An Efficient RFID Tag-Reader Mutual Authentication Scheme

Yansheng Zhang, Dancheng Li, Zhiliang Zhu
School of Software Engineering
Northeastern University
Shenyang, China
{yszhang, ldc, zzl}@mail.neu.edu.cn

***Abstract*—Performing tag-reader mutual authentication to detect cloned fake tags and ward off malicious readers is a basic requirement of various applications of RFID technology. In this paper, we propose an efficient RFID tag-reader mutual authentication scheme that is light weight, simple to implement and can effectively alleviates tracking and replaying threats. This scheme also has good performance and scalability features and can support efficient ownership management of the RFID tag attached objects.**

Keywords-RFID; security; tag; reader; mutual authentication; ownership management

I. INTRODUCTION

RFID (Radio Frequency IDentification) Technology aims at identifying objects in an automated fashion. For this purpose, objects are labeled with basic microchips called RFID tags[1]. An RFID tag transmits data over the air in response to the interrogations by RFID readers. A powerful malicious reader can illegally snoop upon the tags attached to objects, tracking and inventorying individuals. A malicious reader can also scan and copy the data on a genuine tag, embed the same data onto a fake tag, and attach this fake tag to a counterfeit product. Therefore, performing RFID tag-reader mutual authentication to detect cloned fake tags and ward off malicious readers is a basic requirement for various applications of RFID technology.

However, the messages exchanged between tags and readers during the mutual authentication process can be eavesdropped by nearby malicious readers. If identifying information is somehow exposed in these messages, or these messages are relatively static, then tracking and inventorying are still possible. Further more, fake tags or malicious readers can replay the messages eavesdropped from the authentication process between genuine ones and mimic genuine counterparts in subsequent session. Therefore, messages exchanged between readers and tags need to be encrypted and keep changing from session to session.

Solutions for these issues can be based upon the principle of three-pass mutual authentication in accordance with ISO 9798-2, in which both participants in the communication check the other party's knowledge of a secret cryptological key. Even though confidentiality can be provided by implementing suitable application layer encryption, the cost and hardware constraints limit the amount of logic that can be accommodated. Various practical hardware constraints aware solutions have been proposed. But currently available solutions either do not provide satisfying security levels, or suffer from scalability issues when the number of tags issued by the system is very large[1,3].

In this paper, we propose an efficient new RFID tag-reader mutual authentication scheme. This scheme use a novel symmetric key encryption method to encrypt all messages exchanged between tags and readers during the authentication process. This encryption method is simple and lightweight, and combined with a session specific random numbers that keeps changing from session to session, a satisfying security level can be achieved. In this scheme, the online database is accessed by using only static values, so efficient indices can be created on the online database to provide good performance and scalability. And also, through the owner ids and owner keys stored in both the tags and the online database, this scheme can supports efficient ownership management over the tag attached objects.

The remainder of this paper is organized as follows. The principles and derived practical schemes of symmetric key mutual authentication are described in Sect. 2. The new authentication scheme proposed in this paper is described in Sect. 3, and its efficiency is discussed in Sect. 4. Finally, we conclude this paper in Sect. 5.

II. RELATED WORK

A typical three pass mutual authentication procedure begins with the reader sending a GET_CHALLENGE message to the tag, as shown in Fig. 1. A random number R_A is then generated in the tag and sent back to the reader. The reader now generates a random number R_B. Using the common secret key K and a common key algorithm e_k, the reader calculates an encrypted data block (token 1), which contains both random numbers and additional control data, and sends this data block to the tag. The received token 1 is decrypted in the tag and the random number R'_A contained in the plain text is compared to the previously transmitted R_A. If the two figures correspond, then the tag has confirmed that the two common keys correspond.

Another random number R_{A2} is generated in the tag and this is used to calculate an encrypted data block (token 2), which

contains R_B and control data. Token 2 is sent from the tag to the reader. The reader decrypts token 2 and checks whether R_B, which was sent previously, corresponds with R'_B, which has just been received. If the two figures correspond, then the reader is satisfied that the common key has been proven. By choosing suitable encryption method, satisfying security levels can be achieved[3].

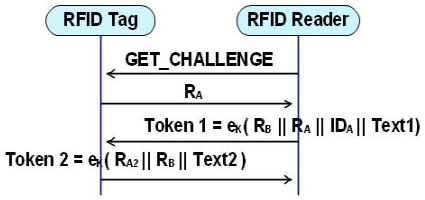

Figure 1. Symmetric Key Mutual Authentication

However, the costs and hardware constraints limit the amount of logic that can be accommodated. Various practical solutions derived from symmetric key mutual authentication have been proposed. These solutions refrain from using traditional cryptographic primitives, and doing just elementary arithmetic in tags. M^2AP (Minimalist Mutual Authentication Protocol) is a simple and practical symmetric key mutual authentication protocol[4].

