Cryptographic Approach to “Privacy-Friendly” Tags

Miyako Ohkubo, Koutarou Suzuki and Shingo Kinoshita

NTT Laboratories
Nippon Telegraph and Telephone Corporation
1-1 Hikari-no-oka, Yokosuka-shi, Kanagawa-ken, 239-0847 Japan
{ookubo.miyako,koutarou,kinosita}@isl.ntt.co.jp

Abstract. Radio frequency identification (RFID) is expected to become an important and ubiquitous infrastructure technology. As RFID tags are affixed to everyday items, they may be used to support various useful services. However, widespread deployment of RFID tags may create new threats to user privacy, due to the powerful tracking capability of the tags.

There are several important technical points when constructing an RFID scheme. Particularly important is ensuring forward security, i.e., data transmitted today will still be secure even if secret tag information is revealed by tampering in the future. Low cost implementation is another key RFID requirement.

This paper discusses and clarifies the requirements and restrictions of RFID systems. This paper also examines the features and issues pertinent to several existing RFID schemes. Finally, this paper suggests the use of our previously proposed scheme, which protects user privacy using a low-cost hash chain mechanism.

1 Introduction

Ubiquitous network connectivity will allow society to conveniently access sophisticated services anywhere, at any time. RFID will most likely be part of such a connected society. RFID tags can be classified based on several points, such as the existence of an internal battery (active or passive), memory, memory capacity, internal circuits for security, e.g., an encryption processor, and the frequency used. Compared to using optical barcodes, RFID tags have many benefits. Especially noteworthy is the feature of mass identification. A group of tags with different IDs can be quickly and remotely polled for batch processing. RFID tags also make counterfeiting more difficult. Barcodes offer none of these properties. For this reason, RFID tags are expected to be the next-generation barcode, leading to new markets in various fields.

However, to receive the benefits of RFID tags, we must define and overcome security problems such as the violation of user privacy. While expectations for RFID are growing, more concerns are being raised about their use [10]. The issue of consumer privacy has drawn considerable attention and caused a attitude about RFID to prevail. Since privacy fears are beginning to gain ground [11][8], it is clear that addressing the privacy problems of RFID tags is essential to their success and proliferation.

The core problem is due to the basic functionality of RFID tags: each ID can be scanned remotely by anyone. Tags respond automatically to any reader and transmit their information indiscriminately. This property can be used to track a specific user or object over wide areas and is viewed as a barrier to the widespread adoption of RFID. Another barrier in RFID adoption is the high cost of current units; lower fabrication costs are essential. What is needed is an RFID identification scheme that offers adequate privacy protection at low cost.

Moreover, a forward secure RFID privacy protection scheme is necessary. Forward security is the property that privacy of messages sent today will be valid tomorrow. A future security compromise on an RFID tag will not reveal data previously transmitted. Any RFID privacy protection scheme must be efficient to ensure low cost, while still offering forward security.

We discuss the RFID scheme requirements for protecting user privacy and suggest the use of our proposed scheme [17] using hash functions. Through the use of a hash chain with two hash functions, the RFID scheme achieves all of the key requirements, even forward security.

This paper is constructed as follows. Section 2 describes the RFID system and its limitations. Section 3 describes the RFID privacy problem and the security requirements. It also examines the existing RFID systems. Section 4 describes our scheme and discusses its security properties and efficiency. Section 5 describes the application of our scheme to an Auto-ID system. The last section provides a summary of our results.

2 RFID System

The RFID system is an information tracking system that consists of wireless tag, T, wireless reader, R, and back-end database, B.
Tag : T is comprised of an IC chip and antenna, and sends information to the RFID reader in response to a wireless probe.
Reader : R is a device that transmits a radio frequency probe signal to T, receives the information sent by T, and sends the information to the back-end database, B.
Back-End : B is a secure server that has a database and manages various types of information related to each T, e.g., ID, reader location, read time, and temperature of sensor. B resolves the ID of T from the information sent by T through authenticated R.

2.1 Physical Conditions

Types of RFID: There are active tags, which have a battery, and passive tags, which have no battery. We focus on the passive tag, which is expected to be the most common type of RFID.

Cost: Generally, it is understood that a passive tag should cost no more than 5 cents. According to [24], to construct a 5- cent tag, the IC cost should not exceed 2 cents. This limits the number of gates to 7.5 to 15 K gates. A 100-bit EPC chip requires approximately 5 to 10 K gates [21]. As a result, the number of gates available for security cannot exceed 2.5 to 5 K gates.

