Defense against Impersonating Attackers: An Efficient RFID Mutual Authentication Protocol Based on Standard

Shiqi Wang
SEIEE, Shanghai Jiao Tong University No.800 Dongchuan Road, Shanghai, P.R. CHINA
+086 134 8279 2901
wangshiqi@sjtu.edu.cn

Linsen Li
SEIEE, Shanghai Jiao Tong University No.800 Dongchuan Road, Shanghai, P.R. CHINA
+086 139 1757 7256
lsli@sjtu.edu.cn

Gaosheng Chen
SEIEE, Shanghai Jiao Tong University No.800 Dongchuan Road, Shanghai, P.R. CHINA
+086 150 2111 9875
chengaosheng@sjtu.edu.cn

Tao Chen
SEIEE, Shanghai Jiao Tong University No.800 Dongchuan Road, Shanghai, P.R. CHINA
+086 188 1827 3810
jeffrey.tchen@foxmail.com

Zeming Wang
UMJI, Shanghai Jiao Tong University No.800 Dongchuan Road, Shanghai, P.R. CHINA
+086 188 1821 3919
cpt_leo@sjtu.edu.cn

***Abstract*--As the RFID based Internet of Things (IoT) gets worldwide attention, to prepare for the rapidly increasing applications in daily life, various security protocols are proposed. But, these protocols, most of which are limited by the tag processing capacity and dangerous exposure during transmission, could only be applied in certain fields. Previously, Chen and Deng's mutual authentication and privacy protection protocol which conforming EPC Class 1 Generation 2 Standards stands out for low cost as well as little requirements of the tag processing capacity. However, currently reported by others, this system faces up with severe dangers of tracking or cloning tags via impersonating attacks. After scrutiny, we found out that these vulnerabilities lie in the insufficient protections of random numbers, and we reconstruct the request and response based on the original protocol by making message unrepeatable, key elements secret and adding small storage for comparisons. The security of our protocol, proved by Ban logic analysis, is ensured by double protections—secret key pairs and dynamic random numbers. Our comparisons show that our protocol not only is safe under traditional attacks guaranteed by the original protocol but also overcomes impersonating attacks which represents the inherent weakness of information exposure in public.**

Keywords--RFID security; authentication; vulnerabilities

I. INTRODUCTION

RFID is the short for radio frequency identification system which includes tags, readers, database, and hosts in its system. The RFID tags receive and send remote commands from readers so that we can authenticate, communication and edit the information of the object through attached tags, which builds up the connections among things, namely IoT via RFID. The low-cost tags made of small chips and antenna has an EPC (Electronic product Code), a unique and symbolic code, to identify the targets the tag is attached.

The leader of MIT Auto-ID Center once presented a heady vision elaborated in [13]: "By creating an open global network that can identify anything, anywhere, automatically, [the Auto-ID Center] seeks to give companies something that, until now, they have only dreamed of: near-perfect supply chain visibility." This shows the hot trend of RFID, signifying the wide acceptance and promising usage in industry. Besides, its potential of highly efficient item-level authentication implementation, as one of the greatest advantages over bar-code, showing its irresistible benefit in our daily life. Currently, some of the RFID applications are widely used and would be widely applied in the future such as access control, wireless commerce and supply chain management [6].

Most characteristics of RFID tags like negligible individual cost, sufficient capacity, high efficiency are absolutely compelling to businesses. However, the wide acceptance of RFID is limited by the security vulnerabilities exists in the current systems. For example, containing highly sensitive and private information in the system makes itself the extremely tempering target for the attackers, from which the attackers could get great benefit if the system was broken down. Besides, the nature of RFID which utilizes the network transmitted in public makes it more easy to be probed and attacked. Unless RFID systems are properly architected and improved, they would cause massive collateral damage to consumer privacy such as the exposition of user's location and his private information.

