情報保護學會論文誌
第16卷 第4號, 2006. 8

# 저가형 RFID 시스템에 강한 프라이버시를 제공하는 자체 재암호화 프로토콜*

박 정 수,† 최 은 영, 이 수 미, 이 동 훈‡
고려대학교 정보보호기술연구센터

# Self Re-Encryption Protocol (SREP) providing Strong Privacy for Low-Cost RFID System*

Jeong-su Park,† Eun-young Choi, Su-mi Lee, Dong-hoon Lee‡
Center for Information Security Technologies (CIST), Korea University

## 요 약

RFID (Radio Frequency Identification) 시스템은 다양한 서비스를 제공함으로써 유비쿼터스 시대에 매우 중요한 역할을 할 것으로 예상되지만, 이와 반대로 이 시스템이 폭넓게 사용될수록 정보 노출과 위치 추적 문제 같은 소비자의 프라이버시 문제가 발생될 것으로 예상된다. RFID 태그는 기존의 암호학적인 방법을 사용할 수 없을 정도로 제한된 연산 능력을 갖기 때문에, 이러한 프라이버시 문제를 해결하는데 많은 어려움이 있었다. [2]의 프로토콜에서는 외부기기를 사용하여 태그를 가진 소비자의 프라이버시를 보호하지만, 태그는 높은 자원이 필요한 지수 연산을 실행해야 했다. 하지만, 제안하는 프로토콜의 태그는 사용자의 프라이버시를 보호하기 위해서 곱셈연산만을 실행하기 때문에 저가형의 RFID 시스템에 적합하다. 또한, 태그는 외부장치 없이 자체적으로 재암호화되므로, 외부장치가 태그에 잘못된 값을 저장함으로써 발생될 수 있는 모든 문제를 근본적으로 해결하였다.

## ABSTRACT

RFID (Radio Frequency Identification) system is expected to play a critical role providing widespread services in the ubiquitous period. However, widespread use of RFID tags may create new threats to the privacy of individuals such as information leakage and traceability. It is difficult to solve the privacy problems because a tag has the limited computing power that is not the adequate resource to support the general encryption. Although the scheme of [2] protects the consumer privacy using an external agent, a tag should compute exponential operation needed high cost. We propose Self Re-Encryption Protocol (SREP) which provides strong privacy without assisting of any external agent. Our SREP is well suitable to low-cost RFID system since it only needs multiplication and exclusive-or operation.

**Keywords** : *Re-encryption, Privacy, Low-cost tag, RFID system, Ubiquitous environment*

접수일: 2005년 12월 27일 ; 채택일: 2006년 7월 4일
* 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 육성, 지원사업의 연구결과로 수행되었음.
† 주저자, standalon@korea.ac.kr
‡ 교신저자, donghlee@korea.ac.kr

## Ⅰ. 서 론

RFID 시스템은 RF통신을 이용하여 원거리에서 개체의 정보를 읽는 자동인식 기술이다. 이 시스템은 물리적인 접촉 없이 사물의 정보를 빠르게 읽을 수

있으므로, 바코드 시스템을 대체하게 될 시스템으로 주목 받고 있다. 최근에 이 시스템은 물류 시스템, 교통관리 시스템, 동물 관리 시스템 등등에서 부분적으로 사용되고 있다.

RFID 리더(이하 "리더"라 한다.)가 RFID 태그(이하 "태그"라 한다.)에게 질의(query)를 보내면, 태그는 자신의 정보를 리더에게 전송하고, 그 정보는 데이터베이스에게 전송된다. 이러한 방식으로 서비스 공급자는 태그를 갖고 있는 소비자의 정보를 언제 어디서나 쉽게 얻을 수 있고, 그러한 소비자에게 여러 가지 편리한 서비스를 제공하게 된다. 그러나 태그와 리더는 공개된 채널(insecure channel)에서 통신하기 때문에, 공격자는 전송되는 메시지를 도청할 수 있고, 이 메시지를 이용하여 신용정보, 구매 패턴, 건강정보와 같은 소비자의 민감한 정보가 노출될 수 있다. 또한 공격자는 이 메시지를 분석함으로써 태그를 가지고 있는 소비자의 위치를 추적할 수 있다.

RFID 시스템의 가장 큰 특징 중에 하나는 태그가 제한된 자원을 갖는다는 것이다.[1] 그러므로 많은 학자들은 태그의 연산량을 낮추는 동시에 소비자의 프라이버시를 보호하는 방법을 연구하였다. 결과적으로 물리적인 방법과 암호학적인 방법이 있고, 암호학적인 방법은 해쉬 함수를 이용하는 방법과 재암호화하는 방법으로 분류된다. 본 논문에서는 재암호화하는 방법으로 프라이버시 문제를 해결하였다. 즉, 리더가 태그에게 질의를 보낼 때마다, 태그는 자신의 정보를 자체적으로 재암호화한 후에 리더에게 전송한다. 그래서 공격자는 태그를 가진 소비자의 프라이버시를 침해할 수 없게 된다. 또한, 제안된 프로토콜에서 태그는 곱셈과 exclusive-OR(이하 "XOR"이라 한다.)과 같은 작은 연산을 사용하므로, 저가형 RFID 시스템에 적용할 수 있다.

### 1.1 기존 RFID 프로토콜

기존 RFID 논문에서 프라이버시 문제는 주요 논점이 되었고, 이러한 소비자의 프라이버시를 보호하기 위해 많은 기법이 제안되었다. 초기 논문에서는 'kill command'[12] 같은 물리적인 기법을 사용하였다. 'kill command'는 구입된 상품에서 태그의 정보를 지우고 다시 활성화시키지 않는 방법이다. 그 외에 물리적인 기술은 'blocker tag'[3], 'faraday cage'[5], 'active jamming'[3]등이 있다.