Size of transmitted data: The transmission rate in the 13.56-MHz and 900-MHz bands available to the RFID tags is approximately 26 Kbps / 50 tags at 13.56 MHz and 128 Kbps /200 tags at 900 MHz. Assuming that tag reading should not exceed 1 second, each tag can transmit about 500 bits.

Strength against tampering: Tamper-resistant memory is too expensive for RFID use, so we must assume that the internal data from a tag may be leaked through physical attacks.

Communication: The wireless communications between the reader and the tag is assumed to be vulnerable to eavesdropping. Communications between the reader and the back-end database are conducted over a secure channel. The data in the database is secured by some form of access control.

Restriction of writing: Writing to the tag memory can be restricted by limiting the distance between the tag and the writing device, restricting the type or number of devices from which writing commands are accepted, or the orientation to the writing device. Moreover, writing data into the tag memory can be restricted by using methods such as passwords or requiring physical contact.

3 RFID Privacy Problem

3.1 Privacy Problems

To minimize cost, Class I tags [6] have no access control function. Thus, any reader can freely obtain information from a tag. Since communication between a tag and a reader is by radio, anyone can access the tag and obtain its output. Moreover, attackers can eavesdrop on the communications between tags and readers, which is a cause of much consumer apprehension.

The RFID privacy problem has two components. One is data leakage from RFID-tagged belongings. The second is behavioural tracking and personal identification by tracing tag IDs. Examples of these problems are given below.
Leakage of information of user belongings: In everyday life, people are prone to carrying various objects around with them. Some of them are quite personal, and provide information that the user does not want anyone to know about. Examples include money, expensive products, medicine (which

may indicate a particular disease), or books (which mirror personal consciousness and avocation). If such items are tagged, various personal details can be acquired without the knowledge of the owner.

Behavioural tracking and personal identification: If the consumer buys an item using a credit card and an adversary can link the credit card details with that purchased tagged item, the identity and movements of the consumer can be traced by tracking the ID of the tag. This problem is especially severe if the items are kept for a long time. To stretch the point a bit, this situation is similar to forcing the user to carry a tracking device.

3.2 Security Requirements

Several technical problems that must be overcome to secure RFID against these threats. One is ID anonymity. If the tag ID can be kept anonymous, the problem of leaking information pertaining to the user belongings would be solved. Another problem is to avoid adversary tracking. If the output of the tag is fixed, the adversary can easily track the tag, and thus the user. Therefore, the output of the tag should not be constant. However, only these ideas may be not enough to ensure privacy. We should consider the worst case, i.e., secret information in the tag, which is not usually output, is acquired by an adversary. We should construct a scheme that can prevent all attacks by the adversary. In any event, tracking of past events should be prevented by all means. Note that preventing future event tracking is still currently impossible. We suggest three requirements to protect RFID privacy:

Indistinguishability : Tag output must be indistinguishable from truly random values. Moreover, they should be unlinkable to ID of the tag. If the adversary can distinguish that a particular output is from a target tag, he can trace the tag. Naturally, this is included in the concept of ID anonymity.

Forward security : Even if the adversary acquires the secret tag data stored in the tag, he cannot trace the data back through past events in which the tag was involved. Needless to say, the adversary who only eavesdrops on the tag output, cannot associate the current output with past output.

One approach to satisfying the above security requirements is to use a public-key algorithm. However, public-key algorithms typically need a significantly powerful CPU, which increases the tag cost. Thus, any scheme that requires public-key calculations in a tag would be hard to adopt. It is very important to satisfy the above requirements at a low cost.

3.3 Related Work

Several papers have examined the protection of user privacy. We describe some of the related studies below.

Kill command feature, by Auto-ID Center [4] : Tags that the Auto-ID Center supports have the following property. Each tag has a unique 8-bit password, and upon receiving the password, the tag erases itself. This function is useful in protecting the user privacy, but a conscious decision is required to initiate the procedure, and it is difficult to ensure that the kill command was properly executed. Moreover, tag suicide prevents any subsequent useful services such as special services for each client. This property actually diminishes the benefits of RFID tags. Moreover, each password is only 8 bits long, so a malicious attacker may be able to determine some passwords in approximately 2^8 computations, and use this command abusively. This feature should be used with other protection schemes.

