International Journal of Future Generation Communication and Networking
Vol. 9, No. 8 (2016), pp. 199-212
http://dx.doi.org/10.14257/ijfgcn.2016.9.8.20

# Untraceable Analysis of Lightweight RFID Ownership Transfer Protocol

Xiuqing Chen[1], Tianjie Cao[2,*], Jingxuan Zhai[2] and Yu Guo[3]

[1]*School of Medicine Information, Xuzhou Medical College, Xuzhou, Jiangsu 221000, P.R. China*
[2]*School of Computer, China University of Mining and Technology, Xuzhou, China*
[3]*School of Mine, China University of mining and technology, Xuzhou 221116, China;*
[*]*corresponding author E-mail {xiuqingchen@126.com}*

***Abstract***

*Nowaday, the worldwide applications of RFID technologies have contributed to the development of supply chain system. However, to be confronted with various security and privacy issues in lightweight RFID protocols, the security analysis of lightweight ownership transfer (OT) protocols has become an important task. First and foremost, the passive attacker can further break down the security and privacy of three new published lightweight RFID protocols. Subsequently, the proposed protocol is designed to prevent the traceability attacks. Then, it is significant for us to show how to prove strong forward untraceable and backward untraceable of the improved scheme in security model.*

***Keywords****: RFID technology, Lightweight RFID protocol, Strong forward untraceable, Backward untraceable, Security model.*

## 1. Introduction

Radio Frequency Identification (RFID) technologies have been extensively deployed in the ubiquitous domain, such as construction object management [1]. The design of a RFID ownership transfer protocol is especially significant in various application fields. To implement quick service and transfer ownership in the service industry and supply chain system is a significant tendency via wireless and mobile reader. Adoption of the lightweight RFID protocol based on the hash function gives a lot of benefits. As a part of the RFID system, mobile devices and low-cost tags have been employed in supply chain systems due to its outstanding mobility. Nevertheless, the messages between tags and mobile devices can be disclosed, casing many privacy and security problems.

In an attack scenario, the passive adversary monitors and traces the messages in insecure channels. Except for the channel between a genuine card/tag and a mobile reader, there is another channel between a mobile reader and the sever (DB) for the assumptions of insecurity channels. Moreover, there are supermarket scenarios where the adversary has the objective to steal more expensive goods rebadged by counterfeited and cheap tags (brands) and deceive the salesclerk.

In order to protect effectively the privacy and security of ownership transfer processes, we develop a lightweight ownership transfer protocol for the supply chain system. It is common for RFID-tagged goods to be owned by another legal owner, when the goods are sold and ownership transfer from the old owner to the new owner. The proposed protocol addresses the dynamic associated with the supermarket and supply chain system scenarios.

ISSN: 2233-7857 IJFGCN

The main objectives of this paper are to critically analyze all known attacks on three published protocols. To the best of our knowledge, there are no attacks presented on Dhal *et al.*'s protocol [2], provably lightweight RFID mutual authentication protocol (PLAP) [3], and Kapoor *et al.*'s protocol [4]. Next, we analyze three protocols from the point of security view.

The article is organized as follows: we present an introduction in Section 2 and set up related work in Section 2. Then we describe the adversary model in Section 3. We demonstrate that two different protocols suffer from key disclosure attack, tag tracing attack and tag impersonation attack in Section 4. Afterwards, we propose lightweight ownership transfer protocol, prove that OT protocol is forward/backward untraceable, and give a detailed performance and security analysis in Section 5. At last, we sum up in Section 6.

## 2. Related Work

There is a wide variety of applications in supply chain system where secure and efficient authentication mechanisms are demanded. Even though current passive RFID tags have rather limited on-chip capabilities, they support some cryptographic functions, especially lightweight ones that have been recently developed for this type of applications.

Over the past four years, the high-performance OT protocols proposed by Kapoor and Piramuthu [4] were high-cost on account of adopting a function encrypted with the key and an encrypted function. However, it is difficult to apply their protocols in the low cost setting. The low-cost RFID technologies adopt lightweight cryptographic methods such as pseudo random number generators (PRNGs), hash functions and bitwise operations on passive tags. Although these new published protocols [5-7] conform to EPC Class 1 Generation 2 standard (EPC C1G2), they are still vulnerable to various attacks at different privacy levels. Tag impersonation attack in protocols [8-10] is a forgery attack that the forged messages of the tag should be verified by a legal reader. In addition, the attack approach of tracing tag is introduced in [10], where the attacker can take steps to run the protocol and obtain effective information of the target tag. If an adversary knows the related keys of the target tag by analyzing the obtained information, then s/he can trace the location of the valid tag. Besides, if the legal tag is distinguished by the malice attacker, the privacy of the tag is revealed.

On the one hand, the OT protocol is a key-update protocol. But on the other hand the existing OT protocol is lack of security and privacy proof in terms of forward untraceable and backward untraceable. As aforementioned, the key-update protocols such as the NRS protocol [11], FSA protocol [12] and LPP protocol [13] suffer from forward traceability attacks. As it will be made more explicit in the paper, such a security model is necessary for the efficiency and security of the RFID system, especially for the security proofs to hold. Various formal privacy models [14-17] are utilized to prove safety property of RFID protocol via different oracles. In this paper, we practically adopt the existing formal privacy model [18]. Thus, we enhance the model to analyze forward/backward traceability of the improved protocol with different assumptions.

Moreover, we demonstrate that the protocols [2-4] suffer from various attacks and lack untraceable analysis. Extant RFID ownership transfer literatures do not adequately achieve the strong privacy performance. Therefore, the purpose of this paper is to fill this gap, propose lightweight ownership transfer protocol, and prove the privacy property using the security model.

## 3. Adversarial Model

For simplicity, these notations through this paper have been explained in Table 1.

