RFID Mutual Authentication Protocol based on Synchronized Secret

Hyunsung Kim

Dept. of Cyber Security, Kyungil University Kyungsan, Kyungbuk 712-701, Korea

kim@kiu.ac.kr

Abstract

Radio Frequency Identification (RFID) tags, due to their ability to uniquely identify every individual item and low cost, are well suited for supply chain management and are expected to replace barcodes in the near future. However, unlike barcodes, these tags have a longer range in which they are allowed to be scanned, subjecting them to unauthorized scanning by malicious readers and to various other attacks, including cloning attacks. Privacy and security concerns inhibit the fast adaption of RFID technology for many applications. A number of authentication protocols that address these concerns have been proposed but real-world solutions that are secure and maintain low communication cost are still needed and being investigated. Recently, Cho et al. proposed a hash-based RFID mutual authentication protocol using a secret value. However, this paper shows that Cho et al.'s protocol is weak against desynchronization attack and proposes a remedy mutual authentication protocol, which offers a high level of security based on hash operation with synchronized secret. The protocol is applicable to resource, power and computationally constraint platforms such as RFID tags. Our investigation shows that it can provide mutual authentication and untraceability as well as resistance to replay, denial-of-service and man-in-the-middle attacks, while retaining a competitive computation cost.

Keywords: *RFID, Privacy, RFID Tag Authentication, Desynchronization Attack, Security Protocol*

1. Introduction

Radio Frequency Identification (RFID) is a technology that enables the non-contact, automatic and unique identification of objects using radio waves [1-2]. RFID technology was first used in the Identify Friend or Foe (IFF) aircraft system during World War II. However, its use for commercial applications has recently become attractive with RFID technology seen as the replacement for the optical barcode system that is currently in widespread use [3]. RFID has many advantages over the traditional barcode. It can be applied to different objects, it provides read/write capability, it does not require line-of-sight contact with readers and more than one tag can be read at the same time [3-4]. These advantages have the potential to significantly increase the efficiency of decentralized business environments such as logistics and supply chain management particularly in the fields of inventory control, distribution and transportation [5-6].

The primary purposes of the security protocols in RFID systems are identification and authentication, which raises many privacy and security concerns [7]. As every compliant RFID reader can query the tag and retrieve its ID, third parties can track the bearers of an RFID tag. This inflicts a loss of privacy to humans, which is commonly referred as tracking or tracing. These security problems originate from objects, *i.e.*, tags, readers, and back-end server, of RFID systems and information leakage during the

communication process. These may be solved if a proper cryptographic algorithm is applied in the communication between the tag and the reader [8-11]. However, as the tag is small and low-cost, the hardware resources are limited. Therefore, it is difficult to apply ordinary cryptographic algorithms to RFID systems, and this blocks the wide usage of RFID systems. Some research efforts were being conducted to address these security problems of RFID systems [12-23]. Ohkubo *et al.*, proposed a high level of security protocol based on hash-based techniques, but the cost of searching tags by the back-end server is very inefficient [9]. On the other hand, the hash lock technique in [17] is sufficiently practical in terms of tag implementation and efficiency of the back-end server, but it has many security weaknesses. Particularly, the technique is vulnerable to indistinguishability. Tsudik proposed an authentication protocol called YA-TRAP, which provides tracking-resistant tag authentication through monotonically increasing timestamps on the tag [18]. YA-TRAP requires a pseudo-random number generator (RNG) from the tag and its basic version is vulnerable to denial of service (DoS) attack through timestamp desynchronization between the tag and the server. Furthermore, Tsudik proposes YA-TAP* as an enhanced version of YA-TRAP, which found that the protocol is still weak against replay attacks due to the lack of proving the reader's authenticity fully [19]. Chatmon *et al.*, proposed anonymous RFID authentication protocols based on YA-TRAP that provide anonymity for authenticated transponders and address some vulnerabilities of the original design, while increasing the server workload [20]. Many modifications of these two techniques have been proposed to solve the problems in them. However, they share common problems that when attackers make specific formats of requests or meaningless requests to tags, the tags make the same response or their information is easily leaked. Synchronization is another problem that needs to be solved. To solve those problems, Cho *et al.*, recently proposed two hash-based RFID tag mutual authentication protocols and argued that their protocols provide security requirements for RFID systems [21-22]. Kim showed that Cho *et al.*, protocol in [21] still has desynchronization problem [23].

