Cryptanalysis and Improvement of Mutual Authentication Protocol for EPC C1G2 passive RFID Tag

Ahmed Maarof[1], Mohamed Senhadji1[2] , Zouheir Labbi [3] and Mostafa Belkasmi [4]

**[1] Laboratory of Information, Communication and Embedded Systems(ICES), Mohammed V University, ENSIAS
Rabat, Morocco**

**[2] Laboratory of Information, Communication and Embedded Systems(ICES), Mohammed V University, ENSIAS
Rabat, Morocco**

**[3] Laboratory of Information, Communication and Embedded Systems(ICES), Mohammed V University, ENSIAS
Rabat, Morocco**

**[4] Laboratory of Information, Communication and Embedded Systems(ICES), Mohammed V University, ENSIAS
Rabat, Morocco**

Abstract

A Radio Frequency Identification (RFID) system is a contactless automatic identification system that uses small and low-cost tags. The restricted computation ability and limited memory capacity of low-cost tags make existing RFID systems vulnerable. EPC Class 1 Generation 2 (EPC-C1G2) is the most popular standard for low cost passive RFID tags, for improving security of this standard; many security schemes are designed since the release of the EPC-C1G2. In 2013 Pang et al.'s, proposed an improved authentication protocol for low cost RFID systems. They claimed that this protocol is secure against various attacks. In this paper, we show that this protocol is still vulnerable against other attacks which didn't discussed by Bezhad et al. Then, an improved protocol is proposed to fix the vulnerabilities, moreover we show that the proposed protocol is conforms to EPC-C1G2.

Keywords: *RFID Security, Lightweight, Authentication Protocol, Privacy, Low Cost RFID, EPC Class-1 Generation-2 Standard.*

1. Introduction

RFID technology has many applications such as inventory control to supply chain management, traffic control, stock control, libraries and so on. As it is shown in Fig. 1, RFID systems involve three main parts: back-end server, reader and tag. In general terms, the tag comprises a wireless microchip with a very limited computational and storage capabilities, and a coupling element, such as an antenna coil for communication that can be used to identify the objects it is attached to. The reader is composed of a radio frequency (RF) module, a control unit and a coupling element. The reader communicates to the server and the tag. The reader can carry out some complex cryptographic operations instead of the tag. The Server is usually composed of a database and a processing logic. It receives data from the readers, stores data into a database, and provides access to the data. Furthermore, the Server is assumed to have an infinite computational power [1], [2], [3]. ISO and the Electronic Product Code (EPC) standard proposed by EPC global has become the main dominant RFID technology on global logistics market in the world. ISO [4] has ratified and published EPC-C1G2 standard as an amendment to ISO/IEC18000-6.

The low-cost passive tags do not have a battery, as they obtained their power source from the reader. Among all of the different types of RFID tags, the passive tags are low cost, and it runs very simple functions which does not support cryptographic.

In order to achieve the low-cost manufacture requirement, EPC-C1G2 RFID tag can only authorize a very primitive computation capability and arithmetic functions. However, the EPC-C1G2 RFID tag specification did not take much account to user privacy and data security issue, so the design of a secure mutual authentication protocol has become one of the most important key factors for the success of EPC- C1G2 RFID systems. The security analysis that carried out on the authentication protocol proposed in the EPC-C1 G2 specification have showed several security vulnerabilities in this standard [5,6]. So, the researchers have been motived to propose new EPC-compliant schemes, or to correct the weaknesses and improve its security level, which is also our concern in this paper. In the history of RFID security research, numerous authentication protocols conforming to EPC-C1G2 standards had been developed. However, all of them are vulnerable to security and privacy threat and do not maintain low computational and communication cost and cannot be integrated into the EPC-C1G2 tag.

The RFID system consists of following components (as shown in Figure1) [7]:

- Tag (attached with an object, unique identification).
- Antenna (tag detector, creates magnetic field).
- Reader (receiver of tag information, manipulator).
- Communication infrastructure (enable reader/RFID to work through IT infrastructure).
- Application software (user database/application/ interface).

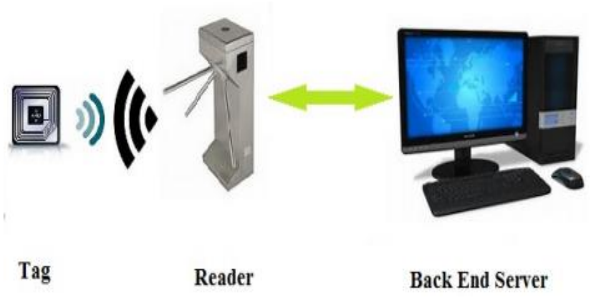

Fig. 1 Model of RFID systems.

