[UCI]I804:24011-200000240302

2010년 8월

석사학위논문

토큰키와 해쉬함수를 이용한 RFID 인증 프로토콜 설계

조선대학교 대학원

정보통신공학과

한 경 헌

토큰키와 해쉬함수를 이용한 RFID 인증 프로토콜 설계

Design of an RFID Authentication Protocol Using Token Key and Hash Function

2010년 8월 25일

조선대학교 대학원

정보통신공학과

한 경 헌

토큰키와 해쉬함수를 이용한 RFID 인증 프로토콜 설계

지도교수 한 승 조

이 논문을 공학석사학위신청 논문으로 제출함

2010년 04월

조선대학교 대학원

정보통신공학과

한 경 헌

한경헌의 석사학위논문을 인준함

위원장 조선대학교 교수 박 종 안 印

위 원 조선대학교 교수 한 승 조 印

위 원 조선대학교 교수 최 동 유 印

2009년 05월

조선대학교 대학원

목　　차

ABSTRACT ····· v

제 1 장　서　　론 ····· 1

제 2 장　RFID 시스템 ····· 3

제 1 절　RFID 시스템 구성 ····· 3

1. 태그 ····· 3

2. 리더 ····· 4

3. RFID 시스템 특징 ····· 5

제 3 장　RFID 보안 ····· 6

제 1 절　RFID 보안 이슈 ····· 6

1. 제약조건 ····· 6

2. 프라이버시 침해 ····· 7

3. 위·변조 문제 ····· 8

제 2 절　RFID 포핸드 보안 기술 ····· 11

1. 도청방지 기술 ····· 11

2. 정보차단 기술 ····· 14

3. 인증 및 접근제어 기술 ····· 16

제 4 장 토큰키와 해쉬 함수를 이용한 인증 방법 ····· 17
제 1 절 기존 RFID 인증 방법 ····· 17
제 2 절 토큰키와 해쉬 함수를 이용한 인증 방법 ····· 19
1. 키 분배 기법 ····· 19
2. 인증 방법 ····· 21
제 3 절 토큰키에 난수를 추가한 인증 방법 ····· 24
1. 키 분배 기법 ····· 24
2. 인증 방법 ····· 27

제 5 장 실험 및 고찰 ····· 30
제 1 절 시뮬레이션 구현 ····· 30
제 2 절 안정성과 효율성 분석 ····· 32
1. 토큰키와 해쉬 함수를 이용한 인증 방법 ····· 32
2. 토큰키에 난수를 추가한 인증 방법 ····· 32
3. 인증기법 비교 ····· 40
제 3 절 효율성 분석 ····· 42

제 6 장 결 론 ····· 45

참 고 문 헌 ····· 48

그 림 차 례

그림 4.1 MIT 해쉬 함수 ····· 17
그림 4.2 암호화 알고리즘 상호인증 키 분배 ····· 20
그림 4.3 태그와 리더간의 인증과정 ····· 22
그림 4.4 난수 발생 키 분배 ····· 25
그림 4.5 태그와 리더간의 인증과정 ····· 27
그림 5.1 시뮬레이션 환경 ····· 30
그림 5.2 재생 공격에 대한 응답 ····· 33
그림 5.3 서비스 거부 공격에 대한 응답 ····· 34
그림 5.4 도청 공격에 대한 응답 ····· 35
그림 5.5 재생 공격에 대한 응답 ····· 37
그림 5.6 서비스 거부 공격에 대한 응답 ····· 38
그림 5.7 도청 공격에 대한 응답 ····· 39
그림 5.8 재생 공격에 대한 응답 ····· 40
그림 5.9 서비스 거부 공격에 대한 응답 ····· 40
그림 5.10 도청 공격에 대한 응답 ····· 41
그림 5.11 도청 공격 중 태그와 리더간의 전송량 ····· 42
그림 5.12 도청 공격 중 태그와 리더간의 지연시간 ····· 43