Hash lock scheme, by MIT [25] : This scheme is low cost, since all it requires is a hash function. Each tag verifies the reader as follows. The reader has key k for each tag, and each tag holds the result $metaID$, $metaID = hash(k)$ of a hash function. A tag receives a request for ID access and sends $metaID$ in response. The reader sends a key that is related to $metaID$ received from the tag. The tag then calculates the hash function from the received key and checks whether the result of the hash function corresponds to the $metaID$ held in the tag. Only if both data sets agree does the tag send its own ID to the reader.

Although this scheme offers good reliability at low cost, since $metaID$ is fixed, the adversary can track the tag via $metaID$. To avoid this, the $metaID$ should be changed repeatedly, however, operating the system in a way that satisfies this requirement in practice is difficult.

Randomized hash lock scheme, by MIT [25] : This is an extension of the hash lock type scheme. It requires the tag to have a hash function and a pseudo-random generator. Each tag calculates the hash function based on the input from pseudo-random generated, r and id, i.e., $c = hash(id|r)$. The tag then sends c and r to the reader. The reader sends the data to the back-end database. The back-end database calculates the hash function using the input as the received r and id for each ID stored in the back-end database. The back-end database then identifies the id that is related to the received c and sends the id to the reader.

The tag output changes with each access, so this scheme deters tracking. However, this scheme allows the location history of the RFID tag to be traced if the secret information in the tag is revealed, i.e., this scheme cannot satisfy the forward security requirement. Additionally it is said that a hash function can be achieved at low cost, however, a pseudo-random generator may be difficult to incorporate at low cost in this paper [25].

Anonymous ID scheme, by NTT [15] : In this scheme, the tag output is an anonymous ID; the adversary can never know the real ID of the tag. This is realized by using public-key encryption schemes or symmetric-encryption schemes or random value linked to tag's ID on external computation units. Since the tags use only RAM to hold the anonymous ID as sent from the reader, they are relatively inexpensive.

To use this scheme in practice, an authentication or secure channel must be established between the reader and the back-end database. Because the anonymous ID is fixed, tracking again becomes possible. Thus, it is necessary to change consciously the anonymous ID. It will be difficult to operate this system in practice.

External re-encryption scheme, by RSA Lab, etc [2] : This scheme uses public-key encryption. Tag data are rewritten at the request of the user using data sent from an external unit. This unit is necessary because public key encryption imposes heavy calculation loads that are beyond the ability of the tag. This task is usually done by the reader. The tags output seems random in each rewrite period, so an adversary who eavesdrops only on the tag output cannot trace the tag over long periods of time.

However, this scheme has the same problem as in [24] and [15]. The difficulty is that the data of each tag must be rewritten often, because the encrypted ID is constant.

XOR based one-time pad scheme, by RSA Lab [1] : This scheme needs only an XOR calculation, and so is very low cost. In this scheme, the reader (actually the back-end database) and the tag share a common list of random keys, and in some interactions they confirm that the partner has the common list. If the check passes, the tag sends its ID.

This scheme is very low cost, however, this scheme requires several interactions between the reader and the tag. Moreover, the common list must be overwritten completely as needed to ensure security. These points may make implementation difficult.

Other approaches : There are some other schemes such as [3] [22][13][20]. For example, [3] proposed a scheme that took a different viewpoint of reader authentication. The tag data are represented in tree form and an anti-collision technique is used. To prevent tracking, however, the user must take along a tool that can isolate the tag.

Besides the technical approaches as described above, sociological studies have been done. Over the last two years, The Auto-ID Center has examined operational policy and rules and has presented an encouraging plan[5]. Garfinkel proposed an RFID Bill of Rights that should be upheld when using RFID [14]. We restate the conditions in [14] as follows.

Users of RFID systems and purchasers of products containing RFID tags have: I) The right to know if a product contains an RFID tag. II) The right to have embedded RFID tags removed, deactivated, or destroyed when a product is purchased. III) The right to first class RFID alternatives: consumers should not lose other rights (e.g., the right to return a product or to travel on a particular road) if they decide to opt-out of RFID or exercise an RFID tag's "kill" feature. IV) The right to know what information is stored inside their RFID tags. If this information is incorrect, there must be a means to correct or amend it. V) The right to know when, where, and why an RFID tag is being read.