The rest of this paper is organized as follows. section 2 covers the related work in this area. Section 3 provides an overview about the EPC-C1G2 standard and security threats related to low cost RFID technology. We review Pang et al.'s protocol and we investigate their vulnerabilities in Section 4. In section 5, the new proposed protocol is presented. The proposed protocol is analyzed in terms of security and performance in Section 6. Finally, conclusions are given in Section 7.

2. Related Work

Motivated by the release of EPC-C1G2 specification, some researchers have recently proposed a lot of schemes [7-11] trying to solve the practical problems due to the fact that the EPC-C1G2 only provides a low security level. In this section, we will summarize some related proposals in this field.

In 2007 Chien et al.[8] proposed a mutual authentication protocol conforming to EPC-C1G2 standard, the security of their scheme heavily relied on the abuse of the cyclic redundancy code (CRC). However, Peris-Lopez et al.[9]showed that the protocol cannot resist to tag impersonation, desynchronization attacking and location tracking. So the Chien et al.'s protocol not only is vulnerable to such attacks, but also does not provide tag privacy.

Chen and Deng [10] proposed an EPC-friendly mutual authentication scheme by using a pseudo-random number generator (PRNG) and CRC which is conformed to the EPC-C1G2 standard. Their protocol tried to apply CRC as cryptographic hash function for message authentication. However, CRC functions are linear and should not be used for any cryptographic purpose—only for detection of random errors in the channel [9]. So attacker is able to impersonate a tag or a reader, to trace a tag, and even to launch a DoS attack. These security vulnerabilities are all due to the misuse of the CRC function.

In 2010 Yeh et al.[11] also proposed a RFID mutual authentication protocol conforming to EPC-C1G2 standard which allows us free from the assumption of the channel's security. The information transmitted between reader and back-end database may also eavesdropped and intercepted by the attacker in actual environment, so the protocol applied a one-way hash function to guarantee the communication security between reader and back-end database. Nevertheless, Yoon [12] pointed out that Yeh et al.'s protocol still had two serious security problems such as DATA integrity problem and forward secrecy problem.

In 2013, Pang et al.[13] proposed a mutual authentication protocol to enhance the security for RFID systems which is conforming RFID systems, but their protocols still suffers from some weaknesses which are discussed by Behzad et al.[14] and other will be discussed in this paper. Behzad et al. proposed to add a hash function to improve the protocol, but the EPC C1G2's tag only support CRC, PRNG functions, simple logic operations and generate random number, then its proposition is not supported by EPC C1G2's Tag.

In this study, we investigate other security and privacy weaknesses of Pang et al. that have not been discussed by Behzad et al. and then, in order to increase the security and privacy of Pang et al. and to eliminate the existing weaknesses, we apply some modifications on the analyzed protocol and proposed an improved lightweight mutual authentication protocol conforming to EPC C1G2 standard. For more evaluation, the improved protocol is analyzed in terms of security and we show that all existing attacks are removed also it is secure against different attacks. We briefly review some related work and examine their weaknesses.

3. EPC-C1G2 Standard and Security Requirements

3.1 EPC-C1G2 Requirements

RFID Class 1 Generation 2 (C1G2) standard has been issued by EPCglobal [4]. It defines RFID standards as follows:

- RFID tag is passive.
- RFID tag communicates on the UHF band (800–960 MHz) and it communicates in the range from 2 m to 10 m.
- A Pseudo-Random Number Generator (PRNG) function.
- A Cyclic Redundancy Code (CRC) function, to produce checksum code to verify the integrity of the transmitted information.
- XOR: Exclusive OR.
- Reserved memory that contains a 32-bit kill password (KP) to permanently disable the tag and a 32-bit access password (AP).

As an EPC-C1G2 tag has very limited resources, thus it is incapable of supporting complex operations like symmetric encryption, public encryption, and hash function.

3.2 Security Requirements of EPC-C1G2 System

Low cost RFID systems generate significant security risks, mainly due to their cost constrained implementations and the insecure communication channels over which tags and readers communicate.

The following security issues are frequently discussed for low cost RFID systems:

- Secret Parameters Reveal
- Replay attack
- Forward secrecy
- De-synchronization (Dos) Attack
- Traceability Attack
- Tag Impersonation Attack
- Server Impersonation Attack

4. Review of Pang's Protocol

In this section we briefly review Pang's protocol. The scenario of the protocol is divided into the initialization and the authentication phases.

