Int. Journal of Math. Analysis, Vol. 7, 2013, no. 52, 2583 - 2593
HIKARI Ltd, www.m-hikari.com
http://dx.doi.org/10.12988/ijma.2013.36146

A New Ultra-lightweight RFID Authentication Protocol Using Merge and Separation Operations

Il-Soo Jeon

School of Electronic Engineering, Kumoh National Institute of Technology,
77 Sanho-Ro, Yangho-Dong, Gumi, Kyungsangpuk-Do 730-701, South Korea

Eun-Jun Yoon[*]

Department of Cyber Security, Kyungil University,
33 Buho-Ri, Hayang-Ub, Kyungsan-Si, Kyungsangpuk-Do 712-701, South Korea

Abstract

Since Low-cost RFID tags have very limited hardware resources, it is difficult to implement an authentication protocol which uses heavy operations such as modern ciphers or hash functions. It has been presented some ultra-lightweight RFID authentication protocols for low-cost RFID tags by using very light operations. Recently, Jeon and Yoon proposed a new ultra-lightweight RFID authentication protocol. They defined and used the merge and separation operations. The merge operation can merge the bits from two bit strings and the separation operation is an inverse operation of the merge operation. However, we found that the protocol cannot serve correctly when the collision of tag pseudonyms is occurred. In this paper, we propose an improved authentication protocol that solves the problem. We show that the proposed protocol can resist various security attacks and is efficient enough to implement low-cost RFID tags.

Keywords: RFID, Low-Cost RFID Tags, Ultra-lightweight Authentication Protocol, Merge, Separation

[*] Corresponding author: E-mail: ejyoon@kiu.ac.kr (Eun-Jun Yoon)

1 Introduction

The RFID systems are very useful and convenient in various applications. Therefore, they are widely used such as access control, supply chain management, inventory control, smart labels, and etc. The usage of application is significantly increasing. Generally, RFID systems are consisted of three entities: tags, the reader, and the server. The reader is connected to the server which has a database. The communications between the reader and the server are safe enough from the various security attacks. However, since the communications between the reader and the tags are performed through wireless channel, they are vulnerable to the security attacks. Therefore, in order to resist the security attacks between the reader and the tags, authentication protocols are generally used.

Since low-cost RFID tags have very limited hardware resources, it is difficult for them to adapt the existing authentication protocols using modern ciphers which require a lot of computation cost and storage space. Thus, several ultra-lightweight authentication protocols for low-cost RFID tags have been proposed recently. These protocols generally use some lightweight operations such as XOR, rotation, AND, OR, permutation, etc. In 2006, Peris-Lopez et al. proposed a family of ultra-lightweight authentication protocols for low-cost RFID, LMAP[1] and M^{2}AP[2], which use bitwise operations, XOR, AND, OR, and modular operation. Since their protocols were very simple, they were suitable for low-lost RFID tags. Unfortunately, their protocols are vulnerable to de-synchronization attack and full disclose attack[3]. In 2007, Chien[4] proposed a new ultra-lightweight authentication protocol, SASI, which supports mutual authentication and tag anonymity. However, Sun et el.[5] showed that SASI cannot resist from the de-synchronization attack. Cao et al.[6] showed that SASI is vulnerable to the de-synchronization attack through the man-in-the-middle attack. Phan[7] used the imbalance of the bitwise OR operation to do the tracking attack for SASI. In 2009, Peris-Lopez et al.[8] proposed another ultra-lightweight authentication protocol called Gossamer. But in 2010, Targa et al.[9] showed that

Gossamer is vulnerable to the de-synchronization attack. In 2011, Tian et al.[10] proposed a new ultra-lightweight authentication protocol (RAPP) for low-cost RFID tags. They defined and used permutation operation in their protocol. However, quite recently, Jeon and Yoon [11] proposed a new authentication protocol, EURFID for low-cost RFID tags. In their protocol, they defined and used the merge and separation operations. The merge operation can merge the bits from two bit strings and the separation operation is an inverse operation of the merge operation. However, we found that the EURFID protocol does not operate correctly when the collision of tag pseudonyms is occurred between one tag and the other tags.

