Anonymous Mutual Authentication Protocol for RFID Tag without Back-end Database

Song Han, Tharam S. Dillon, Elizabeth Chang
DEBI Institute
Curtin Business School
Curtin University of Technology
GPO Box U1987
PERTH, WA 6845 Australia
{s.han, t.dillon, e.chang}@curtin.edu.au

Abstract. RFID, as an emerging technology, has very huge potential in today's social and business developments. Security and Privacy are one of the important issues in the design of practical RFID protocols. In this paper, we focus on RFID authentication protocol. RFID mutual authentication is used to ensure that only an authorized RFID reader can access to the data of RFID tag while the RFID tag is confirmed that it releases data to the authenticated RFID reader. This paper will propose an anonymous mutual authentication protocol for RFID tag and reader. RFID tag is anonymous to RFID reader so that privacy can be preserved. In addition, mutual authentication does not need to rely on a back-end database.

1 Introduction

Radio-Frequency Identification (RFID) is emerging technology in automatic identification and tracking systems [1, 16]. In a classical RFID system, it consists of RFID tags, readers and back-end database server. An RFID tag is connected to an antenna by its integrated circuits. The antenna can receive radio frequency signals for passive tag or receive/emit radio frequency signals for active tag. An RFID reader can make queries to a tag and gain access to related information stored in it. The information can range from static identification numbers to user written data to sensory data. An RFID back-end database server provides tags and readers with secure identification via consistent connection between RFID readers and the back-end database serve. Because of the automatic identification property of RFID systems, RFID have numerous potential applications in automatic identification and tracking purposes [1, 2, 3], such as in supply chain management benefiting industries by increasing the visibility and accuracy of the shipment data. RFID can reduce overhead and errors associated with moving items through the manufacturing steps in industrial automation. RFDI can also help to infer people's current behavior and their actions as an implicit input for computer systems in hospital systems or anti-terrorism system.

Different from the older bar-code technology [15], RFID tags have a number of important advantages:

- The small size can allow them to be implanted within objects;

- Identification by frequency allows objects to be read in large numbers without the need for a visual contact.
- RFID identifiers are long enough so that every object has a *unique* code. Such universal uniqueness means that a product may be tracked as it moves from location to location, finally ending up in the consumer's hands. This may help companies combat theft or improve management of stock and inventories in shops or warehouses [14].
- The introduction of RFID tags in all objects could also directly benefit the consumer: waiting times at checkout lines may be drastically reduced by the use of reader technology hat requires no bar-code scanning.

Thanks to these advantages of RFID technique, RFID tags have been used in transport systems, passports, automotive, animal tracking, Human implants, RFID in library, and so on [1, 4, 6, 14, 16, 17, 18, 19]. The following will introduce the components of an RFID system and the related security and privacy issues.

1.1 RFID System

General back-end database based RFID systems are comprised of four components:

- The RFID tag, or transponder, which is located on the object to be identified and is the data carrier in the RFID system;
- The RFID reader, or transceiver, which may be able to both read data from and write data to a transponder;
- The back-end database server, which helps with registrations and authentication of RFID tags; and
- The data processing subsystem which utilizes the data obtained from the transceiver in some useful manner.

Classification of RFID Tags: RFID tags are classified into three different types by their power resource availability and purposes:

- Passive RFID tag: The passive tag has no power source or battery within itself. The tag uses the energy of the radio frequency signals received from the reader to power its operation. This is the least expensive tag.
- Active RFID tag: The active tag has own power resource to support the entire operations, and can therefore generate radio frequency signals for the corresponding transactions.
- Semi-active tag: The semi-active tag has own power resource within itself. The power resource is used to support internal circuits during communications. This power resource is not strong enough to generate and emit radio frequency signals.