**Table 1. The Notations and the Related Descriptions**

| Notations | Descriptions |
|---|---|
| $Adv^+$ | The wise adversary |
| $R_1$, $R_2$ | The old (new) mobile reader |
| $s_1$,$s_2$ | Shared keys between tag and $R_1$($R_2$) |
| $N_k$ | Random number (the $k^{th}$ bit is 1, other bits are 0) |
| $N_T$,$N'_T$,$N_{R1}$ | Random numbers |
| $N$,$N'$ | Nonces,$N= N_T \oplus N_{R1}$,$N' = N_T \oplus N_{R1} \oplus N_k$ |
| $f_s(\ )$, $H_s(\ )$ | Encrypted Hash function |
| $H(\ )$ | Hash function |
| $TID$ | The tag identication number |
| $K_i$, $P_i$ | The $i^{th}$ authentication key(access key) in tag |
| $C_i$ | The index stored in tag to find the record in DB |
| $N_T$ ($N_R$) | The nonce generated by tag (reader) |
| $A \rightarrow B$ | $A$ forwards a message to $B$ |
| $A_X$ | The $x$ state of message $A$ |

In this paper, we enhance the existing formal privacy Vaudenay model [18]. Thus, we practically define strong forward untraceable and backward untraceable based on Vaudenay model. These definitions are utilized to prove safety property of the improved RFID OT protocol via the different oracles.

The revised Vaudenay's model has strong realistic application background, since the wireless communication between the wireless reader and DB is unsecure, and a portable reader may be stolen and tracked in supermarket setting and supply chain system. Therefore, **SendReader-DB** should be added to identify that the reader whether or not is counterfeited. Then, the following nine queries are illustrated for privacy proof.

• **CreateTag$^x$($ID$)** generates the **free** tag with sole identifier $TID$ with **SetupTag($ID_i$)** oracle ( $1 \leq i \leq n$ ). When $x$ = 1($x$ =0), $T$ is legitimate (illegality).

• **DrawTag()** randomly sends one $T$ from the **free** tags to the ***drawn*** tags as ***vtag***.

• **Free(*vtag*)** returns the ***vtag*** to the set of free tags.

• **SendReader-Tag ($m_R$,$\pi$)** $\rightarrow m'_R$. The reader transmits message $m_R$ to tag and obtains the response $m'_R$.

• **SendReader-DB ($m_D$,$\pi$)** $\rightarrow m'_D$. The reader sends message $m_D$ from reader to DB in the session π.

• **SendTag ($m$, *vtag*)** $\rightarrow m'$. The tag transmits the message $m$ and responds with $m'$.

• **Result(π)** returns 1 when protocol instance π is successfully completed or 0 otherwise.

• **Corrupt(*vtag*)** $\rightarrow$S. The attacker returns the tag's keys**.**

The privacy game contains learning and attack phases. As long as the game's outcome is true with negligible probability, ownership transfer protocol will possess untraceable privacy and the wise adversary's attack will be trivial.

**Definition 1** (Privacy). A wise adversary $Adv^+$ ($A^+$) is able to only call all oracles. Then, if $\left|Pr(Exp_S^A \ succeeds) - Pr(Exp_S^{A^+} \ succeeds)\right|$ is negligible, $A^+$ is called trivial $A^+$ depending on the related adversary class. The strong forward untraceable depends on success probability of wise adversary $A^+$.

**Definition 2** (Strong forward untraceable). It should be impossible for the wise adversary $Adv^+$ to trace the tag at the round $i'$ that $i' \geq i + 1$, even though $Adv^+$ corrupted the $i^{th}$ round keys of the target tag.

**Definition 3** (Backward untraceable) [18]. Even if the wise adversary $Adv^+$ corrupted the $i^{th}$ round keys of the target tag, s/he could not trace the target tag's transactions that occur at the past round $i'$ ($i'<i$-1). Finally, the experiments $Exp_{S,A}^{Backward}(k)$ succeed as long as $Adv^+$ returns true.

## 4. Cryptanalysis of Two Protocols

### 4.1. Cryptanalysis of Kapoor *et al.*'s Protocol

We assume that two owners are mobile devices, such as the old (current) mobile owner ($R_1$) and the new mobile owner ($R_2$). There are three following unreasonable assumptions without considering many insecurity factors of mobile devices in the monitoring actions using the notations in Table 1. Since the first assumption is the absence of any other reader in the vicinity of $R_2$, there is no need to encrypt any messages [19]. Moreover, the other two assumptions are secure communication from $R_2$ to $R_1$ and secure channels between $R_2$ and $R_1$ [20].

If these assumptions were unreasonable, it is necessary to reconsider the safety performances of protocols under the realistic and reasonable assumptions. Our assumption is developed from the application background of mobile devices. Channels between two entities are unsecure (i.e., $R_1$ to $R_2$). An attacker can monitor the messages from $R_1$ to $R_2$ and from $T_i$ to $R_2$ in Figure 1. Then s/he continues with the following attack steps to pass the verification of $R_2$.

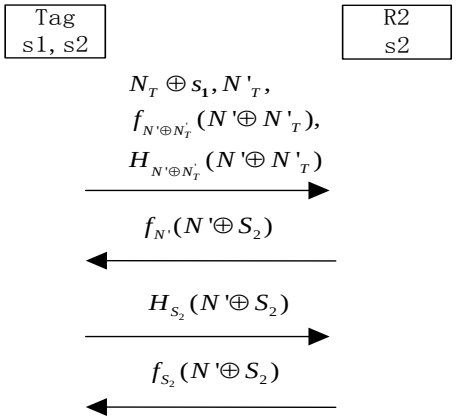



**Figure 1. OT protocol without TTP: Key change**

Kapoor *et al.*'s proposed lightweight ownership transfer protocols without a trusted third party (TTP) [4]. They claimed that the presented protocols have achieved forward security. However, we point out an impersonation attack in protocols under our assumption. After a while, we describe the attack process as follows:

- **Phase 1 (Learning)**: An adversary can monitor the initialization information between $R_1$ and $R_2$ and a legitimate authentication between $R_2$ and $T_i$, and record the second messages.

