International Journal of Control and Automation
Vol.7, No.2 (2014), pp.131-142
http://dx.doi.org/10.14257/ijca.2014.7.2.13

Secure and Private Protocols for Server-less RFID Systems

He Jialiang[1], Xu Youjun[2] and Xu Zhiqiang[3]

*[*1]College of Information and Communication Engineering, Dalian Nationalities University, China*
[2]College of Computer Science and Information Technology, Daqing Normal University, China
[3]Department of digital media technology, Sichuan College of Media and Communications, China

urchin2012@sina.com; xu_youjun@sohu.com; starsep928@yahoo.com.cn

Abstract

Server-less RFID systems are used more and more widespread recently, which allow RFID readers authenticating a specific tag without the help of on-line backend servers, it brings higher design requirements for RFID security protocols. In this paper, a mutual RFID authentication protocol and its corresponding serach ptotocol for server-less systems are proposed. The security properties of these protocols are analyzed as well by comparing with the related protocols.

Keywords: *Server-less RFID system; Authentication Protocol; Search Protocol*

1. Introduction

Radio Frequency Identification (RFID) is a wireless technology to identify objects automatically and remotely [1], and has been used in various application fields. However, the widespread deployment of RFID systems into consumer products identification and correlation of the tagged objects and the principals who carry them may expose the potential security threats and risks either to corporations or to individuals. So before granting access, tags should authenticate the communicating reader, and the reader should authenticate tags to ensure the authenticity of the collected data [2]. Except RFID authentication protocols, the various applications scenarios requirements call for more security protocols such as RFID search protocols.

Conventional RFID systems are based on the central database model [3], a backend server, readers and tags constitute a typical RFID architecture, and a reader identify and authenticate tags via the help of the online backend server. Portable and mobile readers are used more and more widespread; a user is dispatched to an off-site location to collect information of some objects that labeled with RFID tags. He has a mobile RFID reader, but the communication quality cannot ensure the connection to the backend server. A simple solution is to let the user download all the data of the tags will be indentified into his mobile reader from the central database before he heads for his destination, this is a typical application of Server-less RFID Systems. However, unlike a fixed reader which can be well protected, a portable and mobile reader might be lose or stolen, so the information inside might be used to forge the tags and violate the privacy, so it brings higher design requirements for RFID security protocols.

ISSN: 2005-4297 IJCA

Based on wireless communication, signal broadcasting, and non-symmetry between the forward channel and the backward channel, RFID systems are confronted with many security problems. Due to strictly limited calculation resources, small storage capacity and faint power supply of low-cost tags, it is difficult to apply an ordinary and complicated but safe cryptographic algorithm to a RFID system and these factors are hindering the rapid spread of this technology [4]. Presently, lightweight encryption methods such as Hash, PRNG and CRC are used wildly in design of RFID protocols. Especially, for achieving the balance between security and performance, hash-based methods have been researched and used actively [5]. In this paper, based on a few existing RFID security protocols, a mutual RFID authentication protocol and its corresponding search protocol for server-less systems are proposed.

The rest of this paper is organized as follows: in the second section, the privacy and security requirements for server-less RFID systems are generalized; in the third section, the related work is introduced; in the fourth section, a mutual authentication protocol for server-less RFID systems is proposed; in the fifth section, corresponding search protocol for server-less systems is presented; in the sixth section, security properties of the proposed protocols are analyzed carefully; finally, the conclusion of this paper is generalized.

2. Security and Privacy Requirements for Server-less RFID Systems

Based on the special characters, server-less RFID systems are confronted with more kinds of security threats than traditional fixed RFID systems as follows:

(1)Tag untraceability

If responding message in authentication process or search process from a tag always contains a changeless value, namely the response are linkable to each other or distinguishable from those of other tags, an adversary can recognize and locate the tag by intercepting and analyzing. That is to say, the location privacy of the user that attached by the tag could be traced [6].

(2)Tag information protection

A tag is always attached to a specific object, storing data in an encrypted form helps retain its confidentiality. So through all transmission process of information, an unauthorized user should not acquire the holder’s detailed information.

(3)Reader untraceability

In a server-less RFID system, we should consider the privacy of mobile reader privacy holders. Since users commonly handle mobile readers while RFID-tagged objects are attached to goods or products in RFID search systems. Usually, a message from a reader is more easily eavesdrops than a message from a tag [8]. That is to say, the location privacy of the mobile reader holder could be traced.

(4)Spoofing attack

An adversary may feign a legitimate tag and communicate with a reader instead of the tag and be authenticated as the tag but the genuine legitimate tag may be out simultaneously.