1.2 Security and Privacy of RFID

Security and privacy are becoming more and more important in present business, government, industry and individual transactions or activities, especially in the

presence of developed computer networks. RFID as an emerging technology in automatic identification and tracking systems operated on radio frequency, its security and privacy issues are one of those critical concerns of RFID users, companies, and scientists ranging from researchers to implementers. In fact, RFID may disclose personal privacy in daily activity. Consider a supermarket scenario: a supermarket can wave a RFID reader near people's clothes, handbags, and other personal items and retrieve private data about people and their belongings and shopping behavior. The size of their shirts is no longer their personal secret, nor is the amount of cash they are carrying. Therefore, it is possible to create a complete commercial, and, worse, personal profile with the collected tag data on the person. However, this will break the privacy of customers in the supermarket. Hence, customers will concern the privacy protection issue.

In this paper we will focus on the privacy issue and RFID tag authentication. In fact, the mass deployment and acceptance of RFID technology is nowadays mainly limited by privacy concerns [1, 4, 6, 7]. Products labeled with RFID tags reveal sensitive information when queried by readers, and they do it indiscriminately. This may induce the violation of location privacy, i.e. tracking. Besides the privacy and tracking issues, there are some other worth mentioning: impersonating, spoofing, eavesdropping, traffic analysis, etc..

1.3 Advantages of the Proposed Protocol

In this paper, we will propose a secure authentication protocol for RFID tag and reader without back-end database. The advantages of the new protocol are as follows:

- The authentication process does not involve a back-end database server with consistent connection with the RFID reader. Therefore, the authentication is a serverless authentication.
- The authentication is mutual between RFID reader and tag, i.e. the RFID reader is authenticated to the RFID tag while the latter is also authenticated to the former.
- There is an off-line registration authority that is responsible for preparing the initial stages for RFID tag and reader.
- The unique identity ID of RFID tag is encapsulated from authorized RFID readers. Therefore, the privacy of the RFID tag and its owner is preserved. This point is held from two aspects: the identity privacy is preserved from not only the authorized RFID readers but also any adversary from outside.
- The unique identity of the RFID tag can be revealed in case of dispute. This is supported by the fact that the off-line registration authority and the RFID tag share a pre-established secret. Therefore, the off-line registration authority will release the unique identity of the compromised or misfunctioned RFID tag to the RFID reader or a legal third party.

1.4 Organization of the rest of the Paper

The rest of the paper is organized as follows: In Section 2, some related works based on the back-end database model will be introduced. A previous work that was serverless will also be briefly reviewed. In Section 3, a secure anonymous mutual authentication protocol for RFID tag without back-end database will be proposed. This section is composed of three subsections: the first is notation introduction; the second is the initial stage for the RFID tag, the RFID reader and the off-line registration authority; the third will be the proposed mutual authentication protocol. In Section 4, the security analysis and comparison will be provided. Finally, we will conclude our paper.

2 Related Works

In this section, we will review two kinds of existing RFID authentications. One is the back-end database server model based authentication [4, 5, 6, 9, 10, 11, 13], the other is the non-server authentication [12]. The back-end database server based authentication resorts to a back-end database server to help an authorized RFID reader to authenticate RFID tag and vice versa. The non-server authentication does not resort to any online help besides the RFID reader and the RFID rag. On the other hand, the non-server authentication protocol only needs an off-line registration authority. Therefore, a consistent connection between the RFID reader and a trusted third party (say, the registration authority in the non-server model) is removed.

2.1 Back-end database server model based authentication

The authentication protocol in [4] was based on hash-chain. It only requires a hash function in the tag and data management at the back-end. It offers a high degree of location privacy and is resistant to many forms of attacks. Further, only a single message exchange is required, the communications channel needs not be reliable and the reader/third party need not be trusted, and no long-term secrets need to be stored in tags. However, their solution did not provide full privacy guarantees; i.e. the tag is vulnerable to tracing when the attacker interrupts the authentication protocol mid-way.

The proposed protocol in [10] provides specific time-memory trade-off that supports the scalability. The authors also proved that the system could truly offer privacy and even forward privacy. The authors further provided an extension of the scheme which offers a secure communication channel between RFID tags and their owner using building blocks that are already available on the tag.

