Vulnerability Analysis of Two Ultra lightweight RFID Authentication Protocols

Yousof Farzaneh[1], Mahdi Azizi[2], Masoud Dehkordi[1], and Abdolrasoul Mirghadri[2]
[1]School of Mathematics, Iran University of Science and Technology, Iran
[2]Faculty of Communication and Information Technology, IHU University, Iran

Abstract: *Ultra lightweight Radio Frequency Identification (RFID) authentication protocols are suitable for low-cost RFID tags with restricted computational power and memory space. Recently, Lee proposed two ultra lightweight authentication protocols for low-cost RFID tags, namely DIDRFID and SIDRFID protocols. The first protocol is based on dynamic identity and the second one on static identity. Lee claimed that his protocols can resist tracking, replay, impersonation and DOS attacks. In this paper, we show that Lee's protocols are not secure and they are vulnerable against tracking, impersonation, and full disclosure attacks. Specially, an adversary can accomplish an effective full disclosure attack on DIDRFID protocol by eavesdropping two consecutive sessions and gets all the secret information stored on a tag. Also, we demonstrate that an adversary with ability of obtaining secret information of a single compromised tag in SIDRFID protocol, can get the secret information of other tags and she/he can completely control the whole RFID system.*

Keywords: *Low-cost RFID, cryptography, protocol, vulnerability*.

Received August 8, 2012; accepted July 28, 2013; published online August 17, 2014

1. Introduction

Radio Frequency Identification (RFID) systems use radio frequency technology to automatically identify objects, animals or people. A typical RFID system consists of tags, readers and a server with a database as shown in Figure 1. A reader accesses the information contained within a tag via radio transmission. With the accessed information as an index, the reader can retrieve the corresponding record from the database of the server [5, 6, 15]. RFID systems have many applications such as e-passport, security control, supply chain management, inventory control, etc., [8, 11].

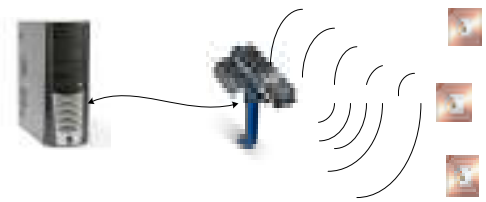

Figure 1. Working of RFID system.

Based on the computational cost and the operations supported on the tags, the RFID authentication protocols are classified by Chien [2] into four classes as follows:

1. The Full-Fledged Class: The protocols that should support conventional cryptographic functions like symmetric encryption, one-way hash functions, or even public key cryptographic algorithms on tags.
2. The Simple Class: The protocols that should install a Pseudo Random Number Generator (PRNG) or one-way hash function on tags.
3. The Lightweight Class: The protocols that require an PRNG and simple functions like Cyclic Redundancy Code (CRC) checksum.
4. The Ultra Lightweight Class: The protocols that only require simple bitwise operations like XOR, AND, OR, etc., on tags.

The low-cost RFID tags with extremely restricted computational power and memory space are the best option in many new RFID applications due to market share consideration. Hence, designing of secure ultra lightweight protocols are interested. In recent years, several ultra lightweight protocols based on simple bitwise operations have been proposed for low-cost tags [2, 4, 8, 9, 12, 13, 14]. However, researchers showed that these protocols were insecure e.g., [1, 3, 7, 10].

Lee [8] proposed two ultra lighweight authentication protocols for low-cost RFID tags, namely DIDRFID and SIDRFID protocols. The first protocol is based on dynamic identity and the second protocol is based on static identity. Lee claimed that both of the protocols have the merits of providing mutual authentication and resisting various attacks such as tracking, replay, DOS and impersonation attacks.

In this paper, we analyze the security of the Lee's protocols and show that they are not secure. Firstly, we demonstrate that DIDRFID protocol is vulnerable against tracking, replay attack, full disclosure attack, and DOS attack. In this protocol, an adversary can track a target tag among other tags and uses replay attack to impersonate a valid reader. Full disclosure attack can disclose all the secret information stored on a tag by eavesdropping two consecutive sessions of DIDRFID protocol. Thus, it completely compromises

the tag and an adversary can accomplish DOS attack on this protocol by using full disclosure attack.

Secondly, we show that SIDRFID protocol is vulnerable against tracking, impersonation and full disclosure attacks. An adversary with ability of obtaining secret information of a single compromised tag in SIDRFID protocol can get the secret information of other tags and she/he can completely control the whole RFID system.

