Cryptanalysis of an EPC Class-1 Generation-2 Standard Compliant Authentication Protocol

Pedro Peris-Lopez[a,*], Julio C. Hernandez-Castro[b], Juan M. E. Tapiador[c], Jan C.A. van der Lubbe[a],

[a] *Delft University of Technology (TU-Delft), Faculty of Electrical Engineering, Mathematics, and Computer Science (EEMCS), Information and Communication Theory group (ICT), P.O. Box 5031 2600 GA, Delft, The Netherlands*
[b] *School of Computing Science, Buckingham Building, Lion Terrace, Portsmouth PO1 3HE, United Kingdom*
[c] *Department of Computer Science, University of York, Heslington, York, YO10 5DD, United Kingdom*

Abstract

Recently, Chen and Deng proposed a mutual authentication protocol [7]. Their scheme is based on a cyclic redundancy code (CRC) and a pseudo-random number generator in accordance with the EPC Class-1 Generation-2 specification. Authors claimed that the proposed protocol is secure against all attacks on RFID systems, offering and increase in security and performance in comparison with their predecessors. However, in this paper we show that the protocol is as insecure as the EPC standard it unsuccessfully tries to improve, which security limitations are well known. An attacker, following our suggested approach, will be able to impersonate both readers and tags. Untraceability is not guaranteed, being possible and even easy to associate a tag with its future broadcast answers with a high probability. Readers are vulnerable to a denial of service attack (DoS), obtaining an incorrect EPC identifier after the successfully authentication of the tag. Finally, for the implementation point of view, the length of variables are not compatible with those of the standard, thus discouraging even further the wide deployment of their protocol.

Key words: RFID, EPC, Security, Authentication, Cryptanalysis

1. Introduction

A Radio Frequency Identification (RFID) system is a set of automated identification technologies in which a small transponder (tag), attached to an object

*Corresponding author
Email addresses: P.PerisLopez@tudelft.nl (Pedro Peris-Lopez), Julio.Hernandez-Castro@port.ac.uk (Julio C. Hernandez-Castro), jet@cs.york.ac.uk (Juan M. E. Tapiador), J.C.A.vanderLubbe@tudelft.nl (Jan C.A. van der Lubbe)

(i.e. person, animal, product), receives and responds to radio-frequency queries from a transceiver (reader). Nowadays, barcodes are the most extended identification systems, but this technology may be replaced by RFID in a near future. The attractiveness of the RFID over the barcodes is twofold. First, the technological advantages: data can be read automatically, without line of sight and through a non-conducting material such as cardboard or paper, at a rate of hundreds of times per second, and at a distance of several meters. Secondly, the unequivocally identification provided by RFID technology: an RFID tag assigns an unique identifier to each tagged item, while a barcode only specifies the type of the labeled product. Despite these benefits, security and privacy concerns are holding up the rapid and widespread adoption of this promising technology.

Due to the heterogeneity of RFID systems, there is a great number of interconnected standards. ISO [2] and EPCglobal [11] have played an important role in harmonization. In 2004, Electronic Product Code Class-1 Generation-2 specification (EPC-C1G2 in short) was adopted by EPCGlobal [9]. Few months later, it was ratified by ISO and published as an amendment to its ISO/IEC 18000-6 [1]. This standard is an important milestone for the standardization of low-cost RFID tags. However, the different analyses of the security carried out on the EPC-C1G2 specification reveal important security flaws [5, 18]. Some researchers have later proposed EPC-friendly schemes trying to correct these weaknesses. One of the most recent proposals following this approach is Chen and Deng's scheme [7], which is our concern in this paper.

The rest of the paper is organized as follows. The related work is reviewed in Section 2. In Section 3, we present Chen and Deng's protocol. The properties of CRC functions are studied in Section 4. We present our attacks and the non-conformity with the standard in Section 5. Finally, we extract some conclusions in Section 6.

2. Related work

Motivated by the low security level of EPC-C1G2 specification, some recent proposals attempt to correct its deficiencies whilst still conforming to the standard. Next, we briefly summarizes the most important proposals in this research direction.

In [5], Juels et al. examined various ways for RFID tags to perform cryptographic functions while remaining EPC-C1G2 compliant. Their main idea is to take an expansive view of EPC tag memory. Instead of considering memory merely as an storage media, they use it as an input/output way of interfacing with a cryptographic module within the tag. Read/write commands may therefore carry out cryptographic values, such as messages in a challenge-response protocol. Their work clearly shows the need for mutual authentication between readers and tags. However, the assumption that a low-cost tag might support on-board cryptographic modules is not realistic, at least at present time.

Karthikeyanand and Nesterenko [12] proposed an efficient tag identification and reader authentication protocol based on simple XOR and matrix operations.

Two matrices and a key are stored in both the tag (K, M_1, M_2^{-1}) and the back-end database (K, M_1^{-1}, M_2). Once the tag is identified, the reader sends to the tag messages Y, Z. The first is used to authenticate the tag and the second to update the key. However, an attacker can substitute the original Z by a random Z'. Upon receiving Y, Z', the tag will be authenticated and will update the key wrongly. So the legitimate reader and the tag will not be able to authenticate each other any more. Additionally, the protocol is vulnerable to replay attacks and privacy location is not guaranteed [8].

Duc et al. [15] later proposed a tag-to-back-end database authentication protocol. The security of Duc et al.'s protocol is based on key synchronization between tags and back-end database. The last message of the protocol is comprised of an *EndSession* command, which is sent to both tags and readers. Interception of one of these messages will cause a synchronization loss between the tag and the server. So the tag and the reader will not be able to authenticate any more, which is an extremely serious problem. This protocol also presents backward secrecy problems, as compromise of the EPC allows an attacker to trace back all past communications.