In this paper, our main contributions are providing cryptanalysis focused on Cho *et al.*, mutual authentication protocol in [22] and proposing a RFID mutual authentication protocol based on hash operation and synchronized secret to solve the security problem in the previous protocol. First of all, security analysis is provided focused on desynchronization attack noted in Cho *et al.*, hash-based RFID mutual authentication protocol, which uses synchronization by using secret value shared between tag and back-end server. Attacker could perform desynchronization attack between two parties by simply modifying the final message from reader to tag. Thereby, legal tag could not be served by the back-end server, which is kind of attack focused on availability. After that, we will propose a RFID mutual authentication protocol based on hash operation and synchronized secret to solve the security problem in Cho *et al.*, protocol and to solve the privacy infringement and forgery problems in the previous RFID authentication protocols.

This paper is organized as follows. Section 2 defines adversarial model and security requirements in RFID system. Cho *et al.*, hash-based RFID mutual authentication protocol is reviewed and analyzed in Section 3. In Section 4, we propose a RFID mutual authentication protocol to cope with the security problem in Cho *et al.*, protocol. Security and performance analyses are provided in Section 5. Finally, Section 6 gives a brief conclusion.

2. Adversarial Model and Security Requirements

The characteristics of RFID can cause serious information leakage in the RFID system and the adversary can engage in various illegal behaviors by using the acquired information. This section defines adversarial model used in [24] and [25] and reviews security requirements for the RFID system used in [26-29].

2.1. Adversarial Model

We use the same adversarial model from [24] and [25] in this paper. An adversary takes advantage of any weakness in RFID systems to gain tag information for malicious use. Typical RFID security threats can be classified into two categories: privacy violation including eavesdropping, traffic analysis and location tracking, and forgery including replay attacks, spoofing attacks and physical attacks [24]. Privacy violation is divided into user privacy violation and location privacy violation. In user privacy violation, an adversary gains the tag's identity (ID) through various attacks and analyzes it to obtain the information about the RFID-tagged objects, such as product price and person's interest and taste. In location privacy violation, an adversary traces the location of people and objects via the illicit tracking of RFID tags. Privacy violation in RFID systems is a serious security risk because sensitive information is read directly from the tag without the knowledge or acknowledgement of the tag holder. Forgery, a primary security concern in RFID systems along with privacy violation, is a type of malicious acts that obtain authentication information to masquerade as a legitimate reader or tag. An adversary impersonating a legitimate reader or tag in the system counterfeits behaviors of a tag or a reader or falsifies tag contents to gain an illegitimate advantage [24-25]. In general, RFID readers suffer mostly from forgery attacks. For example, a reader unawarely communicates with an adversary who mimics an authentic tag or receives tag information that has been altered by unauthorized or unknown means.

2.2. Security Requirements

An RFID security scheme is assessed in various security requirements. Most commonly adopted RFID security requirements in practice are confidentiality, indistinguishability and mutual authentication [26-29]. When an RFID security scheme satisfies confidentiality and indistinguishability, it is considered "resilient to privacy violation." If the scheme satisfies mutual authentication, it is considered "resilient to forgery."

RFID security schemes can be grouped into two categories: schemes that interrupt tag recognition itself and those that utilize tag authentication protocols. Schemes in the first category can resist privacy violation attacks to some extent, but they decrease RFID systems' efficiency and applicability. In addition, adversaries can exploit schemes in the first category for malicious use [26-27]. In the second category, tag authentication schemes can be categorized according to the employed cryptographic primitive. Factors that one must consider in choosing a cryptographic primitive are security, efficiency and applicability. When the priority is given to efficiency and applicability, tag authentication schemes use lightweight operations like bit or mod operations as cryptographic primitives. Not surprisingly, tag authentication protocols employing lightweight cryptographic primitives provide high efficiency and applicability but they are weak in terms of security.

When security is most important, tag authentication protocols use traditional cryptographic operations such as symmetric or asymmetric key algorithms and hash

algorithms. An assumption here is that such traditional cryptographic algorithms are operable in tags or they will be operable in the near future if they are beyond the computational capabilities of current low-cost tags [28-29]. Authentication schemes based on traditional cryptographic primitives are relatively secure but they have increased tag retrieval complexity in the back-end server, *i.e.*, efficiency is decreased.

As mentioned earlier, one major challenge in building an RFID security scheme is a trade-off between security and efficiency. There have been many proposals that aim to address this issue but they often neglect problems associated with the special architecture of RFID systems. Most existing schemes utilize random numbers to provide indistinguishability and mutual authentication. A problem is that attackers can capture authentication messages and random numbers by eavesdropping on tag-reader communications and use the obtained information to carry out a brute-force attack. Moreover, it is possible that the complexity of RFID tag retrieval using authentication messages in the back-end server is equivalent to the complexity for performing the brute force. Despite the fact that the success of this attack depends on the adversary's computational power, this attack can be considered as a valid security threat to the RFID system. In other words, RFID systems are vulnerable to brute force attacks using the observed tag responses.