암호학적인 방법으로는 해쉬 함수(hash function)를 이용한 인증 (authentication) 프로토콜 기법이 있다. 'hash lock'[9-12]은 해쉬 함수의 특징인 일방향성(one-way property)을 이용하여 태그의 ID를 감출 수 있지만, 위치 정보는 노출된다. 'hash-chain'[6,7]은 위치 정보를 숨기면서 전방향 안전성(forward security)을 제공하지만, 태그의 정보를 확인하기 위해서 데이터베이스는 최대 태그의 개수만큼 해쉬 연산을 실행해야 한다. 또한 이 프로토콜은 재사용 공격과 스푸핑 공격에 취약하다.[4] 'challenge-response based RFID authentication protocol'[4]은 위치 정보를 숨기면서 재사용 공격과 스푸핑 공격에 안전하지만, 정당한 태그를 확인하기 위해 데이터베이스는 여전히 태그의 수만큼 해쉬 연산을 실행해야 한다.

또 다른 암호학적 기술은 재암호화 (re-encryption) 기법으로써, 공개키[1] 또는 대칭키[8]를 사용하여 태그의 정보를 주기적으로 재암호화하는 방법이다. 이 기법은 암호화를 통해서 태그 정보의 노출을 막고, 재암호화를 통하여 태그 사용자의 위치 정보를 숨긴다. 이 기법은 암호 시스템을 기반으로 하기 때문에 다른 프로토콜보다 훨씬 더 안전하다. 그러나 태그 자원의 한계성 때문에 태그에서 재암호화를 실행하는 것은 불가능하므로, 태그는 외부장치를 이용하여 재암호화를 실행한다. 그러나 태그는 다음 재암호화 단계를 실행할 때까지 리더의 질의에 대하여 항상 같은 값을 응답하므로, 위치 정보를 감추기 위해서는 외부장치와 빈번하게 재암호화를 실행해야 한다.

'Universal 재암호화'[8]는 키를 사용하지 않고 재암호화를 실행하는 방법이다. Saito 등은 외부장치로 가장한 공격자가 태그에 특별한 값을 저장함으로써, 이 프로토콜이 깨지는 두 가지 공격방법을 발견했다. 이러한 취약점을 보완하기 위해 Saito 등은 외부장치에 의해서 재암호화된 암호문을 태그에 저장하기 전에, 이 암호문의 확인 과정을 태그에 추가하였다[2]. 그러나 이러한 확인 과정을 위해서 태그는 지수연산과 같은 많은 연산이 필요하므로, 제한된 자원을 사용하는 태그에 적용하는 것은 매우 어렵다.

### 1.2 논문의 공헌도

**- 효율성**

앞에서 설명된 것처럼, 태그는 제한된 연산능력을

1) RFID 태그는 능동형 태그(active tag)와 수동형 태그(passive tag)로 나누어지는데, 본 논문의 모든 태그는 수동형 태그라고 가정한다. 수동형 태그의 특징은 제한된 연산 능력, 소형, 경량, 저비용 등등이 있다.

갖기 때문에 RFID 시스템에 적용하기 위한 프로토콜은 태그의 연산량을 최소화 하는 것이 중요하다. [2]의 프로토콜에서 태그는 지수연산과 같은 많은 연산을 해야 하기 때문에, 저가형 RFID 시스템에 부적합하다. 하지만, 제안하는 프로토콜의 태그는 단지 곱셈과 XOR연산만을 사용하여 암호화를 실행하기 때문에, 저가형 RFID 시스템에 적용하는 것이 가능하다.

**- 프라이버시**

기존 논문[1,2]의 재암호화 방식은 태그의 제한된 연산 능력 때문에, 태그 대신 외부 기기가 재암호화를 실행하고 생성된 암호문을 태그에 갱신하는 방법으로 실행되었다. 이처럼, 인증되지 않은 외부기기가 태그에 값을 갱신하므로, 공격자에게 많은 취약점이 노출되었다. 만일 공격자가 목표 태그의 재암호화 세션을 차단한다면, 리더의 질의에 대해서 태그는 항상 같은 값을 전송하므로, 공격자는 태그를 가진 소비자의 위치를 추적할 수 있다. 하지만 제안하는 프로토콜의 태그는 외부장치 없이 자체적으로 재암호화를 실행하기 때문에, 매 세션마다 변경된 형태의 메시지를 리더에게 전송한다. 그러므로 제안하는 프로토콜은 이전 프로토콜[1,2]보다 훨씬 더 강한 프라이버시를 제공한다.

이 논문의 나머지 부분은 다음과 같다. 2장에서는 RFID 시스템에서 발생될 수 있는 프라이버시 문제에 관하여 살펴본다. 그리고 3장에서는 본 논문에서 새롭게 수정된 RFID 시스템 구성요소의 특징과 역할에 대하여 살펴보고, 4장에서는 본 논문에서 제안하는 프로토콜에 대하여 자세히 설명한다. 5장에서는 제안하는 프로토콜의 안전성과 효율성을 평가하고, 6장에서 이 논문의 결론을 맺는다. 또한, 부록에서는 제안된 프로토콜에 시도-응답 기법을 적용하여, SREP의 재사용 공격과 스푸핑 공격에 대한 안전성을 설명한다. 그리고 SREP를 기존 RFID 시스템에 적용시키는 방법과, 마지막으로 시스템 환경에 맞게 그룹의 수를 결정하는 방법에 관하여 설명한다.

