RBAC을 이용한 RFID보안 인증 프로토콜

배우식*, 이종연
충북대학교 컴퓨터교육과
e-mail : bws@motor.ac.kr*

RFID Security Authentication Protocol Using RBAC

Woo-Sik Bae*, Jong-Yun Lee
Dept. of Computer Education, Chungbuk National University

요 약

RFID 시스템은 향후 바코드를 대체하고 우리 생활 전반에 걸쳐 사용될 획기적인 시스템 이지만 태그의 정보가 외부에 노출될 경우 심각한 문제가 발생 할 수 있다. 본 논문에서는 여러 보안 문제중 프라이버시 보호를 위해 RBAC 기반으로 리더의 권한을 배분하여 태그가 데이터를 선별적으로 전송하고 태그가 리더로부터 수신한 난수로부터 매 세션마다 비밀키 및 실시간으로 새로운 해쉬 함수를 생성하는 인증 프로토콜을 제안한다. 제안된 RBAC를 이용한 해쉬 기반 인증 프로토콜은 각종 공격에 대해 안전하며 연산을 최소화하여 다양한 적용성을 제공한다.

1. 서론

RFID(Radio Frequency Identification)는 현재 바코드를 대체하기 위한 기술로서 바코드의 저장 정보는 매우 적고 다시 프로그래밍을 할 수 없는 단점이 있으며 이를 극복하기 위해서 현재 기술로는 반도체에 데이터를 저장하는 방법이 있다. 이 분야는 앞으로 사용의 편리성 향상으로 개인 및 산업 전반에 활용이 예상 되며 국내·외적으로 많은 연구가 진행되고 있다. 그러나 반도체 칩에 내장된 정보를 무선 주파수를 이용하여 읽어내기 때문에 RFID기술은 도청, 트래픽 분석, 서비스거부 공격, 메시지유실, 트래킹 공격, 스푸핑 공격등 많은 취약점 들을 지니고 있어서 보안이나 프라이버시 보호에 심각한 문제를 야기할 수 있다[1]. 따라서 RFID시스템이 활성화되기 위한 한 가지가 바로 보안 문제이며 반드시 해결되어야 하는 분야이다.

본 논문에서는 RFID의 프라이버시 문제를 해결하기 위한 기존 제안된 해쉬락(Hash-Lock)기법[2], 등이 해결하지 못한 문제점을 보완하여 RBAC기반으로 각각의 리더에 등급별 역할을 주어 DB의 정의된 코드를 통해 태그가 리더의 등급별로 저장된 최소 자료만 암호화 하여 전송함으로써 보다 안전하고 효율적으로 사용자의 프라이버시를 보호할 수 있는 인증 프로토콜을 제안 한다.

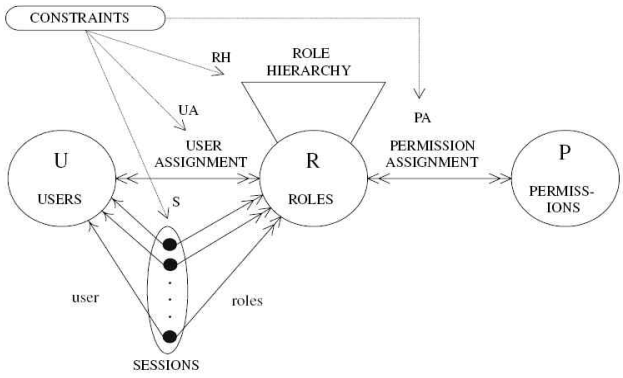
2. 관련연구

2.1 역할기반 접근제어

RBAC(Role-base Access Control)[3][4][5]은 접근제어 관리 작업을 단순화하고 역할기반 접근제어를 제공하기 위해 Ravi S. 등에 의해 제안 되었다. RBAC의 핵심은 권한과 역할을 연관시키고 사용자들이 적절한 역할을 할당 받도록 하는 것이다. RBAC 정책에서 관리자는 접근제어를 통해 권한이 없는 사용자가 불법적으로 중요 정보의 변조를 방지하는 무결성, 기밀 정보가 권한이 없는 사용자에게 유출되는 것을 막는 기밀성, 그리고 권한을 부여받은 사용자가 정보를 사용할 수 있도록 보장하는 가용성을 제공할 수 있다. 접근을 통제하기 위해 역할을 사용하는 것은 특성에 맞는 보안 정책을 시행하고 개발하며 보안 관리를 능률화 하는데 효율적인 기법이 된

다.