3. Review of Cho et al.'s Mutual Authentication Protocol

This section reviews Cho *et al.*, hash-based RFID mutual authentication protocol using a secret value [22]. The goal of their protocol is to design a protocol to demand high costs of acquiring the tag information for attackers.

3.1. Cho et al.'s Mutual Authentication Protocol

In Cho *et al.*, protocol, it is assumed that the communication between the reader and the back-end server is secure, while the communication between each tag and the reader is insecure. The notations used in the protocol are defined in Table 1. It is necessary to explain the method of generating RID_i and the role of s. First of all, RID_i is generated as follows:

- RID_i is the group ID of a random number. Each group is divided into random numbers based on a certain value, after sorting them sequentially. RID_i is made by using the minimum and maximum values of each group. If a random number is 96 bits, RID_i of each group is created using the most significant bit (MSB) (48 bits) of the minimum value and the least significant bit (LSB) (48 bits) of the maximum value. The detailed notations are as follows.

$$R_{0_{min}} \sim R_{0_{max}} : RID_0$$
$$R_{1_{min}} \sim R_{1_{max}} : RID_1$$
$$\vdots$$
$$R_{n_{min}} \sim R_{n_{max}} : RID_n$$

$$RID_i = R_{i_{min}}(0:47) \| R_{i_{max}}(48:95)$$

The range of each group is determined by s, *i.e.*, the secret value. RID_i and s are important factors needed to solve the privacy and forgery problems of the existing protocols. They change the output value of the tag each session without updating the s value.

Another important factor of Cho *et al.*, protocol is the blind factor β, which is computed as $\beta=s(0:47)||ID_k(48:95)$, where β consists of the MSB (48 bits) of s and the LSB (48 bits) of the tag's ID. The role of β is to conceal the random number R_t during transmission to the back-end server, in order to prevent disclosure with the $R_t \oplus \beta$ operation.

Table 1. Notations

Notation	Description
R	Random number (96 bits)
R_r	Random number generated by reader
R_t	Random number generated by tag
RID_i	Group ID of random number (96 bits)
ID_k	ID of tag k
s	Secret value (96 bits) mutually shared by back-end server and tag
s_j	Secret value used in the jth session
$h()$	Hash operation
$\|\|$	Concatenation operation
α	Message generated by tag for authentication
β	Blind factor
$\oplus$	XOR operation

The purpose of Cho *et al.*, protocol is authenticating an RFID tag using a secret value, which is shown in Figure 1. Each phase of this protocol is omitted in this section but the reader could check the detailed phases in [22].

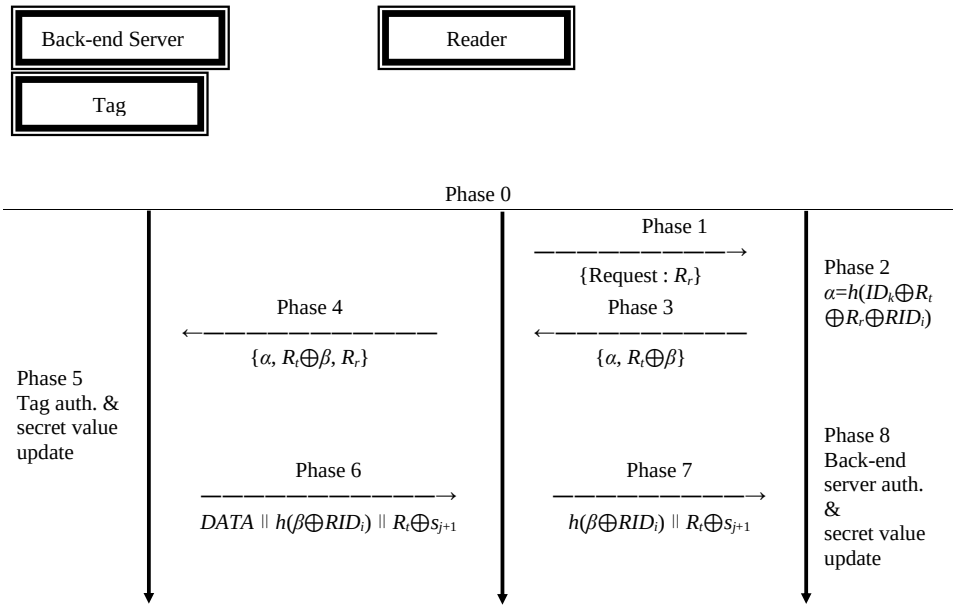

Figure 1. Cho *et al.*'s Mutual Authentication Protocol [22]

3.2. Desynchronization Attach in Cho *et al.*'s Mutual Authentication Protocol

Cho *et al.*, claimed that their protocol provides security requirements for RFID systems. However, we found that the protocol is vulnerable to desynchronization attack.