표 차 례

표 3.1 RFID 보급과 역기능 ····· 9
표 5.1 시뮬레이션 파라미터 ····· 31
표 5.2 재생 공격에 대한 응답 ····· 33
표 5.3 서비스 거부 공격에 대한 응답 ····· 34
표 5.4 도청 공격에 대한 응답 ····· 36
표 5.5 재생 공격에 대한 응답 ····· 37
표 5.6 서비스 거부 공격에 대한 응답 ····· 38
표 5.7 도청 공격에 대한 응답 ····· 39
표 5.8 각 공격에 대한 응답 시간 비교 ····· 41

ABSTRACT

Design of an RFID Authentication Protocol Using Token Key and Hash Function

Han, Kyoung-heon
Advisor : Prof. Han Seung-jo Ph.D
Department of Info. & Comm. Engineering,
Graduate School of Chosun University

RFID is method used on wireless system. However, this mechanism is not used for authentication and security. Therefore, it is very vulnerable to attacks such as dropping attacks and traffic attacks. the RFID Tags are of the limited nature due to environment factors and there is greater constraints in designing authentication protocol. If we do not RFID to secure corporate information and product all the information will be exposed. The attacker will break into the RFID system and stop the distribution system. A few companies used Hash in order to prevent attacks. However, this method has a problem. Hash is based on ID of a tag that it used authentication. But there is a problem that ID was flowed out from this method. As a database is maintained based on Tag's ID. As Tag's ID is exposed attackers can easily

get information of tag. So they can easily

So, this paper proposes a new authentication protocol which provides authentication mechanism by defining the key distribution. And a new authentication protocol mechanism using Hash function.

제 1 장 서 론

근래 각종 매체를 통해 빈번히 보도되고 있는 유비쿼터스 기술은 또 한 번 정보통신 기술의 대혁신을 예고하고 있다. RFID(Radio Frequency Identification) 및 USN(Ubiquitous Sensor Network)기술은 이러한 유비쿼터스 환경의 가장 핵심적인 요소라고 할 수 있으며 지식경제부에서도 미래의 정보통신 기술 구현을 위한 핵심 인프라로 선정된 바 있다. 실제로 RFID기술은 물류, 운송, 유통, 내부 재고관리 등에 획기적인 개선을 가져올 것으로 예상되며 이미 여러 분야에 적용되어 사용되고 있다.

RFID기술은 사물의 식별정보 등을 극소형 태그에 저장하여 사물에 부착하고 해당 사물 및 주변 환경 정보를 무선주파수를 통하여 안테나가 장착된 리더 및 네트워크로 전송하여 필요한 정보처리를 하는 비접촉식 자동식별 기술을 말한다. 그러나 RFID기술은 인간생활에 편리함을 주는 동시에 '언제·어디서나' 개인정보가 누출될 수 있는 환경을 제공하여 프라이버시 침해의 역기능 문제도 동시에 안겨주고 있다.

RFID는 무선네트워크를 기반으로 하는 소규모 네트워크로 이루어지기 때문에 보안적 문제가 많이 발생한다. 이런 보안적 문제를 해결하기 위해 현재 사용중인 높은 보안성을 가지는 알고리즘을 사용하여 문제점을 해결하고자 하지만 RFID 태그의 특성인 제한된 메모리와 제한된 전력 때문에 크기가 큰 암호화 알고리즘을 사용하지 못한다. 매우 좋은 암호화 알고리즘이라도 태그의 제한적 환경에 맞지 설계가 되어 있지 않다면 사용할 수 없다는 것이 현실이다. 전송하는 데이터 보다 암호화하는 과정에 더 많은 전력이 소모된다면 비효율적인 시스템이 되기 때문에 사용하지 못한다. 좋은 암호화 알고리즘을 사용하기 위해서 좋은 태그를 사용하면 되지만 태그의 단가가 높아지기 때문에 많은 기업에서도 태그에는 암호화 알고리즘을 사용하지 않는다.

태그와 리더는 무선 통신을 통해 정보를 교환하기 때문에 불법적인 리더에 의한 도청 및 정보의 변복조가 쉽게 가능하다. 하지만 이를 막는 기술은 매우 미흡한 상

태이며, 사용하는 기술도 많은 문제점을 가지고 있다. 그래서 본 논문은 태그의 제한적 요소를 감안하여 기존의 인터페이스를 계속 사용하는 가운데 기존의 RFID 인증 시스템보다 높은 효율성과 안전성을 가진 RFID 인증 프로토콜을 설계하고, 시뮬레이션을 통해 기존의 보안 방식과 비교·검토함으로서 그 실용성과 타당성을 검증하고자 한다.

