International Journal of Security and Its Applications
Vol. 10, No. 2 (2016), pp.115-128
http://dx.doi.org/10.14257/ijsia.2016.10.2.11

The Lightweight Ownership Transfer Protocol using Physically Unclonable Function

Xuejun Zhang[1,2], Wanlu Huang[1], He Xu[3,4,*] , and Yu Wang[1]

[1] *College of Electronic Science and Engineering, Nanjing University of Posts and Telecommunications, Nanjing, 210003, China*
[2] *Jiangsu Province Engineering Lab of RF integration & Micropackage, Nanjing, 210003, China*
[3]*School of Computer Science & Technology/School of software, Nanjing University of Posts and Telecommunications, Nanjing, 210003, China*
[4] *Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks, Nanjing, 210003, China*
[1,2]*xjzhang@njupt.edu.cn;* [3,4,*]*xuhe@njupt.edu.cn*

Abstract

RFID technology has increasingly used in the various fields of our lives, and in the entire life cycle of the tag, there may be several ownership changes. Therefore, it is very important to protect the security and privacy of customer's information during transferring. Combining with the physical characteristics of the anti-clone tag and the low-cost requirement, we proposed a lightweight ownership transfer protocol using Physically Unclonable Function (PUF) to achieve a safe ownership transfer. Through using formal analysis, it shows that this protocol can also resist reply attack, desynchronization attack, tampering attack, to protect the old and the new owner's private information.

Keywords*: Radio Frequency Identification (RFID), Ownership Transfer, PUF, Security and Privacy*

1. Introduction

RFID is the abbreviation of Radio Frequency Identification, which is one kind of Automatic Identification Technology. It uses the wireless channel to execute non-contact bi-directional data communication, meanwhile identifying the target and getting relevant information [1]. RFID system consists of three important components, tags, readers and back-end databases. The reader which is apart from the tag a few centimeters to several meters sends a radio frequency signal, communicating with the tag within its range. After achieving mutual authentication, the reader reads the memory of tag to get valuable information, and gets the identification of goods or human attached to the tag [2]. Compare with the bar code, Radio Frequency Identification system has many advantages. Obviously, the tag has greater storage capacity, so that it can ensure that all product items attached to a tag have a unique identifier. This breakthrough completely breaks the limitations of bar code.

However, the ownership of the tag is transferred frequently in RFID system, it is very necessary to have a safe and effective ownership transfer protocol. And the premise of the protocol is that a tag can communicate with one or more readers. RFID-tagged product item such as a toy or a pack of sausage, supply chain management among factories, distributors, retailers and customers need to handle ownership transfer issue for product

* Corresponding author, He Xu, Email: xuhe@njupt.edu.cn

ISSN: 1738-9976 IJSIA

item carefully. With RFID technology, ownership transaction for product item can be handled along with product item identification and the overall system efficient on supply chain will be increased. Therefore, a secure and efficient RFID ownership transfer protocol is crucial for stakeholder in supply chain.

Since the RFID tag attached on a product item may contain owner information and access control data, a full-fledged RFID tag ownership transfer protocol should meet the following security and privacy conditions [3]:

(1) New owner privacy: when the ownership of a tag has been transferred to a new owner, only the new owner can identify and control the tag any more.
(2) Old owner privacy: after the ownership of a tag has been transferred to a new owner, the new owner will not be able to trace back past interactions between the tag and its previous owners.
(3) Authorization recovery: in some situations such as after-sales server for an RFID-tagged object, the present owner may need to transfer tag ownership to the previous owner temporarily such that the previous owner of this object can get authority to access information inside the tag.
(4) Resistance to Denial of Server (DoS) attack: synchronization mechanism foe shares secret information between tag and the back-end server is required to prevent DoS attack.
(5) Resistance to replay attack: attackers cannot utilize eavesdropped messages, which were transmitted between reader and tag, to cheat reader or tag in a new communication session.
(6) Resistance to man-in-the-middle attack: adversaries positioning themselves between tag and reader cannot successfully insert false message or modify messages passed between tag and reader.
(7) Prevention of windowing problem: if ownership transfer protocol is not well designed, there is a period of time in which both the new owner and the old owner can access the tag with the same shared secret key during ownership transaction.

The rest of paper is structured as follows: In Section II, we review the related work. In Section III, we present our lightweight ownership transfer protocol. In section IV, we provide the security and performance analysis. In Section V, we use BAN logic to prove the security of the protocol. Section VI concludes the paper.

2. Related Work

2.1. Silicon PUF

Physically Unclonable Function[4][5] is a technology that can extract a unique ‘key’ (secrets) from a chip, these ‘key’ information can be used to verify the authenticity of the chip, and it has a great application prospect in the security field. Each chip will inevitably have a lot of unique characteristics in manufacturing process. It is impossible to clone two identical chips with the same design, packaging and manufacturing process. Therefore a chip can enhance its anti-cloning feature using the PUF technology. As the Arbiter-based PUF shown in Figure 1, the received query command is seen as an input signal of PUF circuit to produce a unique sequence as a response. It is the use of challenge/response authentication mechanism.