In Cho *et al.*, protocol, tag and back-end server share the same ID_k, s_j and s_{j-1}. The synchronized values make them to be authenticated each other by using session independent random numbers.

As a key is strongest at its weakest point, the protocol falls short of desynchronization by other adversarial means. Once the server and the tag are desynchronized then their method of authentication each other is jeopardized. An attacker, *A*, has no access to s_j and s_{j-1} but can still sabotage the protocol due to the message format of Phases 6 and 7, as explained below only focused on the three phases:

[A_Phases 6 and 7 : Back-end server → Reader → Tag**]** $\{DATA \| h(\beta \oplus RID_i) \| R_t \oplus s_{j+1}\} \rightarrow \{h(\beta \oplus RID_i) \| R_t \oplus s_{j+1}\}$

- The back-end server delivers the message $\{DATA \| h(\beta \oplus RID_i) \| R_t \oplus s_{j+1}\}$ to the reader.
- The reader acquires the information of the tagged object from the message received from the back-end server and sends the remaining message $\{h(\beta \oplus RID_i) \| R_t \oplus s_{j+1}\}$ to the tag.

** For the desynchronization attack, *A* can trap s_{j+1} from the reader to the tag in Phase 7, which uses a portion of the message $R_t \oplus s_{j+1}$. After *A* intercepts the message on Phase 7, he/she reforms the message as $\{h(\beta \oplus RID_i) \| V\}$ and submits to the tag as the original message on Phase 7, which *V* is a random number with 96 bits.

[A_Phase 8 : Tag**]** Back-end server authentication and secret value update

- The tag authenticates the back-end server via $h(\beta \oplus RID_i)$.
- The tag extracts the new secret value s_{j+1} from $R_t \oplus s_{j+1}$ with R_t used in the relevant session after the back-end server authentication.
- The protocol is finished after updating the secret value.

** When the back-end server authentication is completed, it extracts the updated secret value $s_{j+1}' = V \oplus R_t$ from *V* and updates not the correct secret values s_{j+1} but the corrupted secret values s_{j+1}'. Due to the attack process, *A*'s attack would be successful to desynchronize between the tag and the reader. Thereby, the legal tag could not be serviced from right after the attack session.

The desynchronization attack could derive the DoS from the server. This is a serious security breach.

4. RFID Mutual Authentication Protocol

The main design goal is to retain the high level of security and low communication costs of Cho *et al.*, while improving on its weakness of desynchronization attack. Cho *et al.*, protocol is susceptible to the attack because the message formats could be easily modified by attacker without noticing by receiver. Although there is no benefit to attacker from the attack, it is impossible for the attacked tag to prove the authenticity of itself to reader/back-end server. This allows a DoS attack due to the desynchronization attack. Providing message integrity code is necessary to avoid such attack. Thereby, we propose a RFID mutual authentication protocol based on secret synchronization and

hash operation by employing additional integrity parameter for message. Therefore, it is possible to transfer data secretly and only allow an authentic entity to success the protocol phases. The proposed protocol is shown in Figure 2. Each phase of this protocol is described in detail below.