The rest of this paper is organized as follows. In the following section 2, we describe preliminaries and notations for this paper and comment the security problem of the EURFID protocol and then describe the proposed RAPLT protocol in section 3. In section 4, the security and efficiency analysis of the proposed scheme are discussed. Finally, the conclusion is given in Section 5.

2 Preliminaries and Notations

RFID systems are composed of three entities: the tags, the reader, and the back-end server containing the database. The communication channel between the reader and the back-end server is generally assumed secure, but the communication channel between the reader and the tags is wireless and insecure. For the sake of convenience, we assume that the reader has the database. So we use two entities, the reader and the tags in the protocols. Each tag and the server share the unique identity of the tag, secret keys, and old and new pseudonyms of the tag to resist de-synchronization attacks. Jeon and Yoon [11] defined the merge operation, *Mer()* and the separation operation, *Sep()* and used them in their protocol, RAPLT. The *Mer()* and *Sep()* are the relation of inverse operation. To describe this paper easily, we use some notations and summarized them in Table 1[11].

Table 1. System Notations

Notations	Description
$\boldsymbol{ID}$	Unique identity for an RFID tag
$\boldsymbol{IDS^o}$, $\boldsymbol{IDS^n}$	Old and new pseudonym of RFID tags, respectively
$\boldsymbol{n_1}$, $\boldsymbol{n_2}$	Random numbers generated by the RFID reader
$\boldsymbol{A, B}$	*l*-bit random strings
$\boldsymbol{K}$	Secret key composed of 2*l* bits, where total counts of 0 and 1 are equal
$\boldsymbol{K_1, K_2}$	Left and right half parts of secret key *K*, respectively
Mer(A, B, K, C)	Merge operation which merges *A* and *B* to *C* according to *K*. If the bit of *K* is 0, then the bit of *A* moved to *C*, otherwise the bit of *B* moved to *C*.
Sep(C, K, A, B)	Separation operation which demerges *C* to *A* and *B* according to *K*. If the bit of *K* is 0, then the bit of *C* moved to *A*, otherwise the bit of *B* moved to *C*.
⊕	Bit wise XOR operator
‖	String concatenation operator
→	Message transmission

3 The proposed RAPLT protocol

This section proposes an improved RFID authentication protocol which solves the flaw that exists in the EURFID protocol [11]. In the EURFID protocol, there is no collision policy for the tag's pseudonyms. Tag's pseudonym, *IDS* is updated each successful session, but the protocol does not check if the *IDS* is unique or not in the server's database. If a tag's *IDS* is equal to other tag's *IDS*, the tag can be considered as the other tag on the reader's side. Therefore, in that case, the tag cannot be authenticated.

In order to solve the flaw in the EURFID protocol, it requires a mechanism that the protocol guarantees every new *IDS* is unique. As a solution mechanism, we use one of the random numbers. If the server generates a random number which can be a unique *IDS* in the database and uses it as a new *IDS*, such mechanism can overcome the problem existing in the EURFID protocol. Therefore, we propose a modified EURFID to eliminate the collision problem.

Jeon and Yoon [11] defined and used *Mer()* and *Sep()* operations in the protocol, EURFID. Therefore, we introduce their definition of *Mer()* and *Sep()* operations below.

Definition 1: Assume *A* and *B* are two *l*-bit strings, *K* and *C* are 2*l*-bit strings,

$$\begin{aligned} A &= a_1a_2\cdots a_l,\ a_i \in \{0,1\}, \quad i = 1,2,\cdots,l \\ B &= b_1b_2\cdots b_l,\ b_i \in \{0,1\}, \quad i = 1,2,\cdots,l \\ K &= k_1k_2\cdots k_{2l},\ a_i \in \{0,1\}, \quad i = 1,2,\cdots,2l \\ C &= c_1c_2\cdots c_{2l},\ c_i \in \{0,1\}, \quad i = 1,2,\cdots,2l \end{aligned}$$

where a total count of 0 = total count of 1=*l*.

