An Improved Ownership Transfer and Mutual Authentication for Lightweight RFID Protocols

Peng-yu Cui
(Corresponding author: Peng-yu Cui)

Information Centre, Liaoning Geology Engineering Vocational College, Dandong 118008, China
(Email: cuipengyu1982@126.com)

(Received Sep. 24, 2015; revised and accepted Dec. 07, 2015 & Jan. 3, 2016)

Abstract

Radio Frequency Identification (RFID) technology is an automated identification technology which is widely used to identify and track all kind of objects. However, it is a challenging task to design an authentication protocol because of the limited resource of Lightweight RFID tags. Recently, a lightweight RFID authentication protocol and an ownership transfer of RFID tags are presented by Kulseng et al. Both protocols use Physically Unclonable Functions (PUF) and Linear Feedback Shift Registers (LFSR) which are well known lightweight operations. The number of gates which the protocols require can be significantly decreased and the most efficient protocol can be obtained with respect to the existing protocols. Unfortunately, their protocols face several serious security issues. In this paper, based PUF and LFSR, we suggest an improved mutual authentication and an improved ownership transfer for low-cost RFID Protocols. Security analysis shows that our protocol owns security and privacy.

Keywords: LFSR, mutual authentication, ownership transfer, PUF, RFID

1 Introduction

RFID (Radio Frequency Identification) is an emerging ubiquitous technology which identifies different kinds of objects based on radio wave signals. It has been widely used in many fields, such as inventory control, transportation payment, supply chain management and so on [11].

As many technologies, RFID faces also similar security concerns: Authentication, Confidentiality and Availability. For insecure RFID system, the user's privacy will face a great threat. An adversary can obtain user's privacy by eavesdropping or trace the tag's holder in such condition [15]. However, as RFID tags are generally low-cost device without tamper resistance, compromising RFID tag can be very easy. The challenge on addressing the security concerns is much harder than conversational technology [5].

Some authentication protocols have been suggested to use in RFID system which aiming to solve the privacy and forgery problems. Generally, we only consider the information security issues in the channel between tags and reader for research convenience because of the special property of tags. In order to promote the great potential of RFID technology, the cost of RFID tags must be competitive with existing solutions such as bar codes, which are very low-cost. Passive RFID tags with no battery have between 200-2000 hardware gates available for security measures. Unfortunately, traditional security mechanisms used in RFID system require a large number of gates. A low-cost version of AES has been shown to require 3,400 gates, while hash functions such as MD5 and SHA-256 have been implemented using between 8,000-10,000 gates. Therefore, it is a key problem for RFID system to design efficient and secure authentication protocol [1, 2, 14].

Many RFID authentication protocols based on Pseudo-Random Number Generator (PRNG operation) have been proposed to achieve security and privacy protection [3, 13]. Also, several light-weight RFID authentication protocols with inexpensive cryptographic primitives, such as XOR and hash functions, are also presented [8, 9, 10, 12]. However, these protocols suffer from either privacy and security issues or efficiency.

Physically Unclonable Functions (PUFs) are known as random functions that map challenges to responses. PUFs are unclonable because it computes random numbers with the help of the inherent variability of wire delays and gate delays in manufactured circuits [4]. The existence of the fact is that no two circuits have exactly the same delay properties, even if they were produced on the same wafer. Given a certain input, the tag's PUF will produce a certain output, while other tag's PUFs will produce different output.

Kulseng et al. [7] present a lightweight mutual authentication and ownership transfer protocol which can be considered as lightweight because their protocols do not require expensive cryptographic operations. Their protocols are basically designed by using Physically Unclonable

Functions (PUFs) and Linear Feedback Shift Registers (LFSRs) which are well known lightweight operations and are particularly suitable for the low-cost RFID tags. Their protocol requires only 784 gates for 64-bit variables. So, this protocol can certainly be considered to have a significant improvement. But, Kardas et al. [6] show that there are in fact several serious security issues with Kulseng et al.'s protocols.