(5)Replay attack

Such an attack in which an adversary repeatedly launches a message that obtained by eavesdropping or intercepting from a regular communication between a reader and a tag during a normal authentication access.

(6)Denial of Service (DoS) attack

An adversary disturbs the communications between a reader and a tag by means of intercepting or blocking messages transmitted, that could cause losing synchronization between the backend server and the tag, so the legitimate tag cannot be authenticated by the backend server again [7].

(7)Privacy of search result

The other privacy requirement to be considered is the search result of a mobile reader. It is undesirable to reveal the search result of a mobile reader. In some circumstances to an adversary, it might be useful information whether a mobile reader holder found a particular tag or not [8]. So a well designed search protocol should protect privacy of search result from an illegal user.

3. Related Work

Many security for server-less systems have been proposed recently [8-15]. Let's review and analyze Tan et al.'s protocol [14] and Ji *et al.*'s protocol [8] in Section 3.1, and show weakness in Section 3.2, the notations used in this paper as follows:

Table 1. The notations used in this paper

Symbol	Meaning
ID	The unique index code of a tag (The length is l)
R_i	The unique index code of a reader (The length is l)
S	Trusted back-end database
H()	An one-way hash function, H: $\{0,1\}^{l*} \rightarrow \{0,1\}^{l}$ (The length of output is l)
F()	An one-way hash function, H: $\{0,1\}^{l*} \rightarrow \{0,1\}^{l}$ (The length of output is l)
PRNG()	The pseudo random number generator (The length of output is l_R, usually $l_R < l$)
$\oplus$	XOR operator
$\|\|$	Concatenation operator
n	The random number generated by the reader (The length is l_R)
T	Temporary value (The length is l)
k_i	The shared secret key between Ti and S
$H()_m$, m	$H()_m$ denotes the left m bits of H(), where m is a pre-set parameter and $m < l$
A→B:M	A sends message M to B

3.1 Review of Tan *et al.*'s RFID security schemes

3.1.1 Tan *et al.*'s server-less RFID authentication protocol

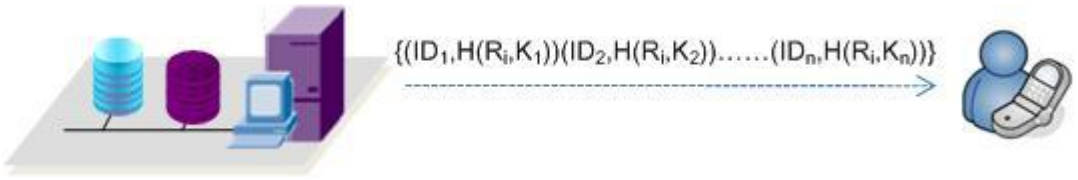

Figure 1. Reader downloads the access list

Before performing the server-less authentication protocol, a reader R_i need identify itself to the backend server and download the access list $\{(ID_1,H(R_i,K_1))(ID_2,H(R_i,K_2))\ldots\ldots(ID_n,H(R_i,K_n))\}$, that is to say, for each tag with its identity ID_j, the reader is given a specific value $H(R_i,K_j)$, then the authentication performing access as follows:

(1)Reader→Tag: request

(2)Tag→Reader: n_j

After receiving the request from the reader， the tag generates random number n_j and responds it to the reader as a challenge.

(3)Reader→Tag: n_i, R_i

The reader generates a random number n_i and then sends R_i with n_i to the tag.

(4)Tag→Reader:$F(H(R_i,k_j))_m$, $F(H(R_i,k_j) \| n_i \| n_j) \oplus ID_j$

The tag uses its secret key kj and the identity IDj to compute $F(H(R_i,k_j))_m$, $F(H(R_i,k_j) \| n_i \| n_j) \oplus ID_j$ and sends them to the reader, where $F(H(R_i,k_j) \| n_i \| n_j)$ is used to hide its identity ID_j.

After receiving the response message from the tag, the reader first uses $F(H(R_i,k_j))_m$ to filter and get the list of candidate tags, and then uses the key of each candidate tag to verify $F(H(R_i,k_j) \| n_i \| n_j) \oplus ID_j$. If there is a match, then the reader identifies and authenticates the tag successfully.

3.1.2 Tan *et al.*'s server-less RFID search protocol

(1)Reader→Tag*: Broadcast $F(H(R_i,k_j) \| n_r) \oplus ID_j$, n_r, R_i

The reader wants to search a specific tag ID_j. Hence, it calculates the encrypted form $F(H(R_i,k_j) \| n_r) \oplus ID_j$ and broadcasts the data in step1 to all the tags in the neighborhood.