Many kinds of protocols are proposed in extant papers, pervading all kinds of directions. For encryption algorithms, ECC [16] and RSA [17] are proposed which gradually make the protocol designs similar to Internet protocol, while the others use hash or Xor, curtailing the requirements of the tag processing capacity [20]. As for the system structure designs, people studied group management [10][19], strategies of back-end database [18], etc. However, most of them, though seems to be safe in their own fields, unfortunately has their own limitations out of their predefined fields like health center and could not be regarded as a standard, widely accepted one. Recently, Chen and Deng proposed a successful mutual authentication and privacy protection protocol [1] which could mostly guarantee the basic security of the systems at low cost. Unlike other protocols using complicated algorithms to encrypt, they just use Xor, CRC and Random numbers. Their simple and efficient design of mutual authentication requires little of the system capability, which could not only curtail the cost but also make the fast efficient authentication possible. Also, this protocol conforms EPC Class 1 Generation 2 standard [5], satisfying diverse applications in all kinds of circumstances. Given its generality as well as exquisite design, we decided to create a protocol based on it, which we expect could push forward the development of RIFD security and basically ensure the safety of daily use.

However, their protocol, though seems to be great, still has several vulnerabilities first proposed in another paper [2]. Especially, the users' tags could be easily cloned as well as dangerously tracked. Through scrutiny, we discover that the unchangeable tag response message of the same request as well as repeatable process of authentication are the two main reasons account for these vulnerabilities. When we dig deeper, we find this was mainly because the response structure lacks the ability of self-alternation. In other words, the random numbers generated and used during authentication are almost useless for keeping the process safe. So we rebuilt an exquisite structure of the request and response messages on the basis of Chen and Deng's protocol. The random numbers r_R, r_T, having the function as nonce, enable the encrypted

978-1-5090-4243-2/17/$31.00 ©2017 IEEE

message unrepeatable. In order to prevent impersonating attacks, these random numbers are kept secret and verified in every conversation. Our Ban logic security analysis as well as comparisons show that our protocol not only inherits the original protocol's universality and concise structure but also overcomes the main vulnerabilities proposed, and eventually ensures the security of the systems.

II. CHEN AND DENG'S PROTOCOL

Since the detailed information is available in [1], we just give a brief sketch of Chen and Deng's protocol. We first introduce the notation used in this paper.

- (N_i, K_i): N_i is a nonce word, K_i is a key. If the tags have registered with the Database, they obtain the (N_i, K_i).
- (N_i^t, K_i^t): the pair stores in the tags
- CRC(): a cyclic Redundancy check function.
- EPC_{Ti}: the ith EPC which conforms C1G2 standards to identify the unique global product.
- ID_{Ri}: the ith reader's identification.
- $\oplus$: exclusive-or operation.
- r_T: a random value which is generated by a tag.
- r_R: a random value which is generated by a reader.
- r_R^t: the r_R sent by tag.
- $r_R(c)$: current r_R
- $r_R(l)$: r_R in last cycle
- M_{Req}: the reader's request message.
- M_{Resp}: the tag's response message.

A. The process of Chen and Deng's protocol

Chen and Deng proposed a mutual authentication protocol, which contains two different phases—registration and communication.

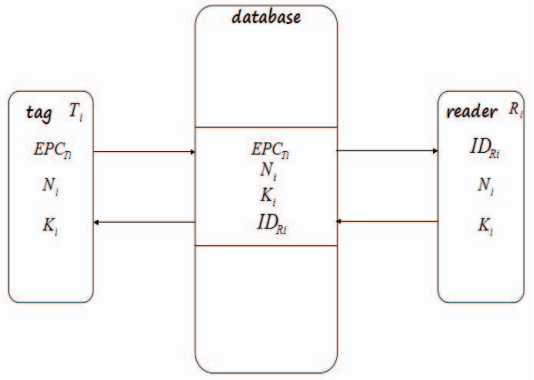

Fig.1 Scenarios of registration

Each tag has a unique EPC identifying itself. In the first registration phase, when the tag is registered to the database, the database will issue n parameter pairs (N_i, K_i) for each registered tag and the EPC_{Ti} as pairs stored in the database. Then, each reader has a unique ID_{Ri}. After the registration of readers, they would be assigned the tag registered before which should be legitimately recognized by storing corresponding (N_i, K_i). This registration not only registers all the legitimate tags and readers, but also establish the relationships between these tags and readers. The scenarios of registered tag and registered reader is shown in Fig.1.