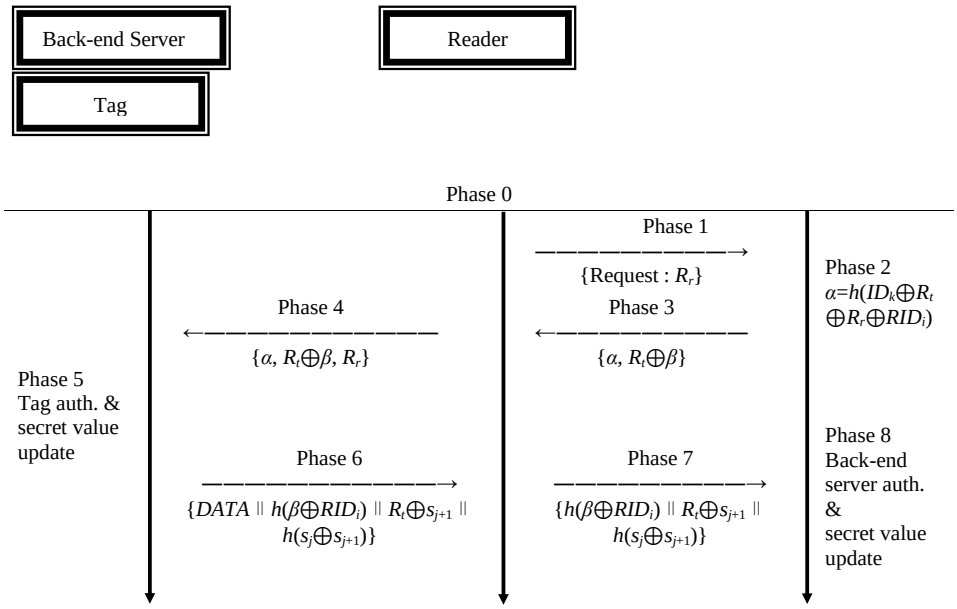

Figure 2. Proposed RFID Mutual Authentication Protocol

[Phase 0] Pre-phase

- The back-end server and the tag decide and share the message composing method for authentication and operating function.
  - The back-end server and the tag share the hash function for an operating function.
- The tag has its own random number generator.
- The reader has its own random number generator.
- The back-end server and each tag share the following information
  - Tag's ID: ID_k
  - Secret value of tag: s_j
- The back-end server composes the following table for each tag and the information shared with it:

Tag-ID	s_j	s_{j-1}	*DATA*

- The contents to be saved in the table are as follows.
  - *Tag-ID*: Unique identification information of the tag.
  - s_j : Secret value s_j of tag to be used in the current jth session.

◦ s_{j-1} : Secret value s_{j-1} of tag used in the previous session. Initially, the value is set to null.

◦ *DATA* : Information of the tagged object.

[Phase 1] Reader's request

- The reader generates a random number, R_r and sends a request to the tag.

[Phase 2] Generation of response message

- The tag generates a random number, R_t and also generates RID_i of group involving R_t. Each group is divided based on s_j, i.e. the secret value contained in the current tag. The RID_i is computed as follows :

$$RID_i=(R_t - R_t \bmod s_j+1)(0 : 47)\|(R_t + s_j - R_t \bmod s_j)(48 : 95)$$

- α is computed as follows:

$$\alpha=h(ID_k \oplus R_t \oplus R_r \oplus RID_i)$$

- It computes $R_t \oplus \beta$ right after β is computed as follows

$$\beta=s(0 : 47)\|ID_k(48 : 95)$$

[Phases 3 and 4] Transmission of response

- The tag sends the response message of $\{\alpha, R_t \oplus \beta\}$ to the reader.

- The reader resends the response message received from the tag after reforming the message by adding R_r, to the back-end server.

[Phase 5] Tag authentication and secret value update

- The back-end server searches the tag information in the DB with the information from the reader. The back-end server performs the following operations based on the saved information of each tag.

(1) It extracts ID_k and s_j of a tag in the DB

(2) It computes β using the extracted ID_k and s_j.

(3) It extracts R_t from $R_t \oplus \beta$ using the computed β.

(4) It computes group ID(RID_i') using the extracted s_j and R_t.

(5) It computes α' using R_r obtained from the reader, the computed RID_i' and the extracted R_t and ID_k.

(6) Repeats steps (1)~(3) until α' is the same as α.

- When the right tag is found, the back-end server must notify the relevant tag data to the reader and update the secret value of the tag. The back-end server performs the following steps.

(1) It updates the secret value s_j of tag. The new secret value s_{j+1} is generated at the back-end server's discretion ($s_j \rightarrow s_{j+1}$).

(2) It updates the DB of the relevant tag. It updates s_{j+1} and s_j into s_j and s_{j-1}, respectively.

(3) It forms a new message as follows

$\{DATA \| h(\beta \oplus RID_i) \| R_t \oplus s_{j+1} \| h(s_j \oplus s_{j+1})\}$

◦ $DATA$ is the tag information, which needs to be transmitted to the reader.

◦ $h(\beta \oplus RID_i)$ is the value used to authenticate the back-end server by the tag.

◦ $R_t \oplus s_{j+1}$ is the value used to securely transfer the secret s_{j+1} securely to the tag.

◦ $h(s_j \oplus s_{j+1})$ is the integrity value used to validate the new secret to the tag.

- If the back-end server fails to find the right tag, it is judged an abnormal authentication message and the session is terminated.