## Ⅱ. RFID 시스템과 프라이버시 문제

RFID 시스템에는 메시지 도청 (eavesdropping), 분석 (analysis), 변경 (alteration) 등등 다양한 공격 방법이 존재한다. 이러한 공격을 이용하여 공격자(adversary)는 태그의 민감한 정보를 노출시키려고 하고, 태그의 위치를 추적하여 태그를 가진 소비자의 프라이버시를 침해 하려고 할 것이다. 또한 제안된 프로토콜은 재사용 공격(replay attack)과 스푸핑 공격(spoofing attack)에 대하여 고려하지 않는다. 부록 1에서는 제안된 프로토콜에 시도-응답 기법(challenge-response technique)을 적용하여 이러한 공격에 안전하다는 것을 자세하게 설명할 것이다.

**- 정보노출 (Information leakage)**

유비쿼터스 환경에서 대부분의 사람들은 정보를 가진 태그를 여러 개씩 몸에 소지하게 될 것이다. 태그의 정보들 중에는 매우 개인적이고, 다른 사람들에게 노출되기를 원하지 않는 정보가 있을 것이다. 예를 들어, 대부분의 사람들은 자신들이 가진 약의 이름, 고가의 제품, 책의 이름 등이 다른 사람들에게 노출되는 것을 원하지 않을 것이다. 그러한 민감한 정보를 가진 태그가 폭넓게 사용된다면, 태그는 자신도 모르는 사이에 공개된 채널을 통해서 인증되었는지 알 수 없는 리더에게 그러한 정보를 전송하게 될 것이다. 그러므로 태그의 정보가 공격자에게 쉽게 노출되어, 태그 사용자의 프라이버시는 침해될 것이다.

**- 위치 추적 (Traceability)**

리더의 질의에 대하여 태그가 응답을 보낼 때, 공격자는 그 응답이 목표 태그의 응답인지 아닌지를 구별할 수 있다면, 태그의 위치 추적이 가능해 진다. 또한 공격자가 태그 응답의 내용을 이해 할 수 없어도, 목표 태그와 응답 사이에 어떤 관계(link)를 만들 수 있다면, 공격자는 목표 태그의 위치를 추적할 수 있다. 다시 말하면, 공격자는 태그를 가진 목표 소비자의 위치를 추적할 수 있게 된다.

## Ⅲ. SREP의 구성요소

일반적인 RFID 시스템은 RFID 태그, RFID 리더, 데이터베이스로 구성된다. 리더가 태그에게 질의를 보내면, 태그는 자신의 정보를 리더를 통해서 데이터베이스에게 전송한다. 그러나 제안된 프로토콜의 모든 태그는 여러 개의 그룹으로 나누어지고, 그 그룹들의 태그 정보는 여러 개의 데이터베이스에서 각각 그룹별로 관리된다. 제안하는 프로토콜은 어떤 그룹의 멤버 태그에서 생성된 암호문을 해당되는 그룹

데이터베이스로 분류하기 위한 새로운 구성요소를 추가하고 이를 "게이트웨이"라고 한다. 이제부터 제안 프로토콜인 SREP의 구성요소는 일반적인 RFID 시스템과 달리, RFID 멤버 태그 (이하 "멤버 태그"라 한다.), RFID 리더 (이하 "리더"라 한다.), 게이트웨이, 그룹 데이터베이스로 구성된다.[2][3] 그림 1은 SREP의 전체적인 모습이고, 각 구성요소들의 역할과 특징은 다음과 같다.

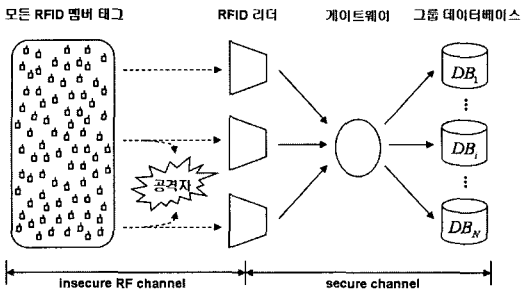


그림 1. SREP의 구성요소

**- RFID 멤버 태그 (RFID member tag)**

멤버 태그는 크기가 작고, 제한된 연산 능력을 가진 마이크로칩이다. 멤버 태그는 주위에 있는 리더의 질의에 대해서 자신의 정보를 전송한다.

**- RFID 리더 (RFID reader)**

리더는 안테나와 칩을 가진 무선통신 장치이다. 리더가 근처에 있는 멤버 태그에게 RF통신으로 질의를 보내면, 그 질의를 받은 멤버 태그는 리더에게 자신의 정보를 보내고, 그 정보를 받은 리더는 게이트웨이에게 단지 전송하는 역할만 한다. 리더는 멤버 태그에 비해 높은 연산 능력을 갖는다.

**- 게이트웨이 (Gateway)**

게이트웨이는 리더와 마찬가지로 높은 연산 능력을 갖는다. 게이트웨이는 메시지를 자신의 키로 각각 연산하여 모든 그룹 데이터베이스에게 전송한다.

**- 그룹 데이터베이스 (Group database)**

제안된 프로토콜에서 각 그룹 데이터베이스는 자신이 관리하는 그룹의 정보와 개인키를 안전하게 관리한다고 가정한다. 그룹 데이터베이스는 게이트웨이로부터 메시지를 받으면 자신의 개인키로 복호화를 실행한다.