While schemes do exist that can partially protect the anonymity of the tag ID, none of them are very practical. Some of them such as [25], [2], and [15] require that the information in the tag be rewritten often. This demands conscious decisions and actions by one or more parties. Another scheme, [25], cannot satisfy the forward security requirement, i.e., once the secret information contained in the tag is known by an adversary, forward (past) tracing becomes possible.

4 Proposed RFID Privacy Protection Scheme

After reviewing the past work, we suggest five points for an approach to RFID scheme design. 1. Keep complete user privacy. 2. Eliminate the need for extraneous rewrites of the tag information. 3. Minimize the tag cost. 4. Eliminate the need for high power of computing units. 5. Provide forward security.

In this section, we describe our forward secure RFID privacy protection scheme.

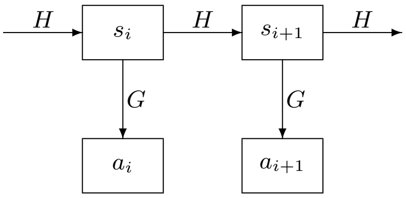

Fig. 1. RFID tag sends answer $a_i = G(s_i)$, and renews its secret $s_{i+1} = H(s_i)$.

To achieve forward security, we use the hash chain technique to renew the secret information contained in the tag. Initially tag has initial information s_1. In the i-th transaction with the reader, the RFID tag

1. sends answer $a_i = G(s_i)$ to the reader,
2. renews secret $s_{i+1} = H(s_i)$ as determined from previous secret s_i,

where H and G are hash functions, as in Figure 1. The reader sends a_i to the back-end database. The back-end database maintains a list of pairs $(ID,\ s_1)$, where s_1 is the initial secret information and is different for each tag. So the back-end database that received tag output a_i from the reader calculates $a'_i = G(H^i(s_1))$ for each s_1 in the list, and checks if $a_i \overset{?}{=} a'_i$. He find a'_i, $a'_i = a_i$, then return the ID, which is a pair of a'_i.

Our scheme satisfies the security requirements, i.e., indistinguishability and forward security, as follows. G is a one-way function, so if the adversary obtains tag output a_i, he cannot know s_i from a_i. G outputs random values, so if the adversary watches the tag output, he cannot link a_i and a_{i+1}. H is a one-way function, so if the adversary tampers with a tag and obtains the secret information in the tag, he cannot know s_i from s_{i+1}.

From the view point of efficiency, the proposed scheme is efficient enough to yield low-cost RFID tags, since it uses only hash operations that require a small gate size. Thus the proposed scheme is quite practical for low-cost RFID tags, while still ensuring privacy even in the face of tampering.

Furher details of our scheme can be found in [17]. The paper describes the security proof of the proposed scheme and discusses implementation issues as well as details of the scheme.

4.1 Security

This section discusses the security of our scheme. We assume that the adversary may eavesdrop on the radio frequency signals between the reader and the tag. He can also access the tag freely and collect the tag output. Moreover, the adversary can acquire the secret information stored in the tag by tampering the tag. The RFID scheme should be able to protect the user privacy against such an adversary.

To ensure the anonymity of the tag ID, obviously the tag should not output its ID nor should output any constant data. Our scheme satisfies this requirement. Moreover, this scheme offers the properties of indistinguishability and forward security.

We define indistinguishability for tag scheme. Adversary tries to distinguish between a random value and the i-th output a_i of tag. He can obtain the output of tag and back-end at any time.

Definition 1 (Indistinguishability). *Adversary $\mathcal{A}_{ind}$ performs the following game.*

- *Accesses tag oracle adaptively, sends i and receives a_i.*
- *Accesses back-end oracle adaptively, sends a_i and receives id or NG.*
- *If $\mathcal{A}_{ind}$ requests the problem with count number t, $b \in_U \{0,1\}$ is chosen randomly. If $b = 0$, $\mathcal{A}_{ind}$ is given a_t, which has been not given to $\mathcal{A}_{ind}$. If $b = 1$, $\mathcal{A}_{ind}$ is given r as a truly random value.*
- *$\mathcal{A}_{ind}$ guesses b and outputs b'.*

The advantage of $\mathcal{A}_{ind}$ is defined as follows.

$$Advantage_{\mathcal{A}_{ind}} = |Pr[b' \leftarrow \mathcal{A}_{ind}, b = b'] - \frac{1}{2}|$$

For the tag scheme, the advantage of any probabilistic polynomial-time adversary $\mathcal{A}_{ind}$ is negligible; therefore, We say that tag scheme is indistinguishable if and only if the advantage of $\mathcal{A}_{ind}$ is negligible.