The second phase is communication phase. When reader wants to access the tag, it sends M_{Req}, r_R and $CRC(r_R \oplus N_i)$. The tag authenticates the reader by calculating $CRC(r_R \oplus N_i) \stackrel{?}{=} CRC(r_R \oplus N_i^t)$. If it holds, then the tag calculates $X = K_i \oplus EPC_{Ti} \oplus r_T$, $Y = CRC(N_i \oplus r_T \oplus X)$ and sends r_T, X, Y to the reader. The reader authenticates the tag by calculating Y. If it holds, then it gets $EPC_{Ti} = K_i \oplus r_T \oplus X$. Finally, the reader sends M_{Resp} to the tag to tell the tag the success of the authentication. The whole authentication process is shown in Fig.2.

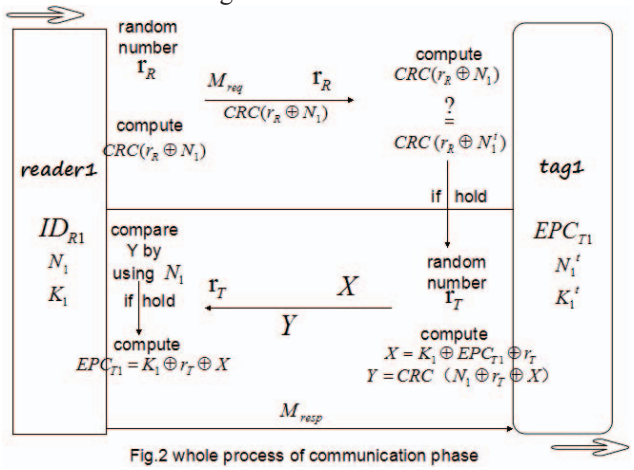

Fig.2 whole process of communication phase

B. The advantage of Chen and Deng's protocol

The tags are limited by computing speed and cost while Chen and Deng's protocol only includes CRC, exclusive-or operations and random numbers generations. The low cost and high speed of efficient authentication shed light on the possibility of wide acceptance of RFID applications.

Besides, Chen and Deng's protocol can satisfy the mutual authentication mechanism between tag and reader. Through the verification of $CRC(r_R \oplus N_i) \stackrel{?}{=} CRC(r_R \oplus N_i^t)$, the tag could authenticate whether the reader is a legal one. At the same time, through the verification of $CRC(N_i^t \oplus r_T \oplus X) \stackrel{?}{=} CRC(N_i \oplus r_T \oplus X)$, the reader can authenticate whether the tag is a legal one. Thus, the mutual authentication ensures the basic security and accurate communication. The detailed proofs against traditional attacks is given in [1].

Though attackers might be able to intercept the messages, they won't be able to get the private information because of the combination of random numbers as well as asymmetric encryption, which prevent the possibility of accessing private messages via brute force.

III. VULNERABILITIES OF CHEN AND DENG'S PROTOCOL

Though Chen and Deng's protocol can ensure the basic security during authentication, it still has the vulnerabilities which could be utilized via impersonating attacks.

Impersonating attacks, which means that the attacker could impersonate the legitimated tags and readers by sending

the legal message intercepted in public, are very similar to replay attacks in the field of Internet protocols but specially benefit from the unsecure design of the request and response. When the attacker fortunately intercepts the legitimated tag request and replays the request to the targeted reader, it would be really difficult for the reader using Chen and Deng's protocol to distinguish and defend. Similarly, reader impersonating signifies that the legal tag isn't able to distinguish the request of legal reader or illegal impersonating one.

Impersonating attacks were first proposed in the paper [2] where figured out these vulnerabilities over the system using Chen and Deng's protocol. The detailed analysis could be found in that paper. Here, we would mainly detail these two major vulnerabilities, pinpointed the possible means of attacks and points out the ensuing serious dangers in Chen and Deng's system.

A. Legal Reader Impersonating:

{Step 1. The attacker passively observes and intercepts legitimated conversation between the legal tag T_j and legal reader R_i.

Step 2. The fake reader replays the request message (M_{req} , r_R , CRC ($r_R \oplus N_i$)) intercepted at the step 1 to the targeted legal tag T_j.

Step 3. The attacker analyzes the response from T_j and successfully tracks T_j.}

It seems to be really unbelievable for the attacker to easily track the legal tag, which means the exposition of the user's location, if the attacker utilizes the vulnerability of reader impersonating. But, after scrutiny, we could see that if the illegal reader request message is the same as a legal one, the Y of response message (r_T, X, Y) of the legal tag is completely the same. For $Y = CRC(K_i \oplus EPC \oplus r_T \oplus N_i^t \oplus r_T)$, the r_T in Y was Xored double times, so Y doesn't change, which directly leads to the existence of the vulnerability. The attacker can easily analyze the message he gets, and continuously track the tag by always sending the same request through illegal reader.