Different PUF circuits create discrepant delay characteristics and signal transmission speeds, therefore the durations of two signals pass through the same PUF circuit are diverse. The arbiter at the end of a PUF circuit determines whether the output is ‘0' or ‘1’ according to the order of signal arrival. If one signal goes through two different PUF circuits respectively, the output sequences are also inequable.

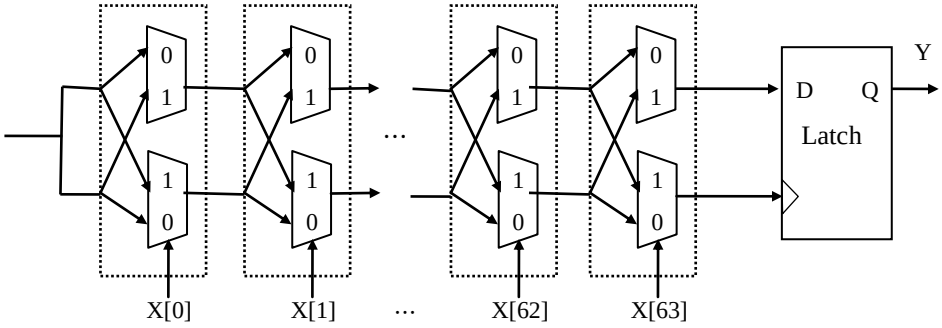

Figure 1. Arbiter-based PUF

A sequence of PUF provides a simple and stable authentication mechanism. As shown in Figure 2, firstly, we put the challenge/response sequence which is extracted from the chip into the database. This work is usually completed in a safe environment where the chip has just been produced. Secondly, when checking the chip, we can take one challenge from the database and send it to the chip as input of its PUF circuit. Then we compare the output to the response stored in the database. If they are equal, the chip through verification [6].

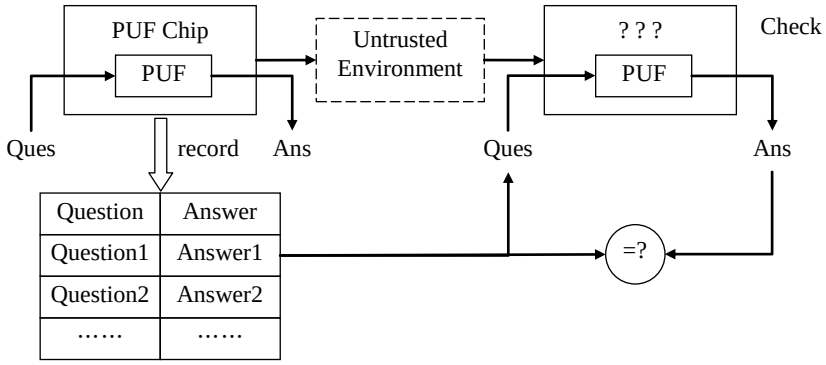

Figure 2. PUF Authentication Process

2.2. Proposed Protocols

(1) Protocol of Kulseng

In this protocol [7], several one-time secrets are used to prevent attacks. The new and previous owners are assumed to be able to exchange information offline without any security violations. The author proposed two ownership transfer protocols, where each tag may communicate with one or more readers.

The first protocol assumes the existence of a trusted authority by both the reader and the tags, called the Trusted Third Party (TTP). But this protocol is vulnerable to DOS attack by an adversary who blocks the message from tag to the new owner. When this happens, the tag has the updated secrets whereas the TTP has the previous secrets. The new owner has neither. The new owner can then, possibly, attempt the process again. However, this time, the tag would be unable to understand the messages from the new owner since it has updated its previous secrets. This leads to de-synchronization, and the adversary is able to accomplish a DoS attack [8].

The second protocol involves no third party, instead, asymmetric communication between the reader and the tag is assumed. However, we find that the privacy of the tag can be elaborated by the old owner. Assume that this protocol transfers the legitimate tag

from the old owner to the new owner. After running the protocol successfully, the old owner can identify that any session between the reader and the legitimate tag [9].

(2) Protocol of Osaka

This protocol [10] attempts ownership transfer with the database as a trusted third party. It assumes that the new owner initially receives the tag secret key. The purpose here is to update the secret and provide information about the tag to the authenticated new owner. An adversary can intercept the first message from reader to tag, modify *NR*, and send (query, *NR*=0) to the tag. The tag then responds with H(fk(*ID*)) as long as k remains the same. This can be used to trace the tag. This protocol is also vulnerable to DoS attack which an adversary can accomplish by adding a small noise to the last message from reader to tag. Now, the tag updates the different secret from the database and reader. This leads to de-synchronization.