In this paper, using PUFs and LFSRs, we give an improved mutual authentication and ownership transfer for lightweight RFID Systems. The remainder of this paper is organized as follows. In Section 2, we describe the lightweight mutual authentication proposed by Kulseng et al. and its drawbacks. Section 3 presents an improved mutual authentication and discusses its security. Section 4 proposes a new ownership transfer protocol. Concluding remarks are presented in Section 5.

2 Kulseng et al.'S Protocol and Its Drawbacks

The notations and steps for the protocol are described as follows.

2.1 Notations

- ID: Tag's ID which is unique.
- IDS: An index to tag's ID and is updated in each round.
- G_n: A greeting number.
- F: A random permutation function mapping within range [1, q], where log q is the bit length of IDS (LFSR can be used as F).
- P: A random permutation function mapping within a range [1, q] (P is implemented based on Physically Unclonable Functions (PUF)).

2.2 Description of Protocols

The initial ID, IDS and a random greeting number G_n are generated for each tag firstly. Then, G_{n+1} is computed by the PUF function stored in the tag as $G_{n+1} = P(G_n)$. The entry of (IDS, ID, G_n, G_{n+1}) are inserted into the backend database. The IDS, ID and G_n are stored in the tag. Kulseng et al.'s protocol consists of five steps as Figure 1.

1) The reader continuously broadcasts *Req* message.

2) Receiving *Req* from the reader, the tag responds with its IDS.

3) The reader looks up the corresponding greeting G_n for this tag. If it finds an entry, it computes $ID \oplus G_n$ and sends it to the tag.

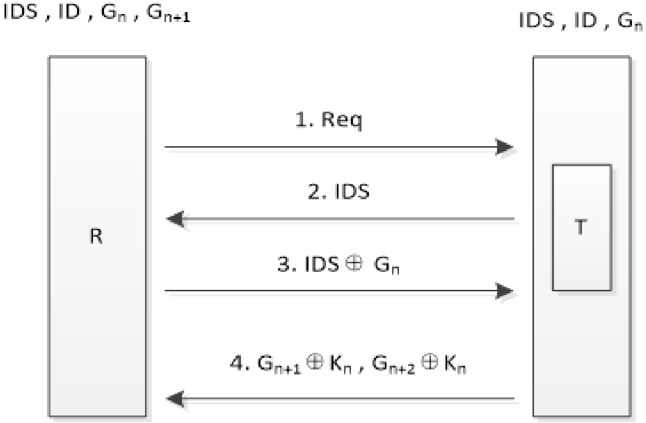
IDS , ID , G_n , G_{n+1}
IDS , ID , G_n
R
T
1. Req
2. IDS
3. $IDS \oplus G_n$
4. $G_{n+1} \oplus K_n , G_{n+2} \oplus K_n$

Figure 1: Kulseng et al.'s authentication protocol

4) Receiving the message $ID \oplus G_n$, the tag verifies the correctness of this response. If it is valid, it computes $G_{n+1} = P(G_n)$, $G_{n+2} = P(G_{n+1})$, $K_n = F(G_n)$ and $K^{'}_n = F(K_n)$. Then, it calculates $K_n \oplus G_{n+1}$, $K^{'}_n \oplus G_{n+2}$ and sends them to the reader. Finally, the tag updates $IDS = F(IDS \oplus G_n)$ and $G_n = G_{n+1}$.

5) The reader verifies $K_n \oplus G_{n+1} = F(G_n) \oplus G_{n+1}$. If it is valid, the reader can get G_{n+2} by $K^{'}_n \oplus G_{n+2} \oplus F(F(G_n))$. At last, it updates $IDS_{new} = F(IDS_{old} \oplus G_n)$, $G_n = G_{n+1}$ and $G_{n+1} = G_{n+2}$.

2.3 Security Analysis

Kardas et al. [6] describe three different security flaws of the authentication protocol above. Here, we introduce them briefly.

Set R as a legitimate reader, T as a legitimate tag and A as an adversary.

Message blocking attack.

