Wireless Pers Commun (2015) 83:2607–2621
DOI 10.1007/s11277-015-2558-0

Security Analysis and Strengthening of an RFID Lightweight Authentication Protocol Suitable for VANETs

Fereidoun Moradi[1] · **Hamid Mala**[1] · **Behrouz Tork Ladani**[1]

Published online: 14 April 2015

Abstract Due to the storage capacity and computational power restrictions of low-cost RFID tags based on the EPC-C1G2 standard, most of the existing authentication protocols seem too complicated to be appropriate for these tags; thus the design of authentication protocols compliant with the EPC-C1G2 standard is a big challenge. Recently, a lightweight mutual authentication protocol for RFID conforming to the EPC-C1G2 standard was proposed by Caballero-Gil et al. aiming to be used in VANETs. This scheme does not rely on RFID readers as they are portable. Instead, it bases security on trust in the server because all shared secrets are stored only by the tag and the server with no possible access by the reader at any time. In this paper, we prove that this scheme is vulnerable to de-synchronization attack and suffers from the information leakage with a complexity of about 2^{16} offline PRNG evaluations which is completely affordable by a conventional adversary. In addition, we present a simple tag impersonation attack against this protocol. To counteract such flaws, we improve the Caballero-Gil et al. scheme to present a new RFID authentication protocol, entitled CG+, so that it provides the claimed security properties.

Keywords RFID · EPC-C1G2 standard · Mutual authentication · De-synchronization · Tag impersonation

1 Introduction

Nowadays, with the development of logistics and ecommerce, the Radio Frequency Identification (RFID) technology is being employed in many applications such as public transportation pass, road traffic systems, supply chain management, e-passport, access

✉ Fereidoun Moradi
fereidoun.moradi@gmail.com; fereidoun.moradi@eng.ui.ac.ir

[1] University of Isfahan, Isfahan, Islamic Republic of Iran

control systems, etc. The RFID technology is a passive identification technology that uses radio signals to automatically identify the target and obtain its relevant data without human intervention and can work in all kinds of harsh environments. Most of the RFID systems consist of tags (transponders), readers (transceivers) and a back-end database. The tags contain a microchip (with antenna) that stores the unique tag identifier and other related information about an object. The reader is a device that can read/modify the stored information of the tags and transfer these data to the back-end database, with or without modification. The back-end database stores this information and will keep track of the data exchanged by the reader [1].

In a common RFID system, the communication channel between the server and the reader is secure, but the reader communicates wirelessly with an RFID tag via radio frequency signals, which makes the RFID system vulnerable to the eavesdropping attack, the tag impersonation attack, the tracing attack, the replay attack, the DoS (denial of service) attack, and so on. [2].

Many concerns have been expressed over the security and privacy of RFID systems. Perhaps the biggest security concerns in these systems are espionage and privacy threats. As organizations adopt and integrate RFID into their supply chain and inventory control infrastructure, more and more sensitive data will be entrusted on RFID tags. As these tags inevitably end up in consumer hands, they could leak sensitive data or be used for tracking individuals. Clearly, tracking someone is trivial if an adversary is able to actively query unique identification from tags. The RFID tags can be embedded in clothes, shoes, books, key cards, prescription bottles, and a slew of other products. Many of these tags will be embedded without the consumer ever realizing they are there [3]. In vehicular communication environments, RFID system is typically used where the tag is resided on the vehicle and the reader is located on the road and through vehicular ad hoc networks (VANET), road safety improved and road traffic optimized. In these environments, it is essential to make sure that life-critical information cannot be illegally inserted or modified by an adversary, and it should also protect the privacy of the drivers and passengers as far as possible [4].

Rather than simply trying to glean data from legitimate tags, adversaries might try to imitate tags to readers. This is a threat to RFID systems currently being used for access control and payment systems. The real risk is someone able to skim tags wirelessly for information that can be used to produce forgeries. For instance, if tags simply respond with a static identification number, skimming is trivial. The United States Food and Drug Administration (FDA) proposed attaching RFID tags to prescription drug bottles as a pedigree [5]. Someone able to produce forgeries could steal legitimate shipments and replace them with valid-looking decoys, or could simply sell counterfeit drugs with fake pedigree labels [3].

So, the security and Privacy problem has become an obstacle to using the RFID technology widely. To heal these security weaknesses existing in the RFID system, many researchers have paid more attention to the RFID authentication protocols based on cryptographic mechanisms in the early research of the RFID security mechanism. Nevertheless, due to the storage and computation limitation of the low-cost tags, it is not realistic to design a protocol based on complicated cryptographic algorithms. The EPC Class-1 Generation-2 (short as EPC-C1G2) standard, which is suitable for low-cost RFID tags and proposed by Electronic Product Code (EPC) global organization [6], defines a much stricter framework including the tags' functions and operations. In the EPC-C1G2 standard, the allowable operations of tags are restricted to some simple operations such as Cyclic Redundancy check Code (CRC), Pseudo Random Number Generator (PRNG), and

bitwise XOR, while the hash operation is not available in this standard [7, 8]. In this case, those RFID authentication protocols based on the hash function are too complicated for EPC-C1G2 tags [9]. Therefore, the design of RFID authentication protocols conforming to the EPC-C1G2 standard becomes one of challengeable topics in RFID security.

