A Simple Secure Communication Protocol for RFID Devices

Dang Nguyen Duc*, Jaemin Park*, Hyunrok Lee*, and Kwangjo Kim*

*CAIS Lab, Information and Communications University(ICU),

International Research center for Information Security(IRIS)

Daejeon, Korea

Abstract

In this paper, we present a synchronization-based communication protocol for RFID devices. We focus on the EPCGlobal Class-1 Gen-2 RFID tag which supports only simple cryptographic primitives like Pseudo-random Number Generator (PRNG) and Cyclic Redundancy Code (CRC). Our protocol is secure in the sense that it prevents the cloned tags and malicious readers from impersonating and abusing legitimate tags, respectively. In addition, in our protocol, each RFID tag emits a different bit string (pseudonym) when receiving each and every reader's query. Therefore, it makes tracking activities and personal preferences of tag's owner impractical to provide the user's privacy.

I. Introduction

Radio Frequency Identification (RFID) technology is envisioned as a replacement for Barcode counterpart and it is expected to be massively deployed in the near future. The advantage of RFID system over Barcode system includes many-to-many communication (*i.e.*, one tag can be read by many readers and one reader can read many tags at once), wireless data transmission (versus optical communication, thus requiring light-of-sight, in case of Barcode) and its computing nature. Those major benefits enable much more wider range of applications including: supply chain management, library management, anti-counterfeiting banknotes, smart home appliances, *etc.*

Despite of many prospective applications, RFID technology also poses several security and privacy threats which could harm its global proliferation. Ironically, the security weakness of RFID technology comes from the most basic operation of a RFID tag, that is to wirelessly release a unique and static bit string (usually known as Electronic Product Code - EPC) identifying the object associated with the tag upon receiving the query request from readers. Using those unique EPCs as reference, one (equipped with a compatible reader) can track the moving history, the personal preferences and the belongings of a tag's holder. Even worse, absence of authentication results in revealing EPCs to malicious readers (referred to as skimming attack). Once capturing EPCs, an attacker can duplicate

genuine tags and use cloned tags for various anomalous purposes. A natural solution to the security vulnerability stated before is to use cryptography in RFID system. Unfortunately, the cost of manufacturing a tag has to be extremely low, *e.g.*, less than 30 cents (according to RFID journal [12], the one RFID tag is expected to cost 5 cents by 2007). Therefore, the costful security protocols known in the cryptographic literature cannot be incorporated into a small chip with tightly constrained computational power.

Lots of researchers have proposed several *lightweight* cryptographic protocols to defend against security and privacy threats. Most of proposed solutions make use of the hash function [7, 8, 9, 10]. Even though the hash function can be efficiently implemented in low-power hardware, it is still beyond current capability of low-cost RFID tag. In particular, current EPCGlobal Class-1 Gen-2 RFID specification does not ratify cryptographic hash function like MD5 and SHA-1. Thus, we need to look for another solution which should use only the available functionalities of current RFID standards. In fact, Juels suggested such a scheme to prevent the cloned tags from impersonating legitimate tags [3]. However, his protocol did not take eavesdropping and privacy issues into consideration, therefore provides no protection against privacy invasion and secret information leakage. In this paper, we present another scheme targeting most of security features for a RFID system including authentication, traffic encryption, privacy protection as well. Our proposed scheme employs only PRNG and pre-shared secrets between tag and reader/backend server (*e.g.*, PIN, seed to PRNG). We call our scheme *synchronization-based* as ours requires session-key synchronization between tag and reader/backend server like Ohkubo *et. al.* hash-based scheme [7].

The rest of this paper is organized as follows: in Section II, we briefly review some background knowledge and related works. In Secion III, we then describe our proposed protocol followed by heuristic security analysis in Section IV. Finally, we conclude with the final remarks and future work.

II. Background and Related Works

1. RFID System

An RFID system consists of three components: RFID tag, RFID reader and backend server. A RFID tag is a small chip attached to an object. It emits an unique bit string serving as the object identity. A RFID reader can be a PDA, a mobile phone or any kind of devices capable of querying object identity stored in a RFID tag. Using object identity as a pointer, a RFID reader can later retrieve detail information about the object stored in backend server's database.

2. EPCglobal Class-1 Gen-2 RFID Specification

EPCglobal Inc is a joint venture between EAN International from Europe and Uniform Council Code Inc. from USA to standardize RFID technology [1]. The latest RFID standard ratified by EPCglobal is named EPCglobal Class-1 Gen-2 RFID specification (Gen-2 RFID for short). We briefly summarise properties of Gen-2 RFID tag as follows [2]:

- Gen-2 RFID tag is passive,

meaning that it receives power supply from readers.

- Gen-2 RFID tag communicates at UHF band (800-960 MHz) and its communication range is from 2 to 10m.
- Gen-2 RFID tag supports on-chip Pseudo-Random Number Generator (PRNG) and Cyclic Redundancy Code (CRC) computation.
- Gen-2 RFID's privacy protection mechanism is to make the tag permanently unusable once it receives the *kill* command with a valid 32-bit *kill* PIN.
- Read/Write to Gen-2 RFID tag's memory is allowed only after it is in *secure* mode (*i.e.*, after receiving *access* command with a valid 32-bit *access* PIN).

