RESEARCH ARTICLE OPEN ACCESS

RFID Tag Ownership Transfer Protocol of Multi-owner and Multi-tag Based on EPC C1G2 Protocol

MENG Ke[1], HUANGXincheng[1],CHEN Shun'er[1]*, HUANG Hongbing[1], LIU Weiping[1, 2]

[1](College of information science and technology,Jinan University,Guangzhou,China,)

[2](ZhongshanAiscent Technologies Ltd,Zhongshan,Guangdong,528437,China)

*(Corresponding author: CHEN Shun'er)

----------**********----------

Abstract:

The mutual authentication and ownership transfer are the core issues in the protocols of Regarding RFID tag, especially those based on EPC C1G2 protocol. At present, most of the multi-owner multi-tag ownership transfer cannot be able to deal with the problem of part ownership transfer. To solve this problem, we here integrate the mutual authentication and ownership transfer of tag to propose an effective multi-owner multi-tag protocol by storing the tags related information into the server. Our protocol characters with low resource consumption and high security compared with now existed protocols.

Keywords **—RFID tag,C1G2 protocol, Multi-ownership tag, Transfer of ownership.**

----------**********----------

I. INTRODUCTION

With the development of Internet of Things, radio frequency identification (RFID) is widely applied in many areas such as supply chain system, military surveillance and medical care [1][2].Due to itscontactless character,RFIDalternates the traditional bar code and QR code, providing an automatic acquisition of related data with low cost. A complete RFID tag system includes three parts: tag, reader and server database. In 2006, the well-known EPC C1G2 protocol is introduced officially. EPC C1G2 is targeted at low-power and resource-limited RFID tags [3][4], and has been extensively applied in Mobile IoT since then.

As the outdoor deployment of RFID device and wireless transmittance of its radio signal, presently it is lack of sufficient security means to prevent RFID system from illegal access and hostile attack [5]. Thus, a lot of security protocols regarding RFID tag, including security authentication protocol and ownership transfer protocol, are proposed to avoid the disclosure of relevant information [6][7]. However, those protocols are based on traditional encryption technology, such as RSA, Hash function and ECC, and thus not competent with passive C1G2 protocol tag [8]. In one hand, for example, the number of gate circuits for the encryption of RFID system is at most 3000 under the limit of the power dissipation, which is far less than that needed for the traditional Hash algorithm (80,000 to 100,000), ECC (8,200 to 15,000) and the simpler AES algorithm (3,400 gate circuits at least). In other hand, early C1G2 protocol mainly concentrates on the problems related with a single tag[9][12], and is proved to be with various security defects, and would be more complicated in a practical situation if related protocols focus on only a single tag. However, in complex commercial problems, such as the problem of supply chain where for instance the ownership of a commodity may belong to multiple owners or one owner has the ownerships of multiple commodities, there must be a multi-owner multi-tag ownership transfer agreement to tackle the transference of the ownership between the owners.

Tosolve problems above-mentioned, a systematical multi-owner and multi-tag ownership transfer agreement is proposed in this paper, with the factors, such as limited resources of C1G2 standard tag, protocol security and possible partial ownership transfer, being entirely considered. This paper is arranged as following: in the first chapter we illustrate relevant background knowledge about

RFID; the second chapter we introduce current research work in this area; the third chapter we explain the regulations of the protocol raised in this paper; and the fourth chapter we discuss protocol security and compares it with on-going work.

II. RELATED WORK

Currently, there are two categories of transfer agreements:1. with trusted party(TTP);2. without trusted party. They all need a safe environment to ensure that the process of key updating is in an absolute safe environment. Many protocols [14] assume that TTP directly communicate with tag, but in fact the effective communication distance of RFID tag is within 2-10m and only applicable to indoor scenes. However, the confinement of its indoor deployment imposes great restriction to its application in the case of the well-developed mobile IoT.

As demonstrated by Kapoor [14] where the concept of ownership transfer of multi-owner multi-tag with TTP was first proposed, the ownership transfer agreements was completed by mutual confirmation of an encrypted message between the current and the new owners after the tags being transferred from the current owners to new owners.