2 Related Works

In this section, we review some previous works related to lightweight RFID authentication protocols. However, all of these proposed schemes have certain flaws and vulnerabilities.

In 2003, Vajda and Buttyan developed a set of five lightweight RFID authentication protocols and also gave a brief analysis. Each one of the protocols is extremely lightweight in terms of resources required, and is considered suitable for resource limited devices, like RFID tags [10]. Defend et al. [11] showed that their XOR and SUBSET protocols provide inadequate protection against passive and active adversaries. Their attack exploits certain statistical properties of the bit string and determines the correct key value with high probability.

Peris-Lopez et al. [12–14] proposed a series of ultra-lightweight authentication protocols which only use the most basic operations such as bitwise XOR, bitwise OR, bitwise AND and an addition of module 2^m. Later it was shown that these protocols are prone to de-synchronization attack and full disclosure attack [15, 16]. In order to improve the security of Peris-Lopes's protocols, Chien proposed a new ultra-lightweight RFID authentication protocol providing strong authentication and strong integrity (SASI). In this protocol, not only the old key but also the next key are stored in the memory of tags to resist the de-synchronization attack [17]. However, Sun et al. [18] have found two de-synchronization attacks to break SASI protocol.

Since the publication of EPC standard, many protocols have been proposed to comply with this standard. Duc et al. [7] proposed a new scheme, which only used CRC, XOR and PRNG to guarantee the interactive information security, and declared it can achieve mutual authentication between the tag and the reader as well as the synchronous updating of secret key. But later, researchers found that Duc et al.'s protocol is prone to the de-synchronization attack and it cannot ensure the forward security.

In 2007, Chien and Chen proposed an improved RFID authentication protocol, that uses two types of keys to defend against DOS attack that cause interruptions of synchronization between the tags and the server [19]. But soon after, Peris-Lopez et al. [20] pointed out that their scheme cannot resist tag and reader impersonation, tracing and de-synchronization attacks.

Konidala et al. [21] proposed a simple and cost-effective RFID tag reader mutual authentication scheme to improve the security level of the EPC-C1G2 RFID standard. This scheme utilizes the tag's *Access* and *Kill* passwords and achieves the following three goals: detecting cloned tags, warding off malicious snooping readers, and enabling the manufacturer to implicitly keep track its genuine products. However, their scheme is known to be flawed and the adversary can retrieve most of the secret password's bits efficiently [22]. In 2010 and 2012, to solve Konidala et al. protocol's weakness two novel protocols have been proposed by Huang et al. [23, 24]. These protocols do not use any standard cryptographic primitives and attempt to provide the desired security by simple logical operations. However, in 2013, Aghili et al. [25] showed that an adversary could determine whole password of these protocols with a good probability at a cost of a single

query to the target tag. Moreover, many other researchers have tried to analyze the security of EPC-compliant schemes, or improve the vulnerable schemes [7, 12, 13, 26–30].

In 2012, Caballero-Gil et al. [31] presented a new solution for mutual authentication conforming to the EPC-C1G2 standard which is completely different from all the aforementioned protocols. Their scheme does not rely on RFID readers, instead it bases the security on trust in server. In this study, we show that Caballero-Gil et al. protocol does not provide the desired security and we present an approach to efficiently retrieve the secret identification value of the tag. The interesting property of this attack is its passiveness. The main cost of this attack is about 2^{16} offline PRNG evaluations which can be easily provided by an ordinary adversary. Moreover, we prove that this scheme is vulnerable to de-synchronization and tag impersonation attacks. At last, we propose a modified version of Caballero-Gil protocol, denoted by CG+ which provides significant security compared to its predecessor.

Paper Organization: In Sect. 3 some preliminaries and notations are introduced. We describe Caballero-Gil et al. protocol in Sect. 4. De-synchronization attack, information leakage and tag impersonation attack against Caballero-Gil et al. protocol are presented in Sect. 5. In Sect. 6 we describe CG+ protocol which is the improved version of Caballero-Gil et al. protocol and investigate its security. Finally, the paper is concluded in Sect. 7.

3 Preliminaries

Throughout the paper, we use the following notations in Table 1.

Table 1 Notations

Notations	Description
T_i:	ith RFID tag
R_j:	jth RFID reader
s:	A 16-bit seed chosen by the reader
N_1:	A 16-bit value which is built by the PRNG-function of reader
N_2:	A 16-bit random number generated by the tag
ID_{T_i}:	The 16-bit identity of tag T_i
$ID_{next_{T_i}}$:	The next identity of tag T_i
$ID_{current_{T_i}}$:	The current identity of tag T_i which is used in the current session
SSK_{T_i}:	The 16-bit secret key which is shared between the server and tag T_i
$SSK_{next_{T_i}}$:	The next secret key which is shared between the server and tag T_i
$SSK_{current_{T_i}}$:	The current secret key of tag T_i which is shared between it and the server, and used in current session
K:	The shared session key
$\oplus$:	Bit-wise XOR operation
$B \leftarrow A$:	To assign the value of A to B
PRNG:	The pseudo-random number generator with 16-bit output length

4 Description of Caballero-Gil et al.'s Protocol

We now give a brief description of Caballero-Gil et al.'s protocol. This scheme is used by reader and tag in order to mutually authenticate each other and to establish a shared session key.