(2)Tag*:

After receiving the search request from the reader， the tag near the reader uses its secret key k_{T^*} to calculate the value $F(H(R_i\ k_{T^*}) \| n_r)$ and XOR $F(H(R_i,k_j) \| n_r) \oplus ID_j$; If ID* = ID_j, then it will derive ID_j which it the identity of the current tag and will go to step(3.a); otherwise it fails to derive its identity and will perform step(3.b).

(3.a)Tag→Reader: $F(H(R_i,k_j) \| n_t) \oplus ID_j$, n_t

If ID_j is in the neighborhood and has successfully performed the checking in step2, then it chooses a random number n_t, calculates $F(H(R_i,k_j) \| n_t) \oplus ID_j$ and sends $F(H(R_i,k_j) \| n_t) \oplus ID_j$, n_t back to the reader.

After receiving $F(H(R_i,k_j) \| n_t) \oplus ID_j$, n_t from the tag, the can verify the data $F(H(R_i,k_j) \| n_t) \oplus ID_j$ to authenticate the tag.

(3.b)Tag→Reader: (rand, n_t) with probalility

For each tag ID* which is in the neighborhood but fails to derive its identity in step2, it will choose and respond two random number (rand, n_t) with probability. This arrangement is to confuse attackers from tracing tag ID_j since not only tag ID_j but also tag ID* will respond.

When the reader receives from either step 3.a or 3.b, it verifies which one is a valid response from tag ID_j.

3.1.3 Weakness of Tan *et al.*'s server-less RFID security mechanism

There are some shortcomings of security and performance in Tan *et al.*'s mechanism as follows:

(1)We note that Tan *et al.*'s server-less RFID authentication protocols only provided unilateral authentication—that is, only the tag authenticates itself to the reader but the reader does not authenticate itself to the tag; therefore, tags can't tell whether the reader is genuine or not [8].

(2)In the step1 of Tan *et al.*'s server-less RFID search protocol and in the step3 of Tan *et al.*'s server-less RFID authentication protocol, request message from a reader always contains a changeless value R_i, namely an adversary can recognize and locate the reader by intercepting and analyzing. That is to say, the location privacy of the user that attached by the reader could be traced.

(3)In the setp4 of Tan *et al.*'s server-less RFID authentication protocol, tag T_j will respond the fixed value $F(H(R_i,k_j))_m$ namely an adversary can recognize and locate the tag by intercepting and analyzing, That is to say, the location privacy of the user that attached by the tag could be traced.

(4)Low-cost passive tags have constraint requirements of limited resources, using less hardware cost is an important research object, we can see that using pseudo random number generator in tags leads to extra hardware cost, usually, about 700-800 logic gates is needed to implementing a pseudo random number generator. More seriously, these two protocols use two hash functions each. So it is unpractical for low-cost RFID systems.

3.2 Review of Ji *et al.*'s RFID security schemes

3.2.1 Ji *et al.*'s server-less RFID authentication protocol

Before performing the server-less authentication protocol, a reader R_i need identify itself to the backend server and download the access list $\{(ID_1,H(R_i,K_1))(ID_2,H(R_i,K_2))\ldots\ldots(ID_n,H(R_i,K_n))\}$, that is to say, for each tag with its identity ID_j, the reader is given a specific value $H(R_i,K_j)$, then the authentication performing access as follows:

(1)Reader→Tag: request, Ri,nr

The reader first generates a random number n_r and broadcast R_i,n_r as a request.

(2)Tag→Reader: n_t, $H(H(R_i,k_j))_m$, $H(H(R_i,k_j) \| n_r \| n_t \| ID_j)$

After receiving the request from the reader, the tag generates random number n_t and calculates $H(H(R_i,k_j))_m$, $H(H(R_i,k_j) \| n_r \| n_t \| ID_j)$, then sends n_t, $H(H(R_i,k_j) \| n_r \| n_t \| ID_j)$, $H(H(R_i,k_j))_m$ back to the reader.

(3)Reader→Tag: $H (ID_j \| n_r \| n_t \| H(R_i,k_j))$

After receiving the response message from the tag, in step2, the reader first uses $H(H(R_i,k_j))_m$ to get the list of candidate tags, and then uses the key of each candidate tag to verify $H(H(R_i,k_j) \| n_r \| n_t \| ID_j)$. If there is a match, then the reader identifies and authenticates the tag successfully. Then the reader calculates $H(ID_j \| n_r \| n_t \| H(R_i,k_j))$ and sends it back to the tag.