Another version protocol proposed by Sundaresan[15] also adopts TTP strategy with an assumption that TTP could directly communicate with tags and owners. The ownership is transferred by four steps: 1.the current users send a transfer request to TTP, 2.TTP creates a key for each of the new owners and send it to them together with relevant tag information in an encrypted way. Then after, new owners will verify the encrypted information and successively send it to TTP, 3.TTP rechecks this returned information and sends a key back to tags. After being decoded, tags will subsequently send a confirmation message back to TTP, 4.TTP confirms this received information from tags and then sends a transfer completion information to all the current owners. Different from Kapoor's protocol which is based on the repeated operation on a single tag, the Sundaresan's protocol is based on a group strategy that groups tags according to whether they are to be transferred or not.

However, by analyzation, we find that both Kapoor's and Sundaresan's protocols cannot fit the application scene of part ownership transfer where part owners of one tag give up their ownership but the rest do not, due to the potential loss of some ownership for some owners.

III. PROTOCOLS OF THIS PAPER

Unlike that as in the relevant protocols mentioned above, our version protocol raises four participants: user, tag, reader and server database. Among them, user carries out operations related to protocol by using reader as an entrance.

A. Introduction of Symbols

The symbols used in this paper are as follows:

- T_{id}: tag id, EPC code in the actual use
- T_r: The last record time that the reader visited
- K_j, K_j' : the new and old access key of tag j
- K_s, K_s': the new and old shared key between the server database and the tag
- R_{id}: reader id
- t: current time
- O_{id}: owner id
- pwd:the owner's password, the owner's password is assumed to bepwd
- $H(*)$:cryptographic relevant data of Hash function
- M_*:cryptographic relevant data of Hash function
- $PRNG(*)$:128-bit pseudo-random number generator
- $\oplus$: XOR operation

B. Introduction of the Protocol

In this paper, the implementation of our version protocol is divided into three stages: initialization; mutual authentication; and ownership transfer.

1.Initialization

We storeT_{id}, K_i, K_s, $K_s'$for each tag and setR_{id} for the reader. On the server, storesT_{id}, K_j, K_j' , K_s for each tag and storesR_{id}, T_r for each reader, as well as storesO_{id}, $T_{id(1\ldots n)}$$pwd$for each owner, $T_{id(1\ldots n)}$ is all the tagsT_{id} owned by the owner.

2. Mutual Authentication

1) Reader → Server: M_1, M_2, N_r, t_r

At the beginning of the certification, the reader first generates a current time and a random number for the calculation of , that is

$$M_1 = H\left(R_{id} \oplus N_r \oplus t_r\right) \quad (1)$$

$$M_2 = H\left(O_{id} \oplus pwd \oplus t_r\right) \quad (2)$$

Then, it sendsM_1, M_2together withN_r, $t_r$to the server.

2) Server → Reader: M_3

First, after receiving the information sent by the reader, the server traverses the database store to obtain the R_{id} of each reader, and judges whether the formula (1) is valid or not with the help of N_r, t_r. If it is valid, the reader is legal and the0 process will go on, otherwise it will be terminated.And then according to the storage time $T_r$of the reader, if $t_r<T_r$, and the process will go on, otherwise it will be terminated. And last determining the formula (2) to be valid or not according to each user'sO_{id} and the received N_r. If it is valid, the user is legal and the process will go on, otherwise it will be terminated. If both the judgments are valid, M_3will be calculated as

$$M_3 = H\left(R_{id} \oplus N_r\right) \quad (3)$$

and subsequently returned to the reader.

3) Reader→ Tag: t_r, query

After the returned message from the server being received by the reader, it will be used to judge the legality of formula (3). If the formula (3) is valid, which means that the server database is legal, the reader will send a request message and its reader time to the tag.

4) Tag→ Reader: M_{4j}, M_{5j}, M_{6j}

For each tag$j$, after it receiving the request messagefrom the reader, a random number$N_t$will be generated to calculateM_{4j}, M_{5j}, M_{6j} as

$$M_{4j} = t_r \oplus PRNG\left(K_j \oplus N_t\right) \quad (4)$$

$$M_{5j} = PRNG\left(K_j\right) \oplus N_t \quad (5)$$

$$M_{6j} = PRNG(K_j \oplus T_{id}) \oplus PRNG\left(N_t\right) \quad (6)$$

which will be sent back to the reader. And last, the stored will be updated

$$T_r = t_r \quad (7)$$

5) Reader → Server: M_{4j}, M_{5j}, M_{6j}

After receiving the returned messages sent back by the tag, the reader forwards them to the server.