The authors of protocol proposed a new method for authentication with privacy protection, in compliance with the standard features described in EPC-C1G2. Their scheme does not rely on RFID readers due to their portability. Instead, their proposal bases its security on trust in the server as all shared secrets are stored only by the tag and the server, with no possible access by the reader at any time. They have supposed that each tag T_i of this protocol keeps a pair $\{ID_{T_i}, SSK_{T_i}\}$ and the server also keeps corresponding pair of each tag T_i as a entry in its database. It is also assumed that both reader and tag are able to use a secure pseudo-random number generator PRNG. The mutual authentication protocol of Caballero-Gil et al. as depicted in Fig. 1 is described step by step as follows.

1. The reader chooses a random seed s to produce the 16-bit value $N_1 = PRNG(s)$, and sends it to tag T_i.
2. Upon receiving N_1, the tag computes $A = PRNG(ID_{T_i} \oplus N_1)$ and reply it to the reader.
3. The reader receives the message A and sends the values A and N_1 to the server. The server proceeds as follows.
   (a) *Tag Identification and Authentication* For any entry in database the server picks ID_{T_i}, computes $A' = PRNG(ID_{T_i} \oplus N_1)$ and compares it with the received value A to identify and authenticate the tag. The protocol aborts if the server reaches the end of records without any match. After successful tag authentication, the server sends SSK_{T_i} to the reader and goes to updating phase.
   (b) *Updating Phase* The server updates pair $\{ID_{T_i}, SSK_{T_i}\}$ of tag T_i as follows.

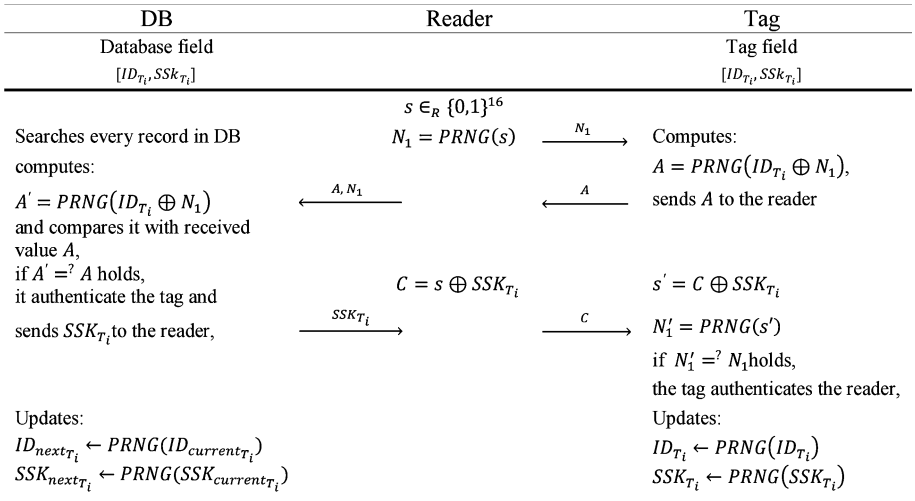

Fig. 1 Caballero-Gil et al.'s mutual authentication protocol

$$ID_{next_{T_i}} \leftarrow PRNG\left(ID_{current_{T_i}}\right)$$

$$SSK_{next_{T_i}} \leftarrow PRNG\left(SSK_{current_{T_i}}\right)$$

4. Once, the reader receives SSK_{T_i} from the server, computes $C = s \oplus SSK_{T_i}$ and then sends it to the tag T_i.
5. Then, the tag receives value C and obtains s by computing $C \oplus SSK_{T_i}$ and checks whether it corresponds to the initially received value N_1 or not. If equation holds, the reader is authenticated, and the tag updates its ID_{T_i} and SSK_{T_i} by applying *PRNG* on the previous values.

After the execution of the above five steps, both the reader and the tag can generate the same secret session key K of length 16 through the XOR operation between the value s chosen by the reader and value of $PRNG(ID_{T_i} \oplus N_1)$ generated by the tag, $K = s \oplus PRNG(ID_{T_i} \oplus N_1)$.

5 Vulnerabilities of Caballero-Gil et al. Protocol

In this section, we present several attacks against Caballero-Gil et al. protocol, including de-synchronization attack, information leakage and tag impersonation attack that rule out every security claims on this protocol. As will be seen in the following, the adversary might eavesdrop and store the messages exchanged between the parties. Such a behavior models *passive* attacks. On the other hand, adversary might intercept/delay/modify messages as he likes. Such a behavior models *active* attack.

5.1 De-synchronization Attack

Caballero-Gil et al.'s protocol cannot defend against the de-synchronization attack. The corresponding attack procedure is described here.

1. Consider a normal situation in which the server and tag T_i are synchronous with the values ID_{T_i} and SSK_{T_i}.
2. The reader and the tag start a new authentication session while the adversary eavesdrops on the channel between the reader and tag T_i.
3. Through this session they exchange the messages N_1 and A. After tag T_i authenticates itself via message A, the server updates shared secret values $\{ID_{T_i}, SSK_{T_i}\}$ of tag T_i and sends SSK_{T_i} to the reader. Upon receiving SSK_{T_i} from the server, the reader computes C and sends it to tag T_i.
4. The adversary blocks the reader's response to prevent the tag updating. As a result, the server authentication fails, and consequently tag T_i does not update its shared secret values $\{ID_{T_i}, SSK_{T_i}\}$ and still keeps previous ones.
5. At next session, once the reader sends a query to tag T_i, the tag replies the message that produced by previous identification value. The server checks whether the tag is a legal one, but it has no entry including the previous value of ID_{T_i} in its database and no match. Hence the server rejects the tag in this session and all future sessions.