[Phases 6 and 7] Delivery of message generated by the back-end server

- The back-end server delivers the message $\{DATA \| h(\beta \oplus RID_i) \| R_t \oplus s_{j+1} \| h(s_j \oplus s_{j+1})\}$ to the reader.

- The reader acquires the information of the tagged object from the message received from the back-end server and sends the remaining message $\{h(\beta \oplus RID_i) \| R_t \oplus s_{j+1} \| h(s_j \oplus s_{j+1})\}$ to the tag.

[Phase 8] Back-end server authentication and secret value update

- The tag authenticates the back-end server based on the message received from the reader.

◦ The tag generates RID_i by computing with the random number, R_r received from the reader.

◦ The tag performs a hash operation using β and RID_i that have been generated.

◦ The tag authenticates the back-end server by confirming that the hashed value is identical to $h(\beta \oplus RID_i)$ that it received from the reader.

- When the back-end server authentication is completed, it extracts the new secret value from $R_t \oplus s_{j+1}$ and updates the secret value only if the extracted secret validation is successful by comparing the hashed operation using s_j and s_{j+1} with $h(s_j \oplus s_{j+1})$.

- If back-end server authentication fails, the secret values are not updated.

5. Security and Performance Analyses

This section provides two analyses focused on security and performance by comparing properties with related protocols in [19, 20], and [22].

5.1. Security Analysis

The proposed RFID mutual authentication protocol based on hash operation and synchronized secret offers a high resistance to most common attacks related to RFID systems. We analyze its security focused on the threat model mentioned in Section 2, which are focused on mutual authentication, eavesdropping and tracing resistance,

replay attack resistance, resistance to DoS attack, man-in-the-middle attack resistance and desynchronization attack resistance.

[F1] Mutual Authentication - The proposed protocol provides mutual authentication between the communicating parties. The reader gets authenticated by the tag if it can correctly computes RID_i from R_r by using s and a tag only gets authenticated if it correctly computes RID_i from R_t by using s. Moreover, it provides authentication between communication parties via $h(\beta \oplus RID_i)$.

[F2] Eavesdropping and Tracing Resistance - The proposed protocol has a high resistance to eavesdropping and tracing. The barrier against tracing is raised through the use of random numbers and anonymity. Therefore, no message that transports crucial information can appear twice due to the challenge/response mechanism by using session independent R_r and R_t. Eavesdropping is always possible but the information that can be gathered is minimized through hash and XOR operations combined with random numbers and synchronized secret related operations. In this way, an attacker would not have any valuable information from the message to pass authentication steps.

[F3] Replay Attack Resistance - The proposed protocol is secure against the replay attack. In the case of the message in Phase 3, the random number R_r from reader will affect α. The random number from reader will be changed every round. Replaying the message in Phase 7 is also impossible to attacker due to the message authentication code $h(s_j \oplus s_{j+1})$, which are related with the current and next synchronized secrets.

[F4] Resistance to DoS Attack - We will show that an attacker cannot simply drop or forge a last message in Phase 7 that is sent to the tag to desynchronize the secret value shared between the tag and the server. Although dropping the last message will make the tag's residents s_{j+1} unchanged, it will not affect the tag's next authentication. Since the server always keeps the previous and the current secret values s_{j-1} and s_j of the tag in the DB. Furthermore, forging the last message will not succeed since only the true server can send a valid acknowledge message $\{DATA \parallel h(\beta \oplus RID_i) \parallel R_t \oplus s_{j+1} \parallel h(s_j \oplus s_{j+1})\}$ to be accepted by the tag. Due to the one-wayness of the hash function and without the knowledge of RID_i, the attacker cannot figure out s_{j+1} in messages.

[F5] Man-in-the-middle Attack Resistance - The proposed protocol could resistance to DoS attack by using the message authentication code by applying hash operation in each message. Man-in-the-middle attack that simply replays messages can only be avoided by using the random numbers and secret related operations in each message.

[F6] Desynchronization Attack Resistance - For the desynchronization attack in the proposed protocol, attacker should know the secret related value s_j and could reform the forthcoming secret value s_{j+1} in the messages. Otherwise, attacker could just reform the forthcoming secret values s_{j+1} in the messages on Phases 6 and 7.

