A Lightweight Authentication Protocol Based on Partial Identifier for EPCglobal Class-1 Gen-2 Tags

Zhicai Shi, Fei Wu, Yongxiang Xia, Yihan Wang, Jian Dai, and Changzhi Wang
School of Electronic&Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, P. R. China
Email: {szc1964, fei-wu1, x-free}@163.com, ehan@sues.edu.cn, {1101196001, 774936933}@qq.com

Abstract—RFID is a key technology that can be used to create the pervasive society. The tag is an important part of the RFID system and most popular tags are some low-cost passive tags. These tags have limited computing and storing resources, and no more attentions are paid to their security and privacy. So the application of these tags is not secure. Lightweight authentication protocols are considered as an effective method to solve the security and privacy of low-cost RFID tags. We propose a novel lightweight authentication protocol by means of some functions provided by EPCglobal Class-1 Gen-2 tags. The protocol enhances the difficulty to reveal the tag’s secrecy by using the tag’s partial identifier to generate the session messages between the tag and the reader. The tag’s partial identifier is generated randomly for each authentication. Otherwise, the tag’s identifier is randomly divided into two separate parts so as to avoid colliding from the 16-bit cyclic redundancy coding function. Some random numbers, which are generated by the tag and the reader respectively, randomize the session messages between the tag and the reader so as to resist against tracing attack and replay attack. Our proposed protocol can assure forward security and it can resist against de-synchronized attack. This protocol only uses some lightweight functions and it is very suitable to low-cost RFID tags.

Index Terms—RFID; Authentication Protocol; Privacy; Security

I. INTRODUCTION

Radio Frequency IDentification(RFID) is a pervasive technology deployed in everyday life and it uses the wireless radio-waves to automatically identify objects, without visible light and physical contact. Today, RFID systems have been successfully applied to manufacturing, supply chain management, agriculture, communication and transportation, electronic-payment, e-passport and other fields [1]. Currently, most of the RFID tags, which are widely deployed, are some low-cost passive tags. These tags are very cheap and they only have very limited storing and computing resources. They are some typical resource-constrained devices and they have not enough resources to solve their privacy and security problems. So the current passive RFID tags are almost insecure. This promising RFID technique may suffer from some privacy leakage and security threats. An adversary can trace and monitor the tagged object, even identify the person carrying the tagged object. Some typical attacks to RFID systems include eavesdropping, tracing, replaying, man-in-the-middle attack, DOS/Denial of Service, forgery (including cloning), and physical attack. To protect the privacy and security of an RFID system, some special techniques are used. Currently, these techniques are divided into two main categories: physical approaches, encryption mechanism and protocol [2,3]. The research results indicate that encryption mechanism and protocol is a more flexible and effective approach for ensuring the security and privacy of RFID systems than any other methods. The RFID authentication protocol is a special encryption protocol which is widely deployed. Now, many authentication protocols for RFID systems have been proposed. Among these protocols, the hash-based authentication protocols and HB family protocols are some typical authentication protocols. These protocols are very suitable to the resource-constrained RFID tags, but some of these protocols use Hash functions and pseudorandom generating functions and they need more computing resources. HB family protocols are based on the computation hardness of the Learning Parity with Noise(LPN) problem. These protocols can resist against passive attacks but they are provable to be vulnerable to some active attacks (e.g. man-in-the-middle attacks). Now we only use cyclic redundancy coding function, pseudorandom generating function and some simple bit operations to construct a novel lightweight authentication protocol. These functions are provided by EPCglobal Class-1 Gen-2 tags. The proposed protocol uses the randomized partial identifier of a tag so as to increase the difficulty to reveal the secrecy of the RFID system. The protocol takes less computing and storing resources than the protocols described above. So it is very suitable for the low-cost passive tags.