4.1 Initialization phase

RFID For each tag, the information kept within the database are [K_{old},$C_{old,}$,K_{new},C_{new},EPC_s,D_i]. The initial information K_0, P_0 and C_0 of tag and database are randomly generated by the manufacturer. Then, we set $K_{old} = K_{new} = K_0$, and $C_{old} = C_{new} = C_0$. Each tag records the information [$K_i = K_0$, $C_i = C_0$, EPC$_s$], whose values are equal to those recorded in the database. The reader does not need any stored identity information, because the communication channel is supposed secure between server and reader.

4.2 Authentication phase

The different steps of authentication phase are as described below:

1) Reader → Tag: The reader *R* generates a random number N_1 and sends it as a query to the tag *Ti*.
2) Tag → Reader: The tag *Ti* generates a random number N_2, and computes $M1 = EPCs \oplus N1 \oplus Ki$, $CN2 = N2 \oplus PRNG(Ki)$, and $M2 = CRC(EPCs \oplus N2 \oplus Ci) \oplus Ki$ and forwards them to the reader.
3) *Reader → Server:* The reader *R* forwards (Ci, M_1, CN_2, M_2, $N1$) to the server.
4) *Server → Reader:* After receiving (Ci, $M1$, $CN2$, $M2$, $N1$), the server performs the following operations:

(i) According to the received value C_i, the server uses C_{inew} or C_{iold} equal to C_i and picks up *EPCs* of the correct tag, and then computes $K_i = EPCs \oplus N_1 \oplus M_1$. If $K_i = K_{old}$ or K_{new}, the server computes also $N_2 = CN_2 \oplus PRNG\ (K_i)$ and checks if the equation $M_2 \oplus K_i = CRC\ (EPCs \oplus N_2 \oplus C_i)$ holds or not. The process is repeated for each entry until the matched tag is found. Otherwise, the server aborts the process.
(ii) Once the tag is authenticated successfully by the server, it computes $M_3 = CRC\ (EPCs \oplus (N_2\ l/4)) \oplus Ki$, and sends $(Di,\ M_3)$ to the reader R.
(iii) The server proceeds to update its records as follows:

$$K_{iold} \leftarrow K_i,\ K_{inew} \leftarrow K_i \oplus (N_2 >> l/4) \qquad (1)$$

$$C_{iold} \leftarrow C_i,\ C_{inew} \leftarrow PRNG\ (N_1 \oplus N_2) \oplus K_i \qquad (2)$$

5) *Reader → Tag:* The reader *R* receives (Di, $M3$) from server and then forwards M_3 to the tag T_i.

6) *Tag* : The tag T_i computes $CRC\ (EPCs \oplus (N_2\ l/4)) \oplus K_i$ and checks whether $M3 \oplus Ki = CRC\ (EPCs \oplus (N2 >> l/4))$ is correct, If yes, then the tag authenticates the server successfully and updates the records as follows:

$$C_i \leftarrow PRNG\ (N_1 \oplus N_2) \oplus K_i \qquad (3)$$

$$K_i \leftarrow K_i \oplus (N_2 >> l/4) \qquad (4)$$

If it does not hold, the tag *Ti* stops the session

4.3 Security Analysis

In addition to the discussed attacks by Bezhad et al., we present an efficient and passive attack that retrieves all secret information of the tag include *EPCs*,*Ki*, *Ci*.

Moreover, we prove by other way, which is different from the one discussed by Bezhad et al, that this scheme is vulnerable to de-synchronization.

a) *Secret Information Disclosure Attack*

Pang et al. claimed that their protocol attempt a complexity of exhaustive search attack equal to O (2^{32}) but in this subsection we aim to show that the complexity is just O (2^{16}), by consequently the Pang's protocol cannot resist against the secret information disclosure attack.
The attacker acts as follows:

Learning phase: Eavesdrops one successful session of the protocol between tag and reader and stores the exchanged messages including M_3, Ci, M_1, CN_2, M_2 and N_1.
Attack phase: Based on the calculated equation M_1, M_2 and M_3 the length of *EPCs* is 16-bit,
Let $\alpha = \{\alpha_1, \alpha_2, \alpha_3, \alpha_{2^{16}}\}$ be the set of all bit string of length 16 and $EPCs \in \alpha$, the attacker runs the following algorithm:

Algorithm 1: attack senario

```
For $1 \leq j \leq 2^{16}$
    Choose: $EPCs_j \in \alpha$
    Computes:
        $K_j = M_1 \oplus N_1 \oplus EPCs_j$
        $N_2 = CN_2 \oplus PRNG\ (K_j)$
  If  $(M_2 \oplus K_j = CRC\ (EPs_j \oplus N_2 \oplus C_j)$ and $M_3 =$
        $CRC\ (EPCs \oplus (N_2 >> (l/4))) \oplus K_j)$
  Return $K_j$ and $EPCs_j$
End For
```

The attacker obtains the secret values *EPCs* and K_j corresponding to a tag.
Concerning the complexity of the given attack, it can be concluded that the attacker needs just one session to collect the exchanged messages between reader and tag, and he/she needs at most 2^{16} *PRNG* and $2 * 2^{16}$ *CRC* computations. Then this attack can be easily performed by an ordinary attacker.

After obtaining the secrets values of the tag, it will be easy that an attacker launches others attacks such as reader impersonation attack, tag impersonation attack, traceability attack and DoS attack.

b) *De-synchronization Attack*

Using CRC linear property, we show how an attacker performs de-synchronization attack between the tag and the server and cause DoS.
The proposed cryptanalysis is described as follows:

Theorem 1*: For any CRC and for any values of x and y* $\in F_2$

$$CRC\ (x \oplus y) = CRC\ (x) \oplus CRC(y) \tag{5}$$

Based on the above property, the attacker can launch the DoS attack:

Learning phase: In step 2 when the tag sent (M_1, M_2 ,Ci, , CN_2) to reader , the attacker intercepts them and generates and random value *RN*.
Attack phase: Attacker replaces the intercepted messages by (M_1, $M_2^{'}$,Ci, $CN_2^{'}$) and send them to reader where:

$$M_2^{'} = M_2 \oplus CRC\ (RN) \tag{6}$$

$$CN_2^{'} = CN_2 \oplus RN \tag{7}$$

The server proceeds to authenticate the forged tag, for that it computes and checks:

$$M_2^{'} \oplus K_j \overset{?}{=} CRC\ (EPCs \oplus N_2 \oplus C_i) \tag{8}$$

Where $N_2 = CN_2^{'} \oplus PRNG\ (K_i)$ (9)

and $CN_2^{'} = CN_2 \oplus RN$ (10)

Then equation (8) becomes as follow:

$$M_2 \oplus CRC(RN) \oplus K_j = CRC\ (EPCs \oplus CN_2 \oplus RN \oplus PRNG\ (K_i) \oplus C_i) \tag{11}$$

By applying the theorem 1, the equation (11) becomes:

$$M_2 \oplus CRC(RN) \oplus K_j = CRC\ (EPCs \oplus CN_2 \oplus PRNG\ (K_i) \oplus C_i) \oplus CRC(RN) \tag{12}$$

Then

$$M_2 \oplus K_j = CRC\ (EPCs \oplus CN_2 \oplus PRNG\ (K_i) \oplus C_i \tag{13}$$

As these parameters are legal, then the above equality holds. As consequently, the server authenticates the forged tag, and it will update its secrets parameters *Ki* and *Ci* using $RN \oplus N_2$.

In the same way, we show how an attacker spoofs a tag to pass its authenticity.

When reader forwards *M*$_3$ to tag, and attacker can intercepts it by eavesdropping the communication channel and replaces *M3 by:*

$$M_3^{'} = M_3 \oplus CRC\ (RN >> \ (l/4)) \qquad (14)$$

Where:

$$M_3 = CRC\ (EPCs \oplus ((N_2 \oplus RN) >> \ (l/4)))) \oplus K_j \qquad (15)$$

Then the tag computes and verifies:

$$M_3^{'} \oplus K_j \overset{?}{=} CRC\ (EPCs \oplus (N_2 >> \ (l/4))) \qquad (16)$$

$$M_3 \oplus CRC(RN >> (l/4)) \oplus K_j = CRC\ (EPCs \oplus ((N_2 \oplus RN) >> (l/4)))) \oplus K_j \oplus CRC\ (RN >> (l/4)) \oplus K_j \qquad (17)$$

By applying the theorem 1, the equation (17) becomes:

$$M_3 \oplus CRC\ (RN >> (l/4))) \oplus K_j = CRC(EPCs \oplus (N_2 >> (l/4))) \oplus K_j \oplus CRC\ (RN >> \ (l/4)) \oplus CRC\ (RN >> \ (l/4)) \oplus K_j \qquad (18)$$

Then the above equation holds. As consequently, the tag authenticates the forged reader, and it updates its secrets parameters *Ki* and *Ci* using *N*$_{2.}$

As showed before the tag and server uses *N*$_2$ and *RN* $\oplus$ *N*$_2$ respectively to update their secrets parameters *Ki* and *Ci*, , then the shared secret between the tag and the server will not be the same , which can cause a de-synchronization between them.