(3) Protocol of Dimitriou

Dimitriou proposed a protocol for ownership transfer without a TTP [11]. The assumption in this protocol is that the new owner updates the secret key in a private environment where adversaries are assumed to be absent. This assumption is questionable since if it were valid, there is no need to encrypt any of the messages between tag and reader in such 'private' environments. This protocol is clearly vulnerable to eavesdropping attacks by previous owner since the current key is known. Once communication between tag and new owner are recorded, it's a trivial matter for previous owner to obtain the new secret key. The protocol is also vulnerable to DoS attacks. An adversary intending to cause de-synchronization between tag and reader can just block the last message from reader to tag. Now, the reader has the updated key while the tag has the previous key. This is an implementation issue that can be easily prevented by the reader storing the previous key in addition to the current key.

3. Ownership Transfer Protocol

In this section, we proposed a protocol LOTPP (The Lightweight Ownership Transfer Protocol using PUF), which combines the feature of PUF circuit and the requirement of low-cost.

In this protocol, the old owner uses its reader to authenticate the tag at first. The purpose is to find the tag and check its legality. In the second step, the old owner transmits the information about the tag through the cable channel which is stored in the database. The final step, the new owner and the tag launch a mutual authentication protocol to exchange information. The relationship between the entities was described in Figure 3.

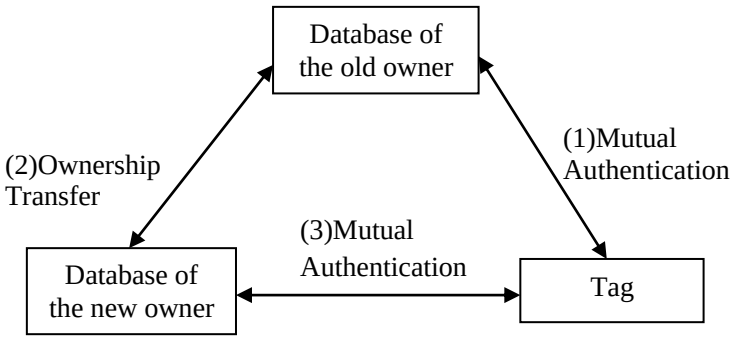

Figure 3. Relationship of the Entities

To distinguish by the general authentication protocol, we set a status bit f in the memory of tag. When f=1, it means the tag has accepted the ownership transfer query and will execute a ownership transfer protocol. However, when f=0, it means the tag is in a normal state. Detailed process of LOTPP is described as follows.

According to the above analysis, LOTPP can be divided into three phases: mutual authentication between the old owner and the tag, Ownership transfer phase, and finally to Mutual authentication phase between the new owner and the tag.

3.1 Mutual Authentication Between the Old Owner and theTag

Before ownership transfer, LOTPP allows the original owner and tag to do a mutual authentication. In this session, the old reader send a query command *OTHello*, which means the tag will change the status bit *f* to 1 at the end of the mutual authentication. Meanwhile, the old reader can verify the legitimacy of the target tag, and the specific steps are as follows:

(1) Tag identification

The reader sends a query command *OTHello* to the tag, and receives *IDS* from it. Then the reader uses the received *IDS* to search the corresponding item of this tag in the back-end database. If the received *IDS* do not match with IDS^{new}, it means the data from database has been updated while the data from tags has not been updated due to some reason. In this case, the *IDS* must match the IDS^{old}, and the reader will use the item of last session $\{IDS^{old}, G_n^{old}, G_{n+1}^{old}, IDT\}$ to authenticate the tag.

(2)Mutual authentication

The reader first generates a 96-bit random number n, combining with pseudonym (*IDS*) and keys (G_n,G_{n+1}), according to the equation (1) and (2) to calculate *A* and *B* separately. The tag makes *B* left rotate *n* mod(48)+1 times, then using this result XOR *B* to calculate the value of *X*. The reader sends the *A*||*X* to the tag.

$$A = IDS \oplus G_{n+1} \oplus n \tag{1}$$

$$B = (G_n \oplus n) + G_{n+1} \tag{2}$$

After receiving the message, according to the equation (4) and (5), the tag calculates the values of *M* and *N*. Then the tag extracts the value of random number *n'* according to the equation (6).

$$X = \mathrm{Rot}(B, n) \oplus B \tag{3}$$

$$M = \mathrm{PUF}(G_n) \tag{4}$$

$$N = \mathrm{PUF}(M) \tag{5}$$

$$n' = A \oplus IDS \oplus M \tag{6}$$

The tag uses G_n, *n'* and *M*' to generate *B*, then makes *B*' left rotate n' mod(48)+1 times, then using this result XOR *B*' to calculate the value of *X'*. Compare *X* to *X'*, if they are unequal, the tag will send a Fail signal to the reader, and if they are equal, the protocol is terminated to combine equation (9), (10) and (11) to calculate the value of *D* and *Y*. The tag sends *D*||*Y* to the reader.