5.2 Information Leakage and Tag Impersonation Attack

We present a simple passive attack which can break the Caballero-Gil et al. protocol. It is possible for a passive adversary to determine the value of ID_{T_i} stored in the server database record of T_i.

Assuming the *PRNG* is a public function and given $Y = PRNG(X)$, where Y and X are 16-bit values, it is possible for adversary to do an exhaustive search and find X as a pre-image of Y at the cost of at most 2^{16} evaluations of *PRNG* function.

Following the above observation and given tag T_i which communicates with reader R_j, the adversary eavesdrops a successful run of the protocol between T_i and R_j, and stores the transferred messages N_1 and $A = PRNG(ID_{T_i} \oplus N_1)$ of the protocol.

Let $L = \{l_1, l_2, \ldots, l_{2^{16}}\}$ be the set of all bit strings with length 16. Since ID_{T_i} is a bit string of length 16, we have $ID_{T_i} \in L$. With N_1 and $A = PRNG(ID_{T_i} \oplus N_1)$, the adversary runs the below algorithm.

(a) Chooses $l_i \in L$, for $1 \leq i \leq 2^{16}$;
(b) Computes $X = PRNG(l_i \oplus N_1)$;
(c) If $X = A$, then returns l_i as ID_{T_i}.

After at most 2^{16} executions of the algorithm, the adversary can find the correct ID_{T_i}. It is easy to see that the Caballero-Gil et al. protocol cannot resist the tag information leakage and adversary can apply PRNG on obtained ID_{T_i} then get updated value $ID_{next_{T_i}}$ for tracing the next transaction of tag T_i.

As a result of this attack and due to knowing the value of ID_{T_i}, also adversary can do tag impersonation attack on Caballero-Gil et al. protocol. The adversary listens to the communication channel between the legitimate reader R_j and the target tag T_i in the next round of the protocol to obtain N_1. Since the adversary has previous value of ID_{T_i}, it computes $PRNG(ID_{T_i})$ to get the next session secret identification $ID_{next_{T_i}}$ of tag T_i, then it computes $PRNG\left(ID_{next_{T_i}} \oplus N_1\right)$ and sends it to the reader. Because this value is calculated correctly, the server accepts the adversary and authenticates him as a legal tag.

6 Improving Caballero-Gil et al. Protocol

In this section, we revise Caballero-Gil et al. protocol with minor changes to make the resulting protocol immune against the attacks described in the previous section. The improved protocol is referred to as CG+. To improve the protocol we use this observation that given $PRNG(X)$ and $PRNG(Y)$, for $X \neq Y$, one needs $O(2^{16})$ evaluations of *PRNG* function to determine X and Y, while given $PRNG(X) \oplus PRNG(Y)$ it is not possible to determine X and Y uniquely and after $O(2^{16})$ *PRNG* evaluations, we come up with 2^{16} possible values for each of X and Y.

So, to prevent easy extraction of the tag secret identification in this protocol, it is enough to randomize and change the message transferred from the tag to the reader. In our modified scheme, tag chooses a random number N_2 and computes A and B as follows, then sends them to the reader.

$$A = PRNG(ID_{T_i} \oplus N_1) \oplus PRNG(SSK_{T_i} \oplus N_2)$$

$$B = N_2 \oplus ID_{T_i}$$

With this enhancement, it is not possible for adversary to extract ID_{T_i} in step 2 anymore. Hence, this solution fixes the basic flaw of the Caballero-Gil et al. protocol.

6.1 CG+ Protocol

Here, we elaborate the improved version of Caballero-Gil et al. protocol, CG+, which is secure against the attacks mentioned in Sect. 5 and other attacks in the context. The CG+ protocol supposes that each tag T_i keeps a pair $\{ID_{T_i}, SSK_{T_i}\}$ in its memory and the server also keeps a record of data for each tag T_i including $ID_{T_i}^{old}$, $SSK_{T_i}^{old}$, $ID_{T_i}^{new}$, $SSK_{T_i}^{new}$. The server keeps the old version of information to provide resistance against de-synchronization attack. The CG+ protocol consists of two phases: registration and mutual authentication. In registration phase, in database of server, the old and the new version of variables are set to the same value, $ID_{T_i}^{old} = ID_{T_i}^{new}$, $SSK_{T_i}^{old} = SSK_{T_i}^{new}$. After registration, tags and readers can mutually communicate. As depicted in Fig. 2, the CG+ mutual authentication protocol is described step by step as follows.

1. The reader chooses a random seed s to initialize *PRNG* in order to produce the 16-bit value N_1, and sends it to tag T_i.
2. Once the tag received N_1, it generates a random number N_2, computes A and B as below and sends these values to the reader.