5. Improved Protocol

Our main contributions can be summarized as follows:

In order to eliminate the mentioned vulnerabilities and the one discussed by Bezhad et al., we propose an improved RFID authentication protocol conforming to the EPC-C1G2 standard.

As EPC-C1G2 standard authorize simple operations which can be performed by a tag, such as Cyclic Redundancy check Code (CRC), Pseudo Random Number Generator (PRNG), and bitwise XOR , an EPCC1G2 tag could not perform the hash function, so the proposed suggestion by Bezhad et al. that used hash function are too complicated and doesn’t conform to EPC-C1G2 tags.

The proposed protocol does not use hash functions or other complex encryption schemes, we use only PRNG and XOR operations. The << shift operation applied by Pang et al. and Bezhad et al will not applied in our protocol because this operation do not conform to EPC-C1G2 standard. Moreover, to avoid the de-synchronization attack due to the linear property of CRC operation, the proposed scenario will not applied CRC operation.

We assume that the communication channel between the reader and the backend server is secure, because a reader has much more resources, such as memory, energy and computation power, than a tag. For the channel communication between reader and tag, we assume that it is insecure.

In order to prevent from exhaustive search attack during the proposed protocol, some exchanged messages are computed by applying separated PRNG functions.

For any PRNG (A) and PRNG (B), if A = B, to determine A and B we needs O (2^{16}) executions of PRNG function, but it is not possible to find A and B after O (2^{16}) PRNG executions for given PRNG (A) $\oplus$ PRNG (B), so we needs O (2^{16}) PRNG executions possible for each A and B, then at least 2^{32} executions.

Moreover to avoid easily retrieve of secret identification parameters, it is necessary to randomize the PRNG function.

Notations used in this paper are defined as follows:

Table 1: Notations

Notation	***Description***
EPCs	The 96 bits of EPC code are divided into six 16-bit blocks, and then the six blocks are XORed to get EPCs
SKi	The authentication key stored in the tag for the database to authenticate the tag at the (i + 1)th authentication phase
SP$_i$	The access key stored in the tag for the tag to authenticate the database at the (i + 1)th authentication phase
SK$_{old}$	The old authentication key stored in the database
SK$_{new}$	The new authentication key stored in the database
SP$_{old}$	The old access key stored in the database
SP$_{new}$	The new access key stored in the database
I$_i$	The database index stored in

	the tag to find the corresponding record of the tag in the database
SMi	Exchanged messages
EPCs	The 96 bits of EPC code are divided into six 16-bit blocks, and then the six blocks are XORed to get EPCs

The stored information in each entity:
Database: (SK_{old}, SK_{new},SP_{old}, SP_{new}, I_{old}, $I_{new,}$ *EPCi*)
Tag: (*SKi*, *SPi*,I_i ,*EPCs*)

5.1 Initialization phase

During the initialization phase, the records stored previously in each party's memory are assumed secure.
Initial information *SP0*, *SK0* and *I0* of tag and database are generated by the constructor, and set the corresponding record in the tag (*SPi* =*SP0*, *SKi* = *SK0*, *Ii* =I_0) and in the database (SP_{old} = SP_{new} =*SP0*, SK_{old} = SK_{new}=*SK0*, I_{old} = I_{new}=I_0).

5.2 Mutual Authentication phase

The authentication protocol operates as follows:

***Step 1**- Reader →Tag*: The reader generates random number RN_R and sends it as a query to the tag.
***Step 2**- Tag → Reader:* After receiving the message, the tag generates RN_T and computes $SM_1 = PRNG\ (EPCs \oplus RN_R) \oplus PRNG\ (SKi \oplus RN_T)$ and $SM_2 = PRNG\ (SKi \oplus EPCs \oplus I_i) \oplus RN_T$ using the local secret parameters.
Then the tag sends SM_1, SM_2 and I_i to the reader.
***Step 3**- Reader → Server:* The reader forwards SM_1, SM_2, RN_R and I_i to the tag.
***Step 4**- Reader → Server:* After receiving (SM_1, SM_2, RN_R and I_i), the database performs the following operations:
a) *Authentication phase*: the server uses *Ii* as an index to find the corresponding record in the database and it will extracts RN'_T from $SM_2 \oplus PRNG\ (SKi \oplus EPCs \oplus I_i)$ and computes $SM'_1 = PRNG\ (EPCs \oplus RN_R) \oplus PRNG\ (SKi_{,x} \oplus RN'_T)$ to authenticate the tag, where x = *old* or *new*.
When a matching is found, the server set x as *old* or *new* according to authentication key SK_{new} or SK_{old} in the record is found matched with the one in the tag.
If the two values RN'_T and SM'_1 match, then the tag is authenticated successfully by Server, otherwise the session aborts.