1) R broadcasts Req and T sends its IDS to R.
2) R computes $ID \oplus G_n$ and sends it to T. Here, blocking attack occurs and transaction between R and T drops.
3) Then, A broadcast Req and T sends IDS to A.
4) A sends $ID \oplus G_n$ to T.
5) T calculates $K_n \oplus G_{n+1}$, $K^{'}_n \oplus G_{n+2}$ and sends them to A. After that, T updates $IDS = F(IDS \oplus G_n)$ and $G_n = G_{n+1}$.
6) T can no longer authenticate R because R will send $ID \oplus G_n$ and T has G_{n+1}. T will not verify ID.

Desynchronization attack.

The protocol can not assure integrity. When A inserts a random message to the second message at Step 4, the synchronization between R and T will be broken.

1) R broadcasts Req and T sends its IDS to R.
2) R computes $ID \oplus G_n$ and sends it to T.
3) T calculates $K_n \oplus G_{n+1}$, $K_n^{'} \oplus G_{n+2}$ and sends them to R. Here, A inserts random number n_x to the message $K_n^{'} \oplus G_{n+2}$, $K_n{'} \oplus G_{n+2} \oplus n_x$. Finally, T updates $IDS = F(IDS \oplus G_n)$ and $G_n = G_{n+1}$.
4) Receiving message $K_n \oplus G_{n+1}$ and the modified message $K_n^{'} \oplus G_{n+2} \oplus n_x$, R verifies whether $K_n \oplus G_{n+1}$ is valid. Because the message is correct, R updates $IDS = F(IDS \oplus G_n)$ and $G_{n+1} = K_n^{'} \oplus G_{n+2} \oplus n_x \oplus K_n^{'} = G_{n+2} \oplus n_x$.
5) In the next section, the R has G_n, G_{n+1} and $G_{n+1}^{'} \neq P(G_n)$. According to the protocol, T can authenticate R but R will not authenticate T.

The misuse of LFSR G.

Kardas et al. point that an adversary can easily find out the secret ID and trace the tag because of the use of LFSR. This attack can be accomplished as follows. Assume that an adversary observes a whole authentication session of a tag. The adversary who has listened the communication between the reader and the tag can obtain the following session messages: Req,IDS_{old},$ID \oplus G_n$,$G_{n+1} \oplus K_n$,$G_{n+2} \oplus K_n^{'}$.

Then, the adversary sends a fake query to the tag. The tag will response $IDS_{new} = F(IDS_{old} \oplus G_n)$ to the adversary. It is critical that $IDS_{old} \oplus G_n$ can be gotten from the value of $F(IDS_{old} \oplus G_N)$ easily. So, the adversary can deduce ID of the tag from IDS_{old}, $ID \oplus G_N$, $IDS_{old} \oplus G_n$ and trace the tag.

3 An Improved Mutual Authentication

In this section, an improved mutual authentication protocol is proposed as Figure 2 and its security is analyzed.

3.1 Notations

The notations are same as Section 2 and contain K_n. K_n is the share key between Reader and Tag.

3.2 An Improved Mutual Authentication Protocol

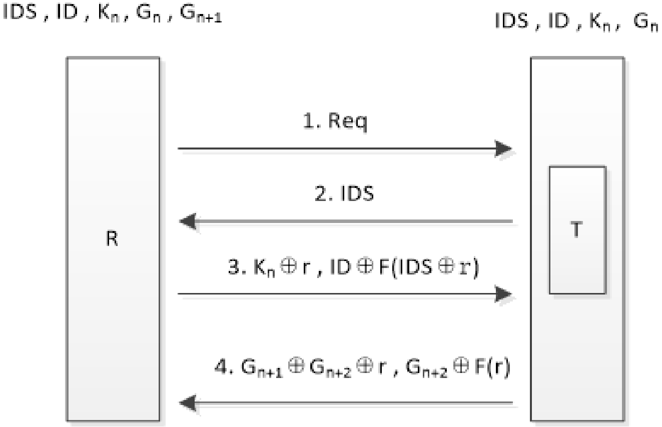
IDS , ID , K_n , G_n , G_{n+1}
IDS , ID , K_n , G_n
R
T
1. Req
2. IDS
3. $K_n \oplus r$, $ID \oplus F(IDS \oplus r)$
4. $G_{n+1} \oplus G_{n+2} \oplus r$, $G_{n+2} \oplus F(r)$

