HGLAP : Hierarchical Group-index based Lightweight Authentication Protocol for Distributed RFID system *

JeaCheol Ha[1], HwanKoo Kim[1], JeaHoon Park[2], SangJae Moon[2], Juanma Gonzalez Nieto[3], and Colin Boyd[3]

[1] Dept. of Information Security, Hoseo Univ., 336-795, Korea
{jcha, hkkim}@hoseo.edu
[2] School of Electrical Eng. and Computer Science, Kyungpook National Univ., 702-701, Korea
{jenoon65, sjmoon}@ee.knu.ac.kr
[3] Information Security Institute, Queensland Univ. of Technology, GPO Box 2434, Brisbane, QLD, 4001, Australia
{juamma, boyd}@isrc.qut.edu.au

Abstract. This paper presents a low-cost and secure authentication protocol to reduce the computational load on both the back-end database and the tags in a distributed RFID system. The proposed protocol is based on a hierarchical group-index to reduce the search time for a tag ID in the back-end database. Thus, when a tag is included in the k-th-level subgroup, the database system takes at most $(k+1) \cdot \sqrt[(k+1)]{m}$ hash operations to find the tag to be authenticated, where m is the number of tags. Furthermore, the proposed protocol also guarantees most security requirements, including robustness against replay and spoofing attacks, synchronization, and indistinguishability.

Keywords: RFID system, authentication, distributed DB, group-index, indistinguishability, traceability.

1 Introduction

A Radio Frequency Identification (RFID) system consists of three parts: the RFID tags, RFID reader, and back-end databases. The insecure channel which is caused by an RF interface between the RFID reader and the tags leaves an RFID system vulnerable to various attacks, such as eavesdropping, spoofing, a replay attack, traceability, and message interrupt attack. One solution to protect tags from attack is authentication between the tag and the reader. However, due to the low computational power and storage space of the tags, a lightweight authentication protocol is needed that takes account of the tag's implementation limitations and back-end server's capacity.

* This research was supported by the MIC of Korea, under the ITRC support program supervised by the IITA(IITA-2006-C1090-0603-0026).

Several attempts to resolve the RFID authentication problem between the tag and the reader have already been made using physical technologies, including the 'Kill command' [13], 'Active jamming' [7], and 'Blocker tag' [7] approaches. Meanwhile, in 2004, Weis *et al.* [12, 13] proposed a hash-lock protocol and randomized hash-lock protocol as cryptographic solutions. Yet, with the randomized hash-lock protocol, the identity of a tag, ID_k, is transmitted in the final step of authentication, making it vulnerable to a replay attack, spoofing attack, and location tracing. Dimitriou [1] also proposed a lightweight RFID authentication protocol that enforces user privacy and protects against cloning, however, there is no method for recovering synchronization when a state of desynchronization occurs. In 2006, Lee *et al.* [9] proposed an RFID mutual authentication scheme that introduces forward security(or forward traceability) to an RFID system, although finding the ID of a specific tag entails a heavy computational load on the back-end database. Plus, while the lightweight and resynchronous mutual authentication protocol proposed by Ha *et al.* [3] resolves the location tracing problem, forward security, a replay attack, and desynchronization attack, this protocol involves updating the tag's ID in each session, which is unsuitable for a distributed database environment.

Rhee *et al.* [11] proposed a challenge-response authentication protocol based on a hash function that is robust against spoofing and replay attacks, plus location privacy is also guaranteed. Meanwhile, Juels and Weis [6] independently suggested improvements to the hash-lock protocol, making it similar to the scheme proposed by Rhee *et al.*[11]. Although both schemes are robust to several attacks, the computational load on the back-end database is heavy when authenticating a tag. Finally, a hash-based efficient authentication protocol for a ubiquitous computing environment was proposed by Choi *et al.* [2]. Nonetheless, even though this protocol only requires one hash operation in a tag, it still has several security weaknesses due to the use of counter information.

Existing authentication protocols can be divided into two categories: ID-constant systems, in which a tag's ID is not updated, and ID-renewable systems, in which the ID can be changed to a new ID value in each session. In literature, the protocols presented by Dimitriou [1], Lee *et al.* [9], and Ha *et al.* [3] can all be categorized as ID-renewable systems, whereas the schemes developed by Rhee *et al.* [11] and Choi *et al.* [2] are ID-constant, making them suitable for a distributed database environment, where all the back-end databases use a unique ID.