B. Legal Tag Impersonating:

{Step 1. The attacker passively observes and intercepts the legitimated conversation between the legal tag T_j and legal reader R_i.

Step 2. When the legal reader R_i sends M_{req} to the legal tag T_j, the fake tag which impersonates the legal T_j reply R_i with $r_T \oplus \delta$, $X \oplus \delta$ and Y. (δ is any number to make the respond look different.)

Step 3. The attacker's fake tags could be completely confounded by the legal reader with legal tag T_j, which means the attacker could successfully clone the legal tag T_j.}

To avoid all of the respond messages looking like the same, the attacker may try to change r_T to $r_T \oplus \delta$, for $r_T \oplus \delta$ just looks like a new random number. Chen and Deng has used random number r_T in the tag respond, but r_T is completely exposed in public and easily intercepted by the attackers. Besides the reader authentication just compares N_i^t which can be calculated by the attacker once the message as well as the random number are intercepted. Via this vulnerabilities, attacker could easily get plenty of legal cloning tag T_j, which would definitely cause disastrous influence in daily use. For example, we originally attached a legal tag T_j to a genuine masterpiece of Picasso. Once the attacker gets the legal authentication message during a legitimated deal, he can then attach hundreds of fake tags of T_j to the fake works, which could send the same response as the legal one. Unfortunately, as a result, the legal reader could definitely never identify which one is real.

IV OUR PROPOSED PROTOCOL

To overcome the vulnerabilities of the Chen and Deng's protocols we stated above, we propose our own protocols, mainly taking full advantage of the random numbers as well as asymmetric encryption. Basically, it still has two phases—registration and communication.

A. Registration Phase

We make a small change of registration sequence during the registration phase to efficiently manage the legal tags and readers. But still the basic registration points are still the same as Chen and Deng's.

The scenarios of the registration are shown in Fig.3.

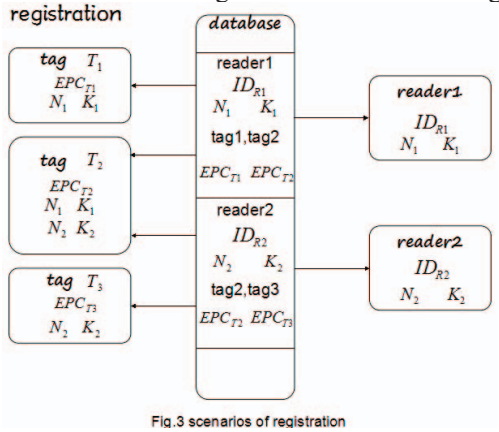

Fig.3 scenarios of registration

The database first registers the reader R_i. The database records the ID_{Ri} and assigns the reader key set (N_i, K_i). Thus, the (N_i, K_i), ID_{Ri} are stored in the reader R_i and are all able to symbolize R_i. Then the database registered tags from Tag T_j to Tag T_{j+k} which could all be recognized by R_i, and assigns the key set (N_i, K_i) of R_i to each of them.

After the registration, as long as tag T_j could be read by R_i, the parameter pairs (N_i, K_i) which belongs to R_i would be stored in T_j. As a result, tag T_j could distinguishes all of its own legal readers through the verification of (N_i, K_i). If we need to add a new tag which should be read by reader R_i, the database just needs to assign (N_i, K_i) and EPC_{Ti} to it.

This small change, though at the cost of a little more storage in tag, has great benefit of easier management for database when register the tags and readers.

B. Reader Request

As what we have analyzed before on vulnerabilities in Chen and Deng's protocols, we could easily draw the conclusion that the key vulnerabilities of impersonating attacks are the unchangeable response of the same request as well as repeatable process of authentication. So it seems to be a better way to make the random number r_R and r_T secret to avoid the replay of the same request and response intercepted by the attacker. Also, these random number r_R and r_T has an expiration time to further avoid replay attacks, which is usually a bit longer than the time needed in one general conversation.