**Step 1**. $R_1$ creats a nonce $N_{R1}$, transmits $f_{s1}(N_{R1} \oplus s_1)$ to $T_i$ and $N_{RI} \oplus s_1$ on an unsecure channel to $R_2$ in the initialization phase.

**Step 2.1**. $T_i$ extracts $N_{R1}$ from $N_{RI} \oplus s_1$, generates $N_T$ and computes $N = N_{RI} \oplus N_T$.

**Step 2.2**. $T_i$ computes $N'= N \oplus N_k$ (the i[th] bit is 1, the rest are 0), creats $N'_T$ and sends the encrypted messages (i.e. $N_T \oplus s_1$, $N'_T$, $f_{N' \oplus N'T}(N' \oplus N'_T)$, $H_{N' \oplus N'T}(N' \oplus N'_T)$) to $R_2$.
**Step 3**.The attacker $Adv^+$ blocks the step from $T_i$ to $R_2$ and stores the transmitted messages from the above steps.
**Phase 2 (Impersonation)**: To impersonate the tag $T_i$, $Adv^+$ freely modifies the transmitted messages ($N_T \oplus s_1 \oplus N_k$, $N'_T \oplus N_k$) using random number $N_k$.
**Step 1**. $Adv^+$ modifies the two messages (i.e. $N_T \oplus s_1 \oplus N_k$, $N'_T \oplus N_k$), and computes other messages as follows:

*a*.$N'_{Adv} = N' \oplus N_k$
*b*.$N'_{T\,Adv} = N'_T \oplus N_k$
c.$N'_{Adv} \oplus N'_{T\,Adv} = N' \oplus N_k \oplus N'_T \oplus N_k = N' \oplus N'_T$
d. $f_{Adv} = f_{N' \oplus N'T}(N' \oplus N'_T)$
e. $H_{Adv} = H_{N' \oplus N'T}(N' \oplus N'_T)$

$Adv^+$ sends the messages (i.e. $N_T \oplus s_1 \oplus N_k$, $N'_T \oplus N_k$, $f_{N' \oplus N'T}(N' \oplus N'_T)$, $H_{N' \oplus N'T}(N' \oplus N'_T)$) from tag to reader.
**Step 2**. (Reader verification)

$R_2$ picks up the received $s_1 \oplus N_T$ and $N_T \oplus s_1 \oplus N_k$ sequentially, computes $N_R = s_1 \oplus N_{R1} \oplus N_T \oplus s_1 \oplus N_k = N \oplus N_k$. $R_2$ uses a brute force technique to determine $N'_R = N' \oplus N_k$ using to decrypt the value sent in $f$ . Since $R_2$ can accomplish that $N'_R = N \oplus N_k$, $R_2$ verifies the solution obtained using the hashed value and compares the computed $H_R$ with the received $H$ to authenticate the tag. If the value calculated $H_R$ is equal to $H$, $R_2$ verifies the illegitimate data as a legal tag. The following equations are given to prove the computational process.

a. $N'_R \oplus N'_{Tadv} = N' \oplus N_k \oplus N'_T \oplus N_k = N' \oplus N'_T$
b. $H_R = H_{N'R \oplus N'Tadv}(N'_R \oplus N'_{Tadv}) = H = H_{N' \oplus N'T}(N' \oplus N'_T)$

From the above compared results, the tag impersonation attack succeeds.

### 4.2. Cryptanalysis of PLAP protocol

In this section, tag impersonation attack aimed to PLAP protocol [3]. The vulnerabilities of the protocol arise because key updating mechanism does not have any contribution in the randomness of the updated keys. The direct way to improve the key updating mechanism is using the hash function and PRNG function.

We analyze the security of PLAP scheme to find out whether it meets the desired requirement. However, we propose two types of attacks against this protocol, such as tag impersonation attack and tracing attack.

(1) Tag impersonation attack

When two entities ($T_i$ and $R_i$) authenticate in communication, it is important for the wise attacker to handle the obtained messages and deceive $R_i$ to authenticate the forged data as a legal tag's information.

So the attacker $Adv^+$ should adopt the following effective phases to create the forged information and drive the illicit tag ($T'_i$) to be authenticated by the reader.
**Phase 1** (Learning): $Adv^+$ plays as a blocker and eavesdrops one successful run of protocol and blocks the **Step 3** Then stores the exchanged messages between $R_i$ and the legitimate tag $T_i$.
**Step 1**. $R_i$ generates a nonce $N_R$ and sends it to the $T_i$.
**Step 2**. $T_i$ generates $N_T$ and computes $A = H(N_R \oplus TID) \oplus K_i$, $B = N_T \oplus K_i$, $D = N_T \oplus H(C_i \oplus K_i)$, then sends { $A$, $B$, $C_i$, $D$ } to $R_i$.
**Step 3**. The attacker $Adv^+$ blocks and stores the information, then stops the session.
**Phase** 2 (Impersonation): To impersonate the tag $T_i$, $Adv^+$ initiates a new session of protocol.
**Step 1**. $Adv^+$ replays the monitored nonce $N_R$ to the $T_i$,

**Step 2**. $T_i$ produces $N'_T$ and computes $A'$, $B'$, $C'_i$, $D'$, then sends them to the $R_i$.
**Step 3**. $Adv^+$ blocks and modifies the transferred messages {$B'$, $D'$} using an arbitrary random number $F$ as follows:

a. $A_{Adv}= A$; b. $B_{Adv}= B' \oplus F$; c. $C_{i\,Adv}= C_i$; d. $D_{Adv}= D' \oplus F$