The rest of this paper is organized as follows: In section 2, we review Lee's protocols. The DIDRFID and SIDRFID protocols are analyzed in sections 3 and 4, respectively. Finally, our conclusions are given in section 5.

2. Review of Lee's Protocols

DIDRFID and SIDRFID are two ultra lighweight authentication protocols which are recently proposed by Lee [8]. These protocols assume that the communications between the reader and the backend server are through secure channels, but the communications between the reader and the tag are susceptible to all possible attacks due to the open nature.

In the proposed protocols, the PRNG is only installed in the server and the tags only perform simple bitwise operations such as: XOR, OR, AND and left rotation. Consequently, the protocols are very practical to be implemented on low cost tags. In this section, we review these two ultra lighweight authentication protocols. The notations used throughout this paper are as follows:

- *IDT*: Tag's static identity.
- IDT_r: Tag's dynamic identity.
- *IDR*: Reader's static identity.
- K_i: The secret key of the tag.
- R_i: A random integer.
- $\oplus$: Bitwise XOR operation.
- $\wedge$: Bitwise AND operation.
- $\vee$: Bitwise OR operation.
- *Rot*(*A, B)*: An $w(B)$- bit left rotation on A, where $w(B)$ denotes the hamming weight of B.
- *A→B: M*: Sends *M* to *B* through a public channel.

2.1. DIDRFID Protocol

DIDRFID is the dynamic identity RFID protocol which is proposed by Lee [8]. In this protocol, the tag and reader share the tag's dynamic identity and secret key. The dynamic identity and secret key are updated after each authentication session. After the i^{th} authentication session, both the tag and the server share two pairs of information, ($DIDT_i$, K_i) and ($DIDT_{i+1}$, K_{i+1}), where ($DIDT_{i+1}$, K_{i+1}) is used for the potential next session.

The protocol consists of two main phases: Authentication phase and key updating phase. In the authentication phase, the reader first inquires the tag, and then the reader and the tag authenticate each other.

In the key updating phase, the reader and the tag update their dynamic identifications and secret keys, respectively. DIDRFID protocol is depicted in Figure 2.

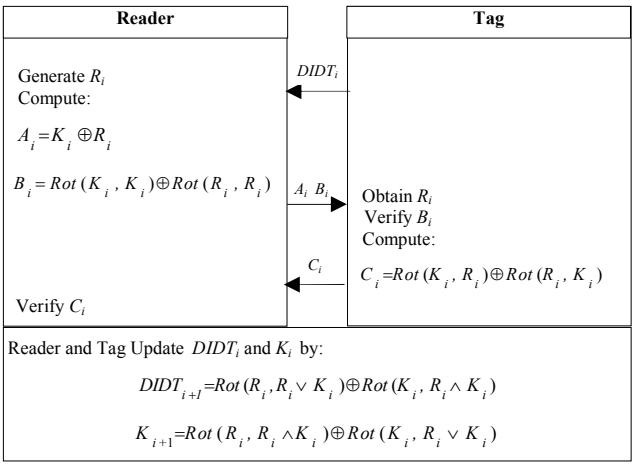

Figure 2. DIDRFID protocol [8].

2.1.1. Authentication Phase

At the i^{th} session, the authentication procedure of the DIDRFID protocol is described as follows:

- *Step L-1. Tag→Reader*, $DIDT_i$: The tag transmits its dynamic identity $DIDT_i$ to the reader after receiving an inquire message from the reader.
- *Step L-2. Reader→Tag*, (A_i, B_i): After receiving $DIDT_i$, the reader finds the tag's corresponding secret key K_i from the database. Then, the reader generates a random number R_i and computes (A_i, B_i) as follows:

$$A_i = K_i \oplus R_i$$

$$B_i = Rot(K_i, K_i) \oplus Rot(R_i, R_i)$$

  Then, the reader sends (A_i, B_i) to the tag.

- *Step L-3. Tag→Reader*, C_i: Upon receiving (A_i, B_i), the tag obtains R_i' by:

$$R_i' = A_i \oplus K_i$$

  Then, the tag computes B_i' with K_i and R_i' as:

$$B_i' = Rot(K_i, K_i) \oplus Rot(R_i', R_i')$$

  The reader will be authenticated if $B_i' = B_i$. Next, the tag computes C_i as follows if the reader is authenticated:

$$C_i = Rot(K_i, R_i) \oplus Rot(R_i, K_i)$$

  Finally, the tag forwards C_i to the reader.