A hash-tree based authentication protocol for RFID tags was proposed in [5]. The authors gave a general scheme for building private authentication with work logarithmic in the number of RFDI tags based on a scheme with linear work as a sub-

protocol. The authors of [5] also did not use any pseudo-random functions or other heavy crypto operations in one of their efficient authentication protocols, where simple bit-wise Exclusive OR operation was used to construct the procedures of identification.

The authentication protocol proposed in [11] aims to solve the desynchronization problem by maintaining a previous ID in the database server. This protocol only needs two one-way hash function operations. During the process of identification, the tag emits its identity after authentication is finished. The protocol also supports local privacy by refreshing an identifier of the tag in each session of authentication.

An anonymous RFID protocol was proposed in [9]. This protocol enforces privacy as it prevents information to be read by unauthorized third parties. This is due to the fact that no single, fixed ID is used throughout the tag's life as tag IDs get refreshed periodically. This protocol also offers a high degree of location privacy and is resistant to many forms of attacks.

A new mutual authentication protocol for RFID tags was recently proposed in [8]. The RFID reader and tag carry out the authentication based on their synchronized secret information. The synchronized secret information is monitored by a component of the database server. Their protocol also supports the low-cost non-volatile memory of RFID tags. This is desirable since non-volatile memory is an expensive unit in RFID tags. However, their protocol still needs the back-end database support.

An efficient lightweight mutual authentication protocol was proposed in [13]. That protocol can be implemented in low-cost tags (say, tags with <1K logic gates) where RFID tags are fitted with a small portion of rewritable memory and another read-only memory. The authentication does not resort to the exhaustive search in the back-end database if an RFID reader wants to identify a registered tag.

2.2 Non-database-server based authentication

All the above authentication protocols were based on back-end database server. Therefore, a consistent connection between the RFID reader and the back-end database server needs to be maintained in those protocols. In order to remove the requirement of such consistent and secure connection, Tan et al. proposed a non-server authentication protocol [12]. However, their protocol did not support mutual authentication between RFID tag and reader. In addition, the anonymity of RFID tags was not maintained while anonymity is one of the important properties concerned in ubiquitous computing environment [2].

In the next section, we will propose a new authentication protocol for RFID tag and reader. This protocol does not need to maintain a back-end database server, and thus can remove the secure and consistent connection between the RFID reader and its

back-end database server. The proposed protocol supports mutual authentication for RFID reader and tag. It also maintains the privacy of the RFID tag and its owner. The privacy can be disclosed by an off-line trusted third party, i.e. the registration authority of the RFID system.

3 New Anonymous Mutual Authentication Protocol for RFID tags

In this section, we will present the anonymous mutual authentication protocol for RFID tags. Some notations will be first presented and then used throughout the rest of the paper. The structure of the RFID reader and the off-line registration authority will be then introduced. Following that, the anonymous mutual authentication protocol for RFID tags will be provided.

3.1 Notations Used in Our Protocol

The following table provides the notations used in the proposed mutual authentication protocol.

Table 1. Notations for the non-database-server authentication protocol for RFID tag and reader

Notation	Representation
h()	One-way hash function available to all parties
$\|\|$	Concatenation of bit-strings
T	A valid RFID tag
CA	Off-line registration authority
R	An authorized RFID reader
id_T	Unique identity of T
PRNG	A pseudorandom number generator
β	The bit-length of the output of h()
$\oplus$	Exclusive-or function (XOR)
id_R	Unique identifier of R
s	A secret of T which is shared with CA
$h(id_T)$	Pseudo-identifier of T
L	Authentication list of R
m	A private integer which is known to R and T, where $0 < m < \beta$

3.2 Initial Preparations

The RFID tag T has a one-way hash function h(). T can calculate the hash value for any input to this function. T can also carry out the XOR calculation. In fact, carrying

out the XOR calculation is an affordable capability for various RFID tags, especially for low-cost RFID tags. The tag T also shares a secret key with an off-line registration authority CA. This secret key is assigned to T while it is registered with the CA. The RFID reader needs to register at the registration authority which will assign a list of hash valuation of identities of RFID tags to the RFID reader. That is to say, the reader R is authorized to have rights to access the data of those RFID tags.