$Adv^+$ sends the messages { $A_{Adv}$, $B_{Adv}$, $C_{iAdv}$, $D_{Adv}$ } to DB.
**Step 4**.Upon receiving the message, the DB checks $C_i$ value. Then, the DB uses $C_i$ to pick up the related records sequentially and compares two computed values { $A_{cur}$, $D_{cur}$ } with the modified messages (received values) as follows:

a. $A_{cur}= H(N_R \oplus TID) \oplus K_i = A = A_{Adv}$; b. $D_{cur}= B' \oplus F \oplus K_i \oplus H(C_i \oplus K_i)= D' \oplus F= D_{Adv}$

$R_i$ compares $D_{cur}$ with the received $D_{Adv}$ and makes sure that both values are equal. Therefore, $Adv^+$ successfully counterfeits $T_i$.

(2) Tag tracing attack
**Phase 1 (Learning)**: In this phase, an attacker $Adv^+$ initiates the protocol with the target tag $T_i$ as follows:
**Step 1**. The attacker sends a nonce $N_R = 0$ to the target tag $T_i$.
**Step 2**. The tag generates $N_T$ and computes $A = H(N_R \oplus TID) \oplus K_i = H(TID) \oplus K_i$, $B = N_T \oplus K_i$, and $D = H(C_i \oplus K_i) \oplus N_T$, then sends ($A$, $B$, $C_i$, $D$) to the attacker.
**Step 3**. The adversary stores these messages and terminates session.

Therefore, the secret keys of target tag do not update and keep the original values.
**Phase 2 (Tracing)**: To trace the tag $T_i$, $Adv^+$ initiates a new round of protocol, where:
**Step 1**. $Adv^+$ transmits a nonce $N'_R = 0$ to the tag $T_i$.
**Step 2**. The tag produces $N'_T$ and uses the original keys { $K_i$, $P_i$, $C_i$, $TID$ } to compute $A' = H(N'_R \oplus TID) \oplus K_i = H(TID) \oplus K_i$, $B' = N'_T \oplus K_i$, and $D' = H(C_i \oplus K_i) \oplus N'_T$, then sends ($A'$, $B'$, $C_i$, $D'$) to the attacker.
**Step 3**. $Adv^+$ analyzes these received messages and compares them with the saved messages of the Learning phase.

$A' = A = H(TID) \oplus K_i$; $B' \oplus D' = B \oplus D = H(C_i \oplus K_i) \oplus K_i$; $C_i = C_i$

If the above equations are hold, $Adv^+$ can trace the tag $T_i$ by computing the above equations. As a result, the privacy of the tag is broken and the complexity of this attack is two runs of the protocol.

## 5. The Improvement of PLAP Protocol

The proposed lightweight ownership transfer protocol (LOTP) is to overcome security problem of ultra-lightweight protocol. In our protocol, the tags need not implement traditional heavyweight encrypted hash compared with Kapoor *et al.*'s protocol. Moreover, the huge computational workloads can be reduced, since the tag computation is restricted to XOR operations, PRNG operations and hash functions. Furthermore, the fixed correlation between messages and random numbers causes forward/backward traceability in PLAP protocol. Furthermore, the random number is encrypted using the keys to prevent the tracing attack. A description of improved protocol is provided later in in Figure 2. The enhanced scheme is described below:
**Step 1**. R $\rightarrow$T: {$A$}

The reader encrypts $N_R$ using $TID$ and sends encrypted signal $V=N_R \oplus TID$ to $T$.
**Step 2**. T $\rightarrow$R: {$A$, $B$, $D$, $F$}

The tag generates a nonce $N_T$ and computes a fresh version of its pseudonym that facilitates its anonymous identification as $B= N_T \oplus TID \oplus C_i$. Then $T_i$ computes the encrypted messages as $A=H(TID \| N_R \| N_T) \oplus K_i$, $D= H(C_i \| K_i \| N_T)$, $C_i=C_i \oplus 1$, $F=H(N_R \oplus C_i)$.
**Step 3**. The tag sends the messages {$A$, $B$, $D$, $F$} to DB.
**Step 4**. R/DB verification and computation.

Upon receiving the messages, the DB looks up the index value $C_i$, calculates $H(C_i \oplus N_R \oplus 1)$, and compares the received $F$ with $H(C_i \oplus N_R \oplus 1)$. If $F = H(C_i \oplus N_R \oplus 1)$, the DB extracts $N_T = TID \oplus B$. Otherwise, the DB terminates the scheme.

a. If $C_i = 0$, the DB checks every record sequentially and computes three values $\{i_{old}, i_{new}, i_{cur}\}$ based on the received $A$ and the stored values $\{K_{old}, K_{new}, TID\}$, such that $i_{old} = A \oplus K_{old}$, $i_{new} = A \oplus K_{new}$, and $i_{cur} = H(TID \| N_R \| N_T)$. When either $i_{cur} = i_{new}$ or $i_{cur} = i_{old}$ is found, the server sets $x$ as new or old accordingly.

b. If $C_i \neq 0$, the DB uses $C_i$ as the index to look up for the related value in the DB. When either $C_i = C_{new}$ or $C_i = C_{old}$ is found, the DB respectively marks $x$ as new or old and recalculates $A$ as $A_{cur}$ where $A_{cur} = H(TID \| N_R \| N_T) \oplus K_x$.

c. If $x$=null, the records between the DB and the stored keys in $tag_x$ are neither found nor matched.

d. If the DB find $C_x$ and $x \neq$null then calculates $D_{cur} = H(C_x \| K_x \| N_T)$ and compares this value with the received $D$. If both values are equal, the DB computes message $E$ as $E = H(N_R \| N_T \| TID) \oplus P_x$. Otherwise, the DB ends the protocol.