Mer(A,B,K,C) operation is as follows.

```
$i,j \leftarrow 1$
for $n$=1 to 2$l$
    if $k_n$=0 then $c_n \leftarrow a_i$, $i \leftarrow i+1$
        else $c_n \leftarrow b_j$, $j \leftarrow j+1$
    end if
end for
```

Sep(C,K,A,B) operation is as follows.

```
$i,j \leftarrow 1$
for $n$=1 to 2$l$
    if $k_n$=0 then $a_i \leftarrow c_n$, $i \leftarrow i+1$
        else $b_j \leftarrow c_n$, $j \leftarrow j+1$
    end if
end for
```

Fig. 1 shows an example that illustrates the movement of bits when *Mer()* and *Sep()* operations are executed.

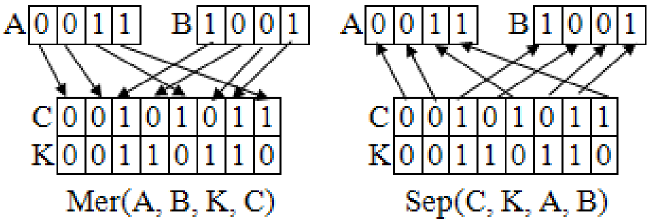

Fig. 1 Execution example of *Mer()* and *Sep()* operations

The detail procedure of the proposed protocol, RAPLT is described below and summarized it in Fig. 2.

Step 1: The reader sends a “Hello” message to the tag to initiate a session.

Step 2: If the “Hello” message is received a second time, the tag sends IDS as IDS^o, otherwise the tag sends IDS as IDS^n to the reader.

Step 3: The reader searches the IDS in the back-end database. If IDS does not exist in the database, the reader sends “Hello” message again, otherwise the reader obtains K_1, K_2, and ID of the matched IDS from the database. The reader generates two l-bit random numbers, n_1 which can be a unique IDS in the database and n_2. Then the reader computes $N_1 = n_1 \oplus ID$, $N_2 = n_2 \oplus IDS$, $Mer(N_1, N_2, K_1 \parallel K_2, A_1 \parallel A_2)$, $M_1 = ID \oplus n_1 \oplus K_2$, $M_2 = IDS \oplus n_2 \oplus K_1$, $Sep(M_1, M_2, K_2 \parallel K_1, B_1 \parallel B_2)$, and $B_3 = B_1 \oplus B_2$. Finally the reader sends the message, (A_1, A_2, B_3), to the tag.

Step 4: The tag extracts n_1', n_2' by computing $Sep(A_1 \parallel A_2, K_1 \parallel K_2, N_1', N_2')$, $n_1' = N_1' \oplus ID$, and $n_2' = N_2' \oplus IDS$. Then it computes $M_1' = ID \oplus n_1' \oplus K_2$, $M_2' = IDS \oplus n_2' \oplus K_1$, $Sep(M_1', M_2', K_2 \parallel K_1, B_1' \parallel B_2')$, and $B_3' = B_1' \oplus B_2'$. If B_3 does not equal to B_3', the tag does not authenticate the reader and terminates the protocol run. Otherwise the tag authenticates the reader and executes $Mer(K_1, K_2, K_2 \parallel K_1, K_1' \parallel K_2')$, $Mer(n_2', N_1', K_1' \parallel K_2', C_1 \parallel C_2)$, and $C_3 = C_1 \oplus C_2$. Then the tag sends the message, (C_3), to the reader. Finally the tag updates IDS^o as IDS, and IDS^n as n_1.

Step 5: The reader computes C_3' by executing of $Mer(K_1, K_2, K_2 \parallel K_1, K_1' \parallel K_2')$, $Mer(n_2, N_1, K_1' \parallel K_2', C_1' \parallel C_2')$, and $C_3' = C_1' \oplus C_2'$. If C_3 does not equal to C_3', the reader does not authenticate the tag and terminates the protocol run. Otherwise the reader authenticates the tag and updates IDS^o as IDS, and IDS^n as n_1.