DB	Reader	Tag
Database field $[ID_{T_i}^{old}, SSK_{T_i}^{old}, ID_{T_i}^{new}, SSK_{T_i}^{new}]$		Tag field $[ID_{T_i}, SSk_{T_i}]$
	$s \in_R \{0,1\}^{16}$ $N_1 = PRNG(s)$	
Searches every record in DB $(ID_{T_i}^x, SSK_{T_i}^x) \in \{(ID_{T_i}^{old}, SSK_{T_i}^{old}),$ $(ID_{T_i}^{new}, SSK_{T_i}^{new})\}$ computes: $N_2 = B \oplus ID_{T_i}^x$ $A' = PRNG(ID_{T_i}^x \oplus N_1) \oplus PRNG(SSK_{T_i}^x \oplus N_2)$ If $A' =^? A$ holds, sends $SSK_{T_i}^x \oplus PRNG(N_2)$ after successful authentication of tag, it goes to updating phase.	$\xrightarrow{N_1}$ $\xleftarrow{A,B,N_1}$ $C = s \oplus SSK_{T_i}^x \oplus PRNG(N_2)$ $\xrightarrow{SSK_{T_i}^x \oplus PRNG(N_2)}$	Generates $N_2 \in_R \{0,1\}^{16}$ computes: $A = PRNG(ID_{T_i} \oplus N_1) \oplus PRNG(SSK_{T_i} \oplus N_2)$ $B = N_2 \oplus ID_{T_i}$ $\xleftarrow{A,B}$ sends A and B to the reader $s' = C \oplus SSK_{T_i} \oplus PRNG(N_2)$ $\xrightarrow{C}$ $N_1' = PRNG(s')$ if $N_1' =^? N_1$ holds, the tag authenticates the reader, then it goes to updating phase.
Updating phase		Updating phase
$if\ x = old \begin{cases} ID_{T_i}^x = PRNG(ID_{T_i}^{old}) \\ SSK_{T_i}^x = PRNG(SSK_{T_i}^{old}) \end{cases}$ $if\ x = new \begin{cases} ID_{T_i}^{old} = ID_{T_i}^{new} \\ ID_{T_i}^{new} = PRNG(ID_{T_i}^{old}) \\ SSK_{T_i}^{old} = SSK_{T_i}^{new} \\ SSK_{T_i}^{new} = PRNG(SSK_{T_i}^{old}) \end{cases}$		$ID_{T_i} = PRNG(ID_{T_i})$ $SSK_{T_i} = PRNG(SSK_{T_i})$

Fig. 2 CG+ mutual authentication protocol

$$A = PRNG(ID_{T_i} \oplus N_1) \oplus PRNG(SSK_{T_i} \oplus N_2)$$

$$B = N_2 \oplus ID_{T_i}$$

3. The reader receives A and B, and sends them along with N_1 to the server.
4. The server receives A, B and N_1, and proceeds as follows.

   (a) *Tag Identification and Authentication* For any entry in database the server extracts N_2 from $B \oplus ID^x_{T_i}$ and computes $A' = PRNG\left(ID^x_{T_i} \oplus N_1\right) \oplus PRNG\left(SSK^x_{T_i} \oplus N_2\right)$, where $x \in \{old, new\}$ and then compares it with the received value A to identify and authenticate the tag T_i. The protocol aborts if server reaches the end of records without any match. Else if the tag is successfully authenticated, the server sends $SSK^x_{T_i} \oplus PRNG(N_2)$ to the reader.

   (b) *Updating Phase* After successful authentication of tag T_i, the server updates corresponding record in database as bellow, where x determines usage of old and new values of record of T_i in the identification and authentication phases.

$$if\ x = old \begin{cases} ID^x_{T_i} = PRNG\left(ID^{old}_{T_i}\right) \\ SSK^x_{T_i} = PRNG\left(SSK^{old}_{T_i}\right) \end{cases}$$

$$if\ x = new \begin{cases} ID^{old}_{T_i} = ID^{new}_{T_i} \\ ID^{new}_{T_i} = PRNG\left(ID^{old}_{T_i}\right) \\ SSK^{old}_{T_i} = SSK^{new}_{T_i} \\ SSK^{new}_{T_i} = PRNG\left(SSK^{old}_{T_i}\right) \end{cases}$$

5. Upon receiving $SSK^x_{T_i} \oplus PRNG(N_2)$, the reader sends to the tag the XOR operation between the seed originally chosen by itself in step 1 and the received message from the server, $C = s \oplus SSK^x_{T_i} \oplus PRNG(N_2)$.
6. The tag receives C and extracts s by XORing C and $SSK_{T_i} \oplus PRNG(N_2)$, then checks whether it corresponds to the initially received N_1. After the server successful authentication, the tag updates its record as follows.

$$ID_{T_i} \leftarrow PRNG(ID_{T_i})$$

$$SSK_{T_i} \leftarrow PRNG(SSK_{T_i})$$

At the end of the execution of these six steps in the above scheme, both the reader and the tag can generate the same secret session key K of length 16, through the XOR operation between the s chosen by the reader and the message A sent by the tag, $K = s \oplus A$.

The established shared secret session key K will be then used both by the tag and by the reader to initialize a stream cipher in order to obtain the same key stream Z to encrypt and decrypt all messages exchanged between them during that session. The key K may be also used by tag and reader during that session for fast challenge-response authentication based on symmetric cryptography.

6.2 Security Analysis of CG+ Protocol

In this subsection, we present a detailed security analysis of CG+ to show that the protocol meets resistance against the attacks presented in this paper and the other known active and passive attacks in the context.