Accordingly, this study presents a low-cost and secure mutual authentication protocol for a distributed database RFID system. The proposed protocol is based on a hierarchical group-index to reduce the search time for a tag ID in the back-end database. When a tag is included in the k-th-level subgroup, the database system only takes at most $(k+1) \cdot \sqrt[(k+1)]{m}$ hash operations to find the tag to be authenticated, where m is the number of tags. In addition, the proposed protocol also guarantees most security requirements, including robustness against replay and spoofing attacks, synchronization, and indistinguishability.

The rest of this paper is structured as follows. Section 2 explains the security properties of an RFID system. Section 3 then analyzes several previous RFID systems as regards their security and efficiency. The proposed a new authentication protocol based on hierarchical group-index is presented in section 4, and its security and efficiency examined in section 5. Some final conclusions are then given in section 6.

2 Security Properties in RFID System

The RFID reader interrogates the tags using an RF signal, then transmits the collected data to the back-end database. As such, the channel between the reader and the tag is insecure. The back-end database then receives data from the reader and transmits certain services to a specific tag, such as product and price information etc. However, the channel between the reader and the database is considered as secure. Thus, an attacker can eavesdrop on the messages between the reader and the tags due to the insecure channel, then use intermediate information or useful responses to perform various enhanced attacks. It is also assumed that an adversary has the capability to transmit various malicious messages to the tag or reader, thereby performing a spoofing or replay attack. The communication messages between the tags and the reader can also be interrupted by an attacker to block the service. As a result, a message interrupt attack can create a state of desynchronization between the tag and the reader, due to an abnormal closing of a session, message blocking, or different ID updating between the tag and the database. Therefore, the various security threats resulting from an insecure channel can be categorized as follows:

- **Information leakage:** One RFID privacy problem is information leakage about a user's belongings. For example, a user may not want certain information known by others, such as ownership of expensive products, identification of personal medicine, and so on.
- **Spoofing and replay attack:** After an adversary sends a malicious query to a target tag, they collect the responses emitted by the tag. The attacker can then impersonate the reader using the messages collected from the tag. Conversely, an adversary can replay the reader's query to impersonate the target tag. An attacker can also impersonate a legal tag or reader by replaying certain useful messages.
- **Desynchronization attack:**
  If the current ID for a tag is different to the one in the database, this is referred to as a state of desynchronization. Thus, if an adversary blocks certain messages transmitted between a tag and the reader, a dysynchronization state can be created in an ID-renewable RFID system. If the ID of a tag is desynchronized, the tag can be easily traced, as one of the values emitted from the tag will be constant, thereby compromising the location privacy.
- **Location tracing attack:** Here, an adversary can obtain some useful information on a tag's location. This attack is essentially applied to a rigid RFID

system in which certain communication messages between the tag and the database are identical to those used in the previous session.

Consequently, various security requirements are needed for secure RFID authentication, as identified in previous literature [5, 9, 12]. The information leakage problem can be easily solved by using an anonymous ID for each product, then checking whether it is in the database or not. Meanwhile, to prevent a spoofing or replay attack, the protocol should satisfy an authentication requirement, whereas a mutual authentication protocol is needed when an adversary has the ability to impersonate a tag or the reader. If a tag's response does not depend on any reader input, as shown in [13], the tag's messages can be used in a replay attack.

One of the aims of a desynchronization attack is to spoil a tag by disturbing the ID search in the database. The other powerful threat is location tracing by successive desynchronization. If an adversary continuously blocks certain legal messages in a wireless channel, they can find a historical trace. Then, even though the adversary does not know the tag's ID, they can still trace the target tag if certain specific message patterns for the tag are found, *e.g.*, transmitted data that is increased by one for every session using a counter. Thus, for perfect location privacy, an RFID system should satisfy both indistinguishability and forward security, where the former means that the values emitted by one tag should not be distinguishable from the values emitted by other tags, while the latter means even if an attacker obtains the secret data stored in a tag, the location of the tag can not be traced back using previous known messages, *i.e.*, disclosed data or communication information.

3 Analysis of Related Works

3.1 Lightweight Challenge-Response Protocol: LCRP

Dimitriou [1] proposed a lightweight challenge-response RFID authentication protocol(LCRP) that guarantees user privacy and protects against cloning. However, since an attacker can block the final message transmitted from the reader to the tag, this means the tag and back-end database update using different keys, where the back-end database renews the secret key, while the tag keeps the old value, resulting in a state of desynchronization and making the target tag useless. In addition, an attacker can trace a tag by successively sending a query from the reader in a desynchronization state. As the tag will respond with the same message $H(ID_i)$, since the ID_i is fixed in a desynchronized session, the tag cannot satisfy indistinguishability.