When the reader wants to recognize a tag, it sends M_{req}, N_i, $CRC(K_i \oplus r_R(c))$. The tag will find whether the public key N_i is stored in itself. Different from the original protocol, random number r_R, which is not exposed to the public, will be stored in the reader, and will update every time the reader sends the request. If $N_i \stackrel{?}{=} N_i^t$ doesn't hold, it means the reader is exactly fake. The authentication stops. If it holds, the reader may be either an impersonating reader or a legal reader. Then the tag uses the corresponding secret key K_i to get the $r_R(c)$ (current r_R) and compares whether the $r_R(c)$ is exactly the same as $r_R(l)$ (r_R in last cycle) last time. If they are the same, the authentication stops, which means the reader is an impersonating one. Else, the reader may be a legal reader. The legal tag will make request.

This phase is shown in the Fig.4.

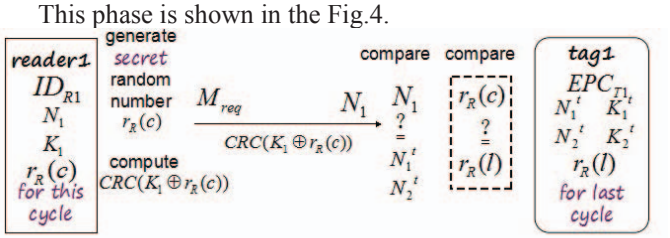

Fig.4 Reader Request phase

C. Tag Request

To ensure the safety and unrepeatable of the whole authentication, r_R should be kept private via the encryption of public key N_i, for it's the symbol of this authentication cycle. Also, we hope to keep the request unique in every conversation and changing continuously after the beginning of new one. We use the help of r_T to keep the request changeable just like a nonce in Internet protocol. Though, in Chen and Deng's protocol, they have used random number r_T in the tag request, but r_T is totally exposed to the public so that its influence could be easily eliminated by the attacker. This is because the original design uses $Y = CRC(K_i \oplus EPC \oplus r_T \oplus N_i^t \oplus r_T)$ to communicate the identity and secret information of tags. However, the r_T in Y was Xored double times so that Y won't be changed when the tag receives the same request message. The unchangeable response leads to the exposure of the identity of the tag and then the location of the user.

Thus, we also encrypt r_T to keep it secret and introduce the parameter Z to check r_T during authentication. In addition, we change the structure of X, Y by using secret parameter r_R to ensure that all of the tag request messages to change as soon as a new conversation begin and r_T is updated by the tag.

When the tag authenticates the reader, it then generates the secret random number r_T. Eventually, the tag computes three key request messages—X, Y, Z and sends them to the reader. The structure of X, Y, Z we designed are listed below:

$$X = K_i \oplus r_T \oplus EPC$$
$$Y = N_i \oplus r_R(c) \oplus X$$
$$Z = N_i \oplus r_T \oplus r_R(c)$$

Now we would explain why we redesigned such kind of request messages:

When the reader receives the tag's request, the reader first get the $r_R^t = Y \oplus N_i \oplus K_i$ (r_R^t is the r_R calculated from the tag request and it may not be equal to $r_R(c)$ or $r_R(l)$). If r_R^t equals to the $r_R(c)$ stored in the reader, then the reader can confirm that the tag is the legal one. Because the K_i is a secret key, the attacker can't get the correct $r_R(c)$ through the reader's request $CRC(K_i \oplus r_R(c))$. If the reader has authenticated the tag, it can get $r_T = Z \oplus N_i \oplus r_R(c)$, which is the dynamic secret random number of the tag.

Finally, the reader computes $X = K_i \oplus r_T \oplus EPC$, and then sends M_{resp} and $r_R(c)$ to the tag, signifying the success of the whole authentication.

The tag updates it's $r_R(l)$ by $r_R(c)$.

The whole process of the tag request is shown in Fig.5.