The paper is organized as follows. In Section II, we introduce and analyze two kinds of typical lightweight authentication protocols for RFID tags: the hash-based...
authentication protocols and HB family protocols. We describe the weakness and vulnerability of these protocols. In Section III, we only use the randomized partial identifier of a tag and propose a novel lightweight authentication protocol by means of the cyclic redundancy coding function embedded in EPCglobal Class-1 Gen-2 tags. In Section IV we give secure analysis of our proposed protocol. In Section V, we conclude our work and point out the advantages of our proposed protocol over other lightweight authentication protocols.

II. SOME TYPICAL LIGHTWEIGHT AUTHENTICATION PROTOCOLS FOR RFID SYSTEMS

An RFID system usually consists of three components: Radio Frequency(RF) tags, RF readers and a backend server, as shown by figure 1. The tag comprises a chip and an antenna that are together attached to an identified object. It has very limited storing and computing resources. Readers query tags using a radio frequency signal to obtain the tag’s identifier. A backend server is simply called a Verifier and it stores the detail information about the tagged objects. Readers have electric power enough to transmit signals over longer distance and tags only have limited electric energy to transmit signals over shorter distance. So the wireless communication channels between readers and tags are asymmetric. We call the channel from readers to tags as forward channel and the channel from tags to readers as backward channel. These channels are open and insecure. Most secure problems of RFID systems are resulted from these insecure wireless channels. The function of an RFID authentication protocol is that the tag can get the mutual authentication with the reader before it is accessed. Then the authenticated reader can get the content of the legitimate tags. Moreover, the private information about the tagged objects would not be leaked to un-authenticated entities.

![Figure 1. The components of an RFID system](image)

An RFID authentication protocol is a special cryptographic protocol, where resource-constrained RFID tags are involved. This kind of protocol is called the lightweight authentication protocol. For the low-cost passive RFID tags, conventional authentication protocols that concern symmetric key encryption or public key encryption are not applicable. Therefore some special lightweight authentication protocols are proposed so as to satisfy the special requirements of passive RFID tags. Among these protocols, the hash-based authentication protocols and HB family protocols are two typical authentication protocols.

The hash-based authentication protocols use the one-way property of Hash functions to assure the integrity and privacy of the exchanged sessions during the authenticating process. S. A. Weis et. al. proposed a typical hash-based authentication protocols [4]. This protocol uses the pseudonym of the tag’s identifier, not its identifier, to complete the authentication so as to protect its privacy. But during the authenticating process the pseudonym is fixed and it is easy to trace a tag or to replay some intercepted sessions. In order to overcome the faults described above, S. A. Weis et. al. proposed another protocol, which is called Hash-lock protocol [4]. This protocol uses a pseudorandom generating function to randomize the exchanged sessions between the tag and the reader. But it still uses plain-text to transfer the tag’s identifier, so it is easy for an adversary to eavesdrop the privacy of the tag. Otherwise, it cannot resist tracing, forgery and replay attacks.

Different from the Hash-lock protocol, which only uses a hash function, M. Ohkubo et. al. proposed a hash-chain protocol and the protocol uses two different hash functions to complete the authentication [5,6]. But this protocol only completes the one-way authentication of the reader to the tag and it cannot resist against forgery and replay attacks. Sang-Soo Yeo et. al. proposed a protocol similar to the hash-chain protocol, which can resist to be traced and assure forward security [7]. Yong Ki Lee et. al. proposed an RFID authentication protocol, which is called Semi-Randomized Access Control (SRAC) [8]. The protocol completes two-way authentication between the tag and the reader and it can resist tracing, cloning and DOS attacks but not resist replay attack. Su Mi Lee et. al. used challenge-acknowledge mechanism to propose an authentication protocol, which is simply called LCAP [9]. The protocol completes two-way authentication between the tag and the reader and it can protect the privacy of an RFID system. It can resist against tracing attack by dynamical updating the tag’s identifier. But it cannot provide forward security.