When the tag ID_j receives $H (ID_j \| n_r \| n_t \| H(R_i,k_j))$, it verifies this value to authenticate the reader.

3.2.2 Ji *et al.*'s server-less RFID search protocol

(1)Reader→Tag*: Broadcast $H(R_i \| k_j \| n_r) \oplus ID_j$, n_r, R_i

The reader wants to search a specific tag ID_j. Hence, it generates a random n_r and calculates $H(R_i \| k_j \| n_r) \oplus ID_j$, then broadcasts $H(R_i \| k_j \| n_r) \oplus ID_j$, n_r, R_i to all the tags in the neighborhood.

(2)Tag*:

After receiving the search request from the reader，the tag near the reader uses its secret key k_{T^*} to calculate the value $H(R_i \| k_{T^*} \| n_r)$, and checks whether $H(R_i \| k_{T^*} \| n_r) \oplus (H(R_i \| k_j \| n_r) \oplus ID_j)$ equals to its identity; If so then it will derive ID_j which it the identity of the current tag and will go to step(3.a); otherwise it will perform step(3.b).

(3.a)Tag→Reader: $H(k_j \| ID_j \| n_r \| n_t \| R_i)$, n_t

If ID_j is in the neighborhood and has successfully performed the checking in step2, then it chooses a random number n_t, calculates $H(k_j \| ID_j \| n_r \| n_t \| R_i)$ and sends n_t, $H(k_j \| ID_j \| n_r \| n_t \| R_i)$ back to the reader.

(3.b)Tag→Reader: (rand, n_t) with probability

For each tag ID* which is in the neighborhood but fails to derive its identity in step2, it will choose and respond two random number (rand, n_t) with probability. This arrangement is to confuse attackers from tracing tag ID_j since not only tag ID_j but also tag ID* will respond.

(4)Reader→T_j or T*: $H(R_i \| k_j \| ID_j \| n_r \| n_t)$ or rand

When the reader receives from either step 3.a or 3.b, it verifies which one is a valid response from tag ID_j. If so, the reader calculates $H(R_i \| k_j \| ID_j \| n_r \| n_t)$ and sends it back to the tag; otherwise, it notes this response invalid but responds rand to confuse the possible tracers.

(5)T_j:

The reader checks whether $H(R_i \| k_j \| ID_j \| n_r \| n_t)$ equals to the value received in step4. If so, then it accepts the reader; otherwise, it terminates this session.

3.2.3 Weakness of Ji *et al.*'s server-less RFID security mechanism

There are some shortcomings of security and performance in Ji *et al.*'s mechanism as follows:

(1)In Ji *et al.*'s server-less RFID search protocol and authentication protocol, request message from a reader always contains a changeless value R_i, namely an adversary can recognize and locate the reader by intercepting and analyzing. That is to say, the location privacy of the user that attached by the reader could be traced.

(2)In the setp3 of Tan *et al.*'s server-less RFID authentication protocol, tag T_j will respond the fixed value $H(H(R_i,k_j))_m$ namely an adversary can recognize and locate the tag by intercepting and analyzing, That is to say, the location privacy of the user that attached by the tag could be traced.

(3)Low-cost passive tags have constraint requirements of limited resources, using less hardware cost is an important research object, we can see that using pseudo random number generator in tags leads to extra hardware cost.

Based on above analysis, we will propose an authentication protocol and its corresponding search protocol for server-less RFID systems as follows.

4. A New RFID Authentication Protocol for Server-less RFID System

4.1 Assumptions

(1)The channel between a reader and a tag is assumed insecure for wireless connection. We assume that an adversary can observe and manipulate communications between insecure channels.

(2)Tags are low-cost passive tags, so the resources of each tag are strictly constrained. In this protocol, each tag only needs to have a one-way hash function H(), XOR operation capability for the reason of hardware cost.

(3)A tag is not vulnerable to compromised with an adversary, that is to say, an adversary cannot acquire the inner information of the tag easily.

(4)The one-way hash function H () is secure enough against brute exhaustive search from an adversary.

4.2 Initialization stage

Before performing the server-less authentication protocol, a reader R_i need identify itself to the backend server and download the access list $\{(ID_1,H(R_i,K_1))(ID_2,H(R_i,K_2))\ldots\ldots (ID_n,H(R_i,K_n))\}$, that is to say, for each tag with its identity ID_j, the reader is given a specific value $H(R_i,K_j)$, the server and tags store information required to perform authentication.