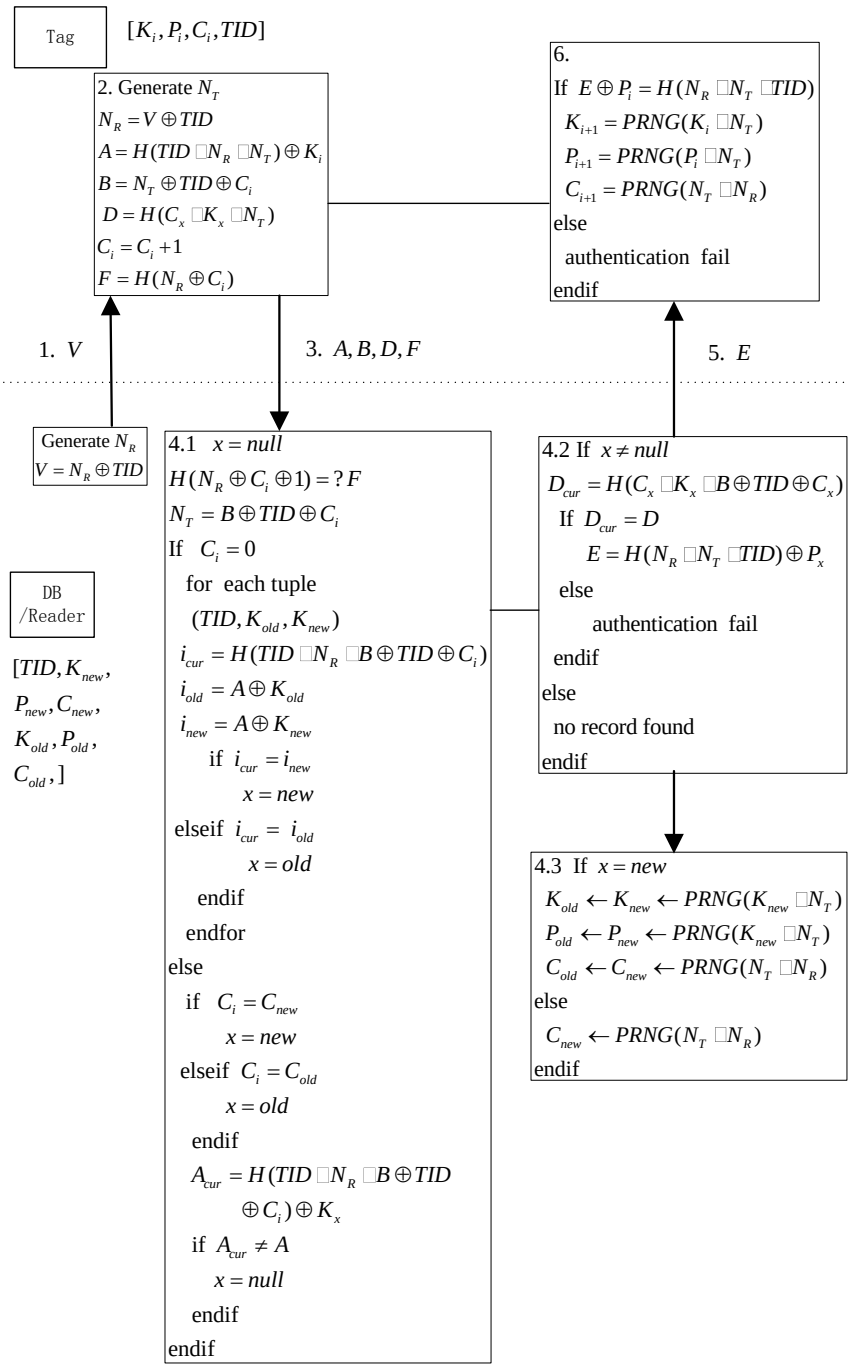



**Figure 2. Provably Security Lightweight Ownership Transfer Protocol**

e. If *x*=old then the keys of DB remain unchanged.
f. If *x*=new then the old keys and the new keys stored in the DB are updated for the next session as $K_{old} = K_{new} = PRNG(K_{new} \| N_T)$, $P_{old}=P_{new}=PRNG(P_{new} \| N_T)$, $C_{old}=C_{new}= PRNG(N_T \| N_R)$.

**Step 5**. The DB forwards *E* to the tag.

**Step 6**. The tag verification and computation.

After receiving *E*, the tag computes $H(N_R \| N_T \| TID)$ and compares $E \oplus P_i$ using the received *E*.

a. If both values have the same result, the authentication has been successfully carried out. The current keys of tag are updated as $K_{i+1}$= *PRNG* ($K_i \| N_T$), $P_{i+1}$= *PRNG* ($P_i \| N_T$),$C_{i+1}$= *PRNG* ($N_T \| N_R$).

b. If the authentication protocol is failed, the old keys of tag are not updated.

### 5.1. The Formal Privacy Proof

We enhance the attacker model and give a proof of the untraceable of the LOTP protocol. To strengthen attack capability and lower assumed conditions, the wise attacker $Adv^+$ has only permission to perform both active and passive attacks in the attack phase with **corrupt** oracle, and only corrupt the updated keys but not the shared key. Then, $Adv^+$ can distinguish the target tag from the set of tags in the analysis phase. In the end, we provide untraceable safety analysis by means of Vaudenay's model communicated with a wise attacker.