Chien et al. pointed out certain weaknesses in the schemes [12] and [15], and then proposed a new EPC-C1G2 compliant mutual authentication protocol [8]. However, Peris et al. [19] showed how none of the expected objectives are met being vulnerable to attacks such as identity impersonation, non-forward security, tracking, etc. Even the correct execution of the protocol results in a desynchronization between the tags and the back-end database.

In [13], Konidola and Kim produced an interesting paper which tried to correct some of the security shortcomings of the EPC-C1G2 specification. The authors hold that the proposed scheme frustrates the access password acquisition by a simple XOR operation, against what happened in the specification. However, Lim and Li [14] showed how a passive attacker can recover the password of the tag by eavesdropping over a single run of the protocol and performing some correlation analysis on the captured information. Then, Konidala and Kim proposed a new version of the TRMA scheme (TRMA+) in which the tag access and kill password are used for authentication. This new version still contains important security flaws, as the key and access password can be acquired by an adversary with non-negligible probability [20].

3. Chen and Deng's Protocol

In [7], Chen and Deng proposed an EPC-friendly scheme (CD-EPC in short) based on the use of a Pseudo-Random Number Generator (PRNG) and a Cyclic Redundancy Code (CRC), as recommended by the EPC-C1G2 standard. In the following, we briefly introduce the CD-EPC protocol, which consists of two phases; registration and initialization. The following notation is used thorough the paper:

$x_{\mathcal{T}_i}(j)$	value x of tag $\mathcal{T}_i$ registered in j^{th} database
$(N_{\mathcal{T}_i}(j), K_{\mathcal{T}_i}(j))$	$N_{\mathcal{T}_i}(j)$ is nonce word and $K_{\mathcal{T}_i}(j)$ is a key of tag $\mathcal{T}_i$
CRC()	a Cyclic Redundancy Code (CRC) function
$EPC_{\mathcal{T}_i}$	EPC identification number of tag $\mathcal{T}_i$
ID_{R_i}	identification number of the i^{th} reader
RND	random number
$\oplus$	exclusive-OR operation
M_{req}	reader's request message
M_{resp}	reader's response message

3.1. Registration phase

Tags and readers must register in the database separately under a secure environment. Tags send their unique $EPCs$ to the database. Then, the database responds with $N_{\mathcal{T}_i}(j)$ and $K_{\mathcal{T}_i}(j)$ to each tag that ask for registering. Once $N_{\mathcal{T}_i}(j)$ corresponds to only one $K_{\mathcal{T}_i}(j)$. In general, each tag may register in several databases obtaining a set of $\{N_{\mathcal{T}_i}(j), K_{\mathcal{T}_i}(j)\}_{j=1}^{n}$. Readers are registered in the database by their unique identification ID_{R_i}. After registration, the database responds with the tuples $(N_{\mathcal{T}_i}(j), K_{\mathcal{T}_i}(j))$ of all the assigned tags that can be accessed by reader ID_{R_i}. Authors also consider that readers may be registered in several databases at the same time. A common scenario depicting tags and readers registration is illustrated in Figure 1.

3.2. Communication phase

Through the registration phase, tags and readers can communicate mutually. Only random numbers, exclusive-OR operations and the lightweight CRC operation are utilized to compose the exchanged messages as the EPC standard demands. The proposed scheme is split into 5 steps (see Figure 2):

Step 1 When the reader wants to access a tag, it sends a request message M_{req}, $CRC(N_{\mathcal{T}_i}(j) \oplus RND_1)$ and RND_1 to the tag.

Step 2 Upon receiving $CRC(N_{\mathcal{T}_i}(j) \oplus RND_1)$ and RND_1, the tag uses the stored $N'_{\mathcal{T}_i}(j)$ to compute $CRC(N'_{\mathcal{T}_i}(j) \oplus RND_1)$. Thus, the tag can authenticate the reader via the following verification:

$$CRC(N'_{\mathcal{T}_i}(j) \oplus RND_1) \stackrel{?}{=} CRC(N_{\mathcal{T}_i}(j) \oplus RND_1) \tag{1}$$

If this equality does not hold, the tag will not perform any further calculations or responses; the request is assumed to be sent from an attacker or from a forbidden list. If it holds, a tag will generate a new random number RND_2 and compute:

$$X = K'_{\mathcal{T}_i}(j) \oplus EPC_{\mathcal{T}_i} \oplus RND_2 \tag{2}$$

$$Y = CRC(RND_2 \oplus N'_{\mathcal{T}_i}(j) \oplus X) \tag{3}$$

Step 3 : Tag sends (RND_2, X, Y) to the reader.