Now the Server proves its authenticity to tag by computing $SM_3 = PRNG\ (EPCs \oplus RN'_T) \oplus PRNG\ (SP_i \oplus RN_R)$ and sends it to the reader.
b) *Updating phase*: If x = *new*, then the database will update the record as follows:

$$SK_{old} \leftarrow SK_{new},\ SK_{new} \leftarrow PRNG(SK_i) \quad (19)$$

$$SP_{old} \leftarrow SP_{new},\ SP_{new} \leftarrow PRNG(SP_i) \quad (20)$$

$$I_{old} \leftarrow I_{new},$$
$$I_{new} \leftarrow PRNG\ (RN_T \oplus SP_x) \oplus PRNG\ (RN_R \oplus SK_x \oplus I_i) \quad (21)$$

If x = *old*, then the server just updates:

$$I_{new} \leftarrow PRNG\ (RN_T \oplus SP_x) \oplus PRNG(RN_R \oplus SK_x \oplus I_i) \quad (22)$$

***Step 6**:* After receiving SM'_3, the tag uses its saved parameters to compute $PRNG\ (EPCs \oplus RN_R) \oplus PRNG\ (SP_i \oplus RN_T)$ and compares it with received *SM'3,* if a match is found, then the server is authenticated successfully by tag and the content kept inside is updated as $SK_{i+1} \leftarrow PRNG\ (SKi)$, $SP_{i+1} \leftarrow PRNG\ (SP_i)$, and $I_{i+1} \leftarrow PRNG\ (RN_T \oplus SP_x) \oplus (RN_R \oplus SK_x \oplus I_i\ )$ for next session.

6. Security Analysis

In this section, we give security analysis of our scheme against different common possible attacks in low cost RFID system.
In order to increase the security level in our protocol, the messages SM_1, SM_2, SM_3 and I_i have been computed from at least three unknown parameters and all linear combinations between at least two of $SM_{1,}$ $SM_{2,}$ SM_3 and $I_{i,new}$ are associated at least three unknown parameters.

6.1 Parameters Reveal

For designing a RFID authentication protocols, it is necessary to keep the secret information (*EPCs*, SP_i , SK_i) of tag secure.
We show how the proposed protocol resists the exhaustive research with complexity O (2^{32}).
To retrieve the secret parameters the attacker needs to intercepts the exchanged messages RN_R ,$SM_{1,}$ $SM_{2,}$ SM_3 and I_i and performs the following algorithm:

Algorithm 2: attack senario

```
For $EPCs_j : 0 \le j \le 2^{16} - 1$
    For $Sk_{i,h} : 0 \le h \le 2^{16} - 1$
        Computes: $RN'_T = PRNG\ (SK_{i,h} \oplus EPCs \oplus I_i) \oplus SM_2$
        The attacker need to check:
        If $SM_1 = PRNG\ (EPCs_j \oplus RN_R) \oplus PRNG(SK_{i,h} \oplus RN'_T)$ is correct
            Return $SK_{i,h}$ and $EPCs_j$
    End for
End For
```

From the above scenario we conclude that the attacker needs to travers at least two variable from 0 to 2^{16} - 1, as consequently the proposed protocol can resist to exhaustive search attack with complexity O (2^{32}).

6.2 Replay Attack

During the authentication phase, new pair of random numbers RN_T and RN_R have been generated in each session, and these random are used to protect the exchanged information from replay attacks in the next session of the authentication.

6.3 Forward Secrecy

In the proposed protocol, if an attacker gets the secret information of the tag in the current session, he/she will not be able to obtain the secret information of previous session, because at the end of each successful session, the authentication and access keys stored in the tag are updated using the PRNG function.