**Theorem 1.** *The improved protocol is untraceable.*

*Proof*

**CreateTag**($ID_0$), **CreateTag**($ID_1$)

*vtag* ← **DrawTag** ($ID_c$), where $c \epsilon \{0,1\}$

$\pi^i$ ← **Launch**

$V$ ← **SendReader-tag**($\pi^i$, *Init*)

*A,B,D,F* ← **SendTag** (*vtag* , *V*)

*E* ←**SendReader**($\pi^i$, *A, B, D, F*)

$Adv^+$ chooses 3 random numbers $r_A$, $r_B$, $r_D$, $r_F$

Null ← **SendTag** (*vtag*, $r_A$, $r_B$,$r_D$,$r_F$)

**Free**(*vtag* )

*Vtag'*← **DrawTag** (*ID'*) between 2 tags

*A',D',F'* ← **SendTag** (*vtag*', $N_R$)

The queries is ended, receives $\tau$( *vtag* ) = $ID_c$

Since $A' \neq A$, $B' \neq B$, $D' \neq D$, $F' \neq F$, A cannot trace *vtag*' and eventually

*Pr*[*Adv succeeds* ]− *Pr*[$Adv^+$ *succeeds* ] << ε .

Indeed, the improved protocol is untraceable, when the related privacy level $Adv^+$ could fail the privacy experiment with negligible probability ε .

■

**Theorem 2** *The improved protocol is strong forward untraceable.*

Moreover, in order to meet strong forward untraceable, the nonces $N_T$ and $N_R$ should be concatenated in the key update procedure. If a scheme successfully, the tag updates its keys as

$K^x_{i+1}$= *PRNG* ($K^x_i \| N_T^i$), $P^x_{i+1}$= *PRNG* ($P^x_i \| N_T^i$), $C^x_{i+1}$= *PRNG* ($N_T^i \| N_R^i$).

Even if $Adv^+$ corrupts {$K_i$, $P_i$, $C_i$} and monitors the $i^{th}$ messages {$A_i$, $B_i$, $D_i$, $F_i$}, the computation of $K^x_{i+1}$= *PRNG* ($K^x_i \| N_T^i$) is impractical. The reason is that it has no access to the random numbers {$N^i_T$, $N^i_R$} encrypted by the unknown key *TID*.

*Proof*

**CreateTag**($ID_0$), **CreateTag**($ID_1$)

*vtag* ← **DrawTag** ($ID_c$), where $c \epsilon \{0,1\}$

$K_i$, $P_i$, $C_i$← **corrupt**() at time interval [$i$-1,$i$+1]

$V$ ← **SendReader-tag**($\pi$, *Init* , *ID*)

*A,B,D,F* ← **SendTag** (*vtag* , *V*)

*E* ←**SendReader**($\pi$, *A,B,D,F*)

**Free**(*vtag* )

$Vtag^x$← **DrawTag** ($ID_x$) between 2 tags

$Adv^+$ chooses another time interval $i$ =[$i$ + 1 ]

$K_{i+1}^x = PRNG(K_i^x \| N_T^i)$, $P_{i+1}^x = PRNG(P_i^x \| N_T^i)$, $C_{i+1}^x = PRNG(N_T^i \| N_R^i)$.

$\pi^{i+1}$ ← **Launch**

$V^{i+1}$ ← **SendReader-tag** ($\pi^{i+1}$, *Init* )

$A^{i+1}, B^{i+1}, D^{i+1}, F^{i+1}$ ← **SendTag** ($vtag^x$, $V^{i+1}$)

The queries is ended, receives τ( *vtag'*)= $ID_x$

Owing to lack of the relation among the messages { $A^{i+1}, B^{i+1}, D^{i+1}, F^{i+1}$} and { $A^i \neq A^{i+1}$, $B^i \neq B^{i+1}$, $D^i \neq D^{i+1}$, $F^i \neq F^{i+1}$}, $Adv^+$ cannot distinguish between *vtag* and $vtag^x$ without the $(i+1)^{th}$ keys { $K_{i+1}^x$, $P_{i+1}^x$, $C_{i+1}^x$}. Since $Adv^+$ cannot compute {$A^{i+1}$, $B^{i+1}$, $D^{i+1}$, $F^{i+1}$} without {$K_{i+1}^x$, $P_{i+1}^x$, $C_{i+1}^x$}, $Adv^+$ is not trivial and

$$\{Adv_A^{Upriv}(k), Adv_A^{strong-Forward-Upriv}(k)\} = 0 = \varepsilon$$

Since $Adv^+$ is allowed to initiate protocol and monitor the information between *T* and *R*, s/he can not trace the $(i+1)^{th}$ messages. Therefore, the improved protocol is strong forward untraceable.

■

**Theorem 3** *The improved protocol is backward untraceable.*

*Proof*

**CreateTag**($ID_0$), **CreateTag**($ID_1$)

*vtag* ← **DrawTag** ($ID_c$), where c∈{0,1}

$K_i, P_i, C_i, TID^c$ ← **corrupt**() at time interval i∈[*i*-1,*i*+1]

**Free(***vtag* )

$Vtag^x$ ← **DrawTag** ($ID_c$) between 2 tags

$Adv^+$ chooses another time interval *i* <=[*i* -1 ]

$K_i^x = PRNG(K_{i-1}^x \| N_T^{i-1})$,

$P_i^x = PRNG(P_{i-1}^x \| N_T^{i-1})$,

$C_i^x = PRNG(N_T^{i-1} \| N_R^{i-1})$,

$K_i^x = PRNG(K_{i-1}^x \| N_T^{i-1})$,

$P_i^x = PRNG(P_{i-1}^x \| N_T^{i-1})$,

$C_i^x = PRNG(N_T^{i-1} \| N_R^{i-1})$,

$\pi^{i-1}$ ← **Launch**

$V^{i-1}$ ← **SendReader-Tag** ($\pi^{i-1}$, *Init* )

$A^{i-1}, B^{i-1}, D^{i-1}, F^{i-1}$ ← **SendTag** ($vtag^x$, $V^{i-1}$)

The queries is ended, receive τ( *vtag*)= $ID_x$

If either $A^{i-1} = A_x^{i-1}$ or $A^{i-1} = A_x^{i-1}$,

Output whether τ( $vtag^x$) = $ID_x$

($Adv^+$ cannot distinguish between *vtag* and $vtag^x$ without the $(i-1)^{th}$ keys {$K_{i-1}^x$, $P_{i-1}^x$, $C_{i-1}^x$}.)