Table 2. Security Features of Various Protocols

Features / Protocols	F1	F2	F3	F4	F5	F6
YA-TRAP* in [19]	No	Strong	Weak	Medium	Yes	Yes
Chatmon et al. in [20]	No	Strong	Weak	Strong	Yes	Yes
Cho et al. in [22]	Yes	Strong	Strong	Weak	Yes	No
Our proposed	Yes	Strong	Strong	Strong	Yes	Yes

Table 3. Performance Features of Various Protocols

Features / Protocols	Computational Cost			Communication Rounds
	Tag	Reader	Back-end server	
YA-TRAP* in [19]	2H+3RNG	RNG	O(n)	4
Chatmon et al. in [20]	3H+RNG	RNG	O(2n)	2
Cho et al. in [22]	2H+2MOD+RNG	RNG	O(2n)	5
Our proposed	3H+2MOD+RNG	RNG	O(2n)	5

- H : Hash operation - MOD : Modular operation - RNG : Random number generation

Table 2 summarizes and compares the security analysis of various recent protocols as well as our protocol. It can be concluded that the proposed protocol provides a much higher level of security than any of the previously presented protocols.

5.2. Performance Analysis

We analyze the efficiency of the proposed protocol by comparing computational cost and communication rounds between related protocols. Table 3 shows the comparison between performance features, which shows that the proposed protocol does not increase the computational cost that much to solve the drawbacks in the existing protocols. In the proposed protocol, the tag requires one more hash operation due to the message integrity check compared with Cho *et al.*, protocol. Chatmon *et al.*, protocol is based on the look-up table and thereby requires two rounds communication, which does not provide mutual authentication. Most hash based protocols basically require five communication rounds for tag and back-end server authentication. The proposed protocol also needs the same rounds with Cho *et al.*'s protocol.

The design of the proposed protocol was based on the frame work of Cho *et al.*, protocol. Thus, the difference in structure and cost is not great. However, the existing protocols have serious vulnerabilities to various attacks. The proposed protocol solves the drawbacks in the existing protocols with minimal cost increases and structural changes.

6. Conclusion

RFID technology has overwhelming importance all over the identification spectrum. Thus, the system is being incorporated in many IT devices but a matter of concern is safeguarding privacy of its usage. There are many protocols proposed to address RFID security issues, one of which is Cho *et al.*, mutual authentication protocol. This protocol uses hash algorithm that supports by exchange of synchronized secret values between server and tag to provide security of their interrogations. They claim that the protocol provides security requirements for RFID systems.

However, this paper showed that Cho *et al.*, protocol is still vulnerable to desynchronization attack. Furthermore, we proposed a RFID mutual authentication protocol based on hash operation and synchronized secret to solve the security problem in Cho *et al.*, protocol. Attacker could perform desynchronization attack between two parties by simply modifying the final message from reader to tag in Cho *et al.*, protocol. Thereby, legal tag could not be served by the back-end server, which is kind of attack focused on availability. To cope with the problem, the proposed RFID mutual authentication protocol uses message authentication code in each message. This

protocol requires little computation and achieves both privacy and authentication, making it sufficient enough for use in supply chain management.

Acknowledgements

This work was supported by the Kyungil University Research Fund and was also partially supported by the National Research Foundation of Korea Grant funded by the Korean Government (MEST) (NRF-2010-0021575).