Jung-Sik Cho et. al. proposed a mutual authentication protocol for RFID systems [10,11]. Their protocol uses a one-way hash function, and is claimed to provide enough security against the most common attacks for RFID systems. To protect privacy, each session of the mutual authentication is randomized by employing two different nonces, which are generated by the reader and the tag respectively. The authors claimed that their protocol can resist against de-synchronized attack, and impersonation attack. But Hyunsung Kim and Masoumeh Safkhani et. al. pointed out that the protocol proposed by Jung-Sik Cho et. al. is vulnerable to some attacks and the successful probability of the de-synchronization attack is 1 while the attack complexity is one run of the protocol, and the successful probability of the impersonation attack is 1/4 for two runs of the protocol [12,13].

J. H. Ha and S. J. Moon proposed another hash-based RFID authentication protocol and proved the protocol can assure forward security [14]. But Da-Zhi Sun and Ji-Dong Zhong found that an adversary can trace a tag by
observing some failed sessions [15]. They pointed out the protocol cannot provide forward security as claimed by J. H. Ha and S. J. Moon.

Liu Yang and Peng Yu proposed a hash-based protocol to complete three party authentication among tag, reader and verifier [16]. But for each authentication the hash function is called more than five times and the computing cost is too high for low-cost tags.

Like these hash-based authentication protocols, HB family protocols are also some typical lightweight authentication protocols. All HB family protocols rely on the computation hardness of the Learning Parity with Noise (LPN) problem to resist against passive attacks. The LPN problem is NP-Hard and currently no polynomial algorithm is known to solve it.

**Definition.** The LPN problem with security parameters \( q, k, \eta \in (0, 1/2) \) is defined as follows: given a random \( q \times k \) binary matrix \( A \), a random \( k \)-bit vector \( x \), a vector \( v \) such that \( |v| \leq \eta q \), and the product \( z = A \cdot x \oplus v \), find a \( k \)-bit vector \( i \) such that \( |A \cdot i + z| \leq \eta q \), where \(|v|\) denotes the Hamming weight of vector \( v \).

Based on LPN problem, Hopper and Blum proposed a secure human identification protocol in 2001, which is called HB protocol [17]. The protocol consists of \( r \) rounds for each authentication and the tag shares the secret key \( x \) with the reader. After \( r \) rounds the reader authenticates the tag if the tag’s response is incorrect less than \( q r \). During the authenticating process the tag injects noise into its response so as to prevent passive eavesdroppers to reveal the secret key \( x \). If an eavesdropper would like to obtain the secret key \( x \) from the observed sessions he has to solve LPN problem. But HB scheme is not secure against active attacks. It is easy for an adversary to reveal the secret key \( x \). The adversary selects the value of the challenges and sends the fake challenges to the reader. Once \( k \) equations with linearly independence have been acquired, the secret key \( x \) can be revealed by Gaussian elimination [18].

To overcome the weakness of HB protocol, Juels and Weis improved HB protocol and proposed HB+ protocol [19]. HB+ protocol added another secret key \( y \) and a blinding vector \( b \). HB+ protocol used the vector \( b \) to blind the secret key \( y \) so as to increase the difficulty to reveal the secret keys. But this protocol was proved vulnerable to some active attacks (e.g. man-in-the-middle attacks) [20, 21]. Henri Gilbert, Matthew Robshaw, and Hervé Sibert proposed a simple active attack against HB+ protocol and this attack is simply called GRS attack, which is a famous active attack.

In 2006, Julien BRINGER, Hervé CHABANNE, and Emmanuelle DOTTAX improved HB+ protocol so as to thwart GRS attack, which is called HB++ protocol [22]. For the protocol, each tag shares a unique secret \( Z \) with the reader. At the beginning phase of each authentication, two challenges are exchanged between the reader and the tag. These challenges are computed under the secret key \( Z \) by a universal hash function \( h(\cdot) \) to obtain the secret keys \( x, x', y \) and \( y' \). These secret keys are then used to perform the subsequent authentication. Julien BRINGER proved that HB++ protocol is at least as secure as HB+

protocol. For HB++ protocol, some new secret keys are generated by a hash function and this guarantees each authentication to use the different secret keys. But this increases the complexity of the protocol. The binary vectors \( A \) and \( B \) are transferred by plaintext. If they are tampered the secret keys between the tag and the reader are different and this will lead to de-synchronized attack.