It is assumed that the function G is random oracle. We have the following theorem.

Theorem 1. *If we assume that function G is a random oracle, the proposed scheme is indistinguishable.*

We define also forward security for tag scheme. Adversary tries to obtain the i-th internal secret information, s_i, of tag. He can tamper with tag and obtain internal secret information s_{i+1} and obtain the output of tag and back-end at any time.

Definition 2 (Forward security). *Adversary $\mathcal{A}_{forward}$ performs the following game.*

- *Accesses tag oracle adaptively, sends i and receives a_i.*
- *Accesses back-end oracle adaptively, sends a_i and receives id or NG.*
- *If $\mathcal{A}_{forward}$ requests to tamper the tag with count number t, $\mathcal{A}_{forward}$ is given s_t.*
- *$\mathcal{A}_{forward}$ output s_k, $k < t$.*

The advantage of $\mathcal{A}_{forward}$ is defined as follows.

$$Advantage_{\mathcal{A}_{forward}} = Pr[s_k \leftarrow \mathcal{A}_{forward}, H^{t-k}(s_k) = s_t]$$

For the tag scheme, the advantage of any probabilistic polynomial-time adversary $\mathcal{A}_{forward}$ is negligible; therefore, We say that tag scheme is forward secure if and only if the advantage of $\mathcal{A}_{forward}$ is negligible.

It is assumed that the function H is one-way, i.e., for given $z = H(s)$, it is intractable to find s', such that $z = H(s')$, and the function G is random oracle. We have the following theorem.

Theorem 2. *If we assume that function H is one-way, and that function G is a random oracle, the proposed scheme is forward secure.*

4.2 Efficiency

We consider the cost of our scheme. As described in Section 2.1, the number of gates available for security cannot exceed 2.5 to 5 K gates.

The proposed scheme requires the computation of two functions: H and G. If they can be implemented within this range at low cost, our scheme well suits the requirements for RFID privacy protection. Actually, interesting studies have examined low-cost hash functions [24][9]. Another study described a hash function that uses symmetric encryption algorithms [18]. It was reported that a symmetric encryption algorithm can be constructed with 6 to13 K gates [12]. So a hash function would be achieved in a similar manner as the symmetric encryption algorithms are achieved.

Furthermore, the performance of the back-end servers is described. A more efficient scheme can be constructed by making the reader send count number i with a_i and making the back-end server memory store the latest results of hash calculation s_i. By doing so, the performance cost of the back-end server can be cut and facilitate the calculation of a_i for all candidates in the database.

One possible threat to the back-end server is access from a corrupt reader. However, this adversary can be prevented by making the back-end server resolve IDs from only authenticated readers. Another possible threat is a Denial of Service attack (DoS attack). The operation cost of the back-end server must include decryption calculation as in [25], and must include exhaustive search as in [2]. In our scheme the back-end server requires an exhaustive search and calculation of hash functions for all candidates in the database, so the degree of adversity caused from a DoS attack maybe greater than that for existing schemes such as [25] and [2].

5 Applicability to Auto-ID System

5.1 Auto-ID System

Auto-ID Center [7], established in 1999, is a national institute with its head office located at MIT. The center believes that RFID tags will become the next-generation barcode and be used in various situations, i.e., manufacturing, distribution, retailing, and by the consumer at home. The Auto-ID Center has examined approaches for achieving low cost, formulated an ID system, and described information management. The Auto-ID Center has adopted a system that uses EPC codes for ID (64 or 96 bits), set only EPC in the IC chip, and other information is controlled by an external server on the network. The standard layout of EPC is shown in Fig 2. Additionally, the Auto-ID Center has examined the design of EPC and the language to describe the data called PML. The center has also implemented a service called ONS to identify the address of the PML server and the basic software to transmit EPC data efficiently.

5.2 Application to Auto-ID System

We show the application of our scheme to the Auto-ID System. In our scheme, the back-end database must hold data for all tags. Of course, database costs rise with the number of tags, therefore, scalability is an important point in our scheme. One solution to this problem is to prepare several back-end servers, divide the tag information among them, and add information that indicates the corresponding back-end server to the header of each tag output. This construction ensures that our scheme remains scalable.