Since the output of PRNG function is random and the seed is unpredictable, $Adv^+$ cannot compute {$K_{i-1}^x$, $P_{i-1}^x$, $C_{i-1}^x$} with {$K_i^x$, $P_i^x$, $C_i^x$}, $Adv^+$ is trivial and

$$\{Adv_A^{Backward-Upriv}(k)\} = 0 = \varepsilon$$ ■

We show that the enhanced protocol is secure against tag/reader impersonation attack, and affords strong forward untraceable and backward untraceable based on the Vaudenay's formal privacy model. Thus, we demonstrate that our scheme provides the required performance and security properties.

### 5.2. The Comparisons of Performance Properties

Therefore, the update process of tag implements PRNG functions. Furthermore, the tag computation only needs XOR bitwise operations, concatenate calculations and hash functions. Since the low-cost tag has a quite restricted hardware for saving and

computation, we only analyze the tag operators. To sum up, the computation overhead are compared with five protocols in Table 2.

According to Table 2, our enhanced scheme provides the lowest computational overload.

**Table 2. The Comparisons of Performance Properties**

| *Protocol* | *T1* | *T2* |
|---|---|---|
| Yoon *et al*'s[21] | 6PRNG | 1 |
| Yeh *et al*'s[22] | 6PRNG | 1 |
| Mohammadali *et al*'s[8] | 7PRNG | 2 |
| Alakrut *et al*'s[3] | 6 hash | 1 |
| Improved NRS[18] | 5hash | 1 |
| ours | 4hash+3PRNG | 1 |

***T1：Type and number of encryption functions on tag; T2：Number of pseudo-random nonces on tag.***

### 5.3. The Comparisons of Performance Properties

(1) Protection against Tag Impersonation Attack

In LPAP, the tag uses the messages *B* and *D* to authenticate the DB. The weakness of LPAP is that an adversary can freely modify the valid B and D without the knowledge of keys. The modified messages *B*' and *D*' can be verified as the legal information by the DB. However, the messages of the proposed scheme are independent. Therefore, there is no linkage between the information of different runs.

In PLAP, sub-messages *A* and *D* are used for the server to authenticate the tag whereas sub-message *B* and *F* are used for ensuring data integrity. By the means of these sub-messages *E*, the tag and *R/DB* communicates with each other thus an attacker cannot impersonate the tag as well as the *R/DB*.

(2) Protection against Secret Parameters Disclosure and Replay Attack

None of these protocols are proven safe, since the tag's keys and random number are sent in plain text. Even if the attacker has revealed the current keys of the tag, s/he cannot calculate the updated key using the public information and the known PRNG function. In LPAP, if the attacker replays $N_R$, obtains the same messages $C_i$ and *A* from the target tag, and can trace the tag. Nevertheless, the revised protocol resolves the questions by encrypting $N_R$ and $C_i$ as $V=N_R \oplus TID$ and $F= H(N_R \oplus C_i \oplus 1)$. Even if the attacker replays *V*, s/he cannot obtain the useful information to trace the target tag.

It can be seen from the comparisons of security in Table 3 that the weaknesses are alleviated with the high security privacy in the improved protocol. Moreover, protective measures can be used to achieve the data integrity of random numbers by means of transmitting the encrypted random sequences.

**Table 3. The Comparisons of Security Properties**

| *Protocol* | *S1* | *S2* | *S3* | *S4* |
|---|---|---|---|---|
| Yoon et al's[20] | NO | NO | NO | NO |
| Yeh et al's[21] | NO | NO | NO | NO |
| Mohammadali et al's[7] | NO | NO | YES | NO |
| Alakrut et al's[2] | NO | NO | NO | NO |
| ours | YES | YES | YES | YES |

***S1：Tag impersonation resistance; S2：Replay attack resistance; S3：Secret disclosure resistance; S4： Security proof.***

Eventually, we proved that the improved scheme provide strong forward untraceable and backward untraceable.

## 6. Conclusion

We have proposed a new RFID ownership transfer protocol for supply chain system environments. Apart from guaranteeing some essential security properties, we solve the trade-off between ownership transfer privacy and security in supply chain system environments. In this paper, we first prove that this protocol is untraceable even if the

scheme allows a wise adversary to corrupt the updated key of the tag's holder (e.g. a consumer). The OT protocol for improving key-update mechanism as it ensures that the illicit owners can be held back, as well as the exceeding access of valid owners. Soon afterwards, strong forward untraceable and backward untraceable of ownership is proven in our enhanced security model.

As a motivating explanation, we have analyzed two questionable protocols. Kapoor *et al.*'s protocol and PLAP protocol both are subject to tag impersonation attack. Moreover, the PLAP protocol undergoes tracking attack. Meanwhile, it is necessary to meet security requirements for analyzing the enhanced protocol. Therefore, our enhanced protocol has achieved mutual authentication, key update mechanism, strong forward untraceable and backward untraceable.

## Acknowledgements

This work is supported by the 333 Project of Jiangsu Province (No. BRA2014047), and the Six Talent Peak Project of Jiangsu Province (No.2014-WLW-023).