To defend the active attack, J. Munilla and A. Peinado introduced the idea of round key to improve the secure performance of HB+ protocol. They proposed a new protocol called HB-MP [23]. HB-MP protocol uses two secret keys \( x \) and \( y \). Each round authentication uses the \( i \)-th bit of \( y \) to simultaneously rotate \( x \) by the tag and the reader. After some rounds \( x \) may be rotated to its initial value. To solve this problem, Xuefei Leng et. al. proposed HB-MP+ protocol [24]. This protocol uses an one-way function to randomize the rotation of the secret key \( x \) and to make the change of the key \( x \) unpredictable. This scheme avoids the repeated occurring of the round key, but the use of the one-way function increases the complexity of the protocol and it takes much time to find another vector \( b \) which equals \( a \).

In 2008, Henri Gilbert, Matthew J. B. Robshaw, and Yannick Seurin proposed their analysis on HB protocol families, and proposed a new protocol called RANDOM-HB# and its optimized version HB# [25]. RANDOM-HB# avoids many practical drawbacks of HB+ protocol and it is also provably resistant to a broader class of active attacks. However, RANDOM-HB# is required to store two random matrices \( X \) and \( Y \) as the secret keys, which is the heavy storage cost for the tags. HB# enhanced RANDOM-HB# by using Toeplitz matrices to reduce storage cost and to improve its performance. But later Khaled Ouafi et. al. [26] presented an effective man-in-the-middle attack against Random-HB# and HB#, and proved that both Random-HB# and HB# are vulnerable against this attack.

There are still new versions for the HB family protocols. Ghaith Hammouri and Berk Sunar used a physical unclonable function to propose PUF-HB protocol [27]. But they did not give any proof of security against man-in-the-middle attacks. Julien BRINGER and Hervé CHABANNE improved HB+ protocol and proposed Trusted-hb protocol to achieve resistance against man-in-the-middle attacks [28]. But D. Frumkin and A. Shamir constructed several complicated attacks and showed that Trusted-hb protocol cannot be trusted [29].

In addition to some protocols described above, other authentication protocols for RFID tags have been proposed and they usually use some different mechanism to assure the privacy and security of RFID systems. Yong-Zhen Li et. al. proposed a protocol based on partial identifier of a tag. But for this protocol, the boundary conditions of the used pseudorandom number aren’t taken into account. It only uses simple XOR operations and it is easy to guess the tag’s identifier [30,31]. Zhang Hui et. al. used CRC function and improved the protocol proposed by Yong-Zhen Li et. al. [32]. But their proposed protocol is easily traced when a reader repeat to send challenges to...
a tag and the tag will return the same acknowledge: $\text{CRC}(\text{preID})$. These protocols use partial identifier of a tag to authenticate and this will increase the difficulty to reveal the tag’s secrecy. Now we will utilize this mechanism to design a novel authentication protocol for RFID systems.

III. LIGHTWEIGHT AUTHENTICATION PROTOCOL
BASED ON RANDOM PARTIAL IDENTIFIER

By analyzing above it is obviously observed that it will increase the difficulty to reveal the secret information stored in a tag if only the tag’s partial identifier is used during the authenticating process. In order to enhance the privacy and security of RFID systems a pseudorandom generating function $\text{PRNG}(\cdot)$ is used to randomize the exchanged sessions between the tag and the reader so as to resist against tracing attack. A circular redundancy coding function $\text{CRC}(\cdot)$ is used to enhance the difficulty to reveal the secret information and to assure the integrity of the exchanged sessions. $\text{CRC}(\cdot)$ and $\text{PRNG}(\cdot)$ are some low-cost functions and it is feasible to complete these functions for low-cost RFID tags. The EPCglobal Class-1 Gen-2 tags provide a 16-bit circular redundancy coding function and a 16-bit pseudorandom generating function. During the authenticating process we suppose the communication channel between the reader and the verifier is secure so that the design of the authentication protocol is simplified.