- *Step L-4. Reader authenticates tag*: Upon receiving C_i from the tag, the reader computes C_i' as:

$$C_i' = Rot(K_i, R_i) \oplus Rot(R_i, K_i)$$

  The tag will be authenticated if $C_i' = C_i$. If $C_i' = C_i$, the reader and the tag obtain mutual authentication.

2.1.2. Key Updating Phase

After mutual authentication is obtained at the i^{th} session, the reader and the tag compute a new dynamic identity $DIDT_{i+1}$ and secret key K_{i+1} for the next session by:

$$DIDT_{i+1}=Rot(R_i, R_i \vee K_i) \oplus Rot(K_i, R_i \wedge K_i)$$

$$K_{i+1}=Rot(R_i, R_i \wedge K_i) \oplus Rot(K_i, R_i \vee K_i)$$

Then, the reader and the tag store ($DIDT_i$, K_i) and ($DIDT_{i+1}$, K_{i+1}) in the memory.

2.2. SIDRFID Protocol

SIDRFID protocol is a Static Identity RFID protocol which is proposed by Lee [8]. In this protocol, the tag's and the reader's secret identities are *IDT* and *IDR*, respectively. *IDT* and *IDR* are installed in the tag's and the reader's memories. SIDRFID protocol is depicted in Figure 3.

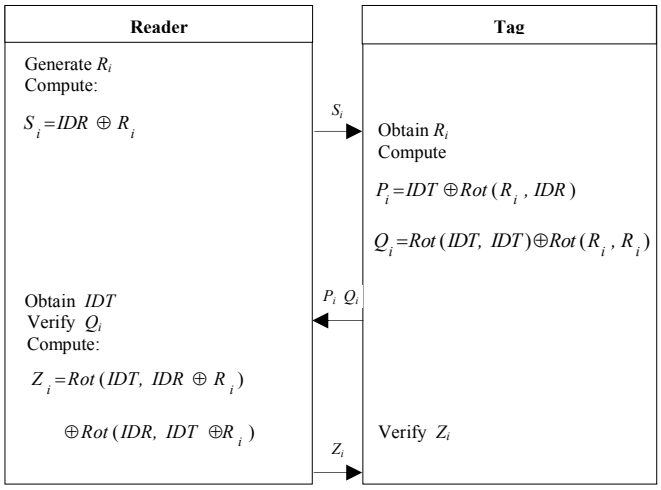

Figure 3. SIDRFID protocol [8].

At the i^{th} session, the authentication procedure of the SIDRFID protocol is described as follows:

- *Step S-1*. *Tag→Reader*, S_i: The reader first generates a random integer R_i and computes S_i by:

  $$S_i = IDR \oplus R_i$$

  The reader sends S_i with a request message to the tag.
- *Step S-2*. *Tag→Reader*, (P_i, Q_i): After receiving S_i, the tag obtains R_i by:

  $$R_i = S_i \oplus IDR$$

  Then, the tag sends P_i and Q_i to the reader, where

  $$P_i = IDT \oplus Rot(R_i, IDR)$$

  $$Q_i = Rot(IDT, IDT) \oplus Rot(R_i, R_i)$$
- *Step S-3*. *Tag→Reader*, Z_i: Upon receiving (P_i, Q_i), the reader computes *IDT'* by|:

  $$IDT' = P_i \oplus Rot(R_i, IDR)$$

  Then, the reader computes Q'_i by:

  $$Q'_i = Rot(IDT', IDT') \oplus Rot(R_i, R_i)$$

  Next, the reader authenticates the tag by checking whether $Q_i = Q'_i$. After the tag is authenticated, the reader computes Z_i by:

  $$Z_i = Rot(IDT, IDR \oplus R_i) \oplus Rot(IDR, IDT \oplus R_i)$$

  Finally, the reader sends Z_i to the tag.
- *Step S-4*. *Reader authenticates tag*: Upon receiving Z_i, the tag computes Z'_i by:

  $$Z'_i = Rot(IDT, IDR \oplus R_i) \oplus Rot(IDR, IDT \oplus R_i)$$

  The tag will be authenticated if $Z'_i = Z_i$. Hereafter, the reader and the tag obtain mutual authentication.