4.3 The authentication access

(1)Reader→Tag: $H(n_r)\oplus R_i$, n_r

The reader first generates a random number n_r and calculates $H(n_r)\oplus R_i$, then broadcasts $H(n_r)\oplus R_i$,n_r as a request to tags.

(2)Tag→Reader: $H(M||n_r)\oplus ID_j$, $H(M||n_r||H(k_j,R_i)||ID_j)$, M

After receiving the request from the reader，the tag first calculates $H(n_r)\oplus(H(n_r)\oplus R_i)$ and gets R_i, then calculates $M = H(T\oplus n_r\oplus ID_j)$, $H(M||n_r)\oplus ID_j$, $H(M||n_r||H(k_j,R_i)||ID_j)$, then sends $H(M||n_r)\oplus ID_j$, $H(M||n_r||H(k_j,R_i)||ID_j)$, M back to the reader, subsequently the tag should updates $T = M\oplus H(M||n_r||k_j||ID_j)$. Especially, we use $M = H(T\oplus n_r\oplus ID_j)$ to substitute pseudo random number of the tag.

(3)Reader→Tag: $H(ID_j \| M \| n_r \| k_j)$

After receiving the response message from the tag, the reader calculates $ID = H(M||n_r)\oplus (H(M||n_r)\oplus ID_j)$, and searches whether there exists certain ID* which equals to ID or not. If there is a match, the reader should verify the tag using received $H(M||n_r||H(k_j,R_i)||ID_j)$, then calculates $H(ID_j \| M \| n_r \| k_j)$ and sends it back to the tag; otherwise, the authentication is failed.

When the tag ID_j receives $H(ID_j \| M \| n_r \| k_j)$, it verifies this value to authenticate the reader, so the mutual authentication access achieves.

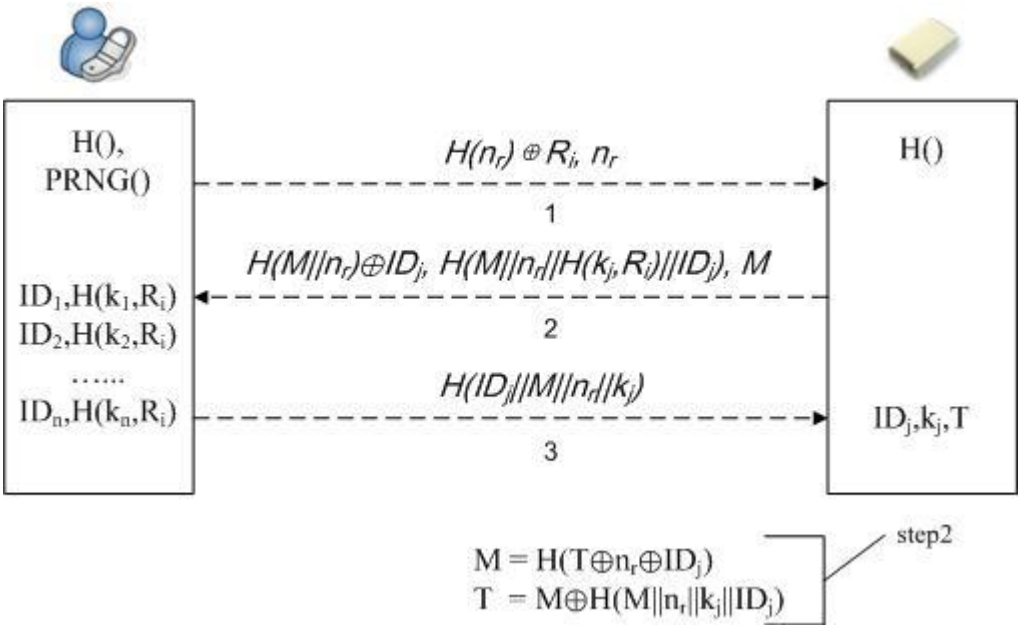

Figure 2. The proposed authentication protocol

5. A new RFID Search Protocol for Server-less RFID System

(1)Reader→Tag*: Broadcast $H(n_r)\oplus R_j$, $H(n_r \| R_i)\oplus ID_j$, n_r

The reader wants to search a specific tag ID_j. Hence, it generates a random value n_r, and calculates $H(n_r)\oplus R_j$, $H(n_r \| R_i)\oplus ID_j$, then broadcasts $H(n_r)\oplus R_j$, $H(n_r \| R_i)\oplus ID_j$, n_r to all the tags in the neighborhood.