본 논문의 구성은 2장에서 RFID 시스템의 구성을 알아보고, 3장에서는 RFID 시스템에서의 보안에 대해 알아본다. 그리고 4장에서는 기존의 해쉬-락 인증 기법의 문제점을 파악하여 새로운 해쉬 인증 기법을 제안한다. 5장에서는 시뮬레이션의 환경을 설명하고 시뮬레이션 결과를 비교 분석 하였다. 마지막 6장에서는 결론으로 구성하였다.

제 2 장 RFID 시스템 구성

제 1 절 RFID 시스템 구성 요소

무선 주파수 인식 시스템이란 라디오 주파수 통신방식을 이용하여 이동식 저장장치와 주 컴퓨터 혹은 PLC간에 정보를 주고받을 수 있는 시스템이며, 일반적으로 RFID시스템은 데이터가 저장되어 있는 태그 및 라벨, PCB 등과 같은 데이터 저장소와 태그와의 통신을 위한 안테나, 그리고 안테나와 PC 혹은 PLC간의 통신을 관리하는 컨트롤러로 구성되어 있다. 가장 중요한 점은, 태그 및 라벨, PCB는 제품 운반 장치 또는 운반 상자, 심지어는 제품 그 자체에 직접 부착하기도 하며, 제품의 이동경로를 원격의 데이터베이스로 제공할 수 있다는 점이다.

시스템의 구성은 크게 태그와 리더 그리고 데이터베이스 서버로 이루어지며 태그와 리더사이에는 무선으로 연결되어 있으며 리더와 데이터베이스 서버에는 유·무선 방식을 선택적으로 사용하여 필요한 정보와 신호를 교환한다.

1. 태그

RFID 시스템은 리더의 요구에 따라 태그의 내장된 칩에서 저장된 데이터를 읽어낼 수 있는 데이터 인식시스템을 가리킨다. 태그는 크기와 메모리 용량, 온도에 대한 내성 및 작동 온도 범위에 따라 다양한 선택이 가능하다. 주사기 바늘을 사용하여 동물의 몸 안에 투입할 수 있을 만큼의 아주 작은 크기의 것부터, 책상 전체를 덮을 수 있을 만큼의 커다란 크기의 것들도 있다. 거의 모든 태그들은 태그의 판독 및 입력 범위가 금속 및 전자기파에 노출되어 있는 환경에 존재하는 경우가 많기 때문에 외부 충격 및 화학 물질, 습기, 먼지 등으로부터 보호될 수 있도록 하기 위해 캡슐 형태로 제작되어진 것도 있다.

태그들은 내부의 전지에 의해 전원이 공급되거나, 유도 커플링에 전원이 공급되기도 한다. 수동 태그의 경우, 태그를 보수 및 수리할 필요가 없으며, 거의 반영구적으로 사용할 수 있다. 능동 태그의 경우, 전지의 수명에 의해 태그의 수명이 정해지고, 전지를 교체 할 수 있으며, 또 대용량의 전지를 사용하고 있는 것들도 있다.

태그는 리더와의 에너지 결합방식에 따라 동작원리가 전혀 다르다. 자계결합방식은 현재 대부분의 저주파 RFID에서 적용되는 원리로서 전원에너지 및 데이터전송이 코일루프안테나 전류에 의해 형성되는 자계 에너지에 의해 전송된다. 즉 구동전류에 의해 리더의 안테나에서 발생한 전력에너지는 자계장을 형성하며 자계장 영역에 위치한 태그는 자계장의 변화에 의한 전류가 유기되어 이로 인한 전압이 발생한다. 이 경우 태그에 요구되는 최소 전력을 리더에서 수신하게 된다. 수신된 전력에 의해 동작하는 태그는 자체 데이터 송출을 동일한 원리에 의해 리더에 전달하게 되는데 태그에 흐르는 전력은 미약 전류이므로 리더에 전송하는 거리는 매우 근거리이다. 참고로 현재 대부분의 저주파 RFID는 이러한 원리에 의해 동작하므로 감지거리 또는 데이터 전송거리는 근거리로 활용이 제한적이며 감지거리를 증대하려면 자속통과 면적을 크게 해야 하므로 안테나 매우 커야 한다. 또한 저주파 RFID는 133[㎑] 또는 13.56[㎒]로서 주파수가 낮기 때문에 데이터 전송속도가 느리고 전송 데이터양이 매우 제한적이다.