6.2.1 Information Leakage Prevention

In CG+ protocol, the private information of tag must be kept secure to guarantee tag privacy. The tag transmits the message A and B which hides the session secret key SSK_{T_i} and secret identification ID_{T_i} to the adversary. The secret parameters included in each session are N_2,ID_{T_i},SSK_{T_i}, s. In step 1 and last step of the proposed protocol, the message$B$ of the tag and the reader response include linear combinations:

$$B = N_2 \oplus ID_{T_i}$$

$$C = s \oplus SSK_{T_i} \oplus PRNG(N_2)$$

Hence, the adversary just needs to guess two unknown values out of the four values N_2,ID_{T_i},SSK_{T_i}, s each of length 16-bit, for disclosing the aforementioned secret parameters in protocol at the cost of 2^{32}. A possible scenario could be as follows.

1. During the number of successful runs of protocol, the adversary eavesdrops on the channel between the reader and tag T_i and stores the transferred sets, e.g. $(N_1, A, B, C)_1$, $(N_1, A, B, C)_2$ and $(N_1, A, B, C)_3$. Then, adversary uses $(N_1, A, B, C)_1$ and does the following computation.
   (a) $For\, SSK_{T_i} : 0, \ldots, 2^{16}$
      (i) $For\, ID_{T_i} : 0, \ldots, 2^{16}$
         (a) $N_2 = B \oplus ID_{T_i}$,
         (b) $s = C \oplus SSK_{T_i} \oplus PRNG(N_2)$,
         (c) The adversary checks if $N_1 = PRNG(s)$ holds or not.

The complexity of this attack is equivalent to eavesdropping exchanged messages of several session of the protocol and bounded by 2^{32} off-line computations.

Although after disclosing secret parameters other attacks, e.g. tag/reader impersonation, traceability, de-synchronization attack, etc. are trivial, we proceed to discuss the security analysis based on different strategies.

6.2.2 Tag impersonation Attack Prevention

In CG+ protocol, the adversary can eavesdrop the communication and store the response messages from the tag, and then retransmit the massage to the legitimate reader in any of the protocol runs in order to impersonate the legal tag. But this is not the case since both the tag and the reader generate different random challenge number in every protocol runs. To impersonate the tag, the adversary has to generate a valid tuple (A, B). However, this tuple includes three unknown parameters, i.e. N_2, ID_{T_i} and SSK_{T_i},where N_2 is refreshed in each session and ID_{T_i} and SSK_{T_i} are renewed after each successful run of protocol. So it is impossible for the adversary to deceive the reader through replay attack.

6.2.3 Reader Impersonation Attack Prevention

To impersonate the reader, the adversary should return a valid C similar to the tag impersonation attack, replay attack does not work and the best strategy for the adversary to impersonate the reader could be sending a random value to the tag. However, since the tag has only one record of the secret parameter, the adversary's success probability in each try is bounded by 2^{-16}.

6.2.4 De-synchronization Attack Prevention

Since the server keeps the record of the old and new pair $\{ID_{T_i}, SSK_{T_i}\}$ in its database, blocking the last message does not de-synchronize the tag and the server. Even if the adversary interferences communication to cause synchronization problem, as the revised mutual authentication protocol is considered two states update procedure for the each tag, keeps the tags synchronized with the server. Hence, to de-synchronize a specific tag, the adversary should either impersonate the reader or change the last message from the reader to the tag such that the tag authenticates the reader. However, the adversary's success probability in each attempt is bounded by 2^{-16}.

6.2.5 Traceability Attack Prevention

The proposed protocol guarantees tag privacy by refreshing secret ID_{T_i} in tag and server for each session. After the successful authentication is finished, the shared secret values are updated. So, the adversary's advantage to impersonate the reader is negligible, and since each session of the protocol is randomized by the reader and the tag, it is not possible to trace the tag.

6.3 Performance Analysis of CG+ Protocol

Here, we evaluate the performance of CG + protocol in terms of computational cost, communication cost, and storage requirement. In Table 2, the performance comparison of Caballero-Gil et al. protocol and CG+ protocol is provided.

In this table l denotes the bit length of parameters which is 16 in our case and N is the total number of tags in database. The protocol of Caballero-Gil et al. denotes by CG. In addition, the secret parameters updating costs are also included in the given values for each field.

Table 2 Performance comparison between Caballero-Gil et al. and CG+ protocols

	No. of *PRNG*	No. of $\oplus$	No. of *stored* bits	No. of *transfered bits*
Server of CG	$N + 2$	N	$2l$	l
Server of CG+	$2N + 3$	$3N + 1$	$4l$	l
Tag of CG	4	2	$2l$	l
Tag of CG+	6	6	$2l$	$2l$
Reader of CG	1	1	l	$4l$
Reader of CG+	1	1	l	$5l$

This table shows that the proposed modification does not increase the computational and communication costs of the protocol extensively while it provides much better security. The descriptions of these features are provided in the following.