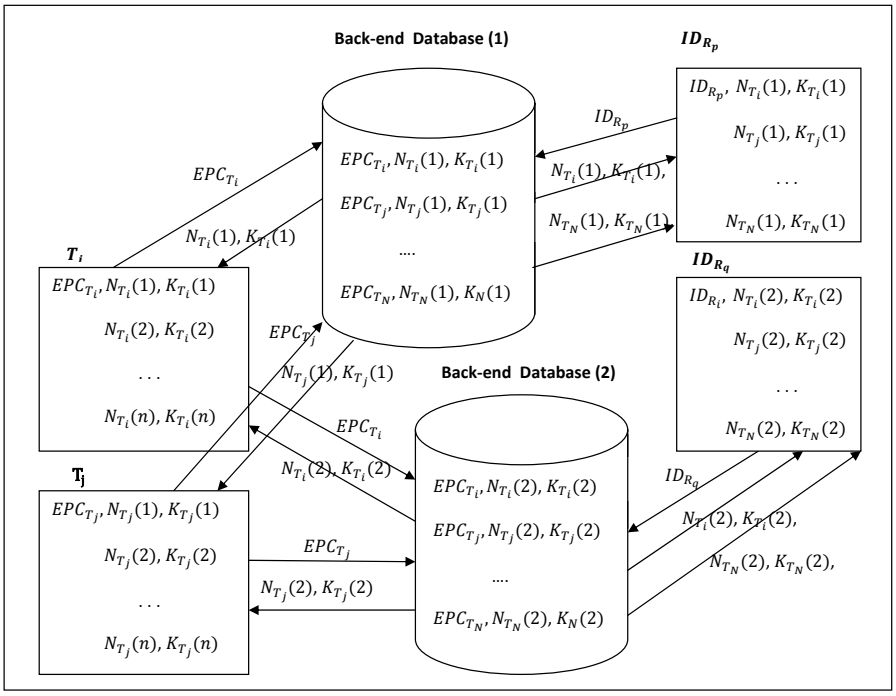

Figure 1: CD-EPC Initialization: Tags & Readers

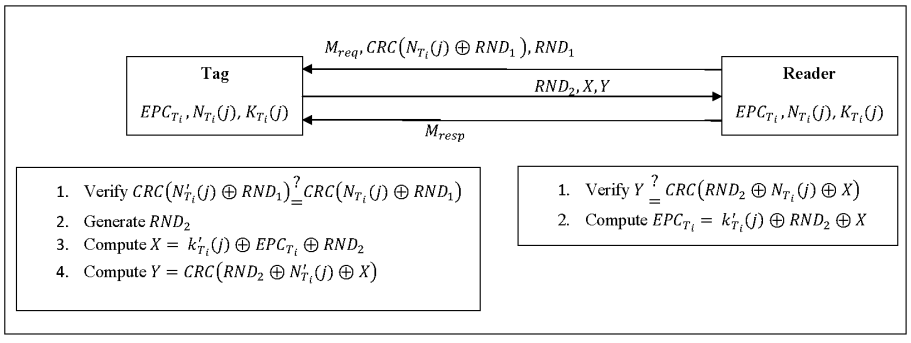

Figure 2: CD-EPC Mutual Authentication Protocol

Step 4 : Upon receiving the tag's response message, the reader computes its local version of $Y = CRC(RND_2 \oplus N_{\mathcal{T}_i}(j) \oplus X)$, using RND_2 and X as obtained from the tag. If the equality does not hold, this response may have been sent by an attacker, and the reader will not perform any further calculations or responses. If it holds, the reader uses $K_{\mathcal{T}_i}(j)$ (linked to $N_{\mathcal{T}_i}(j)$), and (RND_2, X) to obtain the static identifier of tag $\mathcal{T}_i$:

$$EPC_{\mathcal{T}_i} = K_i \oplus RND_2 \oplus X \tag{4}$$

Step 5 : When a reader obtains a tag's $EPC_{\mathcal{T}_i}$ and the authenticity of the tag has been confirmed, the reader sends a response M_{resp} to the tag.

4. Cyclic Redundancy Codes - CRCs

A Cyclic Redundancy Code (CRC) is a checksum algorithm that can be used to detect transmission errors (typically one or two bit flips, or bursts) in a very efficient way. CRCs operate by interpreting input binary sequences as polynomial coefficients that they divide over a prefixed polynomial in order to obtain a remainder, which, in its binary expression, constitutes the *crc* value.

CRCs are linear, so they shouldn't be used in cryptographic or security related applications as they cannot detect malicious changes by a knowledgeable attacker [4, 22, 24, 25]. To illustrate this property, the Hamming Distance (HD) can be used. The HD of a CRC polynomial is the minimum number of error bits that can pass undetected by the CRC. For example, if a CRC has a HD of 3, any combinations of 1 or 2 error bits will be detected, but there is at least one combination of 3 error bits that will pass unnoticed. Cryptographic hash functions, that have very high HD values, should therefore be used for any security related purpose instead.

4.1. Definitions and Notations

Let A be an m-bit string in $\{0,1\}^m$, $A = A_{m-1}||A_{m-2}||...||A_0$. We define $A_{<<n}$ as the $m+n$ bit string A' resulting of left-shift A by n-bits:

$$A'_i = \begin{cases} 0 & \text{for } 0 \leq i \leq n-1 \\ A_{i-n} & \text{for } n \leq i \leq n+m-1 \end{cases} \tag{5}$$

Let B be an n-bit string in $\{0,1\}^n$, where $n \leq m$. We define the exclusive-OR operation $A \oplus B = B \oplus A$ as follows:

$$(A \oplus B)_i = \begin{cases} A_i \oplus B_i & \text{for } 0 \leq i \leq n-1 \\ A_i & \text{for } n \leq i \leq m-1 \end{cases} \tag{6}$$

The set of bit strings $\{0,1\}^\infty$ forms a ground under the exclusive-OR operation. F_2 and $F_2[x]$ symbolize the binary field and the ring of polynomials over F_2, respectively. For a m-bit string A, we define a map $\phi : \{0,1\}^\infty \rightarrow F_2[x]$:

$$\phi(A) = \sum_{i=0}^{m-1} A_i x^i \tag{7}$$

As ϕ is a isomorphic group, there exits inverse (ψ). That is,

$$\psi(\Sigma_{i=0}^{m-1} A_i x^i) = A_{m-1}||A_{m-2}||...||A_0 \tag{8}$$

4.2. CRC Properties

CRC functions are based on polynomial arithmetic in F_2. Computing a crc value for a given binary stream is essentially dividing the polynomial associated with this stream by another fixed polynomial (generator polynomial) and obtaining a remainder. Let G be the generator polynomial used for calculating CRC. Then, the CRC for any bit-string A is computed by:

$$CRC(A) = \psi(\phi(A) \mod G) \tag{9}$$

Due to the linearity, CRCs have the following properties:

Theorem 1. *For any CRC (independent of its generator polynomial), and for any A n-bit and B m-bit string, it holds that:*

$$CRC(A \oplus B) = CRC(A) \oplus CRC(B) \tag{10}$$
$$CRC(A||B) = CRC(A_{<<n}) \oplus CRC(B) \tag{11}$$

Proof From the definition in Equation 9, one can write:

$$CRC(A \oplus B) = \psi(\phi(A \oplus B) \mod G) \tag{12}$$

Since modular operations, ψ, and ϕ are homomorphic, the above equation can be rewritten:

$$\begin{aligned} \psi(\phi(A \oplus B) \mod G) &= \psi((\phi(A) \mod G) \oplus (\phi(B) \mod G)) \\ &= \psi(\phi(A) \mod G) \oplus \psi(\phi(B) \mod G) \\ &= CRC(A) \oplus CRC(B) \end{aligned} \tag{13}$$

The concatenation of any two bit strings ($A||B$), can be viewed as the exclusive-OR between the n-bit shift of the left variable ($A_{<<n}$) and the right value (B). Thus, applying Equation 13:

$$\begin{aligned} CRC(A||B) &= CRC(A_{<<n} \oplus B) \\ &= CRC(A_{<<n}) \oplus CRC(B) \end{aligned} \tag{14}$$

□

5. Vulnerabilities of Chen and Deng's Protocol

In this section we analyze the most relevant weaknesses of the CD-EPC protocol.

5.1. Reader Impersonation

Each tag shares with the reader some private information: $N_{\mathcal{T}_i}(j)$ and $K_{\mathcal{T}_i}(j)$. This information is used to build the exchanged messages between these two devices in order to proof its authenticity. Specifically, the reader is authenticated by checking the following equation: $CRC(N'_{\mathcal{T}_i}(j) \oplus RND_1) \stackrel{?}{=} CRC(N_{\mathcal{T}_i}(j) \oplus RND_1)$.

Theorem 2. *In CD-EPC protocol, upon the eavesdropping of one authentic session, an adversary is able to respond to reader's queries correctly - with the consequent of tag impersonation - by sending* $RND2'$, $X' = X \oplus \Delta$, $Y' = Y$ *message, being* $\Delta = RND2' \oplus RND2$.

Proof

Step 1 The attacker eavesdrops an authentication session between the reader and the tag.

(1) $R \rightarrow T:$ $M_{req}, CRC(N_{\mathcal{T}_i}(j) \oplus RND_1), RND_1$
(2) $T \rightarrow R:$ RND_2, X, Y
(3) $R \rightarrow T:$ $M_{resp},$

Step 2 The attacker can supplant the reader by sending the following message:

(1) $A \rightarrow T:$ $M_{req}, CRC(N_{\mathcal{T}_i}(j) \oplus RND_1) \oplus CRC(\Delta), RND'_1$
where $\Delta = RND_1 \oplus RND'_1$
(2) ...

Upon receiving $CRC(N_{\mathcal{T}_i}(j) \oplus RND_1) \oplus CRC(\Delta)$ and $RND1'$, the tag uses $N'_{\mathcal{T}_i}(j)$ to compute its local value and compare with the received value.

$$CRC(N_{\mathcal{T}_i}(j) \oplus RND_1) \oplus CRC(\Delta) \stackrel{?}{=} CRC(N'_{\mathcal{T}_i}(j) \oplus RND'_1) \qquad (15)$$

From Equation 10, it holds that

$$\begin{aligned} CRC(N_{\mathcal{T}_i}(j) \oplus RND_1) \oplus CRC(\Delta) &= CRC(N'_{\mathcal{T}_i}(j) \oplus RND_1 \oplus \Delta) \\ &= CRC(N'_{\mathcal{T}_i}(j) \oplus RND_1 \oplus RND_1 \oplus RND'_1) \\ &= CRC(N_{\mathcal{T}_i}(j) \oplus RND'_1) \qquad (16) \end{aligned}$$

□

So, the message sent by the attacker is accepted as a valid message. The described attack is quite harmful because once an authentication session is eavesdropped, the attacker is able to supplant this reader indefinitely. This issue could be mitigated by the refreshing of the internal values $(N_{\mathcal{T}_i}(j), K_{\mathcal{T}_i}(j))$, but authors curiously did not opt for this possibility on their protocol design.

5.2. Tracking or Private Location

The untraceability can be viewed as a game $\mathcal{G}$ played between an adversary ($\mathcal{A}$) and a collection of readers (R_i) and tags instances ($\mathcal{T}_i$). The success of $\mathcal{A}$ in winning $\mathcal{G}$ therefore translates to its success in breaking untraceability [21]:

$$Adv_{\mathcal{A}}^{(UNT)} = |Pr(\mathcal{A}\ wins) - \frac{1}{2}| \tag{17}$$

where k is a security parameter (i.e. bit length of secret values). An RFID protocol achieves untraceability if $Adv_{\mathcal{A}}^{(UNT)} < \varepsilon(k)$, being $\varepsilon(.)$ some negligible function.

Theorem 3. *The CD-EPC protocol does not protect against privacy location; an adversary wins the untraceability ($\mathcal{G}$) gain with a significant probability:* $Adv_{\mathcal{A}}^{(UNT)} \simeq 0.499985$.