$$B' = (G_n \oplus n') + M \tag{7}$$

$$X' = \mathrm{Rot}(B', n') \oplus B' \tag{8}$$

$$D = M \oplus N \oplus n' \tag{9}$$

$$C = (IDT + IDS) \oplus (n' + N) \tag{10}$$

$$Y = \mathrm{Rot}(C, n') \oplus C \tag{11}$$

After receiving the message, the reader generates *N'*, *C'* and *Y'* according to the equation (12), (13) and (14) respectively. Compare *Y* to *Y'*, if they are equal, the mutual authentication is successful. The reader will update pseudonym and keys, then send a

Success signal to the tag. If they are unequal, the reader will send a *Fail* signal to the tag to not update any data, the protocol will be terminated.

$$N^{'} = D \oplus G_{n+1} \oplus n \tag{12}$$

$$C^{'} = (IDT + IDS\ ) \oplus (n + N^{'}) \tag{13}$$

$$Y^{'} = \mathrm{Rot}(C^{'}, n) \oplus C^{'} \tag{14}$$

(3) Update pseudonym and keys

The reader:

$$G_n^{old} = G_n, G_n^{new} = G_{n+1}, G_{n+1}^{old} = G_{n+1}, \quad G_{n+1}^{new} = N^{'},$$

$$IDS^{old} = IDS, \quad IDS^{new} = IDS + n + N^{'}.$$

The tag:

$$G_n = G_{n+1}, G_{n+1} = N, IDS = IDS + n^{'} + N.$$

3.2. Ownership Transfer Phase

After the success of first phase, the status of the tag has changed. The old owner will send the information about the tag that is stored in its back-end database to the new owner through cable channel. Of course, the servers also need to authenticate each other. But the server authentication technology is relatively mature, now we can use a symmetric key authentication and digital signature to accomplish security authentication. After successful mutual authentication between the two servers, we could transfer the data. Transfer is completed, the new server creates a new random number, which will be used to update the pseudonym and keys at the end of the protocol, making the old owner do not know the new pseudonym and keys any more. In order to facilitate the graphic display, we consider the old server and the reader as a whole with S_{old}, and the new owner can use S_{new} represented. Detailed steps of ownership transfer protocol are described as follows.

(1) S_{old} uses its reader to send a query command Hello to the tag. Because the status bit *f* of the tag is changed to 1, the tag executives the ownership transfer protocol. And the tag responds its pseudonym IDS to S_{old}.

(2) After receiving the message, S_{old} uses the IDS to search the corresponding information {*IDS*,G_n,G_{n+1},*IDT*} in the old back-end database. Then the searched item is sent to the new server of the new owner.

(3) S_{new} receives the secrets about the tag, and then creates a new random number *R*. According to the equation (15), (16) and (17), S_{new} calculates the value of *E*, *F* and *U* respectively. At last the message *E*||*U* sent to S_{old}.

$$E = IDS \oplus R \tag{15}$$

$$F = G_n \oplus G_{n+1} \oplus R \tag{16}$$

$$U = \mathrm{Rot}(F, R) \oplus F \tag{17}$$

(4) S_{old} sends the message *E*||*U* to the tag.

(5) Firstly, the tag extracts the random number R' according the equation (18). Then U' is calculated using equation (19) and (20). The tag compares *U* to *U'*, if they are equal, received message has not be tampered by the attacker and it is correct. Then the tag generates *J*, *K* and *S* according to the equation (21), (22) and (23) respectively. Send the message *S*||*T* to S_{old} .

$$R^{'} = E \oplus IDS \tag{18}$$

$$F^{'} = G_n \oplus G_{n+1} \oplus R^{'} \tag{19}$$

$$U^{'} = \mathrm{Rot}(F^{'}, R^{'}) \oplus F^{'} \tag{20}$$

$$J = G_n \oplus R^{'} \tag{21}$$

$$K = \mathrm{PUF}(J) \tag{22}$$

$$S = J \oplus K \tag{23}$$

$$T = (IDT + IDS) \oplus (R^{'} + K) \tag{24}$$

(6) S_{old} sends the message $S||T$ to S_{new}.

(7) S_{new} receives $S||T$ and uses equation (25) and (26) to get K' and T', then compares T to T'. If they are equal, it means the received message is correct and credible. Then S_{new} updates its secrets and sends a Success signal to S_{old}. If U and U' are unequal, then S_{new} drops the received message, does not update the data in the database and sends a Fail signal to S_{old}.

$$K^{'} = S \oplus G_{n} \oplus R \tag{25}$$

$$T^{'} = (IDT + IDS) \oplus (R + K^{'}) \tag{26}$$

$$G_{n} = G_{n} \oplus R \tag{27}$$

$$G_{n+1} = K^{'} \tag{28}$$

$$IDS = IDS + R + K^{'} \tag{29}$$

(8) S_{old} sends the signal Success or Fail to the tag.