RFID Reader $\{ID, IDS^o, IDS^n, K_1, K_2\}$	**RFID Tag** $\{ID, IDS^o, IDS^n, K_1, K_2\}$
Hello → 	***If*** *Hello is received a second time* ***then*** $IDS = IDS^o$ ***else*** $IDS = IDS^n$*;*
	← IDS
If *IDS exist in the DB* ***then*** *obtains* K_1, K_2, ID *from the DB* ***else*** *resend Hello;* *Generate* n_1 *which can be a unique IDS in the DB* *Generate* n_2 $N_1 = n_1 \oplus ID$ $N_2 = n_2 \oplus IDS$ $Mer(N_1, N_2, K_1 \parallel K_2, A_1 \parallel A_2)$ $M_1 = ID \oplus n_1 \oplus K_2$ $M_2 = IDS \oplus n_2 \oplus K_1$ $\mathrm{Sep}(M_1, M_2, K_2 \parallel K_1, B_1 \parallel B_2)$ $B_3 = B_1 \oplus B_2$	
A_1, A_2, B_2 →	$Sep(A_1 \parallel A_2, K_1 \parallel K_2, N_1', N_2')$ $n_1' = N_1' \oplus ID$ $n_2' = N_2' \oplus IDS$ $M_1' = ID \oplus n_1' \oplus K_2$ $M_2' = IDS \oplus n_2' \oplus K_1$ $Sep(M_1', M_2', K_2 \parallel K_1, B_1' \parallel B_2')$ $B_3' = B_1' \oplus B_2'$ $Verify\ B_3 = B_3'$ $Mer(K_1, K_2, K_2 \parallel K_1, K_1' \parallel K_2')$ $Mer(n_2', N_1', K_1' \parallel K_2', C_1 \parallel C_2)$ $C_3 = C_1 \oplus C_2$
	← C_3
$Mer(K_1, K_2, K_2 \parallel K_1, K_1' \parallel K_2')$ $Mer(n_2, N_1, K_1' \parallel K_2', C_1' \parallel C_2')$ $C_3' = C_1' \oplus C_2'$ $Verify\ C_3 = C_3'$	
Updating: $IDS^o = IDS$ $IDS^n = n_1$	**Updating:** $IDS^o = IDS$ $IDS^n = n_1$

Fig. 2 The proposed RAPLT protocol

4 Security and Efficiency Analysis

This section analyzed the security and privacy of the proposed RAPLT protocol. The basic structure of the proposed protocol is similar to that of EURFID except that it solves the collision problem of the EURFID protocol. Jeon and Yoon [11] showed that EURFID protocol has resistances to several security attacks and protects the privacy of users.

In their paper, EURFID protocol has resistance to the brute-force attack, the replay attack, the de-synchronization attack, and the disclosure attack. It also has tag anonymity and resistance to the tracking of tag. Therefore, the proposed protocol, RAPLT as well as EURFID can resist the various security attacks and has no collision problem of *IDS*.

The performance of EURFID is evaluated in terms of security, computation operation, storage requirement, and communication cost in each tag. We assume that the hardware and software power of the reader and the server are good enough to run the protocol. So we analyzed only the performance of the tag and summarized it in Table 2 with some other ultra-lightweight authentication protocols.

In Table 2, L denotes the length of each item stored in the tags. As we can see in Table 2, all the protocols except RAPLT do not support the *IDS* collision resistance. Therefore, we can state that RAPLT has the most powerful security among them.

In RAPLT, the operations used in the tags are efficient enough to be performed in low-cost tags. Even though the communication cost of RAPLT for each tag is 3L, it becomes the worst case only when de-synchronization attack occurs. Therefore, we can say that actual communication cost of RAPLT is 2L.

The communication cost of SASI and improved RAPP is also 2L, but they cannot resist the de-synchronization attack. RAPLT requires 5L storage space in the tag, which is less than or equal to those of other protocols.

Table 2. Comparison of Ultra-lightweight Authentication Protocols.