Figure 2: Our authentication protocol

1) The reader continuously broadcasts Req message.
2) Receiving Req from the reader, the tag responds with its IDS and the tag updates $IDS = F(IDS \oplus G_n \oplus K_n)$.
3) According to IDS, if the reader finds an entry, it updates $IDS = F(IDS \oplus G_n \oplus K_n)$ firstly. Next step it generates a random number r and computes $K_n \oplus r$, $ID \oplus F(IDS \oplus r)$. Finally the reader sends $K_n \oplus r$, $ID \oplus F(IDS \oplus r)$ to the tag.
4) Receiving the message $K_n \oplus r$ and $ID \oplus F(IDS \oplus r)$, the tag gets r from $K_n \oplus r \oplus K_n$ firstly. Next, it computes $F(IDS \oplus r)$. Furthermore the tag verifies the correctness of the ID from $ID \oplus F(IDS \oplus r) \oplus F(IDS \oplus r)$. If it is correct, the tag computes $G_{n+1} = P(G_n)$, $G_{n+2} = P(G_{n+1})$. Next step, it sends $G_{n+1} \oplus G_{n+2} \oplus r$ and $G_{n+2} \oplus F(r)$ to the reader. Finally, the tag updates $K_n = K_n \oplus F(K_n \oplus r)$ and $G_n = G_{n+1}$.
5) According to the message of the fourth step, the reader gets G_{n+2} from $G_{n+1} \oplus G_{n+2} \oplus r \oplus G_{n+1} \oplus r$ firstly. Next the reader verifies $G_{n+2} = G_{n+2} \oplus F(r) \oplus F(r)$. If it is correct, the reader updates $K_n = K_n \oplus F(K_n \oplus r)$, $G_n = G_{n+1}$, $G_{n+1} = G_{n+2}$.

3.3 Security Analysis

In this part, we present the security analysis of our scheme. In addition to limited storage capacity, low computational and communicational cost, our protocol withstand against modification attack, de-synchronization attack, disclosure attack, replay attack, man in middle attack, backward security, forward security, cloning attack and also achieve mutual authentication, tag anonymity and indistinguishability.

1) Resistance to modification attack.
   No matter what parts of the messages in our protocol are modified, the reader or the tag can find it because IDS,G_n,G_{n+1},K_n is dynamic and IDS is random for adversary. So the reader or the tag can confirm each message in our protocol is modified and the protocol will be halted and the attacker can not get any valuable information.
2) Resistance to de-synchronization attack.
   An attacker may try to desynchronize IDS,G_n between reader and tag. For this purpose, he blocks

messages from tag to reader in the fourth pass of the protocol. In order to handle this synchronization issue, it is suggested that the previous IDS,G_n value are stored in the tag side. When the IDS is not stored to the database, the reader will ask the tag to use old IDS,G_n.

3) Resistance to disclosure attack.
The key idea of the disclosure attack is that an attacker can slightly modify the challenge from the reader and then infer partial information from the response of the tag. In our protocol, the reader and tag have confidential data contain ID,G_n,G_{n+1},K_n all transmitted messages are random and secrecy. So attacker slightly modifies any challenge in all message, this protocol will be halted and he cannot get any useful information finally. As a result, this attack does not work on our protocol.

4) Resistance to replay attack and man in middle attack.
An attacker may try to do a replay attack by eavesdropping legitimate interactions. If an attacker wants to disguise reader, he replays first and third message. However, he cannot succeed because IDS will be both updated each round and random, and the third message is relation with IDS. So the tag can find this attack quickly. If an attacker wants to disguise tag, he replays second and fourth message. Also, he will not succeed because ID,G_n,G_{n+1},K_n have been updated and the message is relation with these parameters. The reader will not authenticate the disguised tag for the adversary replayed message is outdated. And, when the adversary tries the man-in-the middle attack, he will not succeed because the second message, the third message and the fourth message are dynamic, and lack of necessary parameters such as ID,G_n,G_{n+1},K_n.