6) Server → Reader: M_{7j}, M_{8j}

The information received by the Server will be again sent to readers by which the readers determine whether the formula (9) is valid or not according toT_{id}, K_j andN_{tx} stored in the database after traversing each tag in the database. The variableN_{tx} is got by the formula (8). If the formula (10) is invalid,K_j will be replaced byK_j' and the above steps will be repeated. If it is invalid, the certification will be terminated. However, if the formula (10) is valid, that isK_j orK_j' is valid, it still needs to specify whether there is a corresponding owner. If the answer is yes, the server will generate a random number N_s, work out M_{7j}, M_{8j} from the formula (11) and (12) and send them to the reader. And the access keys $K_s$andK_j' will be updated by formula (13) and (14) if necessary.

$$N_{tx} = M_{5j} \oplus PRNG\left(K_j\right) \quad (8)$$

$$M_{4j} = t_r \oplus PRNG\left(K_j \oplus N_{tx}\right) \quad (9)$$

$$M_{6j} = PRNG\left(K_j \oplus T_{id}\right) \oplus PRNG\left(N_{tx}\right) \quad (10)$$

$$M_{7j} = PRNG\left(k_s \oplus T_{id} \oplus t_r\right) \quad (11)$$

$$M_{8j} = PRNG\left(K_j \oplus T_{id}\right) \oplus N_s \quad (12)$$

$$K_s = PRNG\left(K_s \oplus N_s\right) \quad (13)$$

$$K_j' = K_j \quad K_j = PRNG\left(k_j \oplus N_s\right) \quad (14)$$

7) Reader→Tag: M_{7j}, M_{8j}

Again, the reader will forward what it receives to the corresponding tag.

8) Tag

After the tag receives the relevant information from the reader, it will use the stored keyK_s to judge whether formula (11) is valid. If not, replacingK_s with$K_{.s}$ and repeat the judgement. If it is still invalid, just terminate the process. Note thatN_s can be obtained through the formula (12), and the keysK_s, $K_{.s}$ can be updated according to formula (13) when needed.