If attacker needs to trace the tag's previous communications, he/she should intercept SM_1 and SM_2 and XOR them to obtain $PRNG\ (EPCs \oplus RN_R) \oplus PRNG\ (SKi \oplus RN_T) \oplus PRNG\ (SKi \oplus EPCs \oplus I_i) \oplus RN_T$. Even with the compromised $EPCs$ and intercepted RN_R and $I_{i,}$ he still needs RN_T and SKi to pass the verification. However, RN_T and SKi are transmitted with the protection of the previous authentication key. Then the attacker cannot find the previous secret. So, the forward secrecy is satisfied.

6.4 De-synchronization Attack

If attacker blocks the last message, he/she does not de-synchronize the tag and the server, because the server keeps the old and new values (SK_{new} ,SK_{old} ,SP_{new} ,SP_{old} , I_{new} , I_{old}) which are matched with the tag, and it is still allowed to communicate with the reader in the next session. Hence, the proposed protocol resists against the de-synchronization attack.

6.5 Traceability Attack

In our scheme, the shared secret parameters between the server and tag are updated at the end of every successful session, moreover new random numbers RN_T and RN_R are generated for every session. Then an attacker cannot track a tag based on the intercepted messages, so the proposed resist against traceability attack.

6.6 Tag Impersonation Attack

In the proposed protocol, if an attacker try to lunch tag impersonation attacks, he/she needs to compute a valid messages SM_1 and SM_2 of tag, using $EPCs$, SKi, $I_{i,}$ RN_R, RN_T .since SKi and I_i are protected and updated at the end of each session, moreover RN_R and RN_T are refreshed in next session, an attacker cannot impersonate the tag. As a result the proposed protocol is secure against tag impersonation attacks.

6.7 Server Impersonation Attack

In the proposed protocol, if an attacker try to lunch server impersonation attacks, he/she needs to compute a valid messages SM_3 of server , using $EPCs$, $SPi_{,}$ RN_R ,RN_T .since SPi is protected and updated at the end of each session ,moreover RN_R and RN_T are refreshed in next session, an attacker cannot impersonate the server. As a result the proposed protocol is secure against sever impersonation attacks.

7. Performance Analysis

The performance of the proposed protocol is evaluated in different aspects which are security analysis, computation cost and communication cost.

7.1 Computation Cost

The main problem in designing a secure authentication protocol conforming to RFID EPC-C1G2 standard is the computation restriction on the tags.

The proposed protocol only requires lightweight operations on the tag which are bitwise XOR and PRNG functions. These functions are low-cost and can be efficiently implement in hardware.

7.2 Communication Cost

During the mutual authentication phase, the tag transmit 3 messages (SM_1, SM_2 and I_i) in our protocol against 4 messages (Ci, M_1, CN_2 and $M_{2)}$ in Pang et.al protocol. Then, the tag send 3l (48 bits) against 4l (64 bits) in Pang et. al protocol, with the length of one message is l, in our case l equal 16bits. As consequently, the proposed protocol provides a low communication cost.

7.3 Database Loading

In the proposed protocol, the index I_i is used to access the database, this allows the server to retrieve the data record corresponding to tag in the database, as consequently the performance of the system has not been changed.

8. Conclusions

In this paper, we have shown that Pang et.*al*'s protocol does not achieve the claimed security and not conform to EPC-C1G2.

As EPC-C1G2 low-cost tags have limited storage capabilities and computation power, we have designed a novel lightweight mutual authentication protocol conforming to EPC-C1 G2 standard. Our scheme provides high security level and used simple operator (XOR and PRNG) on the tag, which are suitable for current EPC-C1 G2 standard.

References

[1] Frank B. Song and C. J. Mitchell, Scalable rfid security protocols supporting tag ownership transfer, (2011) Comput. Commun, vol. 34, pp. 556–566, April.

[2] K. Ouafi and R. C.-W. Phan, Privacy of Recent RFID Authentication Protocols, 4th International Conference on Information Security Practice and Experience – ISPEC 2008, ser. Lecture Notes in Computer Science, L. Chen, Y. Mu, and W. Susilo, Eds., vol. 4991. Sydney,Australia: Springer, April 2008, pp. 263–277.

[3] T. Dimitriou, A lightweight rfid protocol to protect against traceability and cloning attacks, Proceedings of the 1st International Conference on Security and Privacy for Emerging Areas in Communications Networks, ser. SECURECOMM '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 59–66.

[4] EPCglobal, EPC radio-frequency identity protocols class-1 generation-2 UHF RFID protocol for communications at 860 MHz– 960 MHz, Version 1.2.0, Specification for RFID Air Interface, EPCglobal, 2008.

