

$P_{new} = PRNG(P_X)$

$C_{new} = PRNG(R_t \oplus R_r)$

– else, updates the database as follows:

$C_{new} = PRNG(R_t \oplus R_r)$

5. Once the reader receipts the message, it forwards M_2 to the tag (which is the adversary here).

The whole idea of the above attack is similar to what we used to impersonate the tag for Luo et al. protocol and ARAP protocol. However, in this protocol the tag and the database update the secret values, but database keeps a record of old values, and the given attack works if the legitimate reader and the target tag have not been evolved on more than one successful run of protocol after the eavesdropping data by the adversary in step 1 of the given attack. The success probability of our tag impersonation attack against Habibi and Gardeshi protocol is "1" and the complexity of attack is two runs of protocol. As a countermeasure, we suggest to include R_r in the calculation of E as $E = R_t \oplus PRNG(C_i \oplus K_i \oplus R_r)$. After that, our attack can be thwarted because it cannot adapt the eavesdropped value of E to the next session. The improved protocol of Habibi and Gardeshi protocol is depicted in Figure 6. For more details see Table V.

VI. CONCLUSION

In this paper, we presented tag impersonation attacks against three recent RFID authentication protocols that have been recently proposed by Luo et al., Shen et al. and Habibi and Gardeshi, respectively. However, the proposed approach may be applicable to other protocols that follow the same strategy to randomize their session, e.g. Song-Mitchell [25], Wei et al. [28] and Yoon [31] protocols. The success probabilities of the presented attacks are "1" and the complexity of them is two runs of protocol. It is strange that several protocols suffer from the same weakness and it is shows that a designer should be careful with the usage of XOR operation in the protocol because it is often vulnerable to active attacks. In this paper, we also proposed the improved version of these protocols to prevent the proposed attacks. At the improved protocols, we almost replaced the XOR operation by concatenation. However, this improvement may enforce the tag and the reader to encrypt (decrypt)/ hash longer messages which reduce the protocol efficiency.

REFERENCES

[1] Y.-Y. Chen, M.-L. Tsai, and J.-K. Jan. "The design of RFID access control protocol using the strategy of indefinite-index and challenge –response", In Computer Communication, vol. 34, 2011, pp. 250–256.

[2] H.-Y. Chien. "SASI: A New Ultralightweight RFID Authentication Protocol Providing Strong Authentication and Strong Integrity. IEEE Transactions on Dependable and Secure Computing", vol. 4, no. 4 , December 2007, pp. 337–340.

[3] J.-S. Cho, S.-S. Yeo, and S. K. Kim," Securing against brute-force attack: A hash-based RFID mutual authentication protocol using a secret value," In Comput.Commun,doi:10.1016/j.comcom.2010.02.029, 2010.

Class-1 generation 2 UHF air interface protocol standard version 1.2.0, Gen2, 2008. http://www.epcglobalinc.org/standards/.

[4] EPC Tag data standar dversion 1.4.2008. http://www.epcglobalinc.org/standards/. Yearly report on algorithms and keysizes, Technical Report D.SPA.13Rev.1.0,ICT-2007-216676,. In Gen2. ECRYPT, 2010.

[5] M. H. Habibi and M. Gardeshi," Cryptanalysis and Improvement on a New RFID Mutual Authentication Protocol Compatible with EPC Standard," In 8th International ISC Conference on Information Security and Cryptology 2011(ISCISC'11), 2011, pp. 49–54.

[6] H.Y.Chien," Secure access control schemes for RFID systems with anonymity", In Proceedings of the 7th International Conference on Mobile Data Management (MDM 2006), 2006.

[7] T. Li and R. H. Deng," Vulnerability Analysis of EMAP - An Efficient RFID Mutual Authentication Protocol", In Second International Conference on Availability, Reliability and Security -AReS 2007, Vienna, Austria, April 2007.

[8] Y. Luo, Q. Chai, G. Gong, and X. Lai," A lightweight Stream Cipher WG-7 for RFID Encryption and Authentication," In IEEE Globecom 2010 proceedings, 2010.

[9] P. Peris-Lopez, J. C. H. Castro, J. M. Est´evez-Tapiador, and A. Ribagorda," Advances in Ultralightweight Cryptography for Low-Cost RFID Tags: Gossamer Protocol," In K.-I. Chung, K. Sohn, and M. Yung, editors, WISA, vol. 5379 of LNCS, Springer, 2008, pp. 56–68.

[10] R. C. Phan," Cryptanalysis of a New Ultralightweight RFID Authentication Protocol-SASI", In IEEE Transactions on Dependable and secure Computing.

[11] A. Sadighian and R. Jalili," FLMAP: A fast lightweight mutual authentication protocol for RFID systems," In ICON,. ,IEEE, 2008, pp. 1–6.

[12] A. Sadighian and R. Jalili," AFMAP: Anonymous Forward-Secure Mutual Authentication Protocols for RFID systems", In R. Falk, W. Goudalo, E. Y. Chen, R. Savola, and M. Popescu, editors, The Third IEEE International Conference on Emerging Security Information, Systems and Technologies(SECURWARE 2009), , Athens, Greece, 2009, pp. 31–36.

[13] M. Safkhani, N. Bagheri, and M. Naderi," Cryptanalysis of Chen et al.'s RFID Access Control Protocol", Cryptology ePrint Archive, Report 2011/194, 2011. http://eprint.iacr.org/.