3.2 Synchronized Secret Information based Protocol: SSIP

Lee *et al.* [9] proposed an RFID mutual authentication scheme that utilizes a hash function and synchronized secret information. This scheme offers the

most enhanced security properties with respect to user privacy, including resistance against tag cloning by allowing an additional hash operation. In particular, they introduce forward security(or forward traceability) to an RFID system, and prove that their scheme is perfectly indistinguishable and almost forward secure. However, the back-end database is required to perform about m hash operations to find the specific ID related to a tag.

3.3 Lightweight and Resynchronous Mutual Authentication Protocol: LRMAP

Ha *et al.* [3] proposed an efficient RFID protocol to reduce the computational load on both the back-end database and the tags, while also guaranteeing most security requirements. Plus, in the case of desynchronization resulting from communication failure or a malicious attack, synchronization can be recovered between the database and a tag. However, the scheme is only suitable for a single database system, as the ID used in this protocol is renewable, as with the above two protocols.

3.4 Challenge-Response based Mutual Authentication Protocol: CRMAP

More recently, Rhee *et al.* [11] independently proposed a challenge-response authentication protocol based on a hash function that is almost the same as the improved randomized hash-lock scheme. This scheme is robust against a spoofing attack, replay attack, and location tracing attack. Nonetheless, the scheme is still vulnerable to forward security, as the ID is not changed with every session. Plus, this protocol is inefficient in terms of the computational load, as the back-end database is required to perform on average $m/2$ hash operations for an ID search, where m is the number of IDs.

3.5 One-way Hash based Low-cost Authentication Protocol : OHLAP

A computationally efficient RFID authentication protocol, OHLCAP, based on a hash function for a ubiquitous computing environment was proposed by Choi *et al.* [2]. Although this protocol only requires one hash operation in a tag, it still has certain security weaknesses, including the possibility of location tracing based on the leakage of counter information, an impersonation attack by maliciously updating a random number, and traceability based on a physically attacked tag [8, 4].

4 Hierarchical Group-index based Lightweight Authentication Protocol: HGLAP

4.1 Notations

The following notations are used for the entities and computational operations to simplify the description.

T	: RFID tag or transponder
R	: RFID reader or transceiver
DB	: back-end database or back-end server
$H(\ )$	: one-way hash function, $H : \{0,1\}^* \rightarrow \{0,1\}^l$
ID_i	: i-th identity of tag, l bits
GI^1	: first group-index of tag, l bits
GI^k	: last k-th depth group-index of tag, $GI^1 \supset GI^2 \supset ... \supset GI^k$
$r_R(r_T)$	: random number generated by reader(tag), l bits
$\{A^k\}$	: set of messages from A^1 to A^k, that is, $(A^1 \| A^2 \| ... \| A^k)$
$Query$	: request generated by reader
$B_R(B_L)$	: right(left) half of message B
$\oplus$	: exclusive-or(xor) operation
$\|\|$	: concatenation of two inputs

4.2 System Model and Assumptions

One of the disadvantages in an ID-constant RFID system is the computational load on the back-end database when searching for a tag to be authenticated. As such, the proposed protocol focuses on two design concepts: 1) an authentication protocol suitable for a distributed database system that is achieved using a constant ID and 2) a low-cost protocol with a low computational load for both the database and the tag.

The DB divides the tag identities into several groups. If the total number of tags is $m(= g_1 \cdot m_1)$, the DB divides the tags into g_1 groups, with m_1 tags in the 1st-level group. Thereafter, if the number of tags in the 1st-level group is $m_1(= g_2 \cdot m_2)$, the DB further divides the tags into g_2 groups, with m_2 tags in the 2nd-level group. Using this way of grouping, up to kth-level groups can be created. A tag is then included in the 1st-level group to the kth-level group, that is, $T \in GI^k \subset GI^{k-1} \subset ... \subset GI^1$ and $m = g_1 \cdot g_2 \cdot \cdot m_k$. For example, Fig. 1 shows the case where the group level $k = 2$, the number of tags $m = 30$, the number of 1st groups $g_1 = 5$, the number of 2nd groups $g_2 = 2$, and the number of tags in a 2nd group $m_2 = 3$. The data field of the DB is composed of the group-index $\{GI^k\}$ and the IDs for each tag. Thus, a tag has a data field, such as $\{GI^k\}$, and an ID.