5) Backward security and forward security.
It is essential that the previously transmitted information cannot be traced using the present transmission tag information, and the future information cannot be confirmed using the present transmission tag information. If the past and future location of the specific tag owner can be traced using the present information, it constitutes a serious privacy infringement. The proposed protocol prevents an adversary from acquiring tag information, by providing confidentiality based on unpredictable variations in the response message of the tag by every session. Moreover, IDS is updated each time and random for an adversary and G_n,G_{n+1},K_n is updated when authentication of the reader is complete and the tag is closed successfully, and the value of r is determined randomly by the reader, thus it guarantees backward security and forward security by disconnecting the relation with both the previously transmitted information and the future information.

6) Cloning attack.
To prevent cloning attacks, our protocol uses a unique PUF in tag. It is infeasible to construct two PUFs with the same challenge-response behavior. So an adversary can copy the PUF and cloning attack is invalid in our protocol.

7) Anonymity and indistinguishability.
Anonymity means that the attacker cannot identify the identity of tag and cannot track tag. Indistinguishability means that information emitted by tag should not be discriminated from other tags. The proposed protocol protects the information necessary for tag authentication by using the PUFs, LFSR function and the Random Number Generator, and guarantees that only the authenticated object knowing ID,G_n,G_{n+1},K_n can verify the information. Furthermore, as mentioned earlier, the proposed protocol is secure against backward security and forward security, and guarantees anonymity and indistinguishability.

8) Mutual authentication.
The proposed protocol provides mutual authentication between reader and tag. The tag authenticates reader by the value of ID and the reader authenticates tag by the value of G_{n+1}. The proposed protocol satisfies all the security requirements, and completely solves the privacy and forgery problems of the RFID system.

Next, we discuss this protocol about attacks described in Section 2.

Message blocking attack.
The way to resist blocking attack is same as that method in resistance to desynchronization attack. IDS,G_n are asked to store in the tag. If the database does not look up IDS, the reader can ask the tag to use old IDS,G_n to continue this protocol.

Desynchronization attack.
If an attacker attempts to insert any message to desynchronize this protocol, it can not gain its ends as all the elements in the message are linked together and any part changed will be found by reader or tag. For example, a random number n_x is inserted to the message in Step 4 as $G_{n+1} \oplus G_{n+2} \oplus r$, $G_{n+2} \oplus F(r) \oplus n_x$. According to our protocol, this attack has no effect because the reader will find $G_{n+2} \neq G_{n+2} \oplus F(r) \oplus n_x \oplus F(r)$.

The misuse of LFSR.
In our protocol, the use of LFSR will not leak any useful information of the tag. If an attacker obtains all messages in a whole authentication session between reader and tag. Then, the attacker sends a fake query to the tag. The tag wills response $IDS_{news} = F(IDS_{old} \oplus G_n \oplus K_n)$ to the adversary. He can obtain $IDS_{old} \oplus G_n \oplus K_n$ from

$F(IDS_{old} \oplus G_n \oplus K_n)$ and further to get $G_n \oplus K_n$. However, it is of no value. It is hard trace the tag only by both $G_n \oplus K_n$ and the previously transmitted information. 1summarizes the comparison of our protocol with Kulseng et al.'s protocol. Y is owns the ability of resistance to attack, N is not owns.

Table 1: Comparison of our protocol with Kulseng et al.'s protocol

Protocol	Our protocol	Kulseng et al.'s protocol
Modification attack	Y	N
De-synchronization attack	Y	N
Disclosure attack	Y	N
Replay attack and man in middle attack	Y	N
Backward security and forward security	Y	N
Cloning attack	Y	Y
Anonymity and indistinguishability	Y	N
Mutual authentication	Y	N

4 Ownership Transfer Protocol

In this section, we introduce Kulseng et al.'s ownership transfer protocol and attacks on it firstly. Then we present improved ownership transfer protocol.