## Ⅳ. 제안 프로토콜 (SREP)

제안하는 프로토콜은 초기화 단계와 실행 단계로 나누어진다. 초기화 단계는 제조업체에 의해서 안전하게 실행된다고 가정한다. 멤버 태그의 정보를 얻기 위해 리더가 멤버 태그에게 질의를 보내면 실행 단계가 시작된다.

### 4.1 초기화 (Setup) 단계

**- 1단계 (제조업체) 그룹화**

제조업체는 모든 멤버 태그를 여러 개의 그룹으로 랜덤하게 (randomly) 나눈다. 그룹의 수($N$)는 안전성과 효율성을 고려하여 적절하게 정해진다.[4]

**- 2단계 (제조업체) 키 생성**

G는 ElGamal 암호 시스템 기반의 그룹이라고 하고, $q$는 그룹 G의 위수(order)라고 하며, $g$, $s$는 그룹 G의 생성자(generator)라고 한다[8]. 제조업체는 각 그룹의 키 4개를 다음과 같이 생성한다.[5]

표 1. 제조업체가 생성하는 각 그룹의 키

| 키 | 표기법 | 계산식 |
|---|---|---|
| 그룹 개인키 | $x_i$ | $x_i \in_R \mathbb{Z}_{q-1}$ |
| 그룹 공개키 | $y_i$ | $y_i = g^{x_i}$ |
| 업데이트 공개키 | $u_i$ | $u_i = s^{x_i}$ |
| 게이트웨이 대칭키 | $GS_i$ | $GS_i \in_R \mathbb{Z}_{q-1}$ |

**- 3단계 (제조업체) 암호화**

$m_{i,j}$는 멤버 태그의 정보라고 한다.[6] 제조업체는

2) 본 논문에서 멤버 태그와 그룹 데이터베이스는 단지 그룹의 의미를 강조하기 위해 이름만 변경하고, 그 외에 기계적인 특성은 일반적인 RFID 시스템의 구성요소와 동일하다.

3) SREP는 기존 RFID 시스템의 구성요소에 게이트웨이를 추가하여 기존 RFID 시스템에 적용시키기 어렵다는 문제점을 갖는다. 부록 2에서는 SREP를 기존 RFID 시스템에 적용시키는 방법에 관하여 설명한다.

4) 부록 3에서는 그룹의 수를 결정하는 방법과 멤버 태그를 그룹으로 분류하는 방법에 관하여 자세하게 설명한다.

5) $key_i$는 $i$-번째 그룹의 키를 의미한다.

6) $info_{i,j}$는 $i$-번째 그룹의 $j$-번째 멤버 태그의 비밀값을 의미한다.

각 멤버 태그마다 새로운 임의의 값 $k_0$, $k_1$를 뽑아서 암호문 $C_{i,j}$와 업데이트 키 $UD_{i,j}$를 식 (1)에 의하여 계산하고, $C_{i,j}$, $UD_{i,j}$, $GS_i$를 각 멤버 태그에 안전하게 저장한다. 또한 제조업체는 모든 $GS_i$를 게이트웨이에 저장하고, 각 그룹의 개인키 $x_i$는 각각 해당되는 그룹 데이터베이스에 안전하게 저장한다.

$$\begin{aligned} C_{i,j} &= (\alpha,\beta) = (m_{i,j}y_i^{k_0}, g^{k_0}) \\ UD_{i,j} &= (\gamma,\delta) = (u_i^{k_1}, s^{k_1}) \end{aligned} \tag{1}$$

제안된 프로토콜의 구성요소에 저장된 비밀값은 표 2와 같다.

### 4.2 실행 (Execution) 단계

**- 1단계 (RFID 멤버 태그) 자체 재암호화**

아래의 식 (2)에 나타난 대로, 멤버 태그는 $UD_{i,j}$를 이용하여 기존의 암호문 $C_{i,j}$를 암호문 $C'_{i,j}$로 재암호화한다.

$$\begin{aligned} & C_{i,j} = (\alpha,\beta) \\ & UD_{i,j} = (\gamma,\delta) \\ & \quad \Downarrow \\ & C'_{i,j} = (\alpha',\beta') \\ & \quad = (\alpha\gamma, \beta\delta) \end{aligned} \tag{2}$$

표 2. SREP의 각 구성요소에 저장된 비밀값

| 구성요소 | $j$-번째 멤버 태그 | $i$-번째 그룹 | 암호문 | 업데이트 키 | 게이트웨이 대칭키 | 그룹 개인키 |
|---|---|---|---|---|---|---|
| RFID 멤버 태그 | $tag_3$ | $group_1$ | $C_{1,3}$ | $UD_{1,3}$ | $GS_1$ | - |
| | $tag_{11}$ | | $C_{1,11}$ | $UD_{1,11}$ | | |
| | ⋮ | | ⋮ | ⋮ | | |
| | $tag_4$ | | $C_{1,4}$ | $UD_{1,4}$ | | |
| | $tag_{23}$ | $group_2$ | $C_{2,23}$ | $UD_{2,23}$ | $GS_2$ | - |
| | $tag_7$ | | $C_{2,7}$ | $UD_{2,7}$ | | |
| | ⋮ | | ⋮ | ⋮ | | |
| | $tag_{52}$ | | $C_{2,52}$ | $UD_{2,52}$ | | |
| | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | - |
| | $tag_{97}$ | $group_N$ | $C_{N,97}$ | $UD_{N,97}$ | $GS_N$ | - |
| | $tag_n$ | | $C_{N,n}$ | $UD_{N,n}$ | | |
| | ⋮ | | ⋮ | ⋮ | | |
| | $tag_{35}$ | | $C_{N,35}$ | $UD_{N,35}$ | | |
| RFID 리더 | - | - | - | - | - | - |
| 게이트웨이 | - | - | - | - | $GS_1$ ≀ $GS_N$ | - |
| 그룹 데이터베이스 | - | $group_1$ | - | - | - | $x_1$ |
| | - | $group_2$ | - | - | - | $x_2$ |
| | - | ⋮ | - | - | - | ⋮ |
| | - | $group_N$ | - | - | - | $x_N$ |