Protocols Comparison factors	LMAP [1]	M^{2}AP [2]	SASI [4]	Gossamer [8]	RAPP [10]	EURFID [11]	RAPLT
Tracking resistance	no	no	no	yes	yes	yes	yes
De-synchronization attack resistance	no	no	no	no	no	yes	yes
Disclose attack resistance	no	no	no	yes	yes	yes	yes
IDS collision resistance	no	no	no	no	no	no	yes
Required storage space	6L	6L	7L	7L	5L	9L	5L
Communication messages	2L	3L	3L	2L	2L	3L	3L
Operation types (Tag)	$\oplus, +, \vee$	$\oplus, +, \vee,$	$\oplus, +, \vee, R$	$\oplus, +, Rot$ $Mixbits$	$\oplus, Rot,$ Per	$\oplus, Sep$	$\oplus, Mer,$ Sep

5. Conclusions

This paper proposed a new authentication protocol for low-cost RFID tags (RAPLT), which solves the IDS collision problem that exists in EURFID. The proposed RAPLT as well as EURFID uses very light operation that can be applied in low-cost RFID tags which have very limited hardware resources. Since the proposed RAPLT can resist the various security and privacy attacks and requires small storage space on the tags, it can be used practically in the application system that uses the low-cost RFID tags.

Acknowledgements: Il-Soo Jeon was supported by Research Fund, Kumoh National Institute of Technology. Eun-Jun Yoon was supported by the MSIP (Ministry of Science, ICT & Future Planning) support program (NIPA-2013-H0301-13-2004) supervised by the NIPA(National IT Industry Promotion Agency).

References

[1] P. Peris-Lopez, J. C. Hernandez- Castro, J. M. E. Tapiador, and A. Ribagorda, LMAP: a real lightweight mutual authentication protocol for low-cost RFID tags, in Proc. 2006 Workshop RFID Security. 2006.

[2] P. Peris-Lopez, J. C. Hernandez- Castro, J. M. E. Tapiador, and A. Ribagorda, M2AP: a minimalist mutual-authentication protocol for low cost RFID tags, in Proc. 2006 International Conference on Ubiquitous Intelligence and Computing, pp. 912–923. 2006.

[3] T. Li and G. Wang, Security analysis of two ultra-lightweight RFID authentication protocols, in Proc. 2007 IFIP RC-11 International Information Security Conference, pp. 109–120. 2007.

[4] H.-Y. Chien, SASI: a new ultralightweight RFID authentication protocol providing strong authentication and strong integrity, IEEE Trans. Dependable and Secure Computing, vol. 4, no. 4, pp. 337–340, 2007.

[5] H.-M. Sun, W.-C. Ting, and K.-H. Wang, On the security of Chien's ultralightweight RFID authentication protocol, IEEE Trans. Dependable and Secure Computing, vol. 8, no. 2, pp. 315–317, 2011.

[6] T. Cao, E. Bertino, and H. Lei, Security analysis of the SASI protocol, IEEE Trans. Dependable and Secure Computing, vol. 6, no. 1, pp. 73–77, 2009.

[7] R. C.-W. Phan, Cryptanalysis of a new ultralightweight RFID authentication protocol—SASI, IEEE Trans. Dependable and Secure Computing, vol. 6, no. 4, pp. 316–320, 2009.

[8] P. Peris-Lopez, J. Hernandez-Castro, J. Tapiador, A. Ribagorda, "Advances in ultralightweight cryptography for low-cost RFID tags: Gossamer protocol," Information Security Applications, pp. 56–68, 2009.

[9] D. Tagra, M. Rahman, S. Sampalli, Technique for preventing DoS attacks on RFID systems, 18th international conference on software telecommunications and computer networks—SoftCOM'10, IEEE Computer Society, 2010.

[10] Y. Tian, G. Chen, J. Li, A New Ultralightweight Authentication Protocol with Permutation, IEEE Communication Letters, Vol. 16, No. 5, pp. 702-705, 2012

[11] I. S. Jeon and E. J. Yoon, Cryptanalysis and Improvement of a New Ultra-lightweight RFID Authentication Protocol with Permutation, Applied Mathematical Sciences, Vol. 7, No. 69, pp. 3433 – 3444, 2013.

Received: June 13, 2013