Kulseng et al. [7] proposed two ownership transfer protocols. The first protocol assumes the existence of a trusted authority by both the reader and the tags, named the Trusted Third Party (TTP). The second ownership transfer protocol involves no third party. The authenticated reader that accesses the tag is called as owner. An ownership transfer protocol should satisfy the following two properties:

1) The old owner should not be able to access the tag after the ownership transfer takes place.

2) The new owner should be able to perform mutual authentication with the tag after the ownership transfer has taken place.

4.1 Kulseng et al.'s Ownership Transfer Protocol with TTP

The communications between the TTP and the readers are assumed to be secure. The old owner first gives its stored tuple (IDS, ID, G_{n+1}) to the new owner. It also transfers the verification pair G_n, G_{n+1} to the TTP. A secret value of PIN is securely shared between the TTP and the tag. The PIN is preinstalled in the tag hardware during production and is not accessible to anyone.

1) The new reader sends G_{n+1} to the TTP via a secure channel.

2) The TTP verifies whether the received G_{n+1} from the new reader equals to the one received from the previous owner, if so, then the new reader gets authenticated. Then the TTP sends $K_n \oplus G_n \oplus PIN$ to the reader, where $K_n = F(PIN)$.

3) The reader forwards the messages to the tag.

4) The tag computes $K_n = F(PIN)$ and gets G_n from $K_n \oplus G_n \oplus PIN$. If the computed G_n equals that it stores, the tag computes $G_n^{'} = P(G_{n+2})$,$G_{n+1}^{'} = P(G_n^{'})$ and $K_n^{'} = F(K_n)$,$K_n^{''} = F(K_n^{'})$. At last, tag calculates $K_n^{'} \oplus G_n{'}$,$K_n^{''} \oplus G_{n+1}^{'}$ and $K_t = F(G_n \oplus G_{n+1})$ and sends them to the reader.

5) The reader forwards these messages to the TTP.

6) Upon receiving the messages, the TTP verifies the correctness of the value K_t. Then it computes the random numbers $K_n{'}$,$K_n{''}$, and obtains the values of the pair value of $G_n{'}$,$G_{n+1}^{'}$ and sends them back the new owner via the secure channel. Now the new reader can start a new mutual authentication with the tag.

7) Both the TTP and the tag can update the PIN internally as $PIN_{new} = F(PIN_{old} \oplus G_n)$.

4.2 Attacks on Protocol

Now we show that the protocol above does not satisfy two secure properties.

- The old owner can access the tag after the ownership transfer takes place.
  Kardas et al. point that privacy of the tag can be elaborated by the old owner [6]. The old owner still knows ID of the tag because ID is constant and unique for each tag. Assume that the old owner A has recorded an successful session between R and T and a subsequent query to the tag T.

  1) A records all messages exchanged between R and T.

  2) A get G_n by $G_n = G_n \oplus ID \oplus ID$(the third message XOR ID).

  3) Next, A derives G_{n+1} by computing $G_{n+1} = (G_{n+1} \oplus K_n) \oplus F(G_n)$.

  4) A sends a fake query to the tag T and T sends back the updated IDS value.

  5) A computes $F(IDS \oplus G_n))$ by using G_n and IDS. Then, A verifies whether this value is equal to IDS which is received from the query. If they are equal, this session belongs to the T.

- The new owner can not perform mutual authentication with the tag after the ownership transfer has taken place.
  We introduce an attack that can make the new owner not implement mutual authentication with the tag. A malicious adversary A injects random numbers n_x and n_y to the message as $K_n^{'} \oplus G_n^{'} \oplus n_x$, $K_n^{''} \oplus G_{n+1}^{'} \oplus n_y$ and $K_t = F(G_n \oplus G_{n+1})$. According to this protocol, what the new reader holds are $G_n^{'} \oplus n_x$, $G_{n+1}^{'} \oplus n_y$, not really $G_n^{'}$, $G_{n+1}^{'}$. So, the new owner is not able to perform mutual authentication with the tag.