Tag*: After receiving the search request from the reader， the tag near the reader calculates $H(n_r)\oplus(H(n_r)\oplus R_i)$ and gets R_i, then calculates ID* by using the value $H(n_r \| R_i)\oplus ID_j$; If ID* = ID_j, then it will derive ID_j which it the identity of the current tag and will go to step(2.a); otherwise it fails to derive its identity and will perform step(2.b).

(2.a)Tag→Reader: $F(H(R_i,k_j) \| n_t)\oplus ID_j$, n_t

If ID_j is in the neighborhood and has successfully performed the checking in step2, then it calculates $M = H(T\oplus n_r\oplus ID_j)$, $H(M||n_r)\oplus ID_j$, $H(M||n_r||H(k_j,R_i)||ID_j)$, and sends $H(M||n_r)\oplus ID_j$, $H(M||n_r||H(k_j,R_i)||ID_j)$, M back to the reader. Subsequently the tag should updates $T = M\oplus H(M||n_r||k_j||ID_j)$.

After receiving $H(M||n_r)\oplus ID_j$, $H(M||n_r||H(k_j,R_i)||ID_j)$, M_t from the tag, the reader calculates $ID = H(M||n_r)\oplus(H(M||n_r)\oplus ID_j)$, and searches whether there exists certain ID* which equals to ID. If there is a match, the reader should verify the tag using received $H(M||n_r||H(k_j,R_i)||ID_j)$.

(2.b)Tag→Reader: V_1, V_2, M with probability

For each tag ID* which is in the neighborhood but fails to derive its identity in step2, it will choose and respond two random number V_1, V_2 with probability, and send V_1, V_2, M back to the reader. This arrangement is to confuse attackers from tracing tag ID_j since not only tag ID_j but also tag ID* will respond.

When the reader receives from either step 2.a or 2.b, it verifies which one is a valid response from tag ID_j.

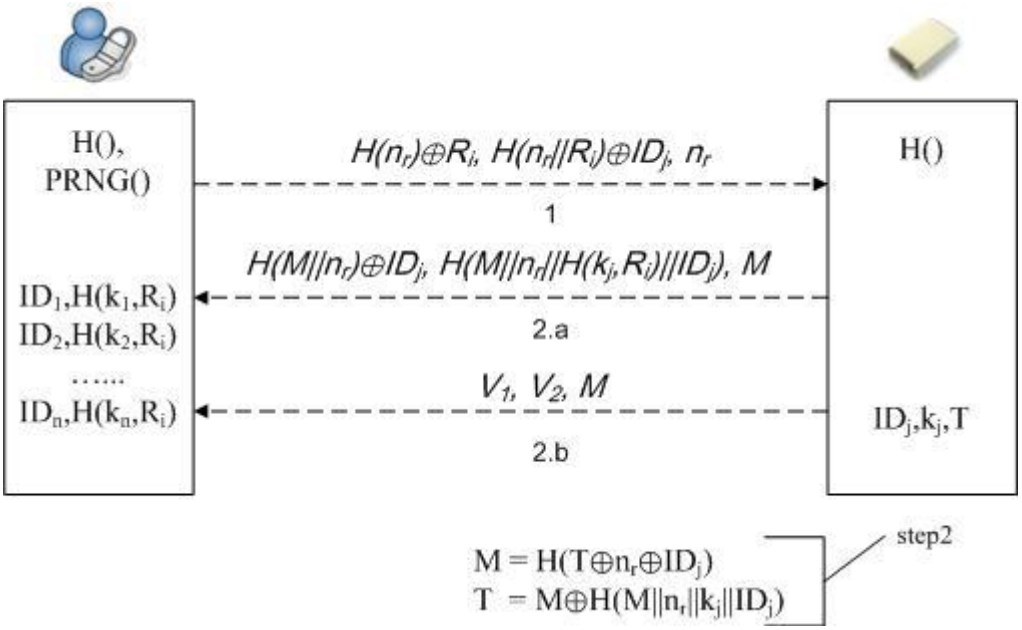

Figure 3. The proposed search protocol

6. Security Analysis

(1)Tag untraceability

An adversary can eavesdrop the response message $(H(M||n_r)\oplus ID_j, H(M||n_r||H(k_j,R_i)||ID_j), M)$ from a tag, and analyze the information carefully and try to detect the user location privacy by tracking the tag. Because the tag generates a new substitute random number $M = H(T\oplus n_r\oplus ID_j)$ during each authentication access, and updates $T = M\oplus H(M||n_r||k_j||ID_j)$ in the step2, so the adversary cannot determine which tag does the response from the message $(H(M||n_r)\oplus ID_j, H(M||n_r||H(k_j,R_i)||ID_j)$. So these two protocols can meet tag untraceability.