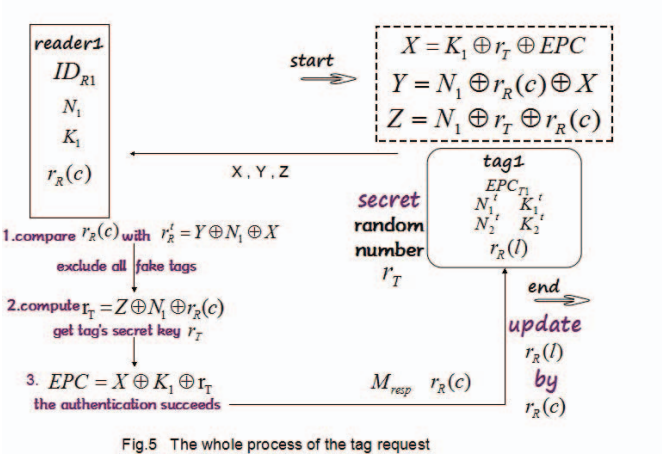

Fig.5 The whole process of the tag request

V. Analysis and Comparison

First of all, our proposed protocol is based on the Chen and Deng's protocol, and we didn't break or reduce any of the key principle of the original protocol, which means our proposed protocol can ensure the basic security guaranteed by the original one. It can mainly prevent four general attack: (a) Counterfeit reader attack tag analysis, (b) counterfeit tag attack reader analysis, (c) man-in-the-middle attack analysis, (d) DoS attack analysis. [1] gave the detailed process of proof.

To ensure the security, most of traditional protocols rely

on secret complicated encryption algorithm, like different kinds of hash [14][15]. So, it will be unsafe if the security system has been deciphered. However, our protocol only use Xor, CRC and random numbers, even if the attacker deciphers the whole security system, he won't be able to threat the privacy security.

We will make comprehensive and deep-going discussion on the effect of impersonating attack, as it's what we target to solve and also is the exact weakness of the RFID technology, especially for Chen and Deng's protocol. And this is because the message, as radio wave, is transmitted throw public area and will be easily intercepted by the attacker. And the possibility of intercepts means the exposure of all of the message without encryption, which would greatly increase the complexity of hardware as well as protocol design. So a good protocol has to consider and solve well the limitation of public transportation and the high cost of complex design.

A. Reader Impersonating

Step 1. The attacker passively observes and replays message between tag T_i and reader R_i. But CRC check will help us find whether the message is lost or changed. Besides, the attacker doesn't have the secret key K_i, so he could not get the secret random number $r_R(c)$, which indicates that it is useless for the attacker to analyze the message he intercepts.

Step 2. The fake reader sends the request message.

(a)If the attacker sends (M_{Req}, N_i, $CRC(K_i \oplus r_R(l))$) intercepts at the step 1 to the legal tag. Of course, the tag will stop the authentication, for the tag will check $r_R(c) \stackrel{?}{=} r_R(l)$.

(b)If the attacker deciphers the request message and sends the tag the request (M_{req}, N_i, $CRC(K_i \oplus r_R(l) \oplus \delta)$) edited from the message intercepted in step 1. The tag will find the same N_i stored in itself, and then send X, Y, Z. But X, Y, Z will be totally different from the tag request sent from the last time intercepted by the attacker. This is because the random number r_T is all included in X, Y, Z and will be updated every time the tag sends the respond message.

Step 3. The attacker won't distinguish which is the tag he attached last time, for nothing of the tag request is the same as the last time. Track fails.

B. Tag Impersonating

Step 1. When the attacker passively observes and intercepts legal tag request message between tag T_j and reader R_i, he gets the X, Y, Z. If the attacker edits the message randomly, CRC check will help the legal reader find the loss or the change of the message and demand the tag to resend. Furthermore, even if the attacker tries to analyze the messages he intercepts, he could not get any useful information, for he doesn't know any of K_i, r_R, r_T.

Step 2. When the attacker's fake tag, which try to impersonate the legal T_j to reply R_i with X, Y, Z he intercepted in step 1, the reader would immediately find it fake after computing $r_R^t = Y \oplus N_i \oplus X$. Now we prove that in detail. We divide all of the possibilities of attack into three different circumstances based on the different times of impersonating tag attack.

(a)When the legal reader requests a fake tag and sends a request message to it, the random number r_R stored in the reader first changed. The r_R^t included in the response message X, Y, Z of the fake tag is the r_R used in the last time. And the attacker's random edition to the random number like $r_R^t \oplus \delta$ could not be the same as the r_R in this new cycle. So the authentication fails.