Two different approaches can be used by the adversary. First, the reader impersonation attack presented in the Section 5.1 may be employed (Proof-A). A passive attacker can trace any given RFID after observing one single authentication session because the secret $N_{\mathcal{T}_i}(i)$ is kept constant and the CRC function do not disguised it well enough. Secondly, the attacker may focus on the answers provided by tags (Proof-B). In that case, we exploit that authors abuse using bitwise operations resulting on the inclusion of a constant value in tags' responds.

Proof-A Specifically, the adversary $\mathcal{A}$ performs the following steps:

Learning Eavesdrop on an authentication session between the reader and the tag $\mathcal{T}_0$.

Challenge Some time later, the adversary chooses two fresh tags $\mathcal{T}_0$ and $\mathcal{T}_1$. Then, one of these tags is randomly selected ($\mathcal{T}_i \in \{\mathcal{T}_0, \mathcal{T}_1\}$) and presented to the adversary. He tries to impersonate it by the procedure described in Section 5.1.

Guessing If an answer message is obtained, $\mathcal{A}$ conjectures it is tag $\mathcal{T}_0$. Otherwise, $\mathcal{T}_1$ is guessed. There is only a negligible probability that by chance the procedure described on Section 5.1 will work for tag $\mathcal{T}_1$. As EPC-compliant tags support on-board a 16-bit CRC function [9], and assuming independence and uniformity in the random number generation and in the CRC output, this probability has a value of 2^{-16}.

So, the $Adv_{\mathcal{A}}^{(UNT)}(k)$ is non-negligible and the proposed protocol does not achieve untraceability:

$$Adv_{\mathcal{A}}^{(UNT)} \simeq |\frac{1}{2} - \frac{1}{2^{16}}| \tag{18}$$

□

Secondly, the attacker may analyze tags' answers. These answers would have to be anonymized to protect privacy location. However, the used of fresh random numbers just by itself does not guarantee this required and important property. Messages have to be carefully designed when messages are built by using modular operations [3]. As in the above case, we can show how privacy location is compromised by means of a game $\mathcal{G}$.

Proof-B We start observing a bad property of tags' answers, which is expressed by the following Lemma.

Lemma 1. *In CD-EPC protocol, tags respond a constant* $Y = CRC(N'_{\mathcal{T}_i}(j) \oplus k'_{T_i}(j) \oplus EPC_{T_i})$ *value despite of the usage of different nonces in each authentication session.*

The verification of the above Lemma is straightforward. After reader authentication, the tag answers the reader sending values, RND_2,X, *and* Y*:*

$$X = K'_{\mathcal{T}_i}(j) \oplus EPC_{\mathcal{T}_i} \oplus RND_2 \quad (19)$$

$$Y = CRC(RND_2 \oplus N'_{\mathcal{T}_i}(j) \oplus X) \quad (20)$$

Combining the above equations, a constant value is obtained:

$$\begin{aligned} Y &= CRC(RND_2 \oplus N'_{\mathcal{T}_i}(j) \oplus k'_{\mathcal{T}_i}(j) \oplus EPC_{\mathcal{T}_i} \oplus RND_2) \\ &= CRC(N'_{\mathcal{T}_i}(j) \oplus k'_{\mathcal{T}_i}(j) \oplus EPC_{\mathcal{T}_i}) \end{aligned} \quad (21)$$

Lemma 1 is used in the untraceability ($\mathcal{G}$) game. Specifically, the adversary $\mathcal{A}$ performs the following steps:

Learning Eavesdrop on an authentication session between the reader and tag $\mathcal{T}_0$ and store Y.

Challenge Some time later, the adversary chooses two fresh tags T_0 and T_1. Then, one of these tags is randomly selected ($\mathcal{T}_i \in \{\mathcal{T}_0, \mathcal{T}_1\}$) and presented to the adversary. The adversary eavesdrops an authentication session between this unknown tag and a legitimate reader.

Guessing If the same Y value is captured, $\mathcal{A}$ conjectures that the unknown tag is in fact $\mathcal{T}_0$. Otherwise, $\mathcal{T}_1$ is conjectured. Note that the probability that tag $\mathcal{T}_1$ and $\mathcal{T}_0$ lead to the same value for $Y = CRC(.)$ value is very low: $1/2^{16}$.

So, the proposed protocol does not offer protection against traceability because the $Adv_{\mathcal{A}}^{(UNT)}(k)$ is significant:

$$Adv_{\mathcal{A}}^{(UNT)} \simeq |\frac{1}{2} - \frac{1}{2^{16}}| \quad (22)$$

□

Summarizing, in this section we show two different (but related) passive attacks on the privacy location of the CD-EPC protocol. An attacker could exploit this vulnerability to associate a tag with its holder, and track him or her as he/she passes through different readers in, for example, different places in a city.

5.3. Tag Impersonation

In this section we show how an attacker is able to impersonate a legitimate tag after only eavesdropping one authentication session between the reader and the tag. The exploitation of this attack can be performed for an indefinite time, as the internal values of the tag tuple (EPC, $K_{\mathcal{T}_i}(j)$, $N_{\mathcal{T}_i}(j)$) remain constant during all its life. In [6], a similar attack is suggested but its proof is not included.

Theorem 4. *In the CD-EPC protocol, upon the eavesdropping of one authentic session, an adversary is able to respond to reader's queries correctly - with the consequent of tag impersonation - by sending* $RND2'$, $X' = X \oplus \Delta$, $Y' = Y$ *message, being* $\Delta = RND2' \oplus RND2$.