[5] D.V. Bailey, and A. Juels, Shoehorning security into the EPC tag standard, In R. D. Prisco, and M. Yung, editors, SCN, of Lecture Notes in Computer Science, Springer, 2006, Vol. 4116, pp. 303–320.

[6] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-T apiador, and A. Ribagorda, RFID specification revisited , In The internet of things: From RFID to The Next Generation Pervasive Networked Systems, Taylor & Francis Group, 2008, pp.311–346.

[7] Chien HY, Chen CH. "Mutual authentication protocol for RFID conforming to EPC Class 1 Generation 2 standards". Computer Standards & Interface Journal 2007; 29: 254-259.

[8] K. Ahsan , and H. Shah , P. Kingston, " RFID Applications: An Introductory and Exploratory Study", IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 1, No. 3, pp. 1-7 , 2010.

[9] P. Peris-Lopez, T. Li, T.L. Lim, J.C. Hernandez-Castro,and J.M. Estevez-Tapiador, "Practical attacks on a mutual authentication scheme under the EPCClass-1Generation-2standard", Computer Communications,vol.32, No7,pp.1185-1193,2008.

[10] Chen CL, Deng YY. "Conformation of EPC Class 1 Generation 2 standards RFID system with mutual authentication and privacy protection". Engineering Applications of Artificial Intelligence 2009; 1284-1291.

[11] T. C. Yeh, Y. J Wang, T. C. Kuo and S. S. Wang, "Securing RFID systems conforming to EPC Class 1 Generation 2 standard", Expert Systems with Applications, vol.37,No 12, pp. 7678–7683 ,2011

[12] E.-J. Yoon, "Improvement of the securing rfid systems conforming to epc class 1 generation 2 standard," Expert Syst. Appl., vol. 39, no. 12, pp. 1589–1594, Dec. 2012.

[13] Pang, L. J., He, L., Pei, Q., & Wang, Y. Secure and efficient mutual authentication protocol for RFID conforming to the EPC C-1 G-2 Standard, In 2013 IEEE wireless communications and networking conference (WCNC): NETWORKS (pp. 1870–1875). IEEE Computer Society.

[14] B. Abdolmaleki, H. Bakhsi, K. Baghery, and M. R. Aref, Analysis of an RFID authentication protocol in accordance with EPC standards, (2014) International Journal of Information & Communication Technology Research, vol. 6, pp. 7-12.

Ahmed Maarof has received M.S. degrees in Telecommunications and Microelectronics from FSTF (Faculty of Science and Technology), Fez, Morocco in 2005. Since 2012, he is working toward a Ph.D. degree at college of Engineering, Ecole Nationale Supérieure d'Informatique et d'Analyse des Systèmes (ENSIAS), Mohammed V University. His interests are security in low cost RFID security and Lightweight cryptography for Internet of things (IoTs), low power circuit design techniques for passive tags. Since 2006, Ahmed Maarof has worked as sub-Contractor hardware and software engineer for several company such as Texas instruments, ON-semiconductor, Wolfson,ST-Ericsson ,Zodiac Aerospace and Freescale , now he is a sub-contractor hardware engineer for NXP.

Mohamed Senhadji is currently an Assistant Professor at the communication networks department of Ecole Nationale Supérieure d'Informatique et d'Analyse des Systèmes (ENSIAS), Mohammed V - Souissi University, Morocco. He gives several courses at ENSIAS school such Computer Architecture, Assembly, Microprocessors and Implementation Networks, Physical security and smart card. His research interests lie with the field of wireless networking, RFID technologies, Internet of Things (IoTs) and Ad-hoc networks.

Zouheir Labbi has received M.S. degrees in Telecommunications and Microelectronics from FSTF (Faculty of Science and Technology), Fez, Morocco in 2005. Since 2012, he is working toward a Ph.D. degree at college of Engineering, Ecole Nationale Supérieure d'Informatique et d'Analyse des Systèmes (ENSIAS), Mohammed V University. His interests are security in low cost RFID and application of RFID technologies in Internet of things (IoTs). Since 2006, Zouheir Labbi has worked as support and software engineer for Alcatel-Lucent, now he is as software engineer at NOKIA.

Mostafa Belkasmi is a professor at ENSIAS (Ecole Nationale Supérieure d'Informatique et d'Analyse des Systèmes, Rabat); head of Telecom and Embedded Systems Team at SIME Lab. He had PhD at Toulouse University in 1991(France). His current research interests include, RFID technologies, Internet of Things (IoTs), mobile and wireless communications, interconnections for 3G and 4G, and Information and Coding Theory.