3. Vulnerability Analysis of DIDRFID Protocol

In this section, we analyze DIDRFID protocol and show that it is vulnerable against tracking, replay, full disclosure and DOS attacks.

3.1. Tracking Attack

After the i^{th} authentication session in the DIDRFID protocol, the tag and the reader share two values $DIDT_{i+1}$ and $DIDT_i$ as tag's dynamic identity, where $DIDT_{i+1}$ is the potential tag's identity in the next session and $DIDT_i$ is the tag's old identity in the next session. In the $(i+1)^{th}$ authentication session, the reader first sends "hello" message to the tag and the tag will respond with $DIDT_{i+1}$. The reader uses the tag's response identity to find a matched entry in the database and goes to the mutual authentication phase if a matched entry is found; otherwise, the reader sends "hello" message again and the tag responds with $DIDT_i$. An adversary can use this condition to track the tag. For this purpose after the i^{th} session, the adversary impersonates a valid reader to send "hello" message to the tag successively, the tag will respond with $DIDT_{i+1}$ and $DIDT_i$ in the 1st round and following round, respectively. The same old value of $DIDT_i$ can be used to track the tag.

Also, in DIDRFID protocol, the dynamic identity is only updated after each successful authentication session by both the tag and the reader. An adversary can abuse this condition to track the tag. For this purpose, she/he can block or change the message B_i in step L-3. This prevents the tag from updating its secrets (i.e., $DIDT_{i+1}$, K_i). Then, in the next session, when the tag receives "hello" message, it responds with $DIDT_i$. So the adversary can track the tag as long as it uses the same $DIDT_i$, since $DIDT_i$ remains constant until the successful mutual authentication is accomplished.

3.2. Replay Attack and Reader Impersonation Attack

Since, the reader's response (A_i, B_i) in step L-2 of DIDRFID protocol is only dependent on the random number generated by reader and K_i, an adversary can impersonate a valid reader in the DIDRFID protocol

without knowing the internal state of the tag. Details of the attack are given below.

The adversary firstly eavesdrops a valid session between the tag and the reader and she/he records the messages ($DIDT_i$, A_i, B_i, C_i). Then, the adversary sends many times only the same value $DIDT_i$ toward the reader and storage its responses as follows:

$$\begin{array}{cc} DIDT_i & (A_i, B_i) \\ DIDT_i & (A_{i+1}, B_{i+1}) \\ \vdots & \vdots \\ DIDT_i & (A_{i+n}, B_{i+n}) \end{array}$$

Then, the adversary initiates a new session with the tag and receives $DIDT_i$ from it by claiming a mismatching for $DIDT_{i+1}$. Now, the adversary replays a recorded message (A_j, B_j), $i \leq j \leq i+n$, to the tag. Since, these values were computed by a valid reader previously, the tag will authenticate the adversary as a valid reader. So, this replay attack leads to impersonation the reader by adversary.

3.3. Full Disclosure Attack

DIDRFID protocol used the data dependent rotation operation; this operation has the following linear property for fixed Z:

$$Rot(X \oplus Y, Z) = Rot(X, Z) \oplus Rot(Y, Z)$$

Here, we use this property to present a powerful full disclosure attack against DIDRFID protocol. In this attack, an adversary can disclose the secret key shared between the reader and the tag by eavesdropping two consecutive authentication sessions with $l(l+1)$ off-line rotations computations, where l is the length of bit strings that used in this protocol.

The adversary first eavesdrop i^{th} and $(i+1)^{th}$ sessions between the tag and the reader. So she/he can obtain ($DIDT_i$, A_i, B_i, C_i) and ($DIDT_{i+1}$, A_{i+1}, B_{i+1}, C_{i+1}) from i^{th} and $i+1^{th}$ sessions, respectively, where:

$$A_i = R_i \oplus K_i$$

$$DIDT_{i+1} = Rot(R_i, R_i \vee K_i) \oplus Rot(K_i, R_i \wedge K_i)$$

$$A_{i+1} = R_{i+1} \oplus K_{i+1}$$

Note that, the secret key of the tag in the $(i+1)^{th}$ session is computed by:

$$K_{i+1} = Rot(R_i, R_i \wedge K_i) \oplus Rot(K_i, R_i \vee K_i)$$

Since,

$$\begin{aligned} A_{i+1} \oplus DIDT_{i+1} &= R_{i+1} \oplus K_{i+1} \oplus DIDT_{i+1} \\ &= R_{i+1} \oplus Rot(R_i, R_i \wedge K_i) \\ &\oplus Rot(K_i, R_i \vee K_i) \oplus Rot(R_i, R_i \vee K_i) \\ &\oplus Rot(K_i, R_i \wedge K_i) \\ &= R_{i+1} \oplus [Rot(R_i, R_i \wedge K_i) \\ &\oplus Rot(K_i, R_i \wedge K_i)] \oplus [Rot(R_i, R_i \vee K_i) \\ &\oplus Rot(K_i, R_i \vee K_i)] \\ &= R_{i+1} \oplus [Rot(R_i \oplus K_i, R_i \wedge K_i)] \\ &\oplus [Rot(R_i \oplus K_i, R_i \vee K_i)] \\ &= R_{i+1} \oplus [Rot(A_i, R_i \wedge K_i)] \\ &\oplus [Rot(A_i, R_i \vee K_i)] \end{aligned}$$

In the above last equation R_{i+1}, $Rot(A_i, R_i \wedge K_i)$ and $Rot(A_i, R_i \vee K_i)$ are unknown values for the adversary. On the other hand, $Rot(X, Y) = X <<< w(Y)$, where $<<<$ is the left rotation operation. Hence, the adversary puts $x = R_i \wedge K_i$ and $y = R_i \vee K_i$. Since, $0 \leq x, y \leq l$ and $y \geq x$, where, l is the length of the bit strings R_i and K_i, in this step of attack the adversary picks up two value of x and y from $\{0, 1, ..., l\}$ iteratively such that $y \geq x$ and for these values computes:

$$R'_{i+1} = A_{i+1} \oplus DIDT_{i+1} \oplus (A_i <<< x) \oplus (A_i <<< y)$$

Where, all terms in the right side of above equation are known. Then, the adversary gets:

$$K'_{i+1} = A_{i+1} \oplus R'_{i+1}$$

Finally the adversary uses K'_{i+1} to calculate:

$$C'_{i+1} = Rot(K'_{i+1}, R'_{i+1}) \oplus Rot(R'_{i+1}, K'_{i+1})$$

Then, she/he checks whether the relation $C'_{i+1} = C_i$ holds. If equality holds for a pair of x and y, the adversary can obtain:

$$R_{i+1} = R'_{i+1}$$

$$K_{i+1} = K'_{i+1}$$

From the above procedure, the adversary obtains K_{i+1} and R_{i+1} without knowing exact K_i and R_i. She/he can compute $DIDT_i$ and K_j for all $j \geq i+2$ by using these values and eavesdropping all massages on the insecure channel. Therefore, the adversary can fake the tag permanently.

This attack can be accomplished by at most $\frac{l(l+1)}{2}$ iterations of the above procedure. Thus, the attack is surely more efficient than a brute-force attack and it can be performed on a single PC.

3.4. DOS Attack

The adversary can easily perform DOS attack after accomplishing full disclosure attack as above. For this purpose, the adversary gets K_{i+1} and R_{i+1} by above full disclosure attack, then she/he communicates with a valid reader in two consecutive sessions and terminates the attack. Thus, the corresponding saved dynamic identity of the target tag in the database are $DIDT_{i+3}$ and $DIDT_{i+4}$. Since, the tag still has $DIDT_{i+1}$ and $DIDT_{i+2}$ for its dynamic identity, the adversary can desynchronize the tag and the reader successfully.

4. Vulnerability Analysis of SIDRFID Protocol

In this section, we analyze SIDRFID protocol and show that it is vulnerable against tracking, impersonation and full disclosure attacks.

4.1. Tracking Attack

Since, the tag's secrets not update in each session of SIDRFID protocol and they are always constant, an adversary can track the tag as follows: An adversary

first eavesdrops one session between the target tag and the reader and she/he obtains the (S_i, P_i, Q_i, Z_i) and records these values, where:

$$S_i = R_i \oplus IDR$$

$$P_i = IDT \oplus Rot(R_i, IDR)$$

$$Q_i = Rot(IDT, IDT) \oplus Rot(R_i, R_i)$$

When the adversary wants to track the target tag among other tags, she/he repeatedly query the tags with the same S_i. Since, *IDR* and *IDT* are fixed the target tags will response with the same value of (P_i, Q_i). Thus, the adversary can track the target tag.