2. 리더

리더는 응용프로그램을 수행하는 PC, PDA 또는 또 다른 단말기와 여러 가지 방법(RS-232/484, USB, W/LAN)으로 연결되며, 태그와 무선으로 송수신하는 역할을 수행한다. 리더는 수동형 태그가 동작할 수 있는 전력과 명령어를 무선 방송파 신호로 태그에 전송하고 태그로부터 응답을 수신하여 신호를 복원하는 기능을 수행한다. 리더는 크게 고정형 리더와 핸드휄드(Hand Held)형 리더로 크게 나누며, 핸드 휄드형 리더에는 안테나가 포함된 일체형이 일반적이며 고정된 리더는 안테나 연결이 가능한 2채널에서 4채널까지 연결할 수 있게 구성되어 있다.

현재 리더는 안테나 성능 및 주변 환경에 의해 인식 거리, 인식 정확도에 영향을 받아 적용 범위가 제한되는 특성이 있으므로, 향후 13.56[㎒], 900[㎒], 433[㎒] 등의 다양한 주파수 대역에서 동작하는 리더와 EPC Code 및 ISO 18000 계열의 코드 등 다중 코드를 인식하는 리더가 필요하게 될 것이다.

3. RFID 시스템 특징

RFID 시스템은 물품과 같이 관리할 사물에 태그를 부착하고 신호를 이용하여 사물의 ID 정보 및 주변 환경 정보를 인식하여 각 사물의 정보를 수집, 저장, 가공 및 추적함으로써 사물에 대한 원격처리, 관리 및 정보교환 등 다양한 서비스를 제공한다. 기본적인 동작원리는 리더가 무선통신을 이용하여 태그에게 질의를 보내면 태그 안에 내장된 안테나가 전파를 수신하다. 태그 안에 내장된 IC칩이 정보를 신호화하여 안테나를 통해 신호를 전송한다. 리더는 전송된 신호를 유무선 방식으로 종단 시스템으로 전달한다. 그리고 리더의 종단 시스템은 태그가 전송한 정보와 종단 시스템의 정보를 비교하여 매치된 정보를 리더에게 전송하며, 리더는 이를 디스플레이 장치를 통해 사용자에게 보여준다.
RFID 시스템의 효율성과 정확성은 바코드 시스템과 비슷한 수준이지만 RFID의 경우에는 바코드에 비해 더 많은 장점을 가지고 있다. [1]

① 태그 내부에 저장되어 있는 정보를 효과적으로 실시간 갱신이 가능
② 비금속 물질을 투과해서 읽을 수 있으며, 비접촉 인식이 가능
③ 태그 설치의 제한을 받지 않아 보이지 않은 곳에 설치 가능
④ 불법 사용자가 특정 상품에 관한 데이터를 변경할 수 없도록 보호가 가능
⑤ 바코드보다 판독시간이 40% 빠름

제 3 장 RFID 보안

제 1 절 RFID 보안 이슈

RFID는 무선네트워크를 이용하여 정보를 전송한다는 편리함을 가지고 있지만 개인 정보나 보안에 대해 취약한 점을 가지고 있다. 특히 물리적인 공격, 위조, 스푸핑, 도청, 트래픽 분석, 서비스 거부 공격등에 대한 취약점을 가지고 있다. 이러한 취약점들은 개인이나 기업의 심각한 보안 문제를 발생시킨다. 또한 RFID는 일반 무선네트워크와 달리 많은 제약조건을 가진다. 태그의 가격문제, 전송데이터의 크기와 판독시간이 한정되어 있으며, 리더 장비나 백앤드 시스템은 성능과 자원 측면에서 한계를 가진다. 따라서 RFID에 대한 보안 대책도 태그와 리더, 백엔드 시스템의 제약사항을 고려하여 설계되고 구현되어야 한다. 또한 태그에는 자체적인 네트워킹 기능이 있어 신뢰할 수 있는 CA(Certificate Authority) 같은 곳에 접속하여 리더를 신뢰할 수 있는지 알아낼 방법이 없다. 고가형 태그의 경우 나름대로의 인증과 암호화 시스템을 가지고 있지만 그렇지 않는 저가형 태그는 어떠한 리더의 요구에도 응답하게 된다.