Normally, it can be assumed that a distributed system is used for a large system, where the DB will include information on a large number of tags. Thus, the time taken to search for an ID in the DB is a very important factor related to the system performance. Therefore, the hierarchical group-index model is useful

1st GI	2nd GI	ID
GI_1^1	$GI_{1,1}^2$	ID_1
		ID_2
		ID_3
	$GI_{1,2}^2$	ID_4
		ID_5
		ID_6
...	...	ID_7
		...
GI_5^1	$GI_{5,2}^2$	ID_{30}

Fig. 1. Hierarchical Group-index in DB ($k = 2,\ m = 30,\ g_1 = 5,\ g_2 = 2,\ m_2 = 3$)

for a fast ID search in a DB, as it provides flexibility between the number of group levels and the computational costs, i.e., the more group levels GI^k, the lower the computational speed for an ID search.

4.3 Protocol Description

Thus, a secure authentication protocol is presented based on a k-level group-index. In the proposed protocol, a tag T is included in the first-level group GI^1, in the second group GI^2, which is a subgroup of the first group, and in the final group GI^k, i.e., $T \in GI^k \subset ... \subset GI^2 \subset GI^1$, where the parameter k means the subgroup level. Simply, if $k = 1$, a tag is only an element of the first-level group. Therefore, the group level is used to find a specific tag in the back-end database. Fig. 2 shows the process of the proposed HGLAP, and the following gives a detailed description of each step:

1. The reader sends a $Query$ and r_R to a tag.
2. The tag generates a random number r_T and computes $A^j = H(GI^j||r_R||r_T)$ for all $j = 1, 2, .., k$ for searching the ID and $B = H(ID||GI^k||r_R||r_T)$ for authenticating the tag in the DB. Then, the tag sends B_R, r_T, and $\{A^j\}$ to the reader.
3. The reader forwards B_R, r_T, and $\{A^j\}$ with r_R to the back-end database.
4. The back-end database finds GI^k by checking $A^j = H(GI^j||r_R||r_T)$ for all $j = 1, 2, .., k$, then finds the real ID in GI^k by checking B_R. The back-end database authenticates the tag by checking that the computed B_R equals the received one. If it is true then the back-end database sends the B_L to the reader as a response.
5. The reader forwards the B_L to the tag
6. The tag authenticates the reader by checking whether the received B_L equals the one computed in Step 2.

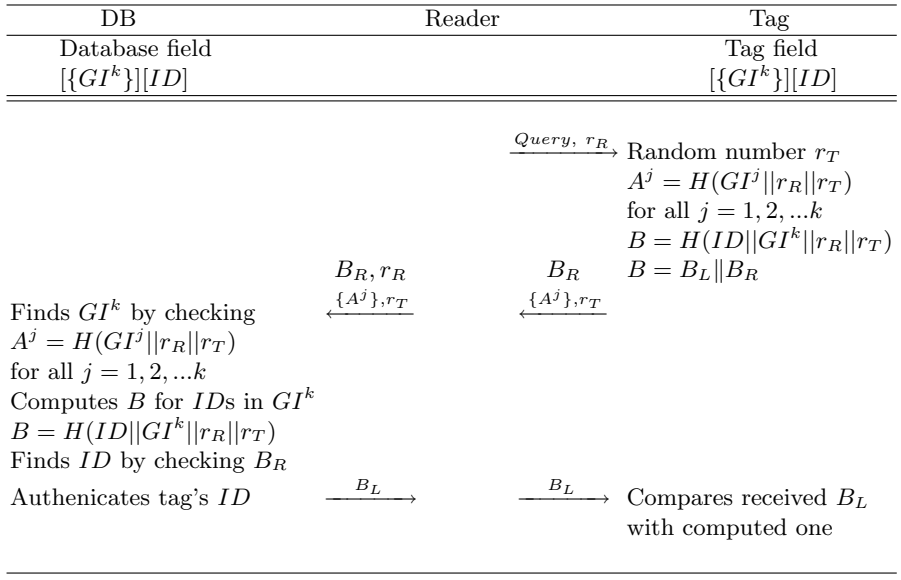

Fig. 2. Hierarchical Group-index based Authentication Protocol

5 Security and Efficiency Analysis

5.1 Security

The security of the proposed HGLAP was evaluated against the threats described in Section 2.