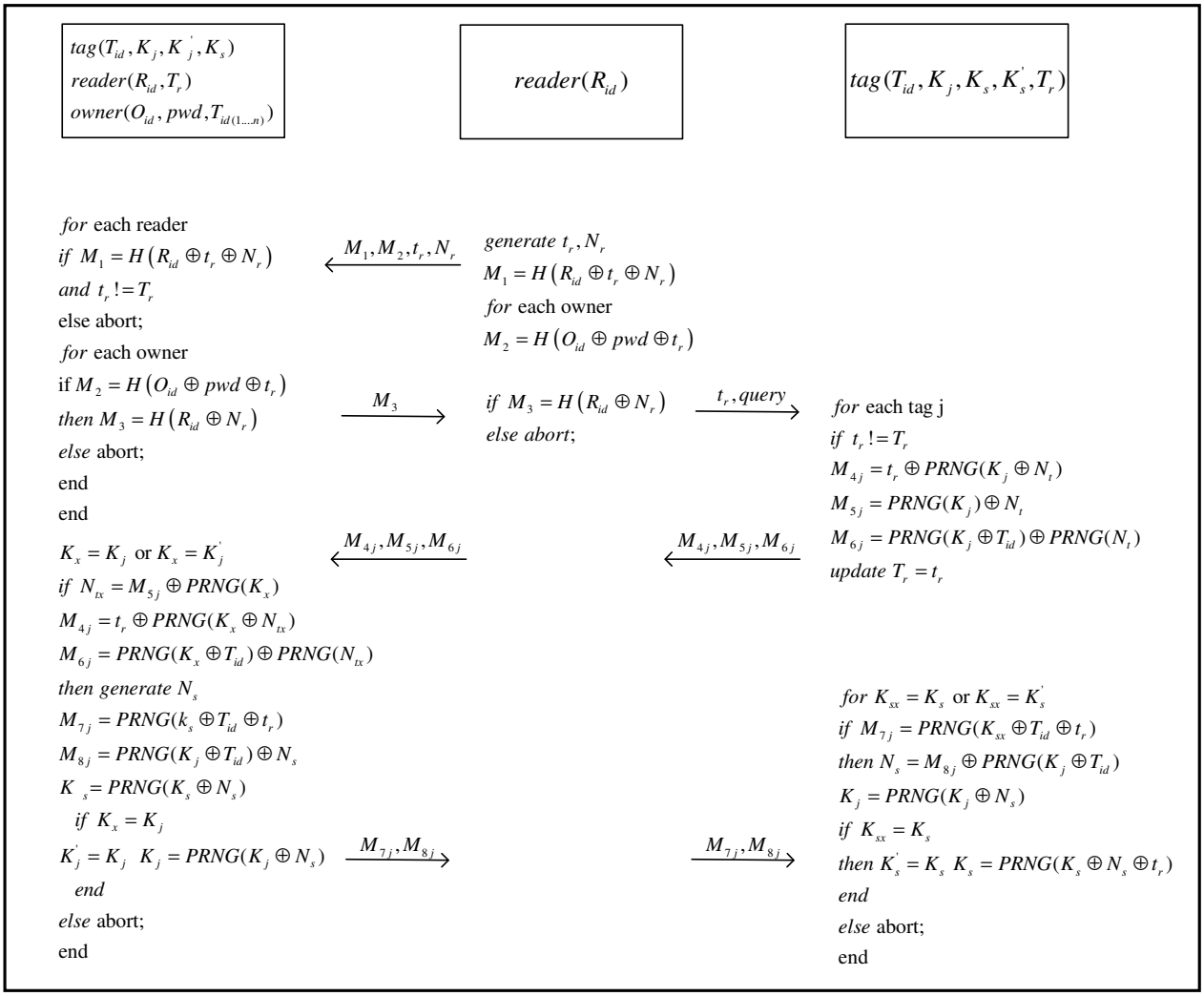

Fig. 1 Authentication Phase

3. Ownership Transfer

Ownership Transfer is implemented after completing the confirmation of the validness of all the tag, reader, server and current owner.

1)*Reader → Server: M_9, t_r*

The reader first generates a current time t_r,works outM_9 after each new owner enters his own O_{id}, *pwd*, and then sends it to the server.

$$M_9 = H\left(O_{id} \oplus pwd \oplus t_r\right) \tag{15}$$

2) Server→Reader: M_{10j}, M_{11j}

After receiving the information sent by the reader, the server judges whether formula (15) is valid based on the storedO_{id}, *pwd*of each user and receipt time t_r.If the stored timeT_r of the reader is the same as t_r,the protocol will be terminated. If not, a new random numberN_s will be generated.ThenM_{10j}, M_{11j} for each tag*j* that needs to be transferred will be worked out from formula (16) and (17) and sent back to the reader. The access key K_j will be updated thereafter by the formula (18).

$$M_{10j} = PRNG\left(K_s \oplus t_r\right) \oplus N_s \tag{16}$$

$$M_{11j} = PRNG\left(K_s \oplus T_{id} \oplus N_s\right) \tag{17}$$

$$K_j = PRNG\left(K_j \oplus N_s\right) \tag{18}$$

3) Reader→Tag: M_{10j}, M_{11j}, t_r

When the reader receives the information of the server, it will send the information along with its time reader $t_r$to the tag.

4) Tag→Reader:M_{12j}

After receiving the relevant information, the tag first judgest_r by the method mentioned above. Then it uses its own$K_s$to work outN_{sx} by the formula (19), and then assesses whether the formula (20) is established. If not, replace$K_s$with$K_{.s}$ to repeat the above steps.If still not, the protocol will be terminated. Otherwise, it works outM_{12j} and sends it to the reader, updates K_j,K_s and$K_{.s}$ if necessary.

$$N_{sx} = M_{10j} \oplus PRNG(K_s \oplus t_r) \quad (19)$$

$$M_{11j} = PRNG(K_s \oplus T_{id} \oplus N_{sx}) \quad (20)$$

$$M_{12j} = PRNG(K_j \oplus K_s \oplus T_{id} \oplus t_r) \oplus N_s \quad (21)$$

$$K_j = PRNG(K_j \oplus N_{sx}) \quad (22)$$

$$K_s' = K_s \quad K_s = PRNG(K_s \oplus N_{sx}) \quad (23)$$

5) Reader →Server: M_{12j}

After the reader receivesM_{12j} from the tag, it will forward it to the server.