- Computational Cost

The main restriction of the computational ability lies on the tags. The Caballero-Gil et al. and CG+ protocols only require bitwise XOR and PRNG function on parties. These operations are very low-cost and can be efficiently implement in hardware. As shown in Table 2, in the server side of the CG+, the correct ID_{T_i} and SSK_{T_i} can be found by at most $2N + 3$ PRNG operations in normal case run of the protocol where N is the total number of tags. In the case of de-synchronization, these secret values can be found based on an average $3N + 1/2$ PRNG operations. However, de-synchronization of a tag is a special and unusual state, and the normal synchronization state only needs $N/2$ PRNG operations.

- Communication Cost

In the proposed protocol, the tag and the reader transmit messages N_1, A, B and C in order to do mutual successful authentication. The length of the total messages transmitted from a tag to the reader and from the reader to a tag is $4l$ where the length of one message is l bits (16 in our case). Hence, the CG+ protocol provides a relatively low communication cost.

- Storage Requirement

We do not change the storage requirements in the tag side. Like the Caballero-Gil et al. scheme, the CG+ protocol stores two secret values as total $2l$ bits in storage of each tag. On the other hand, in the proposed protocol, the database needs more storage to store the old shared secret values to prevent the de-synchronization between the server and the tag when the tag fails to receive the last message. In the practical applications, it is a trade-in measurement.

Therefore, the CG+ is suitable for RFID systems with limited memory space and computational power, and it can be implemented for practical secure vehicular communications.

7 Conclusions

In this paper, we presented de-synchronization attack on Caballero-Gil RFID lightweight authentication protocol and proved that this scheme suffers from information leakage vulnerability. In addition we presented a simple tag impersonation attack against this protocol. To heal the weaknesses in this protocol, an RFID authentication protocol conforming to the EPC-C1G2 standard denoted by CG+ was proposed, which guarantees tag's privacy and satisfies the security requirements. Moreover, the proposed protocol performance was analyzed and compared with Caballero-Gil et al. scheme. In terms of future work, we intend to reduce computational time in the server side and improve the protocol in order to enable the server to find the match in its database entries faster than the current exhaustive search.

Acknowledgments We like to thank the anonymous reviewer of this paper for their valuable comments.

References

1. Hunt, V. D., Puglia, A., & Puglia, M. (2007). *RFID: A guide to radio frequency identification*. NY: John Wiley & Sons.
2. Pang, L., He, L., Pei, Q., & Wang, Y. Secure and efficient mutual authentication protocol for RFID conforming to the EPC C-1 G-2 standard. In *Wireless Communications and Networking Conference (WCNC), 2013 IEEE, 2013* (pp. 1870–1875). IEEE.
3. SA, S. W. (2011). RFID (radio frequency identification): Principles and applications. Retrived from www.eecs.harvard.edu/rfid-article.pdfon,1.
4. Park, Y., Sur, C., Jung, C. D., & Rhee, K.-H. (2009). Efficient anonymous authentication protocol using key-insulated signature scheme for secure VANET. In *Mobile lightweight wireless systems* (pp. 35–44). Springer.
5. Health, U. D. O., & Services, H. (2004). Combating Counterfeit Drugs, A Report of the Food and Drug Administration. *Food and Drug Administration.* http://www.fda.gov/oc/initiatives/counterfeit/report02_04.pdf.
6. Choi, E. Y., Lee, D. H., & Lim, J. I. (2009). Anti-cloning protocol suitable to EPCglobal Class-1 Generation-2 RFID systems. *Computer Standards & Interfaces, 31*(6), 1124–1130.
7. Duc, D. N., Lee, H., & Kim, K. (2006). Enhancing security of EPCglobal Gen-2 RFID against traceability and cloning. *Auto-ID Labs Information and Communication University, White Paper*.
8. Habibi, M. H., Alagheband, M. R., & Aref, M. R. (2011). Attacks on a lightweight mutual authentication protocol under EPC C-1 G-2 standard. In *Information security theory and practice. security and privacy of mobile devices in wireless communication* (pp. 254–263). Springer.
9. Burmester, M., De Medeiros, B., Munilla, J., & Peinado, A. (2009). Secure EPC gen2 compliant radio frequency identification. In *Ad-Hoc, mobile and wireless networks* (pp. 227–240). Springer.
10. Vajda, I., & Buttyán, L. (2003). Lightweight authentication protocols for low-cost RFID tags. In *Second Workshop on security in ubiquitous computing–ubicomp 2003*.
11. Defend, B., Fu, K., & Juels, A. Cryptanalysis of two lightweight RFID authentication schemes. In *Pervasive Computing and Communications Workshops, 2007. PerCom Workshops' 07.* Fifth Annual IEEE International Conference on, 2007 (pp. 211–216). IEEE.
12. Peris-Lopez, P., Hernandez-Castro, J. C., Estevez-Tapiador, J. M., & Ribagorda, A. EMAP: An efficient mutual-authentication protocol for low-cost RFID tags. In *On the move to meaningful internet systems 2006: Otm 2006 Workshops, 2006* (pp. 352–361). Springer.
13. Peris-Lopez, P., Hernandez-Castro, J. C., Estevez-Tapiador, J. M., & Ribagorda, A. (2006). M2AP: A minimalist mutual-authentication protocol for low-cost RFID tags. In *Ubiquitous intelligence and computing* (pp. 912–923). Springer.
14. Peris-Lopez, P., Hernandez-Castro, J. C., Estévez-Tapiador, J. M., & Ribagorda, A. (2006). LMAP: A real lightweight mutual authentication protocol for low-cost RFID tags. In *Proceedings of 2nd Workshop on RFID security* (p. 6).
15. Li, T., & Deng, R. (2007). Vulnerability analysis of EMAP-an efficient RFID mutual authentication protocol. In *Availability, Reliability and Security, 2007. ARES 2007. The Second International Conference on, 2007* (pp. 238–245). IEEE.
16. Li, T., & Wang, G. (2007). Security analysis of two ultra-lightweight RFID authentication protocols. In *New approaches for security, privacy and trust in complex environments* (pp. 109–120). Springer.
17. Chien, H.-Y. (2007). SASI: A new ultralightweight RFID authentication protocol providing strong authentication and strong integrity. *Dependable and Secure Computing, IEEE Transactions on, 4*(4), 337–340.
18. Sun, H.-M., Ting, W.-C., & Wang, K.-H. (2011). On the security of Chien's ultralightweight RFID authentication protocol. *IEEE Transactions on Dependable and Secure Computing, 8*(2), 315–317.
19. Chien, H.-Y., & Chen, C.-H. (2007). Mutual authentication protocol for RFID conforming to EPC Class 1 Generation 2 standards. *Computer Standards & Interfaces, 29*(2), 254–259.
20. Peris-Lopez, P., Hernandez-Castro, J. C., Tapiador, J. M., & Van der Lubbe, J. C. (2011). Cryptanalysis of an EPC class-1 generation-2 standard compliant authentication protocol. *Engineering Applications of Artificial Intelligence, 24*(6), 1061–1069.
21. Konidala, D. M., Kim, Z., & Kim, K. (2007) A simple and cost-effective RFID tag-reader mutual authentication scheme. In *Proceedings of Int'l Conference on RFID Security* (*RFIDSec'07*) (pp. 141–152).
22. Peris-Lopez, P., Li, T., Hernandez-Castro, J. C., & Tapiador, J. M. (2009). Practical attacks on a mutual authentication scheme under the EPC Class-1 Generation-2 standard. *Computer Communications, 32*(7), 1185–1193.