- **Information leakage:** To obtain secret information from a tag, an adversary must be able to guess the ID. However, an adversary cannot compute the ID from $B = H(ID||GI^k||r_R||r_T)$ or $A^j = H(GI^j||r_R||r_T)$ due to the security property of a one-way hash function.
- **Spoofing and replay attack:** Here, an adversary collects a tag's responses, then tries a spoofing attack by impersonating a legitimate tag. However, an adversary cannot compute the hashed messages A^j and $L(B)$ without knowing the ID. Meanwhile, it is also impossible to impersonate a reader, as an adversary must send the correct $R(B)$, which can not be computed without knowing the ID value. Furthermore, even though an attacker can send a constant random number r_R to a tag, a replay attack can not compromise the proposed protocol, as A^j or B is refreshed by including a random number r_T in each session.
- **Desynchronization attack:** In a desynchronization attack, if the message loss occurs due to an adversary, the proposed protocol allows the tag and reader to detect this. First, it is assumed that an adversary blocks the response messages transmitted from a tag, *i.e.*, step 2 in Fig. 2 or a message from the reader, *i.e.*, step 5. However, since desynchronization attacks generally occur in an ID-renewable system, due to desynchronization between

the back-end database and a tag, the proposed scheme is not affected by blocking, as it has a constant ID. Therefore, the back-end database and tag always maintain a synchronized state.

– **Location tracing attack:** The proposed protocol guarantees location privacy by randomizing the transmitted messages in each session. After the authentication is completely finished in the previous session, the tag sends A^j and $L(B)$ in response to a query in the current session. Thus, indistinguishability is satisfied as the values in the previous session have already been refreshed using two random numbers, r_R and r_T. As regards forward security, if it is assumed that an attacker obtains a tag's correct ID at some time, an adversary can then collect all the communication messages up to the time of obtaining the target secret ID, allowing the adversary to trace the past history of B, as the tag ID is not changed. Therefore, the proposed protocol cannot guarantee forward security, which is an inherent property of ID-constant systems.

A security comparison with previous authentication protocols is presented in Table 1. Therefore, with the exception of forward security, the proposed HGLAP was shown to be secure against most attacks, including a replay attack, spoofing attack, desynchronization attack, and location tracing attack.

5.2 Efficiency

When evaluating the computational load and storage costs for the DB and tag, as shown in Table 1, HGLAP exhibited a remarkable improvement in the computational cost for the DB. Although the challenge-response-based protocol [11] satisfies most security items, except forward security, its critical disadvantage is that the DB is required to perform $m/2 + 2$ hash operations to authenticate a tag. Thus, if it is assumed that a distributed RFID system is scalable and appropriate for a large system with lots of tags, the processing time required for the DB is a critical problem. As such, the computational cost in the DB has a trade-off relationship with the group-index level, where the higher the group-index level of a tag, the greater the storage space and computational load.

Yet, the computational cost in the DB can be reduced to at most $(k + 1) \cdot \sqrt[(k+1)]{m}$ hash operations to find a tag, $\frac{(k+1)\cdot \sqrt[(k+1)]{m}}{2}$ on average. Thus, the proposed protocol is very flexible as regards the computational cost in the DB and tag. When the group-index level k is just one, DB in the proposed protocol requires $\sqrt{m}$ hash operations. On the other hand, DB in OHLAP only requires just one hash operation. However OHLAP has serious security flaws, including vulnerability to spoofing attacks and indistinguishability.

With the proposed protocol, the storage size of the DB is $(k+1)l \cdot m$, where k is the length of an ID or hashed value and m is the number of IDs. Plus, a tag requires $(k+1)l$ bits of memory to store an ID and the GI value. The total length of the messages transmitted from a tag to the reader is $(k + 1.5)l$, while that from the reader to a tag is $1.5l$, except for a $Query$. Therefore, the proposed

HGLAP is suitable for a distributed RFID system with limited memory space and computational power.