6) Server

First, judge whether the formula (21) is established according to the storedK_j,K_s, T_{id} and the random number N_s. If it is right,updateK_s.

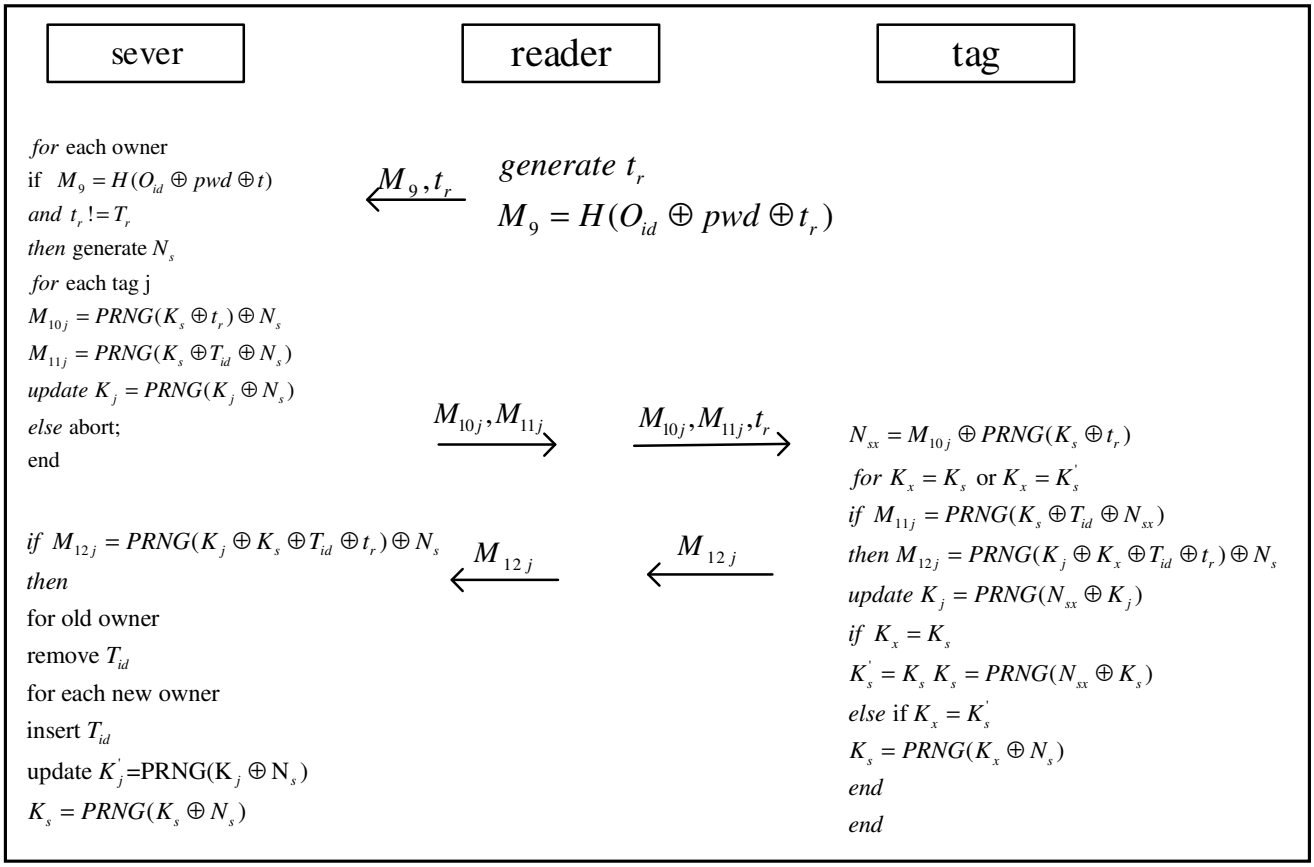

Fig. 2 Ownership Transfer Phase

IV. Safety Analysis and Performance Comparison

In this part, we mainly discuss how our version protocol in this paper deals with all kinds of malicious attacks, and analyze its characteristics in the computation and storage under the limit of limited resource.