(b)When the attacker intercepts the legal tag's request and the reader hasn't resent the request for the next authentication yet, mostly the resend of the fake tag would lead to the failure because of the expiration of the random number r_R. But we have to include the seemingly dangerous exception that the impersonating message is sent before r_R expires. We could easily find out that this situation only occurs when the fake tag was put together with a legal one. It indicates that the reader would easily figure out when this impersonating attack happens, for it gets two same tags. If the cloned tag is used at the next time, it will still be useless just like the situation (a).

(c)When the legal reader doesn't receive the tag's response, it will resend the request again. The fake tag sends the intercepted request message after the resend of the legal reader, the r_R^t included in the fake tag's respond message X, Y, Z is the r_R used in last time which is totally useless in this cycle.

From the situation (a) to (c) above, we can get the conclusion that tag clone fails, and the attacker won't get any useful information through this impersonating attack. The only way for the attacker to break the check of secret random number by brute force in the situation (a). And the probability P of the successful attack depends on the binary length l of the random number r_R and r_T.

We could easily get:

$$P = 2^{-l}$$

So if we generally set 64 or 96 bits which is completely supported by the current tag capacity, the attacker must spend years impersonating one silent reader or tag which has never been used in these years, let alone the system updating the r_R and r_T once the reader and the tag are used.

Table1 Functional Comparisons

Functions	C.D. Protocol	Our Protocol
Conform EPC	Yes	Yes
Database Loading	Low	Low
Avoid Dos Attack	Yes	Yes
Encryption Method	CRC	CRC
Authority Management	Yes	Yes
Mutual Authentication	Yes	Yes
Location Traceable	Yes	No
Tag Cloning	Yes	No

C. Comparisons

To Summarize above, we compare Chen and Deng's protocol with ours from the aspect of basic functions in Table1. We could see that our protocol satisfies all of the basic

security level and could keep safe from most of the common attacks which is ensured by the original one.

Table2 and Table3 show the result of impersonating attack of tag and reader, which mainly design to solve.

Table2 Tag Impersonating Attack		
Protocols	New Conversation	Silent
C.D. Protocol	Succeed	Succeed
Our Protocol	No Response	Fake Identified

Table3 Reader Impersonating Attack			
Protocols	Message	Replay r_T	Replay $r_T \oplus \delta$
C.D. Protocol	X	No Change	No Change
	Y	No Change	No Change
Our Protocol	X	No Response	Totally Change
	Y	No Response	Totally Change
	Z	No Response	Totally Change

D. Ban logic analysis:

In order to analyze the security of the protocol, we use the extended BAN logic to analyze, wherein T represents the tag, R represents the reader, (N, K) represents the key pair of the reader.

The Ban logic analysis could be formalized as follows:

Idealization:

I1: $R \rightarrow T$: N, $\{r_R\}_K$

I2: $T \rightarrow R$: $\{EPC, r_T, r_R\}_K$

Assumptions:

A1: T believes (N, K)

A2: R believes (N, K)

A3: T believes R said $\{r_R\}_K$

A4: R believes T said $\{EPC, r_T, r_R\}_{(N,K)}$

A5: R believes fresh(r_R)

A6: T believes fresh(r_T)

Deduction:

D1: T believes R said r_R ----------A2+A3

D2: T believes R believes r_R ----------A5+D1

D3: R believes T said $\{EPC, r_T, r_R\}$ ----------A4+A1

D4: R believes T believes EPC ----------A6+D3

The verification result shows that our protocol, at the end of success of authentication, could basically ensure the goals of secure communication. All the messages are encrypted by Xor with the key pair (N, K) and would be checked by both sides of the reader and tag, ensuring our features of secure mutual authentication.

VI. Conclusion

RFID technology has the potential to be used widely in our daily life and bring great convenience at a very small cost if we can ensure the security of the system. In this paper, we figure out the vulnerabilities of Chen and Deng's protocol, especially the impersonating attacks of tags and readers. And then we propose our mutual authentication one on the basis of Chen and Deng's, keeping the basic security functionalities as well as solving impersonating attacks. Through security analysis, it could ensure the security under the current existing attack and keeps the original generality and wide range of applications.