[14] M. Safkhani, N. Bagheri, and M. Naderi," Vulnerabilities in a new RFID access control protocol", In 6th International Conference on Internet Technology and Secured Transactions (ICITST 2011), Abu Dhabi, UAE, Dec. 2011.

[15] M. Safkhani, N. Bagheri, M. Naderi, Y. Luo, and Q. Chai," Tag Impersonation Attack on Two RFID Mutual Authentication Protocols", In ARES, 2011.

[16] M. Safkhani, N. Bagheri, M. Naderi, and S. Sandhya," Security analysis of LMAP++, an RFID authentication protocol", In 6th International Conference on Internet Technology and Secured Transactions (ICITST 2011), Abu Dhabi, UAE, Dec. 2011.

[17] M. Safkhani, N. Bagheri, P. Peris-Lopez, M. Naderi, and J. C. Hernandez-Castro," Cryptanalysis of Cho et al.'s Protocol, A Hash-Based Mutual Authentication Protocol for RFID Systems", Cryptology ePrint Archive, Report 2011/331, 2011. http://eprint.iacr.org/.

[18] M. Safkhani, N. Bagheri, S. Sandhya, M. Naderi, and hamid Behnam. On the security of mutual authentication protocols for rfid systems : the case of wei et al.'s protocol. In DPM 2011 , vol. 7122 of LNCS, 2011.

[19] M. Safkhani and M. Naderi," Cryptanalysis and Improvement of a Lightweight Mutual Authentication Protocol for RFID system", In 7th International ISC Conference on Information Security and Cryptology 2010(ISCISC'10), 2010, pp. 57–59.

[20] M. Safkhani, M. Naderi, and N. Bagheri. Cryptanalysis of AFMAP. In IEICE Electronics Express, vol. 7, 2010, pp. 1240–1245.

[21] M. Safkhani, M. Naderi, N. Bagheri, and S. K. Sanadhya," Cryptanalysis of Some Protocols for RFID Systems", Cryptology ePrint Archive, Report 2011/061, 2011. http://eprint.iacr.org/.

[22] M. Safkhani, M. Naderi, and H. F.Rashvand," Cryptanalysis of the Fast Lightweight Mutual Authentication Protocol (FLMAP)", In International Journal of Computer & Communication Technology (IJCCT), vol. 2, 2010, pp. 182–186.

[23] J. Shen, D. Choi, S. Moh, and I. Chung," A Novel Anonymous RFID Authentication Protocol Providing Strong Privacy and Security", In 2010 International Conference on Multimedia Information Networking and Security, 2010.

[24] B. Song and C. J. Mitchell," RFID Authentication Protocol for Low-cost Tags", In WiSec' 08, 2008, pp. 140–147.

[25] H.-M. Sun and W.-C. Ting," A Gen2-Based RFID Authentication Protocol for Security and Privacy", In IEEE Transactions On Mobile Computing, vol. 8, 2009, pp. 1052–1062.

[26] C. C. Tan, B. Sheng, and Q. Li," Secure and Serverless RFID Authentication and Search Protocols", IEEE Transactions on Wireless Communications, vol. 7, no. 4, 2008, pp. 1400–1407.

[27] C.-H. Wei, M.-S. Hwang, and A. Y. Chin," A mutual authentication protocol for RFID", IT Professional, vol. 13, no. 2, 2011, pp.20–24.

[28] R. Xueping and X. Xianghua," A Mutual Authentication Protocol For Low-cost RFID System", In 2010 IEEE Asia-Pacific Services Computing Conference, 2010, pp. 632–636.

[29] T.-C. Yeh, Y.-J. Wang, T.-C. Kuo, and S.-S. Wang," Securing RFID systems conforming to EPC class 1 generation 2 standard", Expert Syst. Appl, vol. 37, no. 12, 2010, pp. 7678–7683.

[30] E.-J. Yoon," Improvement of the securing RFID systems conforming to EPC class 1 generation 2 standard", Expert Systems with Applications, In Press, Corrected Proof, 2011.

Nasour Bagheri is a lecturer at Electrical Engineering Department, Shahid Rajaee Teacher Training University, Tehran, Iran. He is the author of over 40 articles in information security and cryptology. Homepage of the author is available at: http://n-bagheri.srttu.ir .

Masoumeh Safkhani is a Ph.D. candidate at Electrical & Electronics Engineering Department of Iran University of Science & Technology (IUST). She is the author of 10 articles in cryptology. Her current research interest includes RFID security .

Majid Naderi is a professor at Electrical & Electronics Engineering Department of IUST. He is the co-founder of secure communication group of IUST. Homepage of the author is available at: http://ee.iust.ac.ir/Naderi/index.htm.

Yiyuan Luo is a Ph.D. candidate at Department of Computer Science & Engineering, Shanghai Jiao Tong University, China. He is currently interested in all topics about discrete mathematics, especially in algebraic structure, number theory and combinatorics. In the area of cryptography, he is interested in the analysis of cryptography hash functions, block ciphers.

Qi Chai holds a Ph.D. in Computer Science from Department of Electrical & Computer Engineering University of Waterloo in 2011, and is currently a visiting researcher in Department of Electrical & Computer Engineering University of Waterloo. His main research interests are lightweight cryptographic primitives (design and cryptanalysis of block ciphers, stream ciphers, hash functions, etc.), high performance computing for cryptography & cryptanalysis (CUDA, cell broadband engine, FPGA array, etc.), wireless security and physical layer security (wiretap channel, channel coding, etc.), security and privacy in cloud computing, security in RFID systems and wireless sensor networks (WSNs), advanced algorithms and data structures, and information systems (database, CMS, etc.).