A. Safety Analysis

1)Replay Attack

Replay-Attack refers to that an attacker repeatedly sends the same message to the target by capturing a past conversation to paralyze the target

as is seen in Sundaresan[15]. To avoid the Replay-Attack, we assume that the current timet_r must be after the last access time T_r to ensure the access efficiency. In the authentication stage, if the attacker conducts a replay attack to the server or the tag, his reader timet_r obviously does not always meet the condition t_r!= $T_r$due to the existence of the reader time T_r.Thus, the judgment of M_1, $M_{4j}$by the server cannot be completed that leads to the failure of the authentication of tag M_{7j}.Also if the Replay-Attack occurs in the transfer stage, the server can prevent Replay-Attack using M_9, M_{12j}.

2) Desynchronization Attack

Desynchronization attack is a common threat to RFID tag and exists in many of the current RFID security protocols. The principle of desynchronizing attack is to make the tag and server out of synchronization with the keys, resulting in all subsequent failures in authentication. The simple way to prevent is to block an authentication conversation so that the server or the tag updates its key while the other party doesn't complete the update. The protocols proposed by Kapoor [14] and Sundaresan [15] are vulnerable to the desynchronization attack. In this paper, we use two keys (the old key and the new key) to prevent the desynchronization attack. It saves the old and new keys of tagK_j, K_j in the server and stores the old and new shared keys in the tag. The old key is always equal to the last successful key and the new key will be updated according to a random number. In this way, even if the attacker blocks the protocol, it can't hinder the next round of authentication because we use both the old and the new key to try in the new authentication process. So even if the current update is blocked, the old key can still be authenticated.

3) Tag Imitation

As the tag information is transmitted through an encrypted way, the attacker needs to imitate a tag and get the variablesT_{id}, K_j, K_s, K ,s of the tag. However, as our related information is transmitted through the encrypted way, he needs to use the violent crack pseudo-random number generator to crack the tag which greatly reduces his probability of success. In addition, as we know, more bits mean more cracked number. So, the advanced128-bit PRNG of C1G2 with its cracked number 2^{127} is recommended instead of the 16-bit standard PRNG of C1G2 with its cracked number 2^{15}. Sundaresan et al also proposed that 128-bit PRNG resources required only 1500 logic gates. Therefore, in this paper, we adopt the 128-bit pseudo-random number generator.

4) Forward Security and Backward Security

Forward security and backward security are crucial to the ownership transfer protocol because it is necessary to ensure that the previous owner can't access the tag and the later owner can't get the information of the previous owner. The access keyK_j and shared keyK_s will be automatically updated in the process of each authentication and transfer. Neither the previous owner nor the current owner can get the next round of key or the last round of key.

B. Authentication Efficiency

1)Expansibility of Protocol

We assume a situation where a part of the owners of the tag have transferred ownership while the rest haven't. It is obvious that Kapoor or Sundaresan version protocol can't meet this requirement because partially updating the keys of the owners will lead to that the owners who haven't transferred the ownership will lose his authority to the tag. Our version protocol in this paper solve this problem by updating all the keys that all the owners got meanwhile by storing both the access key and shared key of the tag in the database. So, it well deals with the part transfer of ownership while Kapoor or Sundaresan version protocols can't.

2) Storage of Tags

The information of a tag needed to store in our protocol only are T_{id}, K_j, K_j , K_s, T_r,regardless of the number of owners of tag. However, there are at least $2N_t + N_o$ keys and ID information needed to store if there areN_t tags andN_o owners for the Sundaresan version protocol, and at leastN_{og}+ 2 shared keys for the Kapoor version protocol if there areN_{og} owners. So, by comparison, we can see that our version protocol is efficient and economic especially when the number of the owners is growing enough.

C. Comparison of Protocols

Through the above analysis, we conclude that our version protocol is better than others not only in its security and the effective dissipation of resources,but also in its functionality of the transfer of part ownership. Different from the previous protocols, our protocol doesn't require that the tag communicate directly with the TTP. The tag only needs to communicate with the reader, which is more in line with the concepts of Internet of Things with strong practical significance.