**- 2단계 (RFID 멤버 태그) 그룹 대칭 암호화**

멤버 태그는 새롭게 재암호화된 암호문 $C'_{i,j}$를 $GS_i$와 식 (3)과 같이 XOR연산하여 암호문 $P_{i,j}$를 생성한다. 그리고 멤버 태그는 이 암호문 $P_{i,j}$를 리더를 통해서 게이트웨이에게 전송한다.

$$P_{i,j} = C'_{i,j} \oplus GS_i \quad (3)$$

**- 3단계 (게이트웨이) 게이트웨이 전송**

게이트웨이는 그림 2와 같이 전송된 암호문 $P_{i,j}$를 모든 $GS_i$와 각각 XOR연산하여 각 그룹 데이터베이스에게 전송한다.

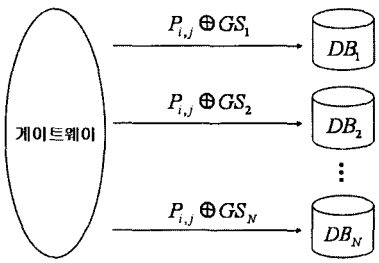


그림 2. 게이트웨이의 전송

**- 4단계 (그룹 데이터베이스) 복호화**

그룹 데이터베이스는 식 (4)와 같이 복호화하여 임시 메시지 $m$을 구하고, 이 메시지와 동일한 정보가 있는지 확인한다.

$$m = \alpha / \beta^{x_i} \quad (4)$$

## V. 제안 프로토콜의 안전성과 효율성 분석

### 5.1 안전성 (Security)

멤버 태그와 리더의 통신에서 수동적 공격자는 모든 암호문 $P$를 도청할 수 있고, 능동적 공격자는 이 암호문 $P$를 분석, 변경하여 정당한 리더로 인증되도록 할 것이다. 이 장에서는 2장에서 소개되었던 위협요소를 바탕으로 안전성을 분석해 본다.

제안된 프로토콜에서 공격자는 암호문 $P$와 그 암호문을 이용하여 계산된 형태만을 알 수 있다. 공격자는 암호문 $P$를 이용하여 멤버 태그의 비밀값 $C^i$, $UD$, $GS$를 알아내려고 할 것이다. 공격자가 활용할 수 있는 정보는 표 3과 같다.[7]

표 3. 공격자가 사용하는 메시지의 형태

| 공개된 채널에서 전송되는 암호문 $P$ | 암호문 $P$를 사용하여 계산된 형태 |
|---|---|
| $P^1 = C^1 \oplus GS$ | $C^1 \oplus C^2 (= P^1 \oplus P^2)$ |
| $P^2 = C^2 \oplus GS$ | $C^1 \oplus C^3 (= P^1 \oplus P^3)$ |
| $P^3 = C^3 \oplus GS$ | $C^1 \oplus C^2 \oplus C^3 \oplus GS$ |
| $P^4 = C^4 \oplus GS$ | $C^1 \oplus C^2 \oplus C^3 \oplus C^4$ |
| ⋮ | ⋮ |

만일 공격자가 어떤 멤버 태그의 비밀값들 $C^i$, $UD$, $GS$중에서 단지 한 가지 값만 알더라도, 그 멤버 태그의 암호문 $P$를 분석하여 다른 비밀값들도 차례로 알아낼 수 있다. 그러나 표 3에서 보는 것처럼, 공격자는 어떠한 비밀값도 개별적으로 알 수 없고, 다만 새롭게 재암호화된 암호문들($C^1$, $C^2$, $C^3 \cdots$)과 $GS$가 XOR로 계산된 형태만 알 수 있다. 그러므로 공격자는 멤버 태그의 비밀값들 중에서 어떠한 값도 얻을 수 없다.

**- 정보노출**

만일 공격자가 암호문 $C^i$를 안다고 할지라도, 제안된 프로토콜은 공개키 기반 구조이기 때문에, 멤버 태그의 정보를 알 수 없다. 즉 개인키 $x$를 가지고 있는 그룹 데이터베이스만이 태그의 정보를 얻을 수 있다.

**- 위치추적**

암호문 $P$에서 그룹 대칭키 $GS$는 고정되어 있지만, 암호문 $C^i$는 매 세션마다 재암호화되어 항상 변하므로 암호문 $P$는 항상 변한다. 공격자가 매 세션마다 변하는 모든 암호문 $P$를 얻더라도, 그 공격자는 2개의 다른 암호문 $P$가 같은 태그에서 왔는지를 결정할 수 없다. 그러므로 제안된 프로토콜에서 공격자는 목표로 하는 태그의 위치를 추적할 수 없다.

제안된 프로토콜은 정보노출과 위치추적 문제에 안전하므로 멤버 태그를 사용하는 소비자의 프라이버시를 보호한다.