(9) If the tag receives the Success signal, then it updates the pseudonym and keys according the equation (30), (31) and (32), because S_{new} has updated the secrets successfully. At the same time, the tag set the status bit f to 0. If the received signal is Fail, the tag does not update any data but set the status bit to 1, because the authentication to S_{new} is fail in this session.

$$G_{n} = J \tag{30}$$

$$G_{n+1} = K \tag{31}$$

$$IDS = IDS + R^{'} + K \tag{32}$$

3.3. Mutual Authentication Between the New Owner and the Tag

The ownership transfer in the second phase has succeeded, at this time, S_{new} and the tag has shared a set of secrets. But S_{old} could not keep synchronization with the tag any longer. At this time, only the new owner can accomplish the mutual authentication with the tag.

4. Protocol Analysis

4.1 Security Analysis

(1)Resistance to modification

Environment of wireless channel is open and complex. Therefore, messages in the wireless channel are particularly vulnerable to the attacker's eavesdropping and tampering. In this protocol, we first use the circle Rotate Left function Rot(.,.) to F, and then make the result XOR with F. This operation can break the correlation between the data bites. If the attacker tampers with E, the corresponding value of U is different to guess. Thus tag or reader will drop the invalid messages, and the mutual authentications will not success.

(2)Tag anonymity and untraceability

The tag does not reveal its *ID* and uses pseudonym as the identity in LOTPP. The pseudonym *IDS* and the keys G_n and G_{n+1} will be updated after every successful protocol run and the update operation involves random numbers. So the tag is anonymous to the adversary. In addition, the messages involve random numbers and thus cannot be tracked either.

(3)Forward security

Even if a tag is compromised at last, the attacker cannot deduce any of the tag's previous interactions because during each session, freshly generated random numbers are used and each time a different input was used in the PUF function.

(4)Resistance to replay attack

All communicating messages between the reader and the tag always contain a dynamically generated fresh random number. If an adversary eavesdrops on transmitted message and replay these messages to the tag, the tag will recognize these messages as invalid ones and drop them, because the random number associated with each message is different every time. Hence, the adversary cannot execute replay attack successfully.

(5)Resistance to de-synchronization attack

In order to prevent the secret de-synchronization, the server stores both the old and the new secrets. Moreover, after verifying the correctness of the received messages, the tag will not update its secrets, because it is vulnerable to message block by attacker. So we do not rush to update the tag's secrets. At the last step of the protocol, the server will transmit a notice to tell whether the tag can update. If there is some problems in new server certification or the attacker intercept the message which is sent to the tag, the tag will not update its secret, and the ownership transfer protocol is not successful. The new owner still does not get the ownership of the tag. By this way, we ensure the synchronization of the protocol.

(6)Resistance to disclosure attack

The adversary may use modify-and-test method to guess the secrets. Such as, the adversary can modify the messages E, U, S and T to test the correctness of the modification. However, we have not found any trapdoor that can help to generate new valid messages without knowing the secrets.

(7)Resistance to tag cloning

The proposed protocol is based on PUF which is physically unclonable. The inclusion of a PUF circuit means that we can protect the tag from these physical attacks, since a PUF will alter its behavior if the hardware itself is altered. Therefore, cloning attack is impossible. Security comparison between LOTTP and some other protocols is listed in Table 1.

Table 1. Security Comparison

	Kulseng	Osaka	Dimitriou	LOTPP
Resistance to modification	Y	N	Y	Y
Untraceability	N	N	Y	Y
Anonymity	N	Y	Y	Y
Forward security	N	Y	N	Y
Resistance to replay attack	N	Y	Y	Y
Resistance to de-synchronization	N	N	N	Y

4.2. Performance Analysis

(1)Computation overhead

In this protocol, we only use three simple bit operators. They are addition (+), XOR and left rotate (Rot (.,.)). Meanwhile, the calculation amount of PUF circuit which was described in the previous section is also very small. Compared with the protocols proposed by Osaka and Dimitriou, whose tag needs random number generator, hash

function and symmetric encryption. All these operations will occupy very large resources of the tag.

(2)Storage overhead

Each tag stores a static identifier (IDT) in the ROM in our protocol. Meanwhile, it needs a pseudonym (96 bits) and two keys (2 * 96 bits) to be stored in the RAM. So the dynamic storage space of the tag needs 288 (3 * 96) bits. The new server also needs to store a set of secrets related to tag, it occupies 384 (4 * 96) bits of space.

(3)Communication overhead

In order to achieve mutual authentication and ensure the integrity of the data, this protocol uses four communication messages. In the phase of tag identification, Query command Hello and pseudonym *IDS* are transmitted in the channel. In the phase of mutual authentication, the reader and the tag response to each other with $E||U$ and $S||T$. In the end of this protocol, the reader needs to send a notification signal, and the tag makes a judgment according to the received signal. Considering the robustness of the protocol, we use more messages back and forth to resist the de-synchronization attack. It is the point needs to be improved.

(4)Cost overhead

This protocol only uses some simple bit operations instead of encryption and random number generator in the process of certification. The PUF circuit is used to generate keys, which is a simple signal delay circuit and needs a limited resource. So this protocol meets the low-cost requirement of the tag.