4.2. Reader Impersonation

Assume a RFID tag (say tag *A*) is compromised and its stored content (IDT_A, *IDR*) is obtained by an adversary, where the tag's and the reader's secret identities are IDT_A and *IDR*, respectively. Thus, the adversary can obtain reader's secret identity, *IDR* and she/he can impersonate the reader and access to all registered tags. For example, the adversary chooses an arbitrary *R* and by using *IDR* sends $S = R \oplus IDR$ to another tag (say tag *B* with secret identity IDT_B) and this tag responds by $P = IDT_B \oplus Rot(R, IDR)$ and *Q* upon receiving *P*, the adversary obtains IDT_B by computing:

$$IDT_B = P \oplus Rot(R, IDR)$$

And sends *Z* to tag *B*, where:

$$Z = Rot(IDT_B, IDR \oplus R) \oplus Rot(IDR, IDT_B \oplus R)$$

Then, tag *B* accepts the adversary as a valid reader because *Z* is verified by it. So, the adversary can obtain the secret identity of each registered tag and impersonates the reader. She/he can control the whole RFID system.

Note that, even though low-cost RFID tags are not tamper resistant and might be compromised when a tag is captured, but RFID authentication protocols should not allow one single compromised tag to disturb the security of the whole system. Unfortunately, the SIDRFID protocol presents this weakness.

4.3. Full Disclosure Attack

In this subsection we present a full disclosure attack on the SIDRFID protocol. An adversary can first find the hamming weight of the reader's identity *w*(*IDR*) and by using it, she/he can find *IDR*. The details of this attack as follows.

An adversary first eavesdrops one session between the tag and the reader and she/he obtains the (S_i, P_i, Q_i, Z_i) and records these values, where:

$$S_i = R_i \oplus IDR$$

$$P_i = IDT \oplus Rot(R_i, IDR)$$

Then, the adversary sends S_i' which is obtained by flipping the j^{th} bit of S_i, i.e., for $0 \le j < l$:

$$S_i' = S_i \oplus e_j$$

Where, the j^{th} bit of e_j is 1 and the other bits of e_j are 0. After receiving S_i', the tag obtains $R_i' = R_i \oplus e_j$ by $R'_i = S'_i \oplus IDR$. Then, the tag sends (P_i', Q_i') to the adversary, where:

$$P_i' = IDT \oplus Rot(R_i', IDR)$$

Now, the adversary computes:

$$\begin{aligned} P_i \oplus P'_i &= IDT \oplus Rot(R_i \oplus, IDR) \\ &\quad \oplus IDT \oplus Rot(R'_i, IDR) \\ &= Rot(R_i, IDR) \oplus Rot(R'_i, IDR) \\ &= Rot(R_i \oplus R'_i, IDR) \\ &= Rot(e_j, IDR) \end{aligned}$$

Since, $P'_i \oplus P_i$ is equal to *w*(*IDR*)-bit left rotation on e_j, the adversary can easily find *w*(*IDR*) Note that, if $P'_i \oplus P_i = e_j$ then, *w*(*IDR*)=0 or *l*. When the adversary finds *w*(*IDR*), she/he picks up a *l*-bit string *K* from $\{0, 1\}^l$ with *w(K)=w(IDR)* iteratively and computes:

$$\overline{R} = S_i \oplus K$$

$$\overline{IDT} = P_i \oplus Rot(\overline{R}, IDR)$$

$$\overline{Q} = Rot(\overline{IDT}, \overline{IDT}) \oplus Rot(\overline{R}, \overline{R})$$

Now, the adversary checks whether a relation $Q_i = \overline{Q}$ holds. If the equality holds, the adversary can obtain the reader's and the tag identities as $\overline{IDT}$ and *K*, respectively.

This attack can be accomplished by at most $\binom{l}{w(IDR)}$ iterations of the above procedure, where *l* is the bit length of *IDR*.

5. Conclusions

Low-cost RFID tags are the best option in many RFID applications. In this paper, we analyzed the security of Lee's Ultra lightweight authentication protocols for low-cost RFID tags and demonstrated several effective attacks against them. We show that these protocols are vulnerable against tracking, impersonation and full disclosure attacks. Thus, these protocols are insecure.

References

[1] Barasz M., Boros B., Ligeti P., Loja K., and Nagy D., "Breaking LMAP," *in Proceedings of International Conference on RFID Security*, Graz, Austria, pp. 69-78, 2007.