이전 프로토콜[1,2]의 멤버 태그는 다음 재암호화 단계까지 항상 같은 값을 전송하기 때문에, 엄밀한 의미로 위치 추적이 가능하다. 하지만 제안된 프로토

7) $C^i$는 $i$-번째 세션에서 재암호화된 암호문이고, $P^i$는 $i$-번째 세션에서 전송된 암호문이다.

콜의 멤버 태그는 외부장치 없이 자체적으로 재암호화를 실행하기 때문에, 멤버 태그는 리더의 모든 질의에 대하여 항상 변경된 형태의 메시지를 전송하게 되고, 공격자는 멤버 태그를 추적하는 것이 완전히 불가능해진다.

공개키 기반의 재암호화 프로토콜은 멤버 태그의 제한된 연산 능력 때문에, 멤버 태그 자체에서 재암호화를 실행하기가 매우 어려웠고, 이를 위해 기존 프로토콜은 인증되었는지 알 수 없는 외부장치를 이용해야만 했다. 이처럼 외부장치가 멤버 태그에 값을 갱신한다는 것은 멤버 태그의 큰 취약점이 될 수 있다. 실제로, [2]에서는 리더로 위장한 공격자가 멤버 태그에 잘못된 값을 갱신하는 방법으로 공격하는 두 가지 모델이 있다. 그러나 제안된 프로토콜은 외부장치 없이 자체적으로 재암호화를 실행하므로, 인증되지 않은 외부장치가 멤버 태그에 잘못된 값을 갱신하여 발생하는 잠재적인 모든 문제들을 근본적으로 해결하였다.

### 5.2 효율성 (Efficiency)

앞에서 설명한대로 RFID 시스템은 멤버 태그의 연산량을 줄이는 것이 가장 중요하므로, 이를 중심으로 제안된 프로토콜을 평가해본다.

**- 멤버 태그의 연산량**

리더가 멤버 태그에 질의를 보내면 멤버 태그는 자체 재암호화와 그룹 대칭 암호화를 실행하고, 리더에게 암호문 $P$를 전송한다. 식 (2)과 (3)에서 알 수 있듯이, 멤버 태그는 자체 재암호화 단계에서 2번의 곱셈 연산을 실행하고, 그룹 대칭 암호화 단계에서 1번의 XOR연산을 실행한다.

[2]의 멤버 태그는 암호문을 확인하기 위해 지수 연산이 필요하다. 그러나 멤버 태그는 제한된 연산 능력을 갖기 때문에 이러한 프로토콜은 RFID 시스템에 적용하기 어렵다. 그러나 제안된 프로토콜은 곱셈 연산을 사용하기 때문에, [2]의 프로토콜보다 훨씬 효율적이다.

표 4. SREP의 효율성

| 프로토콜 | 연산량 |
|---|---|
| [2]의 프로토콜 | 지수 연산 |
| SREP | 곱셈 연산 |

## Ⅵ. 결 론

본 논문은 태그의 제한된 연산 능력을 사용하여 태그 사용자의 프라이버시를 보호하는 프로토콜을 제안하였다. 제안된 프로토콜의 태그는 외부장치를 사용하지 않고 자체적으로 재암호화하므로, 리더의 모든 질의에 대해서 항상 변경된 형태의 메시지를 전송한다. 제안된 프로토콜은 정보 노출 문제와 위치 추적 문제를 해결함으로써, 태그를 가진 소비자의 프라이버시를 보호한다.

제안된 프로토콜은 제한된 연산 능력을 가진 저가형 태그에 적합하도록 설계되었으므로, 멀지 않은 미래에 폭넓게 활용될 것이라고 기대해 본다.

## 참 고 문 헌

[1] Ari Juels and Ravikanth Pappu. "Squealing Euros : Privacy protection in RFID-enabled banknotes." IFCA, Springer-Verlag. *Financial Cryptography - FC'03*, volume 2742 of Lecture Notes in Computer Science, pages 103-121, January 2003.

[2] Junichiro Saito, Jae-Cheol Ryou, and Kouichi Sakurai. "Enhancing privacy of universal re-encryption scheme for RFID tags." *Springer-Verlag.* Embedded and Ubiquitous Computing - EUC 2004, volume 3207 of Lecture Notes in Computer Science, pages 879-890, August 2004.

[3] Ari Juels, Ronald Rivest, and Michael Szydlo. "The blocker tag : Selective blocking of RFID tags for consumer privacy." *ACM, ACM Press.* Conference on Computer and Communications Security - ACM CCS, pages 103-111, October 2003.

[4] Keunwoo Rhee, Jin Kwak, Seungjoo Kim, and Dongho Won. "Challenge-response based RFID authentication protocol for distributed database environment." *Springer-Verlag.* International Conference on Security in Pervasive Com

puting - SPC 2005, volume 3450 of Lecture Notes in Computer Science, pages 70-84, April 2005.

[5] mCloak : Personal/corporate management of wireless devices and technology, 2003. http://www.mogilecloak.com.

[6] Miyako Ohkubo, Koutarou Suzuki, and Shingo Kinoshita. "Cryptographic approach to "privacy-friendly" tags." In RFID Privacy Workshop, November 2003.

[7] Miyako Ohkubo, Koutarou Suzuki, and Shingo Kinoshita. "Efficient hash-chain based RFID privacy protection scheme." In International Conference on Ubiquitous Computing - Ubicomp, Workshop Privacy : Current Status and Future Directions, September 2004.

[8] Philippe Golle, Markus Jakobsson, Ari Juels, and Paul Syverson. "Universal re-encryption for mixnets." *Springer-Verlag*. The Cryptographers' Track at the RSA Conference - CT-RSA, volume 2964 of Lecture Notes in Computer Science, pages 163-178, February 2004.