Performance comparison between LOTTP and some other protocols is listed in Table 2.

Table 2. Performance Comparison

	Kulseng	Osaka	Dimitriou	LOTPP
Operation	⊕ ,PUF	⊕ ,RAND,Hash,E_K	RAND,F_K, ⊕	+,Rot,PUF, ⊕
Storge space (L=96bits)	5L	2L	2L	4L
Communication times	4	5	3	5

5. Formal Analysis Using Ban Logic

As a kind of formal analysis methods, BAN logic [12][13][14] can not only discover the current attack in cryptographic protocols, but also find out flaws comprehensively and profoundly. In this section, we prove the correctness of the proposed protocol based on BAN logic. Specifically, the correctness means that after the protocol execution, the communication parties, tag and the new server believe that they are sharing two fresh secrets; meanwhile, tag and the old server don't share secrets any more.

In the forthcoming description, we consider the reader and the server as an entirety, using S_{old} and S_{new} represent the new owner and the old owner, respectively. Proof process of LOTPP is as follows [15].

5.1. Formalized Protocol

The conventional notations of the generic type of protocol are not convenient for manipulation on logic. In this part, we, at first, simplify the protocol and describe it as a generic type. Then, we formalize the generic type of the protocol for verification goals as shown in Table 3.

Table 3. Generic Type of Protocol

Protocol Generic Type:	**Formalized Protocol:**
m1: $S_{old} \rightarrow T$: Hello	***m3***: $S_{new} \triangle \{IDS, G_n, G_{n+1}, IDT\}$
m2: $T \rightarrow S_{old}$: IDS	***m4***: $T \triangle E \| U$
m3: $S_{old} \rightarrow S_{new}$: $\{IDS, G_n, G_{n+1}, IDT\}$	***m5***: $S_{new} \triangle S \| T$
m4: $S_{new} \rightarrow S_{old} \rightarrow T$: $E \| U$	
m5: $T \rightarrow S_{old} \rightarrow S_{new}$: $S \| T$	

Table 4. Goals of the Proof

Goal 1: $S_{new} \| \equiv T \xleftrightarrow{G_n, G_{n+1}} S_{new}$
Goal 2: $T \| \equiv S_{new} \xleftrightarrow{G_n, G_{n+1}} T$
Goal 3: $T \| \equiv S_{new} \sim \#(R)$
Goal 4: $S_{new} \| \equiv T \sim \#(IDT)$

Table 5. Initial Assumptions

(1) $S_{old} \| \equiv S_{old} \xleftrightarrow{G_n, G_{n+1}} T$ (G_{n+1}=PUF(G_n))
(2) $S_{old} \| \equiv S_{new}$
(3) $S_{new} \| \equiv \#(R)$
(4) $S_{new} \| \equiv S_{old}$
(5) $T \| \equiv T \xleftrightarrow{G_n, G_{n+1}} S_{old}$ (G_{n+1}=PUF(G_n))
(6) $T \| \equiv \#(IDT)$

5.2. Proof Goals and Assumption

The proof goals of correctness are shown in Table 4. The first two goals, (1) and (2), are about shared keys between the tag and the new owner. The goal (3) and (4) are for the shared secrets. Those beliefs are to state that the tag and the new owner shared secrets each other exchanged fresh messages.

Table 5 shows the initial assumption for our protocol. Assumption (1) and (5) are for two fresh shared keys, G_n and $G_{n+1,}$ between the old owner and the tag. Assumption (2) and (4) are based on the assumptions that the channels between reader and server, server and server are safe. Assumption (3) means the new reader has a *Random Number Generator*, which can product a random number R. Assumption (6) means that the identifier *IDT* is embedded into the tag.

5.3. Verification

Protocol Analysis: The logic rules and the assumptions will be used by the messages in the first phase to discover the final beliefs held by the parties in the protocol. If the final beliefs contain the goals of the protocol, the protocol is integrated; else, the protocol has flaws. The proof is as follows:

(1)*Prove:* $S_{new} \| \equiv T \xleftrightarrow{G_n, G_{n+1}} S_{new}$, $T \| \equiv S_{new} \xleftrightarrow{G_n, G_{n+1}} T$

By Assumption (4) : $S_{new} \| \equiv S_{old}$ & *m*3: $S_{old} \rightarrow S_{new}$: $\{IDS, G_n, G_{n+1}, IDT\}$

Get: $S_{old} \| \equiv \{IDS, G_n, G_{n+1}, IDT\}$

By rule: $P \| \equiv Q \| \Rightarrow X$, $P \| \equiv Q \| \equiv X \vdash P \| \equiv X$

Get: $S_{new} \| \equiv S_{old} \| \Rightarrow \{IDS, G_n, G_{n+1}, IDT\}, S_{new} \| \equiv S_{old} \| \equiv \{IDS, G_n, G_{n+1}, IDT\} \vdash S_{new} \| \equiv \{IDS, G_n, G_{n+1}, IDT\}$ (33)

By Assumption (1) : $S_{old} \| \equiv S_{old} \xleftrightarrow{G_n, G_{n+1}} T$ (G_{n+1}=PUF(G_n)) & *Assumption* (5): $T \| \equiv T \xleftrightarrow{G_n, G_{n+1}} S_{old}$ (G_{n+1}=PUF(G_n)) & *Equation* (33): $S_{new} \| \equiv \{IDS, G_n, G_{n+1}, IDT\}$

Get: $S_{new} \| \equiv T \xleftrightarrow{G_n, G_{n+1}} S_{new}$, $T \| \equiv S_{new} \xleftrightarrow{G_n, G_{n+1}} T$

So we can obtain *Goal* (1) and (2).