Proof Step 1 The attacker eavesdrops one authentication session between the reader and the tag.

$$\begin{array}{lll} (1) & R \rightarrow T: & M_{req},\, CRC(N_{\mathcal{T}_i}(j) \oplus RND1), RND1 \\ (2) & T \rightarrow R: & RND2, X, Y \\ (3) & R \rightarrow T: & M_{resp}, \end{array}$$

Step 2 The attacker can impersonate the tag by sending the following message:

$$\begin{array}{lll} (1) & R \rightarrow T: & M_{req},\, CRC(N_{\mathcal{T}_i}(j) \oplus RND1'), RND1' \\ (2) & T \rightarrow R: & RND2',\, X',\, Y' \\ & & \text{where } X' = X \oplus \Delta,\, Y' = Y \\ & & \text{and } \Delta = RND2' \oplus RND2 \end{array}$$

Upon receiving X', Y' and $RND2'$, the reader uses $N_{\mathcal{T}_i}(j)$ to compute its local Y' value and compare it with the received value:

$$\begin{aligned} Y' &= CRC(RND_2' \oplus N_{\mathcal{T}_i}(j) \oplus X') \\ &= CRC(RND_2' \oplus N_{\mathcal{T}_i}(j) \oplus X \oplus \Delta) \\ &= CRC(RND_2' \oplus N_{\mathcal{T}_i}(j) \oplus X \oplus RND_2' \oplus RND_2) \\ &= CRC(RND_2 \oplus N_{\mathcal{T}_i}(j) \oplus X) = Y \end{aligned} \tag{23}$$

Therefore, the reader accepts the RND_2', X', Y' as a valid message and authenticates the adversary. Finally, the reader obtains the static identifier of the impersonated tag:

$$\begin{aligned} EPC'_{\mathcal{T}_i} &= K_{\mathcal{T}_i}(j) \oplus RND_2' \oplus X' \\ &= K_{\mathcal{T}_i}(j) \oplus RND_2' \oplus X \oplus RND_2 \oplus RND_2' \\ &= K_{\mathcal{T}_i}(j) \oplus RND_2 \oplus X = EPC_{\mathcal{T}_i} \end{aligned} \tag{24}$$

□

5.4. Denial of Service

The attack described in the last section can be generalized to perform a Denial of Service (DoS) attack. This exploits the linearity of the CRC function and its lack of resistance against active attacks by the exclusive-OR operation [3].

First, we briefly explain the weaknesses of the exclusive-OR operation. Suppose that a tag and a reader share an unique identifier ($EPC_{\mathcal{T}_i}$) and a secret key ($K_{\mathcal{T}_i}$). Fresh random numbers are denoted as RND. Let $\mathcal{P}$ be the following simple authentication protocol:

$$\begin{array}{ll} R \rightarrow T: & m_1 || m_2 \\ & m_1 = EPC_{\mathcal{T}_i} \oplus RND \\ & m_2 = MAC_{K_{\mathcal{T}_i}} \oplus (RND) = K_{\mathcal{T}_i} \oplus RND \end{array}$$

where MAC symbolizes a message authenticate code.

Upon receiving message m_1 and its MAC m_2, the reader performs a bitwise XOR between m_1 and tag's EPC to extract RND'. Then, the result of computing a bitwise XOR between the obtained RND' value and his version of the secret key is compared with m_2. If this comparison holds authentication process is successful.

Attack: Considering modifying m_1 to $m_1' = m_1 \oplus b$, being b any non-zero bit string. Upon receiving m_1, the extracted RND' is $RND \oplus b$. The attacker will have to modify m_2 to $m_2' = m_2 \oplus b = K_{\mathcal{T}_i} \oplus (RND \oplus b)$ in order to pass unnoticed. Therefore, all the adversary has to do is to disturb both messages m_1 and m_2 by the same non-zero string.

Theorem 5. *In the CD-EPC protocol, after the eavesdropping of one authentic session, an adversary is able to respond to reader's queries correctly and lead the reader to an incorrect EPC value ($EPC''_{\mathcal{T}_i} = EPC_{\mathcal{T}_i} \oplus \delta$) by sending $X' = X \oplus RND_3$, $Y' = Y \oplus CRC(RND_3) \oplus CRC(\Delta)$ message, where $\Delta = RND2' \oplus RND2$ and $\delta = \Delta \oplus RND_3$.*

Proof

Step 1 The attacker eavesdrops one authentication session and blocks or alters tag's answer.

$$\begin{array}{lll} (1) & R \rightarrow T: & M_{req},\, CRC(N'_{\mathcal{T}_i}(j) \oplus RND_1), RND_1 \\ (2) & T \rightarrow R: (blocked) & RND_2, X, Y \end{array}$$

Step 2 The attacker can supplant the tag by sending the following message:

$$\begin{array}{lll} (2') & A \rightarrow R: & RND'_2,\, X',\, Y' \\ & & \text{where } X' = X \oplus RND_3, \\ & & Y' = Y \oplus CRC(RND_3) \oplus CRC(\Delta), \\ & & \text{and } \Delta = RND2' \oplus RND2 \end{array}$$

Upon receiving X', Y' and $RND2'$, the reader uses $N_{\mathcal{T}_i}(j)$ to compute its local Y'' value and compare with the received value:

$$\begin{aligned} Y'' &= CRC(RND_2' \oplus N_{\mathcal{T}_i}(j) \oplus X') \\ &= CRC(RND_2' \oplus N_{\mathcal{T}_i}(j) \oplus X \oplus RND_3) \end{aligned} \quad (25)$$

As $b \oplus b = 0$,

$$\begin{aligned} Y'' &= CRC(RND_2' \oplus N_{\mathcal{T}_i}(j) \oplus X \oplus RND_3 \oplus RND_2 \oplus RND_2) \\ &= CRC(\Delta \oplus N_{\mathcal{T}_i}(j) \oplus X \oplus RND_2 \oplus RND_3) \end{aligned} \quad (26)$$