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

[3] D'Arco P. and De A., "On Ultralightweight RFID Authentication Protocols," *IEEE Transactions on Dependable and Secure Computing*, vol. 8, no. 4, pp. 548-563, 2011.
[4] David M. and Prasad N., "Providing Strong Security and High Privacy in Low-Cost RFID Networks," *in Proceedings of Security and Privacy in Mobile Information and Communication Systems*, Turin, Italy, pp. 172-179, 2009.
[5] European Commission Information Society web site., available at: http://ec.europa.eu/informationciety/policy/rfid, last visited 2012.
[6] Han D. and Kwon D., "Vulnerability of an RFID Authentication Protocol Conforming to EPC Class 1 Generation 2 Standards," *Computer Standards and Interfaces*, vol. 31, no. 4, pp. 648-652, 2009.
[7] Hernandez-Castro J., Peris-Lopez P., Phan R., Estevez-Tapiador M., and Ribagorda A., "Cryptanalysis of the David-Prasad RFID Ultralightweight Authentication Protocol," *in Proceedings of the 6th International Workshop on Radio Frequency Identification: Security and Privacy Issues*, Istanbul, Turkey, pp. 22-34, 2010.
[8] Lee Y., "Two Ultralightweight Authentication Protocols for Low-Cost RFID Tags," *Applied Mathematics and Information Sciences*, vol. 6, no. 2, pp. 425-431, 2012.
[9] Lee Y., Hsieh Y., You P., and Chen T., "A New Ultralightweight RFID Protocol with Mutual Authentication," *in Proceedings of WASE International Conference on Information Engineering*, Shanxi, China, pp. 58-61, 2009.
[10] Li T. and Wang G., "Security Analysis of Two Ultra-Lightweight RFID Authentication Protocols," *in Proceedings of the 22nd IFIP TC-11 International Information Security Conference*, Sandton, South Africa, pp. 109-120, 2007.
[11] Nasir M., Norman A., Fauzi S., and Azmi M., "An RFID-Based Validation System for Halal Food," *the International Arab Journal of Information Technology*, vol. 8, no. 2, pp. 204-211, 2011.
[12] Peris-Lopez P., Hernandez-Castro J., Estevez-Tapiador M., and Ribagorda A., "EMAP: An Efficient Mutual-Authentication Protocol for Low-Cost RFID Tags," available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.69.7922&rep=rep1&type=pdf, last visited 2006.
[13] Peris-Lopez P., Hernandez-Castro J., Estevez-Tapiador M., and Ribagorda A., "LMAP: A Real Lightweight Mutual Authentication Protocol for Low-Cost RFID Tags," *in Proceedings of the 2nd Workshop RFID Security*, Graz, Austria, pp. 1-12, 2006.
[14] Peris-Lopez P., Hernandez-Castro J., Estevez-Tapiador M., and Ribagorda A., "M^{2}AP: A Minimalist Mutual-Authentication Protocol for Low-Cost RFID Tags," *in Proceedings of International Conference on Ubiquitous Intelligence and Computing*, Wuhan, China, pp. 912-923, 2006.
[15] Yeh T. and Wua C., "Improvement of the RFID Authentication Scheme based on Quadratic Residues," *Computer Communications*, vol. 34, no. 3, pp. 337-341, 2011.

Yousof Farzaneh received his MS degree in mathematics from Iran University of Science and Technology (IUST), Iran, in 2004. Currently, he is a PhD student in the School of Mathematics at Iran University of Science and Technology. His research interests include cryptography and network security.

Mahdi Azizi received his MS and PhD degrees in communications, cryptology and information security from the IHU, Iran, in 2006 and 2012 respectively. His research interests include RFID security, authentication protocols and cryptanalysis.

Masoud Dehkordi received his PhD degree in mathematics from Loughborough University, UK, in 1998. He is currently a professor of mathematics at the school of Mathematical Sciences in Iran University of Science and Technology (IUST), Iran. His research interests include number theory, cryptography and other related topics.

Abdolrasoul Mirghadri received his PhD degree in mathematical Statistics, from the faculty of Science, Shiraz University in 2001. He is an associate professor at the faculty and research center of communication and information technology, IHU, Iran since 1989. His research interest includes: Cryptography, statistics and stochastic processes. He is a member of ISC, ISS and IMS.