In this work, we aim at designing a new communication protocol with security properties for EPCglobal Class-1 Gen-2 RFID tag. However, note that our protocol can be applied to other RFID standards as well.

3. Pseudo-random Number Generator and Checksum Code

PRNG is the most frequently used primitive in cryptography as well as in computer science, electrical engineering, statistics, *etc.* In a common setting, a PRNG is modeled as a deterministic function whose next output is computed from previous outputs (usually the last output). The output sequence starts from a (randomly chosen) seed number. The security strength of a PRNG depends on how long the period of the output sequence is. A popular class of PRNG has the congruential form of $x_i = ax_{i-1} + b \bmod N$ where x_0 is the seed number and a, b and N are PRNG's parameters [11]. In this paper, we will use a PRNG to share a new session key between RFID tag and reader for each and every session.

A checksum code is often used to check the integrity of data being sent or received. The popular cryptographic checksum codes are cryptographic hash function, MAC and HMAC. In this paper, we will make use of a well-known, efficient (yet less cryptographically strong) checksum algorithm, namely CRC. This kind of checksum code is currently ratified in EPCglobal Class-1 Gen-2 RFID specification, version 1.09 [2]. Of course, we can always use cryptographically secure checksum algorithms if possible to offer the stronger security than the current standard.

III. Our Proposed Protocol

Main idea. We first think of protecting data transmitted between the tag and reader against eavesdropping. The obvious way is to utilize encryption/decryption and the most simple encryption function that we know of is XOR (which is used in one-time pad and stream cipher). The problem now turns to key management issue: that is to ensure that a new encryption key is used in every session. Solving this issue turns out to be a solution to privacy protection as well since RFID tag can XOR EPC with different key in every session, thus, prevent malicious readers from tracking the tag. And we suggest that the simplest, yet most efficient way of key sharing in this scenario is to use the same PRNG with the same seed at both RFID tag side and reader/backend server side. The session key can be

computed by generating a new pseudo-random number from current session key after every session. This computation is required to be done at both RFID tag and reader/backend server in a synchronous way. Otherwise, subsequent traffic cannot be understood by both sides.

The next security problem that we need to solve is authentication. We argue that, in most cases, a reader just needs to know EPC stored in a tag and then eventually contact the backend server to get/update information about the object carrying the tag. Keeping this in mind, we propose that reader-to-tag authentication can be delegated to tag-to-backend server authentication. More specifically, reader can only receive EPC from RFID tag in an encrypted form. It needs to authenticate itself to backend server first, and then, depending on its privileges, backend server can decide what kind of information to send back to reader (for example, in case of a public reader, only information describing what the referenced object is; and in case of a manufacturer's reader, actual EPC and PIN associated with that tag can be sent). Actual reader-to-tag authentication needs to be carried on when reader wants to access (read/write) other sections of tag's memory bank. To do so, we can use PIN-based approach just like in the original Gen-2 RFID specification.

We also would like to note that, there exists a scheme that allows a reader to be able to decipher EPC without help from backend server for several sessions [6]. We have a different view in this regard. We believe that, in a ubiquitous environment, connectivity is abundant and exercising practical security and simplicity are the key to the successful adoption of new technology. In addition, we think that backend server's database can be partitioned in a hierarchical way, thereby reducing overhead at each backend server. This scenario naturally fits in both DNS-like hierarchical structure of EPCglobal Object Naming System (ONS) and real-life situations (for example, each department in a company manages its own inventories, thus, should have its own backend server). We want to stress that our proposed scheme is simple and provides reasonable security strength within the limit of the low-cost RFID tag's functionalities.

Notations. Before describing our protocol in detail, we give the definition of notations that we use in the description of our protocol.

- T: RFID tag.
- R: RFID reader..
- S: Backend server.
- EPC: Electronic Product Code stored in RFID tag and backend server's database.
- $f(.)$ is a PRNG.
- CRC(.) is CRC function.
- K_i : session key in the i-th session.
- r: pseudo-random number.
- PIN: long-term secret shared between tag and reader/backend server (e.g., access PIN in Gen-2 RFID).

Proposed Protocol. During manufacturing time, manufacturer setup a tag by assigning EPC and other parameters. Then, it chooses a random seed number *seed* and store $K_1 = f(seed)$ to tag's memory and backend server's database entry corresponding to matching EPC. A random PIN is also stored in both tag's memory and backend server

database in a similar way. We also assume that the backend server is highly trustful.

The tag query protocol is as follows:

- R → T: Query Request.
- T → R: Generate a random number r and send M_1 = EPC ⊕ CRC(EPC||r||K_i) ⊕ K_i, CRC(M_1||EPC||K_i).
- R ↔ S: R and S mutually authenticated and then R forwards M_1, r to S.
- S: search its database for entry matching with M_1 and CRC(M_1||EPC||K_i). If found, depending on R's privileges, S sends back appropriate information with indication of success and then updates the session key for the next session by performing $K_{i+1} = f(K_i)$. Otherwise, it notifies R to reject the tag.
- R → T: in case of success, R informs T to update the session key.
- T: $K_{i+1} = f(K_i)$

The above protocol can easily allow R to operate in batch mode, that R collects multiple M's from various tags and send all to S at once.