References

[1] A. Juels, "RFID security and privacy: a research survey", IEEE Journal on Selected Areas in Communication, vol. 24, **(2006)**, pp. 381-394.
[2] K. Finkenzeller, "RFID Handbook, second edition", Wiley & Sons, **(2002)**.
[3] R. Weinstein, "RFID: a technical overview and its application to the enterprise", IT Professionals, vol. 7, **(2005)**, pp. 27-33.
[4] K. Michael and L. McCathie, "The pros and cons of RFID in supply chain management", Proc. of ICMB'05, **(2005)**, pp. 623-629.
[5] X. Gao, Z. Xiang, H. Wang, J. Shen, J. Huang and S. Song, "An approach to privacy and security of RFID system for supply chain", Proc. of IEEE ICETDE04, **(2004)**, pp. 164-168.
[6] H. Gao, Y. Guo, J. Cui, H. Hao and H. Shi, "A Communication Protocol of RFID Systems in Internet of Things", International Journal of Security and Its Applications, vol. 6, no. 2, **(2012)**, pp. 91-102.
[7] A. Juels, "RFID security and privacy: a research survey", Selected Areas in Communications, vol. 24, no. 2, **(2006)**, pp. 381-394.
[8] S. Yeo and S. Kim, "Scalable and flexible privacy protection scheme for RFID systems", Lecture Notes in Computer Science, vol. 3813, **(2005)**, pp. 153-163.
[9] M. Ohkubo, K. Suzuki and S. Kinoshita, "Cryptographic approach to privacy-friendly tag", Proc. of RFID Privacy Workshop 2003, **(2003)**.
[10] J. Yang, J. Park, H. Lee, K. Ren and K. Kim, "Mutual authentication protocol for low-cost RFID", Proc. of the Workshop on RFID and Lightweight Cryptography, **(2005)**, pp. 17-24.
[11] S. A. Sarma, S. E. Weis and D. W. Engels, "RFID systems and security and privacy implications", Lecture Notes in Computer Science, **(2005)**, vol. 2523, pp. 17-24.
[12] A. Mitrokotsa, M. R. Rieback and A. S. Tanenbaum, "Classification of RFID attacks", Proc. of the 2nd International Workshop on RFID Technology, **(2008)**.
[13] M. Langheinrich, "A survey of RFID privacy approaches", Proc. of the IEEE International Conference on RFID, **(2008)**, pp. 58-64.
[14] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador and A. Ribagorda, "RFID systems: a survey on security threats and proposed solutions", Lecture Notes in Computer Science, vol. 4217, **(2006)**, pp. 159-170.
[15] M. Feldhofer, "An authentication protocol in a security layer for RFID smart tags", Proc. of the 12th IEEE Mediterranean Electrotechnical Conference 2004, vol. 2, **(2004)**, pp. 759-762.
[16] I. Vajda and L. Buttyan, "Lightweight authentication protocols for low-cost RFID tags", Proc. of the second Workshop on Security in Ubiquitous Computing 2003, **(2003)**.
[17] S. A. Weis, S. E. Sarma, R. L. Rivest and D. W. Engels, "Security and privacy aspects of low-cost radio frequency identification systems", Security in Pervasive Computing, vol. 2802, **(2003)**, pp. 201-212.
[18] G. Tsudik, "YA-TRAP: Yet another trivial RFID authentication protocol", Proc. of International Conference on Pervasive Computing and Communications 2006, **(2006)**, pp. 640-643.
[19] G. Tsudik, "A family of dunces: trivial RFID identification and authentication protocols", Proc. of the Symposium on Privacy-Enhancing Technologies, **(2007)**, pp. 45-61.
[20] C. Chatmon, T. v. Le and M. Burmester, "Secure anonymous RFID authentication protocols", Technical Report TR-060112, Florida State University, **(2006)**.
[21] J. S. Cho, Y. S. Jeong and S. O. Park, "Consideration on the brute-force attack cost and retrieval cost: A hash-based radio-frequency identification (RFID) tag mutual authentication protocol", Computers and Mathematics with Applications. doi:10.1016/j.camwa.2012.02.025.
[22] J. S. Cho, S. S. Yeo and S. K. Kim, "Securing against brute-force attack: A hash-based RFID mutual authentication protocol using a secret value", Computer Communications, vol. 34, **(2011)**, pp. 391-397.
[23] H. S. Kim, "Enhanced hash-based RFID mutual authentication protocol", Communications in Computer and Information Science, **(2012)**, vol. 339, no. 5, pp. 70-77.

[24] A. Mitrokotsa, M. R. Rieback and A. S. Tanenbaum, "Classification of RFID Attacks", Proc. of the 2nd International Workshop on RFID Technology, **(2008)**, pp. 73-86.
[25] M. Burmester and J. Munilla, "A Flyweight RFID Authentication Protocol", Proc. of Workshop on RFID Security, **(2009)**.
[26] M. Langheinrich, "A Survey of RFID Privacy approaches", Personal and Ubiquitous Computing, **(2009)**, vol. 13, no. 6, pp. 413-421.
[27] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. Estevez-Tapiador and A. Ribagorda, "RFID Systems: A Survey on Security Threats and Proposed Solutions", Lecture Notes in Computer Science, vol. 4217, **(2006)**, pp. 159-170.
[28] I. Syamsuddin, T. Dillon, E. Chang and S. Han, "A Survey of RFID Authentication Protocols Based on Hash-Chain Method", Proc. of Third International Conference on Convergence and Hybrid Information Technology, **(2008),** pp. 559-564.
[29] M. Hutter, M. Feldhofer and T. Plos, "An ECDSA Processor for RFID Authentication", Lecture Notes in Computer Science, vol. 6370, **(2010)**, pp. 189-202.

Authors

Hyunsung Kim he is an associate professor at the Department of Cyber Security, Kyungil University, Korea from 2012. He received the M.S. and Ph.D. degrees in Computer Engineering from Kyungpook National University, Republic of Korea, in 1998 and 2002, respectively. From 2000 to 2002, he worked as a senior researcher at Ditto Technology. He had been an associate professor from 2002 to 2012 with the Department of Computer Engineering, Kyungil University. His research interests include cryptography, VLSI, authentication technologies, network security and ubiquitous computing security.