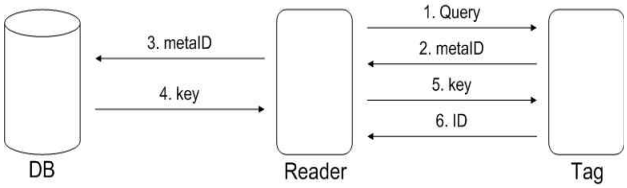
RBAC96 이라고 불리는 기본적인 RBAC모델은 Sandhu에 의해 정의 되었으며 [그림 1]과 같다



[그림 1] RBAC 모델

2.1 해-쉬락 기법

태그와 데이터베이스는 태그의 ID, 키 값을 공유하여 저장하게 되며 데이터베이스와 연결된 모든 리더는 태그에 대한 키를 알 수 있으며 태그는 mataID를 저장하고 있는 상태로 기본적으로 잠겨 있게 된다. 태그가 리더의 범위에 들어오면 mataID를 전송하며 이때 리더는 태그의 mataID를 데이터베이스에 전송하고 데이터베이스는 이에 대응하는 키를 리더에게 전송한다. 이어서 태그에게 보내어 태그가 해쉬값을 계산하며 자신의 mataID와 일치하는 경우 풀림상태로 되어 리더에게 자신의 ID를 전송하는 방식이다. 태그의 식별 값인 metaID가 고정되어 있으며 출력되는 데이터가 같아 전송 되었는지 확인할 수 있다. 아래의 [그림 2]은 해-쉬락 기법의 구조도 이다.



[그림 2] 해쉬-락 기법

3. 제안 프로토콜

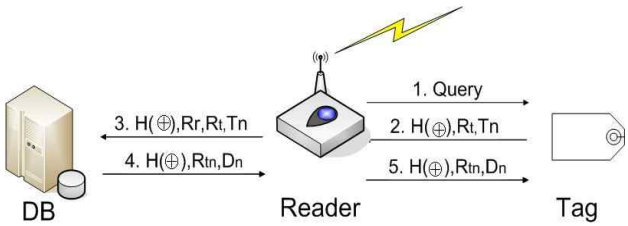
3.1 구조

RBAC에 바탕으로 그룹별로 구성된 리더를 DB에서 정의해주며 모든 리더에게 태그의 데이터가 전송되지 않고 리더의 권한 내의 태그 데이터를 읽을 수 있도록 구성 되었으며 리더가 처음 태그에게 질의를 할 때 난수와 실시간을 함께 전송하고, 태그는 리더로부터 수신한 난수와 실시간을 자신이 가지고 있는 ID, 비밀키 및 실시간으로 해쉬한 값을 이용하여 RBAC 권한모델에 알맞은 내용만 응답하게 된다.

제안 프로토콜에서 사용되는 파라미터는 다음과 같으며, [그림 3]는 제안하는 프로토콜의 기본 구조를 나타낸 것이다.

[파라미터]

- Query : 질의, 태그의 응답을 요청
- ID : 태그 고유의 비밀 인증 정보
- H( ) : 일 방향 해쉬 함수
- H(⊕) : $H(ID \oplus R_r \oplus T_t \oplus key)$ 연산
- R_t : 리더가 태그에게 전송하는 DB시간(μs)
- R_r : 리더가 생성하여 태그에게 전송하는 난수
- T_t : 태그에 저장되어 있는 시간(μs)
- R_{tn} : 태그에 기록될 시간(μs)
- D_n : 데이터베이스에서 태그에게 전송되는 명령
- T_n : 리더의 등급별로 전송하는 태그 신호
- ⊕ : Exclusive OR
- key : DB, 리더, 태그의 공통 비밀키



[그림 3] 제안 프로토콜의 구조

3.2 인증과정

① 리더는 태그들에게 Query를 브로드캐스팅 한다.

리더 → 태그 : Query,

② 태그는 리더의 등급을 확인 후 ID와 자신이 가지고 있던 T_t를 R_r와 XOR연산 후 해쉬 하여, R_t, T_n,와 함께 Query에 대한 응답으로 리더에게 전송한다.

태그 → 리더 : $H(\oplus), R_t, T_n$

③ 리더는 R_r와 $H(\oplus), R_t, T_n$를 백-엔드 데이터베이스로 전송한다.

리더 → 백-엔드 데이터베이스 : $H(\oplus), R_r, R_t, T_n$

④ 백-엔드 데이터베이스에 저장된 ID를 R_t, R_r,key와 해쉬한 값과 리더로부터 수신한 $H(\oplus), R_r, R_t$ 를 비교하여 태그를 인증한다.

백-엔드 데이터베이스 → 리더 :

계산된 $H(\oplus), R_r, R_t$ =

수신한 $H(\oplus), R_r, R_t$

인증이 성공하면 $H(\oplus), R_{tn}, D_n$를 리더에게 전송한다.

⑤ 리더는 백-엔드 데이터베이스로부터 수신한 $H(\oplus), R_{tn}, D_n$를 태그에게 전송한다.