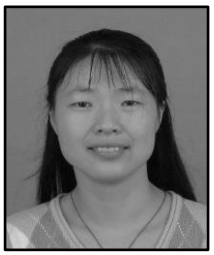
## References

[1] C. H. Ko, "3D-Web-GIS RFID Location Sensing System for Construction Objects", The Scientific World Journal, (**2013)**.
[2] S. Dhal and I. Sengupta, "Handling Authentication and Detection Probability in Multi-tag RFID Environment", Cryptology ePrint Archive, Report 2013/486, (**2013)**, pp.1-20.
[3] R. H. E. Alakrut, A. Samsudin and A. Syafalni, "Provably Lightweight RFID Mutual Authentication Protocol", International Journal of Security & Its Applications, vol. 7, no. 4, (**2013)**, pp. 71-87.
[4] G. Kapoor and S. Piramuthu, "Single RFID tag ownership transfer protocols", Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, vol. 42, no. 2, (**2012)**, pp. 164-173.
[5] C. Y. Ng, W. Susilo, Y. Mu and R. S. Naini, "Practical RFID ownership transfer scheme", Workshop on RFID securit (RFIDSec Asia) volume 4 of cryptology and information security, IOS press, **(2010)**.
[6] J. Saito, K. Imamoto and K. Sakurai, "Reassignment scheme of an RFID tag's key for owner transfer", Proceedings of IFIP Int. Conf. Embedded Ubiquitous Comput. Workshop, LNCS 3823, **(2005)**, pp. 1303-1312.
[7] K. Osaka, T. Takagi, K. Yamazaki and O. Takahashi, "An efficient and secure RFID security method with ownership transfer", Proceedings of Int. Conf. Comput. Intell. Security, LNAI 4456, **(2007)**, pp. 778-787.
[8] A. Mohammadali, Z. Ahmadian and R. Aref, "Analysis and Improvement of the securing RFID systems conforming to EPC Class 1 Generation 2 standard", IACR Cryptology ePrint Archive, **(2013)**, pp. 66.
[9] M. Safkhani, N. Bagheri and M. Naderi, "Strengthening the security of EPC C-1 G-2 RFID standard", Wireless Personal Communications, vol. 72, no. 2, **(2013)**, pp. 1295-1308.
[10] M. Akgun and M. U. Caglayan, "Server Impersonation Attacks and Revisions to SLAP, RFID Lightweight Mutual Authentication Protocol", Proceedings of the Systems and Networks Communications (ICSNC), 2010 Fifth International Conference, **(2010)**.
[11] H. Fernando and J. Abawajy, "Mutual authentication protocol for networked RFID systems", IEEE TrustComm, **(2011)**.
[12] H. Zhu, Y. Zhao, S. Ding and B. Jin, "An improved forward-secure anonymous RFID authentication protocol", Wireless communications, networking and mobile computing (WiCOM), **(2011)**, pp. 1-5.
[13] X. Fan, G. Gong, D. W. Engels and E. M. Smith, "A lightweight privacy-preserving mutual authentication protocol for RFID systems", IEEE GLOBECOM workshops (GC Wkshps), **(2011)**, pp. 1083-1087.
[14] J. Hermans, A. Pashalidis, F. Vercauteren and B. Preneel, "A new RFID privacy model", V. Atluri, C. Diaz (Eds.), ESORICS 2011. LNCS, 6879, **(2011)**, pp.568-587.
[15] I. Coisel and T. Martin, "Untangling RFID privacy models", Journal of Computer Networks and Communications, vol. 26, **(2013)**, doi: 10.1155/2013/710275.
[16] G. Avoine, I. Coisel and T. Martin, "Time measurement threatens privacy-friendly RFID authen-tication protocols", RFIDSec. Sprinfer LNCS, vol. 6370, **(2010)**, pp. 138-157.
[17] S. Vaudenay, "On privacy models for RFID", K. Kurosawa (Ed.), ASIACRYPT 2007. LNCS, vol. 4833, Heidelberg: Springer, **(2007)**, pp. 68-87.
[18] M. R. Alagheband and M. R. Aref, "Simulation-Based Traceability Analysis of RFID Authentication Protocols", Wireless Personal Communications, **(2013)**, pp. 1-20.
[19] B. Song, "RFID tag ownership transfer", Proceedings of the Proceedings of Workshop on RFID Security, Budapest, Hungary, **(2008)**.

[20] R. Doss, W. Zhou and S. Yu, "Secure RFID tag ownership transfer based on quadratic residues", Information Forensics and Security, IEEE Transactions on, vol. 8, no. 2, **(2013)**, pp. 390-401.
[21] E. J. Yoon, "Improvement of the securing rfid systems conforming to epc class 1 generation 2 standard", Expert Syst. Appl., vol. 39, no. 12, **(2012)**, pp. 1589-1594.
[22] T. C. Yeh, Y. J. Wang, T. C. Kuo and S. S. Wang, "Securing RFID systems conforming to EPC Class 1 Generation 2 standard", Expert Systems with Applications, Available online 10 May **(2010)**.

## Authors



**Xiuqing Chen**, she received her bachelor's degree and master's degree from the China University of Mining and Technology. She has been a Ph.D. degree candidate in applied computer Technology from the China University of Mining and Technology. Her research interests include security protocols and network security. Email: xiuqingchen@cumt.edu.cn

**Tianjie Cao**, he received the BS and MS degree in mathematics from Nankai University, Tianjin, China and the PhD degree in computer software and theory from State Key Laboratory of Information Security of Institute of Software, Chinese Academy of Sciences, Beijing, China. He is a professor of computer science in the School of Computer Science and Technology, China University of Mining and Technology, Xuzhou, China. From 2007 to 2008, he has been a visiting scholar at the Department of Computer Sciences and CERIAS, Purdue University. His research interests are in security protocols and network security.Email: tjcao@cumt.edu.cn

**Jingxuan Zhai**, he received his bachelor's degree and master's degree from the China University of Mining and Technology. he has been a Ph.D. degree candidate in applied computer Technology from the China University of Mining and Technology. His research interests include network security and security protocols. Email: zhaijx@cumt.edu.cn