M^2AP provides mutual authentication by using a common secret shared between authorized readers and tags. Each tag in M^2AP has a unique identification number (ID) that never changes. Also, each tag has an index-pseudonym (IDS) and four secret keys (K_1, K_2, K_3 and K_4) that must be updated after every authentication session. Before each authentication session, the reader generates 2 random numbers (n_1 and n_2). This protocol provides bitwise XOR ($\oplus$), bitwise OR ($\vee$), bitwise AND ($\wedge$) and addition modulo 2^{96} (+).

After receiving a hello message from a reader, a tag sends out its IDS to the reader. By means of IDS, the reader will be able to access the tag's 4 secret keys, and the tag's ID from the online database. Then messages are exchanged between the tag and reader to check mutually if they have common knowledge of the tag's 4 secrete keys. These messages are generated by calculating IDS, one of the 4 secrete keys, one of the 2 random numbers, with the operations provided by M^2AP. Among these 4 secrete keys, K1 and K2 are used to authenticate the reader, K3 and K4 are used to authenticate the tag. If the authentication is succeeded, new values for IDS, K1, K2, K3 and K4 are calculated and conferred between the tag and the online database.

However, M^2AP will suffer from the scalability issues. As above mentioned, in M^2AP, the reader use IDS to access the secrete keys of the tag stored in the online database. While the IDS value of a tag keeps changing from session to session, it is very difficult for the online database to maintain an efficient indexing structure to facilitate the process to search the secrete keys associated with a particular IDS. So, when the number of tags issued by the system is very large, performance can be significantly reduced. And also, because M^2AP only use simple bitwise operations to generate messages, it is relative vulnerable to attacks. The research represented in [2] shows that after two eavesdropping runs of M^2AP, the attacker can learn the identification number of the tag and some of the common secrets shared by the tag and the reader.

Other practical mutual authentication protocols currently available either do not provide a satisfying security level, or suffer from scalability issues as M^2AP. Some of the currently available schemes use static values to access the online database, but identifying information is contained in these values and directly exposed to eavesdroppers. Other schemes do not expose identifying information, but they also do not use static values to access the online database, therefore, suffering from performance and scalability issues as M^2AP.

III. New Mutual Authentication Scheme

In this section, we describe the new tag-reader mutual authentication scheme proposed in this paper. This scheme includes the specification of the sequence of messages exchanged between tags and readers, readers and the online database. The communication channel between tags and readers is insecure and attack vulnerable, and the communication channel between readers and the online database is highly secured. A novel encryption method is included in this scheme, for encrypting the messages exchanged between tags and readers during authentication processes.

A. Specification of the Messages

In this scheme, each tag is associated with an identification number (TID) that is static, so never change. A tag is also associated with a session random number (TSN) that is session specific and must be updated after each successfully completed authentication session. Each tag's TID and TSN are stored in the tag itself, and can also be accessed from the online database. Each individual or organization that is an owner of tag attached objects is associated with an identification number (WID) and a secrete key (WK). The WID and WK can be accessed from the online database by using the owner's user id (UID) and password (PWD). The WID and WK is also contained in each of the tags that are attached to the objects owned by the corresponding individual or organization. In the online database, relationships between WID and TID are created to represent the ownerships of owners to tag attached objects.

This new tag-reader mutual authentication scheme can be divided into four stages: reader initialization, reader authentication, tag authentication and new TSN value conferring. Fig. 2 shows the sequence of messages exchanged during the authentication process.

1) Reader Initialization

As shown in Fig. 2, step 1 to step 3 are included in the reader initialization stage. In this stage, the RFID reader uses the UID and PWD of an individual or organization to connect to the online database and retrieve the associated WID and WK. The WID and WK are used by the reader to execute mutual authentication with the tags. The communication channel between the reader and the online database is highly secured (SSL-TLS, EAP-TLS, and X.509 Authentication Framework). The PWD and UID are given by the individual or organization through an interface provided by the reader.