4.3 Improved Ownership Transfer Protocol with TTP

Here, we present an improved ownership transfer protocol with TTP.

1) The new reader sends G_{n+1} to the TTP via a secure channel.
2) The TTP verifies whether the received G_{n+1} from the new reader equals to the one received from the previous owner, if so, then the new reader gets authenticated. Then the TTP sends $PIN^{'} \oplus G_n \oplus PIN$ to the reader, where $PIN^{'} = F(PIN)$.
3) The reader forwards the messages to the tag.
4) The tag computes $PIN^{'} = F(PIN)$ and gets G_n from $K_n \oplus G_n \oplus PIN$. If the computed G_n equals that it stores, the tag computes $G_n^{'} = P(G_{n+2})$, $G_{n+1}^{'} = P(G_n^{'})$ and $K_n^{'} = F(G_n^{'})$, $K_{n+1}^{'} = F(G_{n+1}^{'})$. At last, tag calculates $PIN \oplus G_n^{'}$, ${K_n}' \oplus G_n^{'}$, $PIN \oplus G_{n+1}^{'}$, $K_{n+1}^{'} \oplus G_{n+1}^{'}$ and $K_t = F(G_n \oplus G_{n+1})$ and sends them to the reader.
5) The reader forwards these messages to the TTP.
6) Upon receiving the messages, the TTP verifies the correctness of the value K_t. Then it derives $G_n^{'}$, $G_{n+1}^{'}$ from $PIN \oplus G_n^{'}$, $PIN \oplus G_{n+1}^{'}$. Finally, it verifies $F(G_n^{'}) \oplus G_n^{'} \overset{?}{=} K_n^{'} \oplus G_n^{'}$ and $F(G_{n+1}^{'}) \oplus G_{n+1}^{'} \overset{?}{=} K_{n+1}^{'} \oplus G_{n+1}^{'}$. If both of them are correct, the TTP and sends $G_n^{'}$ and $G_{n+1}^{'}$ to the new owner via the secure channel. Now the new reader can start a new mutual authentication with the tag.
7) Both the TTP and the tag can update the PIN internally as $PIN_{new} = F(PIN_{old} \oplus G_N)$.

It is suggested that the improved mutual authentication protocol in Section 4 and the improved ownership transfer protocol are used together. Thus, in the protocol in Section 4, the old owner can not get any relationship between IDS and ID of tag because the random number r makes the system obscure. Therefore, property 1) is satisfied. In terms of improved ownership transfer protocol, all the parts in the message are linked together and any element changed will be found by the TTP or tag. So, the new owner can receive and accurately and start performing a normal mutual authentication with the tag. Therefore, property 2) can be satisfied.

4.4 Two-party Ownership Transfer

A two-party ownership transfer solution is a protocol without a TTP can be constructed using improved mutual authentication protocol directly. The setup phase is similar to that in the TTP protocol, the old owner gives the tuple stored (IDS, ID, G_n, G_{n+1}) to the new owner. The online authentication phase is shown in Figure 1 We will not discuss the details again.

5 Conclusion

RFID technology can provide great benefits in several areas and has many applications for both business and individuals. As many technologies, RFID faces also similar security concerns: Authentication, Confidentiality and Availability. For insecure RFID system, the user's privacy will face a great threat. At the same time, it is necessary to avoid expensive cryptographic computations because of low-cost devices and less capability. At INFOCOM 2010, Kulseng et al. gave a lightweight RFID authentication protocol and an ownership transfer protocol which is claimed the most efficient protocols among the existing protocols. However, Kardas et al. point that the protocols have several serious security issues. In this paper, an improved mutual authentication protocol is proposed based on PUF functions and LFSR functions. This paper proves that the proposed protocol is secure against various types of attacks and can solve the problems of the previous works. Furthermore, by satisfying all of the security requirements, the proposed RFID mutual authentication protocol completely solves the privacy and forgery problems. The protocols not only can defeat security attacks but also require small number of gates. Finally, improved ownership transfer is proposed.