Since the Auto-ID Center has examined the EPC code, we considered its impact on our scheme. The application of our scheme to the Auto-ID system is shown in Fig.2 We provide an example below. The application of our scheme to the Auto-ID system is shown in Fig.2

EPC Manager: Originally this area is for the manufacturer code. In our scheme this area can be used for information related to the back-end server.
Object Code and Serial Number: Originally this was for the tag ID. In our scheme this area can be used for a_i, the hash chain output.

By the following operation, we can resolve Extended-EPC.

1. The reader sends an extended-EPC to the ONS server.
2. The ONS server resolves the address of the back-end server and responds to the reader.
3. The reader sends extended-EPC to the back-end server.
4. The back-end server resolves the original-EPC and returns it to the reader.
5. Next, the basic protocol in our scheme is performed.

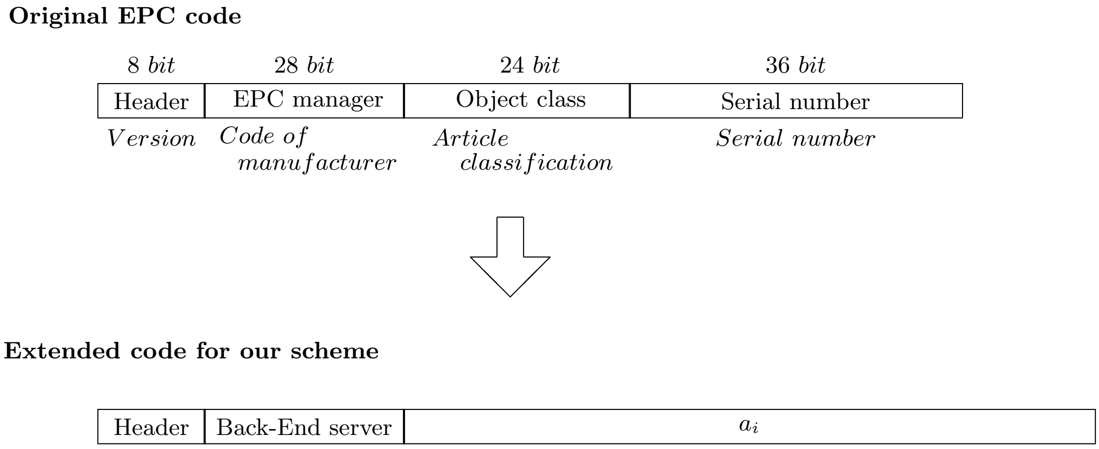

Fig. 2. Original EPC code and extended code for our scheme

In the extended code for our scheme, the code for the back-end server is a constant data. The area is just for load sharing and a back-end server is shared by an enormous number of tags, so the information of this area will not cause a breach of RFID privacy.

The EPC layout can be constructed without the area for the back-end server information, if a single back-end server controls all the tag IDs. However, the number of tags will steadily increase and result in a huge number. For scalability, it is preferred that the tag IDs are controlled by several back-end servers. Admittedly, an adversary can know an accessible back-end server to access a tag. However, invasion of privacy regarding the information on the accessible back-end server can be prevented by distributing the tag IDs uniformly among several back-end servers for control. So there is a trade-off relationship between security and scalability, Which is a common problem to existing schemes.

6 Conclusion

RFID tags have the possibility of revolutionizing society. While bringing to fruition their convenience, we must understand their risks. Implementing ubiquitous network connectivity in society will demand a close examination of personal privacy from both the technical aspects and social aspects. The privacy problems raised by their indiscriminate nature are, however, serious enough to demand a comprehensive and effective technique that can ensure user privacy while retaining their benefits.

While there are several existing schemes, none provide a complete solution. Some of them allow tag output to include relatively constant information, but not tag ID, which allows some form of tracking to take place. Some of them demand that data in the tag memory by rewritten to avoid tracking, and this requires a conscious decision by the user. Others fail to satisfy the forward security requirement.

We should design an RFID scheme that protects user privacy against an adversary who can eavesdrop, collect tag output, and acquire the internal secret data of a tag. As a recommended scheme, we described our scheme, which can protect user privacy while satisfying forward security, i.e., preventing backward tracking. Moreover, our scheme does not require arbitrary rewriting, and operation does not depend on external participants. Additionally only two hash functions are needed in our scheme, so the privacy of each tag can be protected at low cost.