Supposed $\text{ID}$ is a 32-bit identifier and it is used to identify a tag. Each tag shares its identifier $\text{ID}$, $\text{psID}$, $\text{CRC}()$, $\text{PRNG}()$ and $\text{f}(x,m,n)$ with the verifier. $\text{psID}$ is the pseudonym of $\text{ID}$. $\text{f}(x,m,n)$ is a function and it selects some bits of $x$ from the location of $m$ to the location of $n$. Before the authentication begins $\text{ID}$ and $\text{psID}$ are stored in the tag, $\text{psID}$$=$$\text{CRC}(\text{f}(\text{ID},0,16))$$||$$\text{CRC}(\text{f}(\text{ID},16,31))$. And $\text{oldID}$, $\text{oldpsID}$, $\text{newID}$, and $\text{newpsID}$ are stored in the verifier. $\text{oldID}$ and $\text{oldpsID}$ are the values of $\text{newID}$ and $\text{newpsID}$ for last authentication, which is used to prevent de-synchronized attack. The initial values of $\text{oldID}$ and $\text{oldpsID}$ are set to $\text{newID}$ and $\text{newpsID}$. The initial values of $\text{newID}$ and $\text{newpsID}$ are $\text{ID}$ and $\text{psID}$ respectively. Because $\text{ID}$ and $\text{psID}$ are 32 bits, $\text{CRC}(\cdot)$ and $\text{PRNG}(\cdot)$ are two 16-bit functions. Only part of $\text{ID}$ and $\text{psID}$ is extracted as the parameters of $\text{CRC}(\cdot)$ and $\text{PRNG}(\cdot)$ so as to reduce the colliding probability. The used symbols during the authenticating process are listed in Table I. Our proposed protocol is shown in Figure 2. $\text{st}$, $\text{sl}$, $\text{tl}$ are some temporary variables. $\text{L}$ is the length of $\text{ID}$. The protocol is described as follows:

Step 1: Verifier to reader and tag

The Verifier calls the pseudorandom generator, $\text{PRNG}(\cdot)$, to generate a random number $r$. Then it sends its challenge and $r$ to the reader, and the reader transfers them to the tag.

Step 2: tag to reader

After the tag receives the challenge and $r$ from the reader it uses $\text{PRNG}(\cdot)$ to generate two different random numbers $s$ and $t$. Let $\text{sl}$$=$$\%\text{L}$ and $\text{tl}$$=$$\%\text{L}$. $\text{L}/2$$<=$$\text{sl}$$+$$\text{tl}$$<=$$\text{L}$. $\text{st}$$=$$(\text{sl}$$+$$\text{tl})/2$. It calculates $\text{m1}$ and $\text{m2}$ as follows:

\begin{align*}
\text{m1} & = \text{CRC}(\text{f}(\text{ID},0,\text{sl})\oplus r\oplus t) || \text{CRC}(\text{f}(\text{ID},1/2 \times \text{L} - 1)\oplus r) \\
\text{m2} & = \text{CRC}(\text{f}(\text{psID},0,\text{sl})\oplus t) || \text{CRC}(\text{f}(\text{psID},\text{sl},\text{L} - 1)\oplus r)
\end{align*}

Then it sends $\text{m1}/\text{m2}$ to $\text{r}$ to the reader.