Acknowledgments

This work is supported by the Project of The Research of License Plate Recognition System (No.2014j4100201).

References

[1] I. Agudo, R. Rios, and J. Lopez, "A privacy-aware continuous authentication scheme for proximity-based access control," *Computers & Security*, vol. 39, no. 1, pp. 117–126, 2013.

[2] B. Alomair, A. Clark, J. Cuellar, and et al., "Scalable RFID systems: A privacy-preserving protocol with constant-time identification," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 8, pp. 1536–1550, 2012.

[3] G. Avoine, C. Lauradoux, and T. Martin, "When compromised readers meet RFID," in *Workshop on Information Security Applications*, vol. LNCS 5932, pp. 36–50, 2009.

[4] L. Bolotnyy and G. Robins, "Physically unclonable function-based security and privacy in RFID systems," in *Fifth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom'07)*, pp. 211–220, White Plains, NY, March 2007.

[5] M. Chen, W. Luo, Z. Mo, and et al., "An efficient tag search protocol in large-scale RFID systems," in *Proceeding of the 32nd IEEE International Conference Computer Communications*, pp. 899–907, Turin, Italy, April 2013.

[6] S. Kardas, M. Akgun, M. S. Kiraz, and H. Demirci, "Cryptanalysis of lightweight mutual authentication and ownership transfer for RFID systems," in *Workshop on Lightweight Security Privacy: Devices, Protocols and Applications (LightSec'10)*, pp. 20–25, San Diego, CA, March 2011.

[7] L. Kulseng, Z. Yu, Y. Wei, and Y. Guan, "Lightweight mutual authentication and ownership transfer for rfid systems," in *Proceedings IEEE Conference on INFOCOM*, pp. 1–5, San Diego, CA, March 2010.

[8] N. W. Lo and K. H. Yeh, *An Efficient Mutual Authentication Scheme for EPCglobal Class-1 Generation-2 RFID System.* Springer Berlin Heidelberg, 2007.

[9] L. Lu, J. Han, L. Hu, Y. Liu, and L. M. Ni, "Dynamic key-updating: Privacy-preserving authentication for RFID systems," in *Fifth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom'07)*, pp. 13–22, White Plains, NY, March 2007.

[10] D. Moriyama, S. Matsuo, and M. Ohkubo, "Relations among notions of privacy for RFID authentication protocols," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 97, no. 1, pp. 225–235, 2014.

[11] L. M. Ni, Y. Liu, Y. C. Lan, and et al., "LANDMARC: Indoor location sensing using active RFID," *Wireless Networks*, vol. 10, no. 6, pp. 701–710, 2004.

[12] B. Song and C. J. Mitchell, "RFID authentication protocol for low-cost tags," in *First ACM Conference on Wireless Network Security*, pp. 140–147, White Plains, NY, 2008.

[13] Y. Tian, G. Chen, and J. Li, "A new ultralightweight RFID authentication protocol with permutation," *IEEE Communications Letters*, vol. 16, no. 5, pp. 702–705, 2012.

[14] L. Wang, X. Yi, L. V. Chao, and Y. Guo, "Security improvement in authentication protocol for Gen-2 based RFID system," *Journal of Convergence Information Technology*, vol. 6, no. 1, pp. 157–169, 2011.

[15] J. Zhang, W. Wang, J. Ma, and X. Li, "A novel authentication protocol suitable to EPC class 1 generation 2 RFID system," *Journal of Convergence Information Technology*, vol. 7, no. 3, pp. 259–266, 2012.

Peng-yu Cui received a master's degree in College of Electrical and Control Engineering from North China University Of Technology (China) in June 2011. He is a lecturer in Information Centre in Liao-ning Geology Engineering Vocational College. His current research interest fields include information security and computer application.