References

1. Ari. Juels, “Privacy and Authentication in Low-Cost RFID Tags”, *submission 2003*
2. Ari. Juels, Ravikanth. Pappu, “Squealing euros: Privacy protection in RFID-enabled banknotes”, In Proceedings of Financial Cryptography - FC’03, 2003.
3. Ari. Juels, Ronald. L. Rivest and Michael. Szydlo, “The Blocker Tag : Selective Blocking of RFID Tags for Consumer Privacy”, In Proceedings of 10th ACM Conference on Computer and Communications Security(CCS 2003), Oct. 2003
4. Auto-ID Center, “860MHz-960MHz Class I Radio Frequency Identification Tag Radio Frequency & Logical communication Interface Specification Proposed Recommendation Version 1.0.0”, Technical Report MIT-AUTOID-TR-007, Nov. 2002
5. The detail in http://www.autoidcenter.org/research/CAM-AUTOID-EB-002.pdf, and http://www.autoidcenter.org/research/CAM-AUTOID-EB-006.pdf
6. UHF wireless tag, that Auto-ID Center has suggested to standardize http://www.autoidcenter.org/research/mit-autoid-tr007.pdf
7. MIT Auto-ID Center, http://www.autoidcenter.org.
8. Benetton undecided on use of ’smart tags’. *Associated Press*, 8 April 2003
9. S. Bakhtiari, R. Safavi-Naini, and J. Pieprzyk, “Cryptographic Hash Function: A Survey”, Technical Report 95-09, Department of Computer Science, University of Wollongong, July 1995
10. C.A.S.P.I.A.N, http://www.nocards.org
11. CNET, “Wal-Mart cancels ’smart shelf trial”, http://www.cnet.com, Jul. 2003
12. CRYPTOREC reports, published 2002 (in Japanese)
13. Philippe. Golle, Markus. Jakobsson, Ari. Juels and P. Syverson, “Universal re-encryption for mixnets”, 2002. In submission
14. Simson. L. Garfinkel, “Adopting Fair Information Practices to Low Cost RFID Systems”, Ubicomp 2002, Sep. 2002
15. Shingo Kinosita, Fumitaka Hoshino, Tomoyuki Komuro, Akiko Fujimura and Miyako Ohkubo, “Non-identifiable Anonymous-ID Scheme for RFID Privacy Protection”, *to appear in CSS 2003* in Japanese.
16. Michael Krause and Stefan Lucks, “On the Minimal Hardware Complexity of Pseudorandom Function Generators.”, In *Theoretical Aspects of Computer Science*, volume 2010, pp419-435, Lecture Notes in Computer Science, 2001
17. Miyako Ohkubo, Koutarou Suzuki and Shingo Kinoshita, “Forward-secure RFID Privacy Protection using Hash Chain”, submitted 2003.
18. Bart. Preneel, “Analysis and Design of Cryptographic Hash Functions”, PhD thesis, Katholieke University Leuven, Jan 1993
19. Michael Luby and Charles Rackoff, “How to Construct Pseudorandom Permutations from Pseudorandom Functions”, *SIAM Journal on Computing*, 17(2):pp373-386, Apr 1988
20. RSA Laboratories, “What is Secure ID ?”, 2003. *Available at http://www.rsasecurity.com/rsalabs/faq/5-2-5.html*
21. Sanjay E.Sarma, Stephen A. Weis and Daiel W. Engels, “Radio-Frequency Identification : Secure Risks and Challenges”, RSA Laboratories Cryptobytes, vol. 6, no.1, pp.2-9. Spring 2003
22. Sanjay E.Sarma, Stephen A. Weis and Daiel W. Engels, “Radio-frequency identification systems”, In Proceeding of *CHES ’02*, pp454-469. Springer-Verlag, 2002. LNCS no. 2523.
23. Sanjay E.Sarma, “Towards the five-cent tag”, Technical Report MIT-AUTOID-WH-006, MIT Auto ID Center, 2001. *Available from http://www.autoidcenter.org*
24. Stephen A. Weis, “Security and Privacy in Radio-Frequency Identification Devices”, Masters Thesis. MIT. May, 2003
25. Stephen A. Weis, Sanjay E.Sarma, Ronald L. Rivest and Daiel W. Engels, “Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems”, First International Conference on Security in Pervasive Computing, 2003. http://theory.lcs.mit.edu/ sweis/spc-rfid.pdf