(2)*Prove*: $T \| \equiv S_{new} \sim \#(R)$

By Assumption (3) : $S_{new} \| \equiv S_{old}$ & Rule: $P \| \equiv \sharp(X) \vdash P \| \equiv \sharp(X,Y)$

Get: $S_{new} \| \equiv \#(R) \vdash S_{new} \| \equiv \#(R, IDS)$ (34)

*By m*4: $T \triangle E \| U$ ($E = \{R, IDS\}_{G_n, G_{n+1}}$) & *Prove*(1): $T \| \equiv S_{new} \xleftrightarrow{G_n, G_{n+1}} T$ & *Rule*:

$P|\equiv Q \xleftrightarrow{k} P,\ P \vartriangleleft \{X\}_K \vdash P|\equiv Q|\sim X$

Get: $T|\equiv S_{new} \xleftrightarrow{G_n, G_{n+1}} T,\ T \vartriangleleft \{R, IDS\}_{G_n, G_{n+1}} \vdash T|\equiv S_{new}|\sim (R, IDS)$

By Rule: $P|\equiv Q|\sim (X,Y) \vdash P|\equiv Q|\sim X$

Get: $T|\equiv S_{new}|\sim (R,IDS) \vdash T|\equiv S_{new}\sim R$ (35)

By Equation (34): $S_{new}|\equiv \#(R) \vdash S_{new}|\equiv \#(R,IDS)$ & *Equation* (35): $T|\equiv S_{new}\sim R$

Get: $T|\equiv S_{new}\sim \#(R)$

So we can obtain *Goal* (3) .

(3)Prove: $S_{new}|\equiv T\sim \#(IDT)$

By *Assumption* (6) , *Get*: $T|\equiv \#(IDT)$ (36)

By *m*5: $S_{new} \vartriangleleft S \| T\ (S = \{G_{n+2}\}_{G_n, R})$ & *Rule*: $P|\equiv Q \xleftrightarrow{k} P,\ P \vartriangleleft \{X\}_K \vdash P|\equiv Q|\sim X$

Get: $S_{new}|\equiv T \xleftrightarrow{R, G_n} S_{new},\ S_{new} \vartriangleleft \{G_{n+2}\}_{G_n, R} \vdash S_{new}|\equiv T|\sim G_{n+2}$

By Rule: $P|\equiv Q|\sim (X,Y) \vdash P|\equiv Q|\sim X$ & $R \vartriangleleft \{R, IDS, IDT\}_{G_{n+2}}$

Get: $S_{new}|\equiv T \xleftrightarrow{G_{n+2}} S_{new},\ S_{new} \vartriangleleft \{R, IDS, IDT\}_{G_{n+2}} \vdash S_{new}|\equiv T|\sim (R,IDT,IDS)$ (37)

By Rule: $P|\equiv Q|\sim (X,Y) \vdash P|\equiv Q|\sim X$

Get: $S_{new}|\equiv T|\sim (R,IDT,IDS) \vdash S_{new}|\equiv T|\sim IDT$

By Equation (36): $T|\equiv \#(IDT)$ & *Equation* (37): $S_{new}|\equiv T|\sim (R,IDT,IDS)$

Get: $S_{new}|\equiv T\sim \#(IDT)$

So we can obtain *Goal* (4) .

6. Conclusion

In supply chain management, RFID system has to handle ownership transfer issue for consumer products because the ownership of a product item will be changed many times in its life cycle. In this paper, we have presented our ownership transfer protocol that enables perfect RFID tags ownership transfer. Moreover, we give the proof of protocol correctness based on BAN logic. The merit of our protocol is that it does not only well protect owner's privacy but also achieves high-security and high-efficiency.

Acknowledgments

This work was financially supported by the National Natural Science Foundation of P. R. China (No.61572260, No.61373017, No.61572261), Scientific & Technological Support Project of Jiangsu Province (No. BE2015702), Natural Science Foundation of Jiangsu Province, China (Grant No. BK20140886, No. BK20140888), China Postdoctoral Science Foundation (Grant No. 2014M561696, No. 2014M551636), Natural Science Foundation of the Jiangsu Higher Education Institutions of China (Grant No. 14KJB520030), Jiangsu Planned Projects for Postdoctoral Research Funds (Grant No. 1401005B, No. 1302090B), Jiangsu Postgraduate Scientific Research and Innovation Projects (SJLX15_0381) and NUPTSF (Grant No. NY213034, No. NY214060 and No. NY214061).