TABLE I. THE RELATED SYMBOLS FOR THE PROPOSED PROTOCOL

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r,s,t$</td>
<td>pseudorandom numbers</td>
</tr>
<tr>
<td>$\text{ID}$</td>
<td>the tag’s identifier</td>
</tr>
<tr>
<td>$\text{psID}$</td>
<td>the pseudonym of a tag</td>
</tr>
<tr>
<td>$\text{PRNG}(\cdot)$</td>
<td>a 16-bit pseudorandom generating function</td>
</tr>
<tr>
<td>$\text{CRC}()$</td>
<td>a 16-bit CRC function</td>
</tr>
<tr>
<td>$x % y$</td>
<td>modular operation, its value is equal to the remainder which $x$ is divided by $y$</td>
</tr>
<tr>
<td>$f(x,m,n)$</td>
<td>a function to select some bits of $x$ from the location of $m$ to the location of $n$</td>
</tr>
<tr>
<td>$\oplus$</td>
<td>XOR operator</td>
</tr>
<tr>
<td>$</td>
<td></td>
</tr>
<tr>
<td>$x \oplus y$</td>
<td>comparing operator, judge whether $x$ does not equal $y$</td>
</tr>
</tbody>
</table>

Step 3: reader to verifier

After the reader receives $\text{m1}/\text{m2}$ from the tag it transfers them to the verifier.

Step 4: verifier to reader

After the verifier receives $\text{m1}/\text{m2}$ from the reader it searches its database. It gets $\text{oldID}$, $\text{oldpsID}$, $\text{newID}$ and $\text{newpsID}$ from each record. Then it calculates as follows:

\begin{align*}
\text{m2} & = \text{CRC}(\text{f}(\text{psID},0,\text{sl})\oplus \text{r}) || \text{CRC}(\text{f}(\text{psID},\text{sl},\text{L} - 1)\oplus \text{r}) \\
\text{where} & \text{ psID is replaced with newpsID and oldpsID respectively. If the verifier calculates all } m2 \text{ none of them equals } m2. \text{ Then the current authentication fails. If there exists } m2 \text{ which equals } m2, \text{ the verifier calculates as follows:}
\end{align*}

\begin{align*}
\text{m1} & = \text{CRC}(\text{f}(\text{ID},0,\text{sl})\oplus \text{r}) || \text{CRC}(\text{f}(\text{ID},1/2 \times \text{L} - 1)\oplus \text{r})\oplus s
\end{align*}

where $\text{ID}$ is replaced with $\text{newID}$ or $\text{oldID}$ respectively, and this depends on which, $\text{newpsID}$ or $\text{oldpsID}$, is used to calculate $m2$. If $m1$ does not equal $m2$ the current authentication fails. Otherwise the verifier completes the authentication to the tag. The verifier calculates as follows:

\begin{align*}
\text{m3} & = \text{CRC}(\text{f}(\text{ID},0,\text{sl})\oplus \text{r}) || \text{CRC}(\text{f}(\text{ID},1/2 \times \text{sl},\text{L} - 1)\oplus \text{r}) \\
\text{m4} & = \text{CRC}(\text{f}(\text{psID},0,\text{sl})\oplus \text{r}) || \text{CRC}(\text{f}(\text{psID},\text{sl},\text{L} - 1)\oplus \text{r})
\end{align*}

Then the verifier sends $\text{m3}/\text{m4}$ to the reader. After the verifier completes the operations above it begins to update its secret parameters as follows:

\begin{align*}
\text{oldpsID} & = \text{psID} \\
\text{oldID} & = \text{ID} \\
\text{newpsID} & = \text{PRNG}(\text{f}(\text{psID},0,\text{sl})\oplus \text{r}) || \text{PRNG}(\text{f}(\text{psID},\text{sl},\text{L} - 1)\oplus \text{r}) \\
\text{newID} & = \text{PRNG}(\text{f}(\text{ID},0,\text{sl})\oplus \text{r}) || \text{PRNG}(\text{f}(\text{ID},\text{sl},\text{L} - 1)\oplus \text{r})
\end{align*}

Step 5: reader to tag

After the reader receives $\text{m3}/\text{m4}$ it transfers them to the tag.