(2)Tag information protection

ID is stored in the reader only and is not transmitted in plaintext from the reader to tag or from tag to the reader all the performing process of these two protocols, ID is shield by $H(M||n_r)$, an adversary cannot calculate $H(M||n_r)$ so as to can't acquire ID, so this protocol can meet tag information protection.

(3)Spoofing attack

An adversary feigns a legitimate reader that sends a query with $H(n_r)\oplus R_i$, n_r to tags through the forward channel, and obtains the response of a tag $(H(M||n_r)\oplus ID_j, H(M||n_r||H(k_j,R_i)||ID_j), M)$. In the next authentication access, when a legitimate reader sends query with $H(n_r')\oplus R_i$, n_r', the adversary feigns the tag and responds the legitimate reader with the obtained message $(H(M||n_r)\oplus ID_j, H(M||n_r||H(k_j,R_i)||ID_j), M)$ through the backward channel. However, the reader generates a new random number during each authentication access, namely $n_r \neq n_r'$, so the adversary cannot perform tag impersonation.

(4)Replay attack

Replay attack can be prevented in this protocol due to the message transmitted for each session is different. Different value of $H(M||n_r)$ is utilized in individual session and T that stored in a tag plays a key role in providing different value of $H(M||n_r)$ to conceal ID of the tag. An adversary cannot hold H() and then acquire $H(M||n_r)$, so it is impossible for an adversary to apply replay attack.

(5)Denial of Service (DoS) attack

As the ID of a tag is fixed, even if loss of message, power failure or loss of connection with the reader happens during a performing access, it would not affect ID data that stores in the reader, namely it would not lose the synchronization between the reader and the tag, only resetting a new access is well, so these two protocols can shield DoS attack well.

(6)Reader untraceability

An adversary can eavesdrop the request message $(H(n_r)\oplus R_i, n_r)$ or $(H(n_r)\oplus R_j, H(n_r \| R_i) \oplus ID_j, n_r)$ from the reader and get $(H(n_r)\oplus R_i)$, and analyze the information carefully and try to detect the user location privacy by tracking the reader. Because the reader generates a new random n_r during each authentication access, and calculates $H(n_r)$ to shield R_i in the step1, so the adversary cannot determine the user location privacy by tracking the reader. So these two protocols can meet reader untraceability.

(7)Privacy of search result

This requirement usually is needed in RFID search protocol, in the step2 of our proposed search protocol, we will choose and respond two random numbers V_1, V_2 with probability and M. This arrangement is to confuse attackers from tracing tag ID_j since not only tag ID_j but also tag ID* will respond. So it protects the privacy of search result.

(8)Mutual authentication

This requirement usually is needed in RFID authentication protocol, in step2 and step3 of our authentication protocol, it achieves the mutual authentication objects.

Table 2 indicates a comparison of results among our authentication protocol and the related authentication protocols [8, 14, 15] in terms of security.

Table 2. Comparison of security

Security requirement	[8]	[14]	[15]	New
Tag untraceability	X	X	X	O
Reader untraceability	X	X	X	O
Tag information protection	O	O	O	O
Spoofing attack	O	O	X	O
Replay attack	O	O	O	O
DoS attack	O	O	O	O
Mutual authentication	O	X	X	O

'O' denotes satisfied, 'X' denotes not satisfied

Table 3 indicates a comparison of results among our search protocol and the related search protocols [8, 14, 15] in terms of security.

Table 3. Comparison of security

Security requirement	[8]	[14]	[15]	New
Tag untraceability	O	O	O	O
Reader untraceability	X	X	X	O
Tag information protection	O	O	O	O
Spoofing attack	O	X	X	O
Replay attack	O	O	O	O
DoS attack	O	O	O	O
Privacy of search result	O	O	O	O

7. Conclusion

Server-less RFID systems bring higher design requirements for RFID security protocols. In this paper, a mutual RFID authentication protocol and its corresponding search protocol for server-less RFID systems are proposed, these two protocols only requires O(1) work to identify and authenticate a tag in the reader. The careful security analysis shows that these two protocols can meet common privacy and security requirements for RFID systems. The next work we should do is design security protocols based on dynamic ID scheme for server-less RFID systems.

Acknowledgements

This work was partially supported by Research Projects of State Ethnic Affairs Commission No.12DLZ001; Heilongjiang Province Science and Technology Research Grant of the Education Department No.12533002.