Applying Equation 10,

$$\begin{aligned} Y'' &= CRC(\Delta) \oplus CRC(N_{\mathcal{T}_i}(j) \oplus X \oplus RND_2) \oplus CRC(RND_3) \\ &= CRC(\Delta) \oplus Y \oplus CRC(RND_3) = Y' \end{aligned} \quad (27)$$

The tag is authenticated and the reader obtains the static identifier by computing the following equation:

$$\begin{aligned} EPC''_{\mathcal{T}_i} &= X' \oplus K_{\mathcal{T}_i}(j) \oplus RND_2' \\ &= X \oplus RND_3 \oplus K_{\mathcal{T}_i}(j) \oplus RND_2' \end{aligned} \quad (28)$$

Applying Equation 2,

$$\begin{aligned} EPC''_{\mathcal{T}_i} &= K_{\mathcal{T}_i}(j) \oplus EPC_{\mathcal{T}_i} \oplus RND_2 \oplus RND_3 \oplus K_{\mathcal{T}_i}(j) \oplus RND_2' \\ &= EPC_{\mathcal{T}_i} \oplus RND_3 \oplus \Delta \end{aligned} \quad (29)$$

□

An incorrect $EPC_{\mathcal{T}_i}$ is obtained, but the reader is not able to detect the trick. The reader will associate an incorrect identifier ($EPC''_{\mathcal{T}_i} = EPC_{\mathcal{T}_i} \oplus \delta$, where $\delta = \Delta \oplus RND_3$) with the tuple $(N_{\mathcal{T}_i}(j), K_{\mathcal{T}_i}(j))$. So, the proposed protocol is vulnerable to a DoS by means of a man-in-the middle attack.

5.5. Standard Compatibility

Authors claim that the designed protocol is compliant with the EPC-C1G2 specification. However, important technical aspects, necessary to its successful implementation, were ignored. In the following, we briefly summarize the most important properties of tags compliant with this standard:

- Tags are passive, so they receive all their operating energy from readers RF waveform.

- Tags operate on the UHF band (860-960 MHz). Generally, their effectiveness will be poor around metals and water. Their read range is up to 9 m.
- The very constrained resources and storage capabilities dictates that EPC-C1G2 tags can not afford traditional cryptographic primitives.
- Tags include on chip a 16-bit PRNG and a 16-bit CRC checksum.
- Tags have two 32-bit PINs:
  - Kill PIN: The kill password is a 32-bit value stored in reserved memory. A reader shall use a tag's kill password once, to kill the tag and render it silent there after.
  - Access PIN: The access password is a 32-bit value stored in reserved memory. Tags with a nonzero access password shall require a reader to issue this password before transitioning to the secure state, which will allow to read or write in the password protected fields.

The CD-EPC protocol is described by the following three equations:

$$(1) \qquad CRC(N_{\mathcal{T}_i}(j) \oplus RND_1) \tag{30}$$

$$(2) \qquad X = K_{\mathcal{T}_i}(j) \oplus EPC_{\mathcal{T}_i} \oplus RND_2 \tag{31}$$

$$(3) \qquad Y = CRC(RND_2 \oplus N_{\mathcal{T}_i}(j) \oplus X) \tag{32}$$

Equation 30 sets $K_{\mathcal{T}_i}(j)$, $EPC_{\mathcal{T}_i}$ and RND_2 to the same l-bit length. As X is a component of Equation 29, RND_2 and $N_{\mathcal{T}_i}(j)$ are l-bit length just as the above mentioned variables. Finally, as $N_{\mathcal{T}_i}(j)$ is l-bit length, RND_1 is forced to the same length by Equation 28. Summarizing, all the variables must have the same length to run the CD-EPC protocol. As EPC unique identifier must have a length of 96 or 198 bits for compatibility with all encoding schemes (i.e. GID, SGTIN, SSCC) defined by EPCGlobal [10], the value l would have to be fixed to one of these two values. However, the lengths of RND_i, $K_{\mathcal{T}_i}(j)$ and $N_{\mathcal{T}_i}(j)$ are not conforming with the EPC-C1G2 standard. Although not explicitly mentioned by the authors, we can assume that $N_{\mathcal{T}_i}(j)$ and $K_{\mathcal{T}_i}(j)$ are equivalent to the access and kill password as proposed in the standard. A comparison between the length of the variables defined in the standard and the variables used in the proposed protocol is shown in the following table:

Variable	**EPC-C1G2 standard**	**CD-EPC scheme**
$EPC_{\mathcal{T}_i}$	64 or 96 bits	96 or 198 bits
RND_i	16 bits	96 or 198 bits
$N_{\mathcal{T}_i}(j)$	32 bits	96 or 198 bits
$K_{\mathcal{T}_i}(j)$	32 bits	96 or 198 bits

So, CD-EPC protocols triples or sextuples the memory demands ($password\text{-}lenghth = 32 * m$, where $m = 3$ or 6) in comparison with the standard. Additionally, the 16-bit PRNG has to be invoked 12 or 6 times each time a new nonce is necessary, which means a significant reduction in the number of answers/sec that those tags can provide.