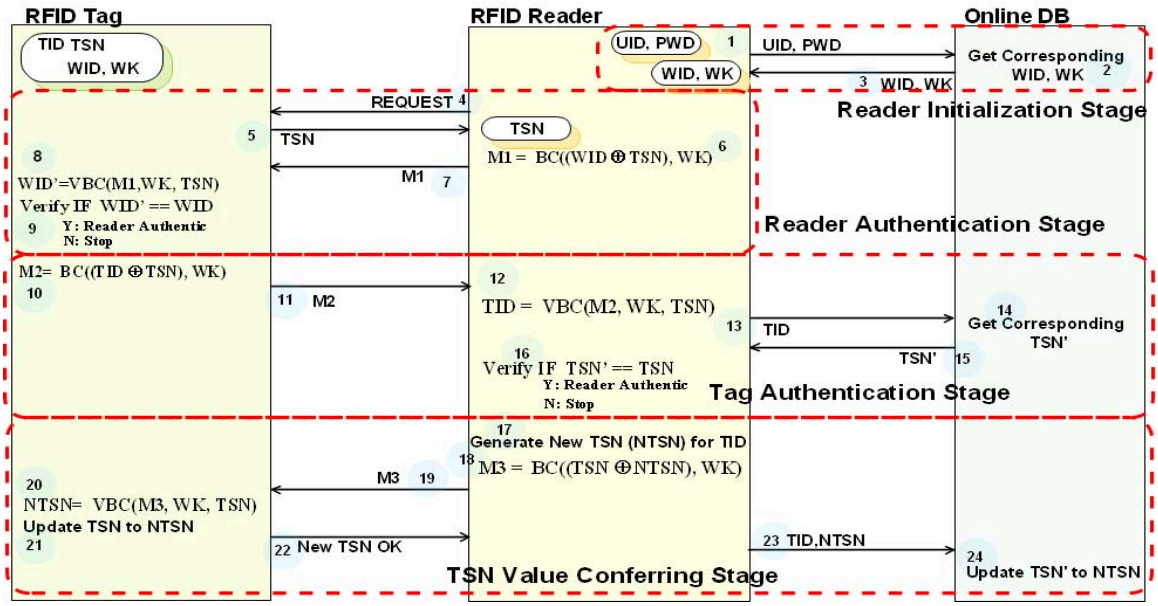

Figure 2. New Mutual Authentication Scheme

2) Reader Authentication

Step 4 to step 9 are belong to the reader authentication stage. To authenticate a reader, a tag responds to the reader's "REQUEST" message with its TSN. A tag's TSN value keeps changing from session to session, so tracking and replaying by eavesdropping on this message can be avoided. To authenticate itself to the tag, the reader need to send it's WID to the tag. The simple encryption method proposed in this paper, named BC in Fig. 2, is used to encrypt the WID. To avoid tracking threats, WID is randomized by TSN, and then encrypted by BC, with WK as the encryption key. M1 is generated as the result of the encryption and being sent out to the tag.

As mentioned above, each tag contains the TID and TSN associated with itself, and the WID and WK associated with its owner. M1 is decrypted in the tag by the reverse process of BC, named VBC in Fig. 2, with the tag's knowledge about WK and TSN. The resulted value WID' is compared with the WID. If they have the same value, then we can say that the reader is a legal one, otherwise, the tag should ignore the reader.

3) Tag Authentication

Step 10 to step 15 are belong to the tag authentication Stage. Once the reader is authenticated as a legal one, the mutual authentication process enters the tag authentication stage. In order to authenticate itself to the reader, the tag needs to send its TID to the reader. The TID is randomized by using the TSN, and then encrypted by BC, with the WK as the encryption key. Generated message M2 is sent to the reader. Because of the same reason as M1, M2 is safe to tracking and replaying threats. The reader then use VBC to decrypt M2 by using the TSN and the WK, and sends out obtained TID to the online database.

The online database is searched using the TID, and if the TID is a legal one, there would be an entry for it in the online database, and the corresponding TSN can be retrieved. This TSN is named TSN' to distinguish it from the TSN that is contained in the tag. TSN' is sent to the reader and compared with the TSN the reader obtained from the tag. If they have the same value, then we can say that the tag is a genuine one. Otherwise, the tag is a faked one and any following action should be discarded.

4) TSN Value Conferring

After a successfully completed mutual authentication, the TSN associated with the tag are needed to be updated to a new value. As have been noted above, TSN is stored in both the corresponding tag and the online database. So, to update the value of a TSN, both places are needed to be updated. A new TSN value named NTSN is generated in the reader. For sending NTSN to the tag, it is randomized by the TSN, encrypted by BC using the WK as the encryption key. Generated message M3 is sent to the tag, and is decrypted in the tag by VBC, based on the knowledge of the TSN and WK. Generated new TSN value is used to update the original TSN. If the tag have been successfully updated its TSN, it sends "NEW TSN OK" message to the reader. The reader then can update the copy of the TSN in the online database. After this update, the mutual authentication session is completed.

B. Novel Encryption Method

A novel encryption method is included in this scheme for encrypting the messages exchanged between tags and readers during the authentication process. This encryption method is very simple, but can provide a certain level of security while dose not impose heavy burden to tags. The essential idea of this encryption method is to change the values of adjacent bits in the data, according to the value of the corresponding bits in the encryption key. Fig. 3 shows how data is encrypted by using this encryption method.

As shown in Fig. 3, one bit in the encryption key determines whether or not to exchange the values of two adjacent bits in the data. If the bit in the encryption key is 1,

then exchange the values of the corresponding pair of bits in the data, otherwise, do not exchange.