Step 6: tag

After the tag receives $\text{m3}/\text{m4}$ from the reader it uses $\text{ID}$ and $\text{psID}$ stored in it to calculate $m3$ and $m4$ as follows:

\begin{align*}
\text{m3} & = \text{CRC}(\text{f}(\text{ID},0,\text{sl})\oplus \text{r}) || \text{CRC}(\text{f}(\text{ID},1/2 \times \text{sl},\text{L} - 1)\oplus \text{r}) \oplus \text{t} \\
\text{m4} & = \text{CRC}(\text{f}(\text{psID},0,\text{sl})\oplus \text{r}) || \text{CRC}(\text{f}(\text{psID},\text{sl},\text{L} - 1)\oplus \text{r})\oplus s
\end{align*}

Then the tag compares $\text{m3}$ and $\text{m4}$ with $\text{m3}$ and $\text{m4}$ respectively. If one of them is not equal the current authentication fails. Otherwise the current authentication succeeds and the tag completes the authentication to the
To calculate \( m_3' \) and \( m_4' \): 
\[
\text{m3'} = \text{CRC}(f(ID,0,t1) \oplus r \oplus s) || \text{CRC}(f(ID,1/2 \times s1,L-1) \oplus r \oplus t)
\]
\[
\text{m4'} = \text{CRC}(f(psID,0,t1) \oplus r) || \text{CRC}(f(psID,s1,L-1) \oplus r)
\]

To update its secret keys: 
\[
\text{psID}=\text{PRNG}(f(psID,0,st) \oplus r) || \text{PRNG}(f(psID,st,L-1) \oplus r)
\]
\[
\text{ID}=\text{PRNG}(f(ID,0,st) \oplus r) || \text{PRNG}(f(ID,st,L-1) \oplus r)
\]

Figure 2. The diagram of our proposed protocol

verifier. Then the tag begins to update its secret information as follows:

\[
\text{psID}=\text{PRNG}(f(psID,0,st) \oplus r) || \text{PRNG}(f(psID,st,L-1) \oplus r)
\]

Then the tag begins to wait for next authentication.

IV. ANALYZING FOR THE PRIVACY AND SECURITY OF OUR PROPOSED PROTOCOL

Our proposed protocol only uses random part of the tag’s identifier during each authenticating process and this increases the difficulty to reveal the secret information of the RFID systems. Each session message between the tag and the reader is randomized by different random numbers. After each authentication is finished all secret information are updated. So our proposed authentication protocol can assure the privacy and forward security of the RFID systems, and it can prevent eavesdropping attack, tracing attack, replay attack and de-synchronized attack.

- Resist against eavesdropping attack

For our proposed protocol, some secret information about the tag is not transferred by plaintext. An adversary can intercept and eavesdrop the exchanged sessions between the tag and the reader. But these sessions are processed by \( \text{CRC}(\ ) \) and \( \text{PRNG}(\ ) \) after the tag’s partial identifiers are randomized. It is very difficult for an adversary to reveal the tag’s secret information from the intercepted sessions. An adversary cannot acquire any useful information by eavesdropping.

- Resist against tracing attack

If a reader repeats to send challenges to a tag and the tag reply the same acknowledge to the reader each time, the reader can trace and locate the tag. For our proposed protocol, the tag receives the challenge from the reader and it will generate different random numbers \( s \) and \( t \) to randomize the acknowledge which will be sent to the reader. Hence the tag sends different acknowledges to the reader for each challenge and the reader cannot distinguish which tag sends these acknowledges to it. So it is impossible for the reader to trace a tag.

- Resist against replay attack

Supposed an adversary is able to intercept and eavesdrop the exchanged sessions between the tag and the reader. An adversary can disguise a tag to re-send the intercepted sessions to a reader. But for each authentication the verifier generates a different random number \( r \) and the tag generates two random numbers \( s \) and \( t \). All sessions between the reader and the tag are randomized by these random numbers. If an adversary re-sends the intercepted sessions and it cannot be
authenticated by the verifier. So replay attack is prevented effectively.