Our mutual authentication protocol is based on double encryption--one is the dynamic random numbers r_R, r_T, and another is the key pair (N_i, K_i). Random number r_R stands for one conversation and will be updated as soon as the beginning of a new one. So the attacker could not get any useful information through analysis if he intercepts last messages. Another random number r_T generated by tags not only ensures the security of EPC, but also ensures the constant change of the tag request to prevent the exposure of host's location. (N_i, K_i) guarantees basic encryption of mutual authentication as well as keeps the r_R and r_T secret. To a certain extent, our proposed protocol overcomes the inherent weakness of information exposure in the process of electromagnetic wave transmission.

The RFID technology such as tag capacity and processing ability has developed rapidly, which could satisfy the basic need of daily use. As a result, security has become one of the leading limitations for its wide application. As long as the security problems are generally overcome, we have every reason to believe the possibility of wide applications of RFID technologies.

VII. Acknowledgement

This work was supported by the NSFC-Zhejiang Joint Fund for the Integration of Industrialization and Informatization under grant No. U1509219, Priority Development Field Project of Doctoral Fund under grant No. 20130073130006, and Shanghai Municipal Science and Technology Project under grant No. 16511102605 and No. 16DZ1200702.

References

[1] C. L. Chen, Y. Y. Deng. conformation of EPC Class 1 Generation 2 Standards RFID System with mutual authentication and privacy protection. Engineering Applications of Artificial Intelligence 22, 2009

[2] G. Kap, S. Piramuthuoor. Vulnerabilities in Chen and Deng's RFID mutual authentication and privacy protection protocol. Engineering Applications of Artificial Intelligence 24, 2011

[3] S. L. GARFINKEL, A. Juels, R. Pappu. RFID privacy: An overview of problems and solutions, IEEE Security & Privacy, the Claremont Resort, Berkeley, Oakland, California, 2005

[4] D. Z. Sun, J. D. Zhong. A Hash-Based RFID Security Protocol for Strong Privacy Protection, IEEE Transactions on Consumer Electronics, 2012

[5] EPC (Electronic Product Code) Class1 Generation 2 standard by EPC global, 2008. (accessavailableon15March2008).

[6] X. Gao, Z. Xiang, H. Wang, J. Shen, J. Huang, S. Song. An approach to security and privacy of RFID system for supply chain. IEEE International Conference on E-Commerce Technology for Dynamic E-Business, 2004

[7] K. Osaka, T. Takagi, K. Yamazaki, O. Takahashi. An efficient and secure RFID security method with ownership transfer, IEEE International Conference on Computational Intelligence and Security, 2006

[8] S. Piramuthu. RFID mutual authentication protocols. Decision

Support Systems, 2010
[9] Y. Kobayyashi, T. Kuwana, Y. Tanigushi, N. Komoda, Group Management of RFID Passwords for Privacy Protection, Electronics and Communications in Japan, 2009
[10] K. Sakai, W. S. Ku, R. Ximmermann, M. T. Sun. Dynamic Bit Encoding for Privacy Protection against Correlation Attacks in RFID Backward Channel, IEEE TRANSACTIONS on computers, Vol. 62, 2013
[11] S. Spiekermann and S. Evdokimov, "Critical RFID Privacy-Enhancing Technologies", IEEE Security & Privacy, 2009
[12] Vince Stanford, "Pervasive computing goes the last hundred feet with RFID systems", IEEE pervasive computing, 2003
[13] J. Meng and Z. Wang, "A RFID Based Security Based on Hash Chain and Three-Way Handshake", International Conference on Computational and Information Sciences, 2013
[14] H. Wu, S. Qing, H. Li, "A hash-based RFID security protocol for strong privacy protection", CECNET, 2012
[15] S. Kim, Y. Kim, S. Park, ‘‘RFID Security Protocol by Lightweight ECC Algorithm", ALPIT, 2007
[16] G. Bao, M. Zhang, J. Liu, Y. Li, "The design of an RFID security protocol based on RSA signature for e-ticket", IEEE International Conference on Information Management and Engineering, 2010
[17] Y. Hao, S. Ren, "Design and analysis of security protocol for RFID without back-end database", IEEE International Conference on Information and Automation, 2015
[18] C. Piao, Z. Fan, C. Yang, X. Han, "Research on RFID security protocol based on grouped tags and re-encryption scheme", IEEE International Conference on Wireless Communications, Networking and Information Security, 2010
[19] L. Gao, Z. Lu, "Low-cost RFID security protocols survey", Cross Strait Quad-Regional Radio Science and Wireless Technology Conference, 2010