References

[1] Z. Shijie, Z. Zhen, L. Zongwei and E. C. Wong, "A lightweight anti-desynchronization RFID authentication protocol", Information Systems Frontiers, vol. 12, (**2010)**, pp. 521-528.
[2] C. -F. Lee, H. -Y. Chien and C. -J. Laih, "Server-less RFID authentication and searching protocol with enhanced security", International Journal of Communication Systems, vol. 25, (**2012**), pp376 -385.
[3] B. Alomair, A. Clark, J. Cuellar and R. Poovendran, "Securing low-cost RFID systems: an unconditionally secure approach", The 2010 Workshop on RFID Security-RFID sec'10 Asia, Singapore, **(2010)**.
[4] A. Juels, "RFID security and privacy: a research survey", Journal of Selected Areas in Communications, vol. 24, (**2006**), pp. 381-394.
[5] H. Jialiang, O. Dantong and X. Youjun, "An Efficient RFID Authentication Protocol Supporting Tag Ownership Transfer", International Journal of Advancements in Computing Technology, vol. 4, no. 4, (**2012**), pp. 244-253.
[6] B. Song and C. J. Mitchell, "Scalable RFID security protocols supporting tag ownership transfer", Computer Communications, vol. 34, **(2010),** pp. 556-566.
[7] H. Jialiang, O. Dantong and Y. Yuxin, "An Efficient Lightweight RFID Authentication Protocol for Low-cost Tags", Advances in Information Sciences and Service Sciences, vol. 3, no. 9, **(2011)**, pp. 331-338.
[8] J. Y. Chun, J. Y. Hwang and D. H. Lee, "RFID tag search protocol preserving privacy of mobile reader holders", IEICE Electronics Express, vol. 8, no. 2, (**2011**), pp. 50-56.
[9] C. -F. Lee, H. -Y. Chien and C. -S. Laih, "Server-less RFID authentication and searching protocol with enhanced security", International Journal of Communication Systems, vol. 25, (**2012**), pp. 376-385.
[10] Y. Zuo, "Secure and private search protocols for RFID systems", Information Systems Frontiers, vol. 12, (**2010**), pp. 507-519.
[11] M. E. Hoque, F. Rahman, S. I. Ahamed and J. H. Park, "Enhancing Privacy and Security of RFID System with Serverless Authentication and Search protocols in Pervasive Environments", Wireless Personal Communications, vol. 55, no. 1, (**2009**), pp. 65-79.
[12] S. I. Ahamed, F. Rahman, E. Hoque, F. Kawsar and T. Nakajima, "S3PR: Secure Serverless Search Protocols for RFID", Proceedings of the 2th International Conference on Infrormation Security and Assurance, (**2008**), pp. 187-192.

[13] S. I. Ahamed, F. Rahman, E. Hoque, F. Kawsar and T. Nakajima, "Secure and Efficient Tag Searching in RFID Systems using Serverless Search Protocol", Security and Its Applications, vol. 2, no. 4, (**2008**), pp.57-66.
[14] C. C. Tan, B. Sheng and Q. Li, "Secure and server-less RFID authentication and search protocols", Proceedings of IEEE Transactions on Wireless Communications, vol. 7, no. 4, (**2008**), pp. 1400-1407.
[15] I. C. Lin, S. C. Tsaur and K. P. Chang, "Light weight and server-less RFID authentication and search protocol", Proceedings of the 2009 Second International Conference on Computer and Electrical Engineering, vol. 2, IEEE: New York, (**2009**), pp. 95-99.

Authors

He Jialiang

Was born in 1977, received the PhD degree in computer software and theory from Jilin University of China in 2012 and the Master degree in computer application from Jilin University of China in 2004. Now he is an associate professor at College of Information and Communication Engineering, Dalian Nationalities University, China. His papers have been published in some well-known international Journals and IEEE conferences. His main interests include Mobile Internet, Internet of Things, and Intelligent Business Information Processing.

Xu Youjun

Was born in 1977, received the Master degree in computer application from Jilin University in 2005 and the PhD degree in computer software and theory from Jilin University in 2011. Now he is a lecturer at College of Computer Science and Information Technology, Daqing Normal University, China. His papers have been published in some well-known international Journals. His main interests include Automated Reasoning, Internet of Things.

Xu Zhiqiang

Was born in 1981, received the Master degree in Electronics & Communication Engineering from Communication University of China in 2012. At present, he is an assistant professor of Communication & Media Institute of Sichuan, China. He is experienced the fields of Mobile Internet, Internet of Things, Intelligent Information Processing, etc., he also is a candidate of MSc of Technopreneurship & Innovation Program in Nanyang Technological University in Singapore.