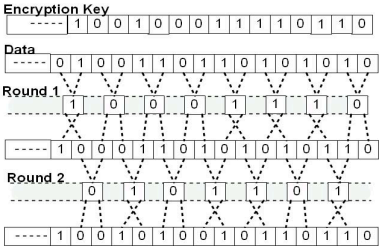

Figure 3. Encryption Method

For example, the values of the bits in position [1, 3, 5…] of the encryption key determine whether or not to exchange the values of the adjacent bits in position [(1,2), (3,4), (5,6)…] of the data, and the values of the bits in position [2, 4, 6…] of the encryption key determines whether or not to exchange the values of the adjacent bits in position [(2.3), (4,5), (6,7)…] of the data. The encryption is achieved through two round of bit value exchange, first round try to exchange the values of bits in position [(1,2), (3,4), (5,6)…], second round try to exchange the values of bits in position [(2.3), (4,5), (6,7)…].

IV. DISCUSSION

A. Security Level

The main security concerns in RFID tag-reader mutual authentication process are tracking and replaying threats. To alleviate these two threats, messages exchanged between tags and readers must satisfy two requirements. Namely, not expose any identifying information, and must keep changing from session to session.

As described in Sect. 3, in the proposed scheme, messages exchanged between tags and readers are "REQUEST," TSN, M1, M2, M3 and "NEW TSN OK." Message "REQUEST" and "NEW TSN OK" are common to all tags using this scheme, do not contain any information specific to a single tag, so can not be utilized for tracking and replaying. TSN is a session specific random number, do not contain identifying information and is updated after each successfully completed authentication session. So, TSN is also safe from tracking and replaying threats.

M1, M2 and M3 are encrypted messages, identifying information contained in these messages aren't exposed to the out world. And also, M1, M2 and M3 are all randomized by TSN values, so keep changing from session to session. If the encryption method is a fully secured method, then M1, M2, M3 is also safe from tracking and replaying threats. The novel encryption method proposed in this paper is light weight and simple to implement, while providing a certain level of security. But because it is not a fully secured method, M1, M2 and M3 are not fully free from tracking and replaying threats.

B. Performance and Scalabilty

Most of mutual authentication schemes access an online database when authenticating tags. If the values used to access the online database is static, then efficient indices can be created on theses values to facilitate the process for searching the online database. However, currently available schemes for mutual authentication either do not use static values to access the online database, so suffering from performance and scalability issues, or expose identifying information in the static values, so can not provide satisfying security level. In this scheme, the reader use TID to access the online database. TID is static for each tag, and when TID is transferred to the reader, it is randomized and encrypted. Therefore, good performance and scalability features can be provided while ensuring certain level of security.

C. Ownership Management

The mutual authentication scheme proposed in this paper also supports efficient ownership management. In this scheme, a tag authenticating a reader, is actually authenticating the owner of the object where the tag is attached. Each tag in this scheme contains a WID and WK, used to authenticate the owner. Each owner's WID and WK are stored in the online database and the ownerships between tags and owners are represented in the online database by relationships between the WIDs and TIDs. Ownership transfer can be easily managed in this scheme by update the WID and WK value stored in the tag, and then, updates the relationships between WIDs and TIDs to reflect the changes to the online database.

V. CONCLUSION

A new RFID tag-reader authentication scheme is proposed in this paper. This scheme is designed to effectively avoid tracking and replaying threats during mutual authentication process. This scheme use a novel symmetric key encryption method to encrypt all messages exchanged between tags and readers to ensure that no identifying information is exposed to eavesdroppers. Combined with a session specific random numbers that keeps changing from session to session, a satisfying security level can be achieved. And also, by using static tag ids to access the online database, this scheme can provides good performance and scalability feature. An owner id and owner key is contained in a tag and is used to identify the owner of the object where the tag is attached. Combined with the relationships between tag ids and owner ids established in the online database, efficient ownership management can be provided.

REFERENCES

[1] A. Juels, "RFID Security and Privacy: A Research Survey," IEEE JSAC, 24(2), pp. 381–394, February 2006.

[2] B. Michaly, B. Balazs, L. Peter, "Passive Attack Against the M^2AP Mustual Authentication Protocol for RFID Tags," The First International EURASIP Workshop on RFID Technology, Sep. 2007.

[3] F. Klaus, RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification, 2nd ed., John Wiley & Sons.

[4] P. L. Pedro, H. C. Julio, E. T. Juan and R. Arturo, "M^2AP: A Minimalist Mutual Authentication Protocol for Low-cost RFID Tags," International Conference on Ubiquitous Intelligence and Computing – UIC'06, vol. 4159, pp. 912–923, 2006.