[9] Stephen Weis. "Security and privacy in radio-frequency identification devices." Master thesis, Massachusetts Institute of Technology (MIT), May 2003.

[10] Sanjay Sarma, Stephen Weis, and Daniel Engels. "RFID systems, security and privacy implications." AutoID Center, MIT, Technical Report MIT-AUTOID-WH-014, 2002.

[11] Sanjay Sarma, Stephen Weis, and Daniel Engels. "RFID systems and security and privacy implications." *Springer-Verlag*. Cryptographic Hardware and Embedded Systems - CHES 2002, volume 2523 of Lecture Notes in Computer Science, pages 454-469, August 2002.

[12] Stephen Weis, Sanjay Sarma, Ronald Rivest, and Daniel Engels. "Security and privacy aspects of low-cost radio frequency identification systems." *Springer-Verlag*. International Conference on Security in Pervasive Computing - SPC 2003, volume 2802 of Lecture Notes in Computer Science, pages 454-469, March 2003.

## 부 록

### 1. 재사용 공격과 스푸핑 공격에 강한 SREP

제안된 프로토콜에서 재사용 공격과 스푸핑 공격은 멤버 태그와 데이터베이스의 상호 인증(mutual authentication)에 의해서 쉽게 해결될 수 있기 때문에, 앞에서 고려하지 않았다. 이 절에서는 SREP에 시도-응답 기법(challenge-response technique)을 적용하여 재사용 공격과 스푸핑 공격에 대한 안전성을 설명한다. 그림 3은 이러한 과정을 전체적으로 설명한 그림이다.

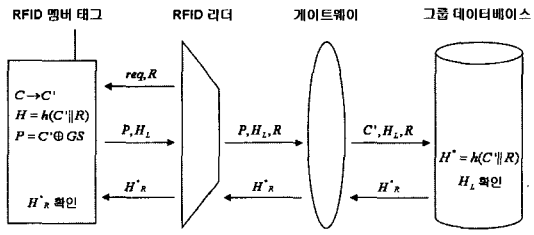


그림 3. 시도-응답 기법을 적용한 SREP

1. 리더가 멤버 태그에게 질의를 할 때, 임의의 값 $R$을 생성하여 함께 전송한다.
2. 4.2절에서 설명한대로, 멤버 태그는 자체 재암호화 단계를 실행하여 암호문 $C$를 $C'$로 재암호화한다. 태그는 $H=h(C'\|R)$을 계산하고,($h$는 일방향 해쉬 함수이다.) 그룹 대칭 암호화를 실행하여 암호문 $P$를 생성한다.
3. 멤버 태그는 암호문 $P$와 $H$의 왼쪽 반값인 $H_L$을 리더에게 전송한다. 리더는 임의의 값 $R$을 추가하여 $P$, $H_L$, $R$을 게이트웨이에게 전송한다.
4. 게이트웨이는 암호문 $C'(=P\oplus GS)$, $H_L$, $R$을 그룹 데이터베이스에게 전송한다.[8]

8) 이 절에서는 메시지를 전송한 멤버 태그의 그룹 데이터베이스만을 고려한다.

5. 그룹 데이터베이스는 게이트웨이로부터 받은 값들을 이용하여 $H^* = h(C' \| R)$을 계산한 후, 태그로부터 받은 $H_L$과 $H^*$의 왼쪽 반값을 비교하여 멤버 태그를 인증한다. 그리고 그룹 데이터베이스는 $H^*$의 오른쪽 반값인 $H^*_R$을 멤버 태그에게 전송한다.
6. 멤버 태그는 $H^*_R$을 자신이 계산한 $H$의 오른쪽 반값과 비교하여 그룹 데이터베이스를 인증한다.

멤버 태그는 매 세션마다 자체 재암호화 단계를 실행하여 암호문 $C$를 $C'$로 변경하기 때문에, 공격자는 멤버 태그와 리더사이의 모든 메시지를 안다고 할지라도, $H$를 구할 수 없다. 제안된 프로토콜은 시도-응답 기법을 사용하여 멤버 태그와 그룹 데이터베이스가 상호인증하기 때문에 재사용 공격과 스푸핑 공격에 대하여 안전하다.

## 2. 기존 RFID 시스템과의 호환되는 SREP

SREP는 게이트웨이와 그룹 데이터베이스를 통합함으로써 기존 RFID 시스템에 적용될 수 있다. 현재 표준화되어 있는 RFID 시스템의 구성요소는 RFID 태그, RFID 리더, 데이터베이스이지만, SREP는 이 세 가지 구성요소에 게이트웨이를 추가했기 때문에 기존 RFID 시스템과 상이하다는 문제점을 갖는다. 하지만, 게이트웨이와 그룹 데이터베이스들을 각각 모듈(module)로 구현하여 하나의 시스템으로 통합한다면,(이를 "게이트웨이 데이터베이스"라 한다.) SREP는 기존의 RFID 시스템과 마찬가지로 RFID 태그, RFID 리더, 게이트웨이 데이터베이스로 구성되므로 기존 RFID 시스템에 적용될 수 있다. 변형된 SREP의 구성요소는 그림 4와 같으며, 그림 1과 비교하여 살펴보자.