TABLE 1 COMPARISON OF RELATED PROTOCOLS

	Protocol in this paper	Kapoor protocol	Sundaresan protocol
Prevent replay attack	Y	N	N
Prevent desynchronization Attack	Y	N	N
Forward security and backward security	Y	Y	Y
Prevent tag imitation	Y	Y	Y
Transfer of part of ownership	Y	N	N
Storage of tags	5	$N_{og}+2$	$2N_t+N_o$

V. CONCLUSIONS

The development of Internet of Thing has driven the development of RFID, but its security problems have always plagued scholars. Currently, security protocols of tags based on the EPC C1G2 standard are mostly based on a single tag, which has great limitations in the shared resources society. This paper proposes a multi-owner multi-tag ownership transfer protocol for RFID tags based on the C1G2 standard, which uses the traditional reader connecting the tag and TTP communication so that the tag doesn't need to directly connect to the TTP. As the key information of owners and tags are stored in the serve, this ownership transfer protocol is more adaptive to the outdoor scene r. Besides, it also can be a good solution to the transfer of part ownership and can reduce the requirement to the operation and storage of tags, which more fits low power dissipation characteristics of C1G2 tag.

ACKNOWLEDGMENT

This work is supported by National High Technology Research and Development Program of China (863 Program, No. 2015AA015501).

REFERENCES

[1] Li H, Hu J, He L, et al. Mutual Authentication and Ownership Transfer Scheme Conforming to EPC-C1G2 Standard[C]// Eighth International Conference on Computational Intelligence and Security. IEEE, 2012:678-682.

[2] Pokala J P, Reddy C M, Abdul J S, et al. A secure RFID protocol for Telecare Medicine Information Systems using ECC[C]// International Conference on Wireless Communications, Signal Processing and NETWORKING. 2016:2295-2300.

[3] Zhang J, Wang W, Ma J, et al. A Novel Authentication Protocol suitable to EPC Class 1 Generation 2 RFID system[J]. Journal of Convergence Information Technology, 2012, 7(3).

[4] Xiao F, Zhou Y, Zhou J, et al. Security protocol for RFID system conformin0g。12 to EPC-C1G2 standard[J]. Journal of Computers, 2013, 8(3).。

[5] Chien H Y, Chen C H. Mutual authentication protocol for RFID, conforming to EPC, Class 1 Generation 2 standards[J]. Computer Standards & Interfaces, 2007, 29(2):254-259.

[6] Kulseng L, Yu Z, Wei Y, et al. Lightweight mutual authentication and ownership transfer for RFID systems[C]// Conference on Information Communications. IEEE Press, 2010:251-255.

[7] Cong, ZHANG, Zi-jian, et al. A novel secure group RFID authentication protocol[J]. Journal of China Universities of Posts & Telecommunications, 2014, 21(1):94-103.Osaka K, Takagi T, Yamazaki K, et al. An Efficient and Secure RFID Security Method with Ownership Transfer[C]// Computational Intelligence and Security. Springer-Verlag, 2007:778-787.

[8] Sundaresan S, Doss R, Zhou W, et al. Secure ownership transfer for multi-tag multi-owner passive RFID environment with individual-owner-privacy[J]. Computer Communications, 2015, 55(C):112-124.

[9] Mohammadi M, Hosseinzadeh M, Esmaeildoust M. Analysis and Improvement of the Lightweight Mutual Authentication Protocol under EPC C-1 G-2 Standard[J]. Advances in Computer Science An International Journal, 2016, 3(2):E417-E423.

[10] Wu K, Bai E, Zhang W. A Hash-Based Authentication Protocol for Secure Mobile RFID Systems[C]// International Conference on Information Science and Engineering. IEEE, 2009:2440-2443.

[11] Chen C L. An Ownership Transfer Scheme Using Mobile RFIDs[J]. Wireless Personal Communications, 2013, 68(3):1093-1119.

[12] Huang Y C, Jiang J R. Efficient Ultralightweight RFID Mutual Authentication[C]// Internet of Things. IEEE, 2014:102-108.

[13] Munilla J. Cryptanalaysis of an EPCC1G2 Standard Compliant Ownership Transfer Scheme[J]. Wireless Personal Communications, 2013,72(1):245-258.

[14] Kapoor G, Zhou W, Piramuthu S. Multi-tag and multi-owner RFID ownership transfer in supply chains[J]. Decision Support Systems, 2012,52(1):258-270.

[15] Sundaresan S, Doss R, Zhou W. Secure ownership transfer in multi-tag/multi-owner passive RFID systems[C]// Global Communications Conference.IEEE,2013:2891-2896.