Table 1. Comparison of security and efficiency

Protocol	LCRP[1]	SSIP[9]	LRMAP[3]	CRMAP[11]	OHLAP[2]	Proposed
Information leakage	O	O	O	O	O	O
Spoofing attack	O	×	O	O	×	O
Replay attack	O	O	O	O	O	O
Indistinguishability	×	O	O	O	×	O
Forward security *	△	△	△	×	×	×
Resynchronization	×	O	O	O	O	O
Hash # of DB	4	$\frac{m}{2}+3$	3^{**}	$\frac{m}{2}+2$	1	$\frac{(k+1)\cdot\sqrt[(k+1)]{m}}{2}$
Hash # of tag	4	3	3	2	1	$k+1$
DB's storage	$2l \cdot m$	$3l \cdot m$	$3l \cdot m$	$l \cdot m$	$4l \cdot m$	$(k+1)l \cdot m$
Tag's storage	l	l	$l+1$	l	$5l$	$(k+1)l$
Comm. load	$5l$	$4l$	$4l$	$4l$	$4l$	$(k+3)l$
Database	single	single	single	distributed	distributed	distributed

O : secure or supported △ : partially secure × : insecure or not supported
* : Systems marked with △ are ID-renewable, those marked with × are ID-constant.
** : $m+3$ required on average to recover synchronization

6 Conclusion

A lightweight and flexible authentication protocol, HGLAP, was proposed to reduce the search time in the DB. The proposed protocol is suitable for a large-scale distributed RFID system, as it uses a constant ID for a tag. Furthermore, the proposed scheme is based on a hierarchical group-index for a fast tag-search operation in the DB. As regards the computational cost, HGLAP is designed to reduce the computational load on both the back-end database and the tags. When analyzed for security against existing attacks, the proposed protocol was shown to guarantee untraceability, authentication, and robustness against replay and spoofing attacks.

References

1. T. Dimitriou.: A lightweight RFID protocol to protect against traceability and cloning attacks. *Security and Privacy for Emerging Areas in Communications Networks-2005. SecureComm 2005*, pp. 59-66, Sept., 2005

2. E. Choi, S. Lee, and D. Lee.: Efficient RFID Authentication Protocol for Ubiquitous Computing Environment, In *EUC Workshops 2005*, LNCS 3823, pp. 945-954, Springer-Verlag, 2005
3. J. C. Ha, J. H. Ha, S. J. Moon and C. Boyd.: LRMAP : Lightweight and Resynchronous Mutual Authentication Protocol for RFID System , In *proceeding of International Conference of Ubiquitous Convergence Technology, ICUCT'06*, Dec., 2006
4. J. C. Ha, S. J. Moon, J. Gonzalez Nieto, and C. Boyd.: Security Analysis and Enhancement of One-Way Hash based Low-Cost Authentication Protocol (OHLCAP), SSDU-2007 Workshop, Nanjing China, May, 2007
5. A. Juels.: RFID Security and Privacy: A Research Survey. *RSA Laboratories*, 2005.
6. A. Juels and S. A. Weis.: Defining strong privacy for RFID, *Cryptology ePrint Archieve, Report 2006/137,* Referenced 2006 at http://eprint.iacr.org, 2006.
7. A. Juels, R. L. Rivest and M. Szydlo.: The Blocker Tag: Selective Blocking of RFID Tags for consumer Privacy. In *Proceeding of 10th ACM Conference on Computer and Communications Security'03*, pp. 103-111, 2003.
8. D. Kwon, D. Han, J. Lee, and Y. Yeom.: Vulnerability of an RFID Authentication Protocol Proposed in at SecUbiq2005, EUC workshops 2006, LNCS 4097, pp. 262-270, 2006
9. S. Lee, T. Asano and K. Kim.: RFID Mutual Authentication Scheme based on Synchronized Secret Information. In *proceedings of the SCIS'06*, 2006.
10. Y. K. Lee and I. Verbauwhed.: Secure and Low-cost RFID Authentication Protocols, *2nd IEEE International Workshop on Adaptive Wireless Networks (AWiN)*, November, 2005.
11. K. Rhee, J. Kwak, S. Kim and D. Won.: Challenge-Response Based on RFID Authentication Protocol for Distributed Database Envirionment. *SPC'05*, LNCS 3450, Springer-Verlag, 2005.
12. S. E. Sarma, S. A. Weis and D. W. Engels.: Radio-Frequency Identification: Security Risks and Challenges. *RSA Laboratories*, Volume 6, No. 1, Spring, 2003.
13. S. A. Weis, S. E. Sarma, R. L. Rivest and D. W. Engles.: Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems. *Security in Pervasive Computing'03*, LNCS 2802, Springer-Verlag, 2004.