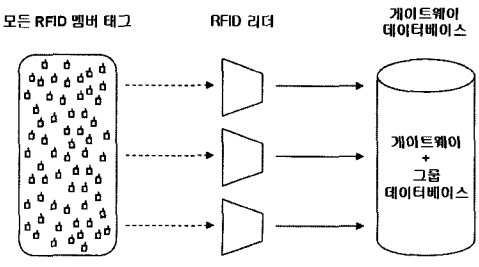


그림 4. 기존 RFID 시스템과 호환되는 SREP

게이트웨이 데이터베이스는 이름에서 알 수 있듯이 게이트웨이와 그룹 데이터베이스를 통합하여 두 가지 구성요소의 역할을 모두 수행한다. 가령, 초기화 단계에서는 게이트웨이가 저장하고 있는 모든 그룹의 게이트웨이 대칭키 $GS$와 각 그룹의 데이터베이스가 각각 저장하고 있는 그룹 개인키 $x$를 모두 게이트웨이 데이터베이스에 저장한다.(4.1절의 3단계 참조) 또한 실행단계에서는 게이트웨이 전송 단계와 복호화 단계를 게이트웨이 데이터베이스에서 실행한다.(4.2절의 3,4단계 참조)

## 3. 멤버 태그의 그룹화

그룹의 수($N$)를 정할 때는 안전성과 효율성을 고려하여 시스템 환경에 맞도록 결정되어야 한다. 표 5에서 볼 수 있듯이, 그룹의 개수를 증가시키면, 안전성은 높아지지만, 효율성은 낮아지고, 반대로 그룹의 개수를 낮추면, 안전성은 낮아지는 대신, 효율성은 높아진다.

표 5. 그룹의 수에 따른 안전성과 효율성

| 그룹의 수 ($N$) | 안전성 | 효율성 |
|---|---|---|
| 증가 | 증가 | 감소 |
| 감소 | 감소 | 증가 |

예를 들어 그룹의 개수를 증가시킬 경우, 각 그룹에 속해있는 멤버 태그의 수는 줄어들게 되므로 어떤 멤버 태그의 게이트웨이 대칭키 $GS$가 노출되었을 경우, 영향을 받는 멤버 태그의 수는 줄어들기 때문에 안전성이 높아진다. 하지만, 게이트웨이가 전송하는 메시지의 수는 늘어나고 이 메시지를 복호화하는 그룹 데이터베이스의 수도 증가하게 되므로 효율성이 감소하게 된다. 가장 높은 안전성을 위해서 그룹의 수를 멤버 태그의 수만큼 증가시킨다면, 어떠한 멤버 태그의 게이트웨이 대칭키 $GS$가 노출되더라도 영향을 받는 다른 멤버 태그는 없기 때문에, 즉 멤버 태그의 게이트웨이 대칭키 $GS$가 모두 다르기 때문에 높은 안전성을 보장한다. 반대로 그룹의 개수를 낮출 경우, 한 그룹에 속해있는 멤버 태그의 수가 상대적으로 많기 때문에, 한 멤버 태그의 게이트웨이 대칭키 $GS$가 노출되었을 경우, 영향을 받는 멤버 태그의 수가 많기 때문에 안전성은 떨어지게 된다. 하지만,

게이트웨이에서 전송하는 메시지의 수가 적어지고 필요한 그룹 데이터베이스의 수도 적기 때문에 효율성이 향상된다.

제조업체는 모든 멤버 태그를 그룹으로 나눌 때 멤버 태그의 정보에 상관없이 무작위로 (randomly) 나누어야 한다. 그래서 공격자가 어떤 멤버 태그의 정보를 보고 같은 그룹에 속한 다른 멤버 태그를 추측할 수 없어야 한다. 만약 멤버 태그의 정보를 그 정보의 유형대로 나누어서 공격자가 한 멤버 태그와 같은 그룹의 다른 멤버 태그를 추측할 수 있다면, 단지 하나의 멤버 태그의 비밀정보가 공격자에게 노출되었다 하더라도, 그 공격자는 노출된 멤버 태그의 정보를 보고 같은 그룹에 속한 다른 멤버 태그를 쉽게 예측함으로써, 그 그룹의 모든 멤버 태그의 비밀정보는 노출된다.

## 〈著者紹介〉


**박 정 수 (Jeong Su Park) 학생회원**
2002년 2월: 고려대학교 전산학과 학사
2006년 8월: 고려대학교 정보보호대학원 공학 석사
〈관심분야〉 정보보호, RFID 시스템

**최 은 영 (Eun Young Choi) 학생회원**
2001년 8월 : 고려대학교 수학과 학사
2003년 8월 : 고려대학교 정보보호대학원 공학 석사
2004년 3월~현재 : 고려대학교 정보보호대학원 박사과정
〈관심분야〉 암호 이론, 정보보호 이론, RFID 정보보호 기술, 유비쿼터스

**이 수 미 (Su Mi Lee) 학생회원**
1995년 2월 : 순천향대학교 화학과 학사
2003년 2월 : 고려대학교 정보보호대학원 공학 석사
2003년 3월~현재 : 고려대학교 정보보호대학원 박사과정
〈관심분야〉 암호 이론, 그룹 키 교환, RFID 인증 시스템

**이 동 훈 (Dong Hoon Lee) 종신회원**
1983년 8월 : 고려대학교 경제학사
1987년 12월 : Oklahoma University 전산학 석사
1992년 5월 : Oklahoma University 전산학 박사
1993년 3월 ~ 1997년 2월 : 고려대학교 전산학과 조교수
1997년 3월 ~ 2001년 2월 : 고려대학교 전산학과 부교수
2001년 2월~현재 : 고려대학교 정보보호대학원 교수
〈관심분야〉 암호프로토콜, 암호이론, USN 이론, 키 교환, 익명성 연구, PET 기술