If R desires to perform read/write operations to T's memory, it needs T's EPC from S, and send M_2 = EPC ⊕ CRC(EPC||PIN||r) ⊕ PIN to T. T receives M_2 and computes its own version of M_2 based on its knowledge (of PIN, r and EPC). If this two are not matched, T rejects R's request and accepts it otherwise.

IV. Security and Complexity Analysis

In this section, we give a heuristic security analysis of our proposed scheme. We claim that, our scheme achieve the following security requirements:

- Tag-to-Reader authentication: in order to clone a tag, an attacker needs to capture both EPC and session key stored on the tag. Since session key is never sent out to other parties including the reader itself, an attacker needs physical access to the tag which is considered to be brute-force attack in RFID environment.
- Reader-to-Tag authentication: this type of authentication is delegated to Reader-to-Server authentication where we can make use of advanced authentication protocols in the cryptographic literature. Furthermore, a valid PIN is required if reader wants to access tag's memory (note that, PIN is sent from reader to tag in scramble form).
- Traffic encryption: by using XOR encryption function with session key, eavesdropping threat is eliminated.
- Privacy protection: tag never directly emits EPC in a plaintext form. Each and every session, tag sends out a different bit string because of new session key. Therefore, it is infeasible for malicious parties to use a compatible reader to track tag's owner's activities, movement, belongings and preferences.

Our scheme also exhibits efficient computational complexity. We compare complexity and security features of our protocol with Juels' protocol in [3] which also targets Gen-2 RFID specification.

[Table 1] *Security and Complexity Comparison.*

	Ari Juels' protocol [3]	Our protocol
Backend Server's Complexity	O(qN)	O(N)O(CRC)
Tag's Complexity	O(q)	2CRC+2PRNG
Reader's Complexity	O(q)	2CRC+2PRNG
Tag Authentication	O	O
Reader Authentication	O	O
Privacy Protection	X	O
Traffic Encryption	X	O

Note: N - number of tags in tag population; O(CRC) - computational complexity of CRC algorithm; q - number of PIN-test round in Juels' protocol resulting in $1/2^q$ security margin; Complexity of authentication protocol between Tag and Server is not counted.

V.Conclusion and Future Work

We have presented a simple communication protocol for RFID devices, especially EPCglobal Class-1 Gen-2 RFID devices. Our protocol achieves all desirable security features of a RFID system including: implicit reader-to-tag authentication, explicit tag-to-reader authentication, traffic encryption and privacy protection (against tracking). Our scheme makes use of only PRRN and CRC which are all ratified in current Gen-2 RFID specification (version 1.09).

For future work, we are now studying on more rigorous security analysis of our protocol. Computational complexity of our scheme at backend server side also needs improvement. In addition, the ownership transfer of tag is not currently considered in out protocol.

References

[1] EPCglobal Inc - http://www.epcglobalinc.org/index.html.

[2] EPCglobal Class-1 Gen-2 RFID Specification - Available at http://www.epcglobalinc.org/standards_technology/EPCglobalClass-1Generation-2UHFRFIDProtocolV109.pdf.

[3] Ari Juels, "Strengthening EPC Tag against Cloning", to appear in WiSe'05.

[4] Ari Juels, "RFID Security and Privacy: A Research Survey", to appear in IEEE JSAC 2006.

[5] Stephen Weis, "Security and Privacy in Radio Frequency Identification Devices", Master thesis, available at http://theory.lcs.mit.edu/~sweis/masters.pdf..

[6] D. Molnar, A. Soppera and D. Wagner "A Scalable, Delegatable Pseudonym Protocol Enabling Ownership Transfer of a RFID Tag", In the Proceedings of Selected Areas in Cryptography (SAC)'05, LNCS. Springer-Verlag, to appear.

[7] Miyako Ohkubo, Koutarou Suzuki, and Shingo Kinoshita, "Efficient Hash-Chain Based RFID Privacy Protection Scheme, In the Proceedings of International Conference on Ubiquitous Computing, Workshop Privacy, September 2004.

[8] Gildas Avoine, Etienne Dysli, and Philippe Oechslin, "Reducing Time Complexity in RFID System", In the Proceedings of Selected Areas in Cryptography (SAC)'05, LNCS. Springer-Verlag, to appear.

[9] Jeongkyu Yang, "Security and Privacy on Authentication Protocol for Low-cost Radio Frequency Identification", Master thesis, available at http://caislab.icu.ac.kr/Paper/thesis_files/2005/thesis_jkyang.pdf.

[10] Gildas Avoine and Philippe Oechslin, "A Scalable and Provably Secure Hash-Based RFID Protocol", In the Proceedings of Workshop on Pervasive Computing and Communications Security - PerSec, March 2005.

[11] NIST, "Random Number Generation and Testing", Available at http://csrc.nist.gov/rng/.

[12] RFID Journal, Available online at http://www.rfidjournal.com/.