23. Huang, Y.-J., Lin, W.-C., & Li, H.-L. (2012). Efficient implementation of RFID mutual authentication protocol. *Industrial Electronics, IEEE Transactions on, 59*(12), 4784–4791.
24. Huang, Y.-J., Yuan, C.-C., Chen, M.-K., Lin, W.-C., & Teng, H.-C. (2010). Hardware implementation of RFID mutual authentication protocol. *Industrial Electronics, IEEE Transactions on, 57*(5), 1573–1582.
25. Aghili, S. F., Bagheri, N., Gauravaram, P., Safkhani, M., & Sanadhya, S. K. (2013). On the security of two RFID mutual authentication protocols. In *Radio frequency identification* (pp. 86–99). Springer.
26. Burmester, M., & De Medeiros, B. The security of EPC Gen2 compliant RFID protocols. In *Applied cryptography and network security, 2008* (pp. 490–506). Springer.
27. Lo, N.-W., & Yeh, K.-H. (2007). An efficient mutual authentication scheme for EPCglobal class-1 generation-2 RFID system. In *Emerging directions in embedded and ubiquitous computing* (pp. 43–56). Springer.
28. Yoon, E.-J. (2012). Improvement of the securing RFID systems conforming to EPC class 1 generation 2 standard. *Expert Systems with Applications, 39*(1), 1589–1594.
29. Yeh, T.-C., Wang, Y.-J., Kuo, T.-C., & Wang, S.-S. (2010). Securing RFID systems conforming to EPC Class 1 Generation 2 standard. *Expert Systems with Applications, 37*(12), 7678–7683.
30. Habibi, M. H., Gardeshi, M., & Alaghband, M. R. (2011). Practical attacks on a RFID authentication protocol conforming to EPC C-1 G-2 standard. *arXiv preprint arXiv:1102.0763*.
31. Caballero-Gil, C., Caballero-Gil, P., Peinado-Domínguez, A., & Molina-Gil, J. (2012). Lightweight authentication for RFID used in VANETs. In *Computer aided systems theory–EUROCAST 2011* (pp. 493–500). Springer.

Fereidoun Moradi received his B.S. degree in Computer Engineering from Payame Noor University of Hamedan in 2012. Now he is an M.S. student at University of Isfahan since 2012. His main research interests include lightweight cryptography and cryptanalysis, RFID security, RFID authenticaton protocols.

Hamid Mala received his B.S., M.S. and Ph.D. degrees in Electrical Engineering from Isfahan University of Technology (IUT) in 2003, 2006 and 2011, respectively. He joined University of Isfahan (UI) in September 2011 as an Assistant Professor in the Department of Information Technology Engineering. His Research interests include the design and cryptanalysis of block ciphers, digital signatures and cryptographic protocols.

Behrouz Tork Ladani holds a bachelor in Computer Engineering from University of Isfahan, M.S. in Software Engineering from Amir Kabir University of Technology and a Ph.D. in Software Engineering from the University of Tarbiat Modarres. Dr. Tork Ladani joined University of Isfahan in 2005. He is currently Associate Professor and head of Department of Information Technology in this University. His research interests include Cryptographic Protocols, Access Control, Trust, and Formal verification. He is currently member of executive council of Iranian Society of Cryptology (ISC).