References

[1] L. Gao, M. Ma, Y. Shu and Y. Wei, "A security protocol resistant to intermittent position trace attacks and desynchronization attacks in RFID systems", Wireless personal communications, vol.68,no.4,(2013), pp. 1943-1959.

[2] D. R. Thompson, J. Di and M. K. Daugherty, "Teaching RFID Information Systems Security", IEEE Transactions on Education, vol.57, no.1, (2014), pp.42-47.

[3] Y. Q. Gui and J. Zhang. "A new authentication RFID protocol with ownership transfer", ICT Convergence (ICTC), 2013 International Conference on IEEE, Jeju Island,Korea,(2013) October 359 - 364.
[4] S. W. Jung and S. Jung,"HRP: A HMAC-based RFID mutual authentication protocol using PUF", Information Networking (ICOIN), 2013 International Conference on IEEE, Bangkok, Tailand,(2013) January 578-582.
[5] W. Choi, S. Kim, Y. Kim, Y. Park, and K. Ahn, "PUF-based Encryption Processor for the RFID Systems", Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on IEEE, Bradford, UK, (2010) June 2323-2328.
[6] X. J. Zhang, W. Q. Cai and S. P. Wang, "One Anti-Collision Algorithm Based on Improved Adaptive Multi-Tree Search", Acta Electronica Sinica, vol.40,no.1,(2012)pp.193-198.
[7] L. Kulseng, Z. Yu, Y. Wei and Y. Guan, "Lightweight mutual authentication and ownership transfer for RFID systems". INFOCOM, Proceedings IEEE, San Diego,USA,(2010) March 1-5.
[8] G. Kapoor and S. Piramuthu, "Vulnerabilities in some recently proposed RFID ownership transfer protocols", IEEE 1st International Conference on Networks and Communications (NETCOM'09), Chennai, India, (2009) December 354-357.
[9] S. Kardas, M. Akgun, M. S. Kiraz and H. Demirci, "Cryptanalysis of lightweight mutual authentication and ownership transfer for RFID systems", IEEE Workshop on Lightweight Security & Privacy: Devices, Protocols and Applications (LightSec), Istanbul, Turkey, (2011) March 20-25.
[10] K. Osaka, T. Takagi, K. Yamazaki and O. Takahashi, "An efficient and secure RFID security method with ownership transfer", IEEE International Conference on Computational Intelligence and Security, Guangzhou , China, (2006)November 1090-1095.
[11] T. Dimitriou, "RFIDDOT: RFID Delegation and Ownership Transfer Made Simple", Proceedings of the 4th International Conference on Security and Privacy in Communication Networks, Istanbul, Turkey, (2008) September: 1-8.
[12] K. Liu, J. Ye and Y. Wang, "The Security Analysis on Otway-Rees Protocol Based on BAN Logic", IEEE 4th International Conference on Computational and Information Sciences (ICCIS), Chongqing, China,(2012) August 341-344.
[13] W. Alhakami, A. Mansour, G. Safdar and S. Albermany, "A Secure MAC Protocol for Cognitive Radio Networks (SMCRN)", IEEE Science and Information Conference (SAI), London,UK,(2013) October 796-803.
[14] J. Meng and Z. Wang, "A RFID Security Protocol Based on Hash Chain and Three-Way Handshake[C]. IEEE 5th International Conference on Computational and Information Sciences (ICCIS), Shiyan, China (2013) June 1463-1466.
[15] X. J. Zhang, Y. Wang, S. P. Wang and Z. X. Sun, "Reach on the Cyclic Shift Lightweight Mutual Authentication Protocol", Acta Electronica Sinica, vol.40,no.11,(2012) pp2270-2275.

Authors

Xuejun Zhang, He received the Ph.D. degree from the Nanjing University of Posts and Telecommunications, Nanjing, in 2011 in communication and information systems. He currently is a professor of College of Electronic Science and Engineering. He serves as the director of Engineering Training Center. His research interests are in the area of wireless communications, including RFID, Cognitive Radio and Wireless Sensor Networks.

Wanlu Huang, She is the master student in the College of Electronic Science and Engineering, Nanjing University of Posts and Telecommunications, Nanjing. Her research interests mainly focus on RFID and EEG signal analysis.

He Xu, He was born in Anhui Province, China, in 1985. He received the Ph.D. degree from the Nanjing University of Posts and Telecommunications, Nanjing, in 2012 in information network. He currently is an associate professor of School of Computer Science & Technology. His research interests include information security, and Internet of things.

Yu Wang, She received the M.S. degree in the College of Electronic Science and Engineering, Nanjing University of Posts and Telecommunications, Nanjing, in 2014 in circuit and system. Her research interests mainly focus on RFID and wireless communication security.