리더 → 태그 : $H(\oplus), R_{tn}, D_n$

태그는 자신의 ID와 인증 세션에서 생성한 R_t, T_t를 XOR하여 해쉬한 값과 리더로부터 수신된 $H(\oplus), R_{tn}$를 확인하여 인증하고 R_{tn}을 기록하며 필요에 따라 D_n명령을 수행하고 인증세션을 성공적으로 종료 한다.

3.3 제안프로토콜의 안전성

3.3.1 스푸핑 공격에 대한 안전성

악의의 공격자가 정당한 리더로 가장하여 Query를 전송하면, $H(\oplus), R_t$를 획득할 수 있다. 그러나 정당한 리더의 암호화 비밀 KEY 값을 알 수가 없으며 알아낸다 할지라도 이후 데이터베이스에 응답으로 보내지게 되면 이미 시간이 지나간 상태의 정보 $H(\oplus), R_t$로는 데이터베이스에서의 인증을 할 수가 없어 결국은 스푸핑 공격이 불가능 하게 된다.

3.3.2 재전송 공격에 대한 안전성

정당한 리더의 Query에 대한 응답은 매 세션마다 변하기 때문에 $H(\oplus), R_t$도 매 세션마다 바뀌게 된다. 공격자는 매번 바뀌는 시간과 난수 및 비밀 key 값을 알아야만 공격에 필요한 자료를 얻을 수 있다. 그러므로 도청으로 획득한 $H(\oplus), R_t$를 다음 세션에서는 응답으로 사용할 수 없으므로 재전송 공격에 안전하다.

3.3.3 위치 추적에 대한 안전성

제안 프로토콜에서는 태그의 값이 매 세션 때마다 지속적으로 업데이트 되며 새로운 값으로 생성 된다. 때문에 세션이 바뀔 때는 물론 리더의 질의에 대한 태그의 응답이 바뀌므로 예측 하거나, 공격자가 Query를 태그에게 전송하여도 다음 세션에서 태그는 매 세션마다 변하는 응답 $H(\oplus), R_t$를 전송하게 된다. 그러므로 공격자는 인증이 안 되어 트래픽 분석이 불가능 하고 태그의 위치도 추적할 방법이 없게 된다.

3.3.4 제안 방식의 효율성

제안 방식은 태그의 데이터가 리더의 역할 등급에 따라 전송하는 데이터양이 다르기 때문에 불필요한 데이터를 리더신호에 모두 전송하는 방법이 아니라 꼭 필요한 정보만을 리더 등급별로 보내므로 만일 공격자가 데이터를 습득한다 하여도 태그의 모든 데이터를 습득하기는 어려워 효과적이며 간단하게 구현할 수 있다.

4. 결론

본 논문에서 제안한 프로토콜은 태그에서의 연산을 최소화 하며 비밀 키를 이용하여 기존의 연구와는 달리 리더의 역할을 배분하여 데이터접근을 제어한다. 이 경우 태그의 모든 데이터를 전송하지 않아 가볍게 동작하며 공격자의 도청, 재전송 공격, 스푸핑 공격 등에 효율적으로 대처하여 보안 효과는 최대한의 효과를 낼 수 있도록 하였다. 또한 안전성 면에서도 현재까지 여러 공격 가능한 상황에 대해 안전함을 보였고 추후에는 안정성을 해치지 않은 범위 내에서 태그의 연산량을 더욱 줄일 수 있는 방안에 대해 연구가 지속 되어야 하겠다. 또한 불필요 태그 데이터에 의한 서버부담을 줄이고 효과적으로 태그를 처리할 수 있는 방법이 될 것으로 기대된다.

참고문헌

[1] 한국정보보호진흥원 “개인정보보호 백서”, 2003.

[2] S. Weis et al., "Security and Privacy Aspects of Low-cost Radio Frequency Identification Systems," Security and Pervasive Computing 2003, LNCS2802, pp. 201-202.

[3] Ferraiolo D. F., R. Sandhu, S. Gavrila, D.R. Kuhn, and R. Chandramouli, “A Proposed Standard for Role-Based Access Control”, Proposed 4/4/2003 Draft NIST, 2003, URL: http://csrc.nist.gov/groups/SNS/rbac.

[4] R. Laborde, B. Nasser, F. Grasset, F. Barr`ere, A. Benzekri, "A Formal Approach for the Evaluation of Network Security Mechanisms Based on RBAC Policies" Electronic Notes in Theoretical Computer Science 121 2005, pp. 117-142

[5] Andrea Omicini Alessandro Ricci Mirko Viroli, "RBAC for Organisation and Security in an Agent Coordination Infrastructure" Electronic Notes in Theoretical Computer Science 128, 2005, pp. 65-85