- Resist against de-synchronized attack

In order to resist against de-synchronized attack we add oldID and oldpsID in the backend database and they are the values of ID and psID for the last successful authentication. If ID and psID stored in the tag cannot be updated synchronously with the verifier our proposed protocol will use oldID and oldpsID to complete the authentication. So our proposed protocol can resist against de-synchronized attack.

- Forward security

Forward security means that an adversary cannot reveal the previous intercepted sessions even if he gets the current secret keys of the RFID system. For our proposed protocol, the secret keys of the RFID system are updated after each successful authentication. Although an adversary can reveal the current secret keys he cannot derive any useful information from the previous intercepted sessions. Because the sessions from different authentication procedures are generated by different secret keys. The sessions of the previous authentications are not revealed by the current secret keys. So forward security is assured.

- Data confidentiality and privacy

As described above, our proposed protocol uses CRC( ) and PRNG( ) to encrypt the secrecy of the tag and some random numbers are generated to randomize the sessions. It can resist against tracing attack and security leakage. Thence the protocol confirms the confidentiality and privacy of the RFID system.

- Data integrity

For our proposed protocol all exchanged sessions between the tag and the reader are generated by CRC( ). If the sessions are tampered, the authentication protocol can find out the changes and terminate itself. Thence data integrity is assured.

Table II shows the summary on the security aspects of several typical authentication protocols and their comparison with our proposed protocol.

### Table II. The Comparison of the Different Lightweight Authentication Protocols

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Tracing attack</th>
<th>Replay attack</th>
<th>Confidentiality</th>
<th>De-synchronized attack</th>
<th>Forward security</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hash-Lock protocol</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>SRAC protocol</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>LCAP protocol</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>The protocol (Ref. [32])</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Our proposed protocol</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
</tbody>
</table>

V. conclusion

RFID systems based on low-cost passive tags are some typical resource-constrained systems and their computing and storing resources are very limited. So some lightweight authentication protocols have to be proposed to meet the special requirements of the RFID systems. Our proposed protocol randomly selects partial identifier of a tag to complete the mutual authentication between the tag and the reader. This increases the difficulty to reveal the tag’s secrecy. At the same time, our proposed protocol uses random numbers to randomize the exchanged sessions between the tag and the reader so as to resist against tracing attack and replay attack. After each successful authentication the tag’s secret keys are updated so that forward security is assured. While updating the tag’s secret keys the current and last secret keys are reserved so as to resist against de-synchronized attack. So our proposed protocol can assure forward security and it can resist against eavesdropping, tracing attack, replay attack, de-synchronized attack. It only uses circular redundancy coding function and pseudorandom generating function. It is feasible to complete these functions for low-cost passive tags. So our proposed protocol is suitable for low-cost FRID systems.

Acknowledgment

We are grateful for the anonymous reviewers who made constructive comments so that we can improve and refine our paper. The relative work about this paper is supported by National Natural Science Foundation of China (No. 61272097), the Science and Technology Innovation Project of Shanghai Education Committee (No. 12ZZ182).

### References


Zhicai Shi Born in 1964, He Received Master degree in computer science and engineering from Harbin Institute of Technology, Harbin, China, in 1988 and Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2000, respectively. Since Sept. 2007, he has been a professor of the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, China. He has published more than 80 research papers in some conferences and journals. He has served as program committee members and reviewers in numerous international conferences and some journals. His main research interests are computer network and information security, embedded system and RFID authentication. His research was supported by The Education Committee of Shanghai.

Fei Wu Born in 1968, He Received Master degree and Ph.D. degree in computer science and engineering from National Defense Technology University, Changsha, China, in 1994 and 1998 respectively. Since Sept. 2013, he is a professor of the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, China. He has published more than 40 papers in some conferences and journals. His main research interest is green computing and his research has been supported by National Natural Science Foundation of China and The Education Committee of Shanghai.