6. Conclusions

EPC-C1G2 is one of the most relevant RFID standards. In fact, it seems to become de facto standard for low-cost RFID tags. Due to its low security, some authors have proposed enhanced schemes, but EPC compliant. However, all these schemes have proven to be as insecure as the standard [8, 14, 19, 20]. In 2009, a new mutual authentication protocol was proposed by Chen and Deng that claimed to offer better security margins. In this paper, the security of this scheme is scrutinized and we show how an 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 use of the CRC and take advantage of its linearity. Additionally, the DoS attack is possible due to the lack of resistance against active attacks of the exclusive-OR operation. Finally, we additionally show how the protocol has non-trivial implementation difficulties because of the length of the involved variables, which is not compatible with that of the standard.

While the design of a secure EPC-C1G2 compliant protocol is a thought-provoking challenge, the use of CRC functions should be confined to detect transmission errors. CRCs are linear functions and can not be use as a one-way function. For security purposes, a cryptographic function such as a lightweight hash-function (i.e. PHF [16], Tav-128 [17]), or some kind of MAC (i.e. Squash [23]) should be used instead.

References

[1] Information technology – Radio frequency identification for item management – Part 6: Parameters for air interface communications at 860 MHz to 960 MHz. http://www.iso.org, 2005.

[2] ISO - International Organization for Standardization . http://www.iso.org/, 2009.

[3] B. Alomair and R. Poovendran. On the authentication of RFID systems with bitwise operations. In *Proc. of the second IFIP conference on New Technologies, Mobility and Security - NTMS'08*, pages 1–6, 2008.

[4] Anarchriz. CRC and how to reverse it. http://www.woodmann.com/fravia/crctut1.htm, 1999.

[5] D. Bailey and A. Juels. Shoehorning security into the EPC standard. In *International Conference on Security in Communication Networks – SCN'06*, volume 4116 of *LNCS*, pages 303–320. Springer-Verlag, September 2006.

[6] M. Burmester, B. de Medeiros, J. Munilla, and A. Peinado. Secure EPC Gen2 compliant radio frequency. Cryptology ePrint Archive, Report 2009/149, 2009. `http://eprint.iacr.org/`.

[7] C.-L. Chen and Y.-Y. Deng. Conformation of EPC Class 1 Generation 2 standards RFID system with mutual authentication and privacy protection. *Engineering Applications of Artificial Intelligence*, DOI: 10.1016/j.engappai.2008.10.022, 2009.

[8] H.Y Chien and C.H Chen. Mutual authentication protocol for RFID conforming to EPC class-1 generation-2 standards. *Computer Standards and Interfaces, Elseiver Science Publishers*, 29(2):254–259, 2007.

[9] Class-1 generation 2 UHF air interface protocol standard version 1.2.0: "Gen 2". http://www.epcglobalinc.org/standards/, 2008.

[10] EPC Tag data standard version 1.4. http://www.epcglobalinc.org/standards/, 2008.

[11] GS1 - EPCglobal. http://www.epcglobalinc.org/, 2009.

[12] S. Karthikeyan and M. Nesterenko. RFID security without extensive cryptography. In *Proc. of SASN'05*, 2005.

[13] D.M. Konidala and K. Kim. RFID tag-reader mutual authentication scheme utilizing tag's access password. Auto-ID Labs White Paper WP-HARDWARE-033, Jan 2007.

[14] T.L. Lim and T. Li. Addressing the weakness in a lightweight RFID tag-reader mutual authentication scheme. In *Proc. of the IEEE Int'l Global Telecommunications Conference - GLOBECOM'07*, pages 59–63. IEEE Computer Society Press, 2007.

[15] D. Nguyen Duc, J. Park, H. Lee, and Kwangjo K. Enhancing security of epcglobal gen-2 RFID tag against traceability and cloning. In *Proc. of Symposium on Cryptography and Information Security*, 2006.

[16] K. Nohl and D. Evans. Feasible privacy for lightweight RFID systems. http://www.cs.virginia.edu/evans/talks/spar07, 2007.

[17] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda. An efficient authentication protocol in RFID systems resistant to active attacks. In *Proc. of SecUbiq'06*, volume 4809 of *LNCS*, pages 781–794. Springer-Verlag, 2007.

[18] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador, and A. Ribagorda. *The Internet of Things: From RFID to the Next-Generation Pervasive Networked Systems*, chapter RFID Specification Revisited, pages 311–346. Auerbach Publications, Taylor & Francis Group, 2008.

[19] P. Peris-Lopez, J.C. Hernandez-Castro, J. Estevez-Tapiador, and A. Ribagorda. Cryptanalysis of a novel authentication protocol conforming to EPC-C1G2 standard. Hand. of RFIDSec'07, 2007.

[20] P. Peris-Lopez, T. Li, T. L. Lim, J. C. Hernandez-Castro, and J. M. Estevez-Tapiador. Vulnerability analysis of a mutual authentication scheme under the EPC Class-1 Generation-2 standard. In *Hand. of Conference on RFID Security*, 2008.

[21] R. Phan. Cryptanalysis of a new ultralightweight RFID authentication protocol - SASI. *Dependable and Secure Computing, IEEE Transactions on*, DOI: 10.1109/TDSC.2008.33, 2008.

[22] D. C. Ranasinghe. *Networked RFID Systems and Lightweight Cryptography*, chapter Lightweight Cryptography for Low Cost RFID, pages 311–346. Springer, 2007.

[23] A. Shamir. SQUASH - a new MAC with provable security properties for highly constrained devices such as rfid tags. In *Proc. of FSE'08*, volume In Press of *LNCS*. Springer-Verlag, 2008.

[24] M. Stigge, H. Pltz, W. Mller, and J.-P. Redlich. Reversing CRC theory and practice. Technical Report SAR-PR-2006-05, Humboldt-Universitat Berlin, 2006.

[25] B. Westerbaan. Reversing CRC. http://blog.w-nz.com/archives/2005/07/15/reversing-crc/, 2005.