The structure of the authentication list of the RFID in the j-th authentication process for the i-th tag T

...	...	...
$h(s,id_R)$	$h(id_T)$	j
...	...	...
...	...	...

Fig. 1. The structure of the authentication list. We use one RFID tag instance T to demonstrate the components of the authentication list of RFID reader. id_T is the unique identifier of T. The secret key of T is s which is assigned by the off-line registration authority. id_R is the unique identifier of the RFID reader.

3.3 The Proposed Protocol

This authentication protocol is working without the timestamps. The details of the mutual authentication protocol are presented as:

1. RFID reader R sends an access Request to RFID tag T.
2. T checks the request and generates a nonce r_1. T then sends r_1 back to R.
3. After receiving r_1, R generates a new nonce r_2 and sends r_2 and its identifier id_R to T.
4. After receiving r_2 and id_R, T uses hash function h(), its secret s, and R's identifier id_R to get $h(h(s, id_R))$, chooses the first m bits of it to get $t_3 = [h(h(s,id_R))]_m$, and then computes $t_1 = \{[h(h(s,id_R))]_m \oplus (r_2)_m\} \| \overbrace{\{1,1,\cdots,1\}}^{\beta-m} \oplus [r_1]_\beta$. Finally, T sends t_1 to R.
5. After receiving t_1, R first uses his own random number r_2 to retrieve $t_2 = [[r_1]_\beta \oplus t_1]_m \oplus [r_2]_m$. R then searches his authentication list L for

finding a $h(x, id_R)$ such that the first m bits of $h(h(x, id_R))$ is identical to t_2. R then computes $f_1 = h(h(s,id_R) \| t_1 \| r_2)$ and sends it to T.

6. After receiving t_1, T sets $f_2 = h(h(s,id_R) \| t_1 \| r_2)$ and compares the received f_1 with f_2. If they are equal, then R is authenticated and T believes R is authorized to access to T. Following that, T first encapsulates its unique identity id_T to get a pseudo-identifier and then encodes the pseudo-identifier to get $f_3 = h(h(s,id_R) \| r_1 \| r_2) \oplus h(id_T)$. T finally forwards f_3 to R.
7. After receiving f_3, R computes $f_4 = h(h(s,id_R) \| r_1 \| r_2))$ and then sets $f_5 = f_3 \oplus f_4$. If the tag is a valid tag which R is authorized to access, then this f_5 is the encapsulated identity of T. To confirm this point, R checks his list L. If f_5 is in L, then T is authenticated and R believes T is a valid RFID tag.

The serverless mutual authentication protocol for RFID tag with encapsulated ID is summarized in the following figure.

Reader (PRNG, h(), L)		Tag (PRNG, h(), s)
	1. Request →	
		2. generates r_1= PRNG
	← r_1	
3. generates r_2 = PRNG		
	id_R, r_2 →	4. sets $t_1 = \{[h(h(s,id_R))]_m \oplus (r_2)_m\} \| \overbrace{\{1,1,\cdots,1\}}^{\beta-m} \oplus [r_1]_\beta$
		← t_1
5. Retrieve $t_2 = [[r_1]_\beta \oplus t_1]_m \oplus [r_2]_m$ Search the authentication list for an entry whose first element's first m-bit is equal to t_2; If so, then computes $f_1 = h(h(s,id_R) \| t_1 \| r_2)$		
	f_1 →	6. Sets $f_2 = h(h(s,id_R) \| t_1 \| r_2)$ Checks whether $f_1 = f_2$ If yes, then R is an authorized reader, and sets $f_3 = h(h(s,id_R) \| r_1 \| r_2) \oplus h(id_T)$