1. 제약조건

낮은 가격을 유지하면서 보안 및 프라이버시 위험을 고려한 태그 및 리더의 설계가 중요한 문제가 되고 있다. 저렴한 태그는 기본적으로 패시브 형태의 사용을 요구하고 있으며, 저가형 태그의 비용 요구 사항은 태그에 사용할 수 있는 전력, 처리시간, 저장 공간, 게이트 수 등의 자원을 제한한다. 저렴한 태그를 만들기 위한 IC 칩 비용은 300원를 넘으면 안되며 이는 게이트 수를 7,500~15,000개로 제한한다. 현재 100비트의 EPC 칩은 약 50,000~10,000개의 게이트를 요구함에 따라 안정성 측면에서 요구되는 게이트 수인 2,500~5,000를 넘어서면 안되는 제약 조

건을 가지고 있다. [2]

CRYPREC 보고서에 따르면 대칭키 암호알고리즘의 구현은 6,000~13000게이트로 알려져 있으며 대칭키를 기반으로 설계할 수 있는 해쉬함수도 유사한 수의 게이트가 요구될 것으로 기대된다.[3] 앞으로 더 적은 게이트 수가 요구되는 Tiny Encryption Algorithm이 저렴한 RFID 태그에 사용될 수 있는 가능성이 있으나 현재 사용하기에는 비싸다.[4] 또한 대칭키 암호에 비해 더 많은 게이트 수가 요구되는 공개키 암호를 RFID에 사용하기 위해서는 더 비싼 비용이 소요될 것으로 보인다. 현재 NTRU에는 NTRU 공개키 암호화기법을 RFID와 비접촉형 스마트 카드에 적용한 솔루션 GenuID를 개발하였으며, 비접속형 스마트 카드에 NTRU 암호기법을 구현할 경우 약 2달러의 비용이 소요될 것으로 예측하고 있다.[5] 그러나 NTRU 공개키 암호도 저렴한 RFID 태그에서 이용할 수 있는 리소스 이상을 사용하기 때문에 현재 5센트 이하의 저렴한 태그에 적용할 수 없으며, 더 비싼 스마트 카드에 사용하여야 한다.[6]

2. 프라이버시 침해

RFID 시스템에서 태그와 리더 사이는 무선 환경에서 서로 정보를 전달한다. 그래서 공격자는 쉽게 도청을 할 수 있다. 만약 태그와 리더사이에 어떠한 정보보호기술이 적용되지 않는다면 태그 정보가 공격자에게 모두 넘어가게 된다. 이러한 정보들을 통해 공격자는 개인 및 기업의 정보를 쉽게 습득하게 된다. 그래서 태그와 리더사이에는 태그의 정보를 숨길 수 있는 암호화 기술이나 그러한 보안기술 적용이 필요하다. 그런데 태그의 정보를 암호화 한다고 해도 또 다른 문제점이 발생하게 된다. 태그는 리더의 쿼리에 대해 반드시 어떠한 형태로든 응답을 해야 한다. 그런데 응답이 일정하게 반복하게 되면 공격자는 리더를 갖고 사용자가 현재 위치정보를 알 수 있게 된다. 즉 공격자가 다수의 리더를 이용하게 되면 사용자의 위치추적이 가능하게 된다. 이러한 위치추적을 방지하기 위해서 태그는 리더의 쿼리에 대해 매번 응답을 다르게 해야 한다.

RFID의 개인 프라이버시 침해는 다음과 같은 2가지 유형으로 나타난다.

① RFID 태그 부착 물품을 들고 가거나, RFID 태그를 부착하고 있는 개인에 대한 위치 추적이 가능하다.
② 개인이 소유하고 있는 태그부착 물품의 ID를 읽은 후, 그 ID에 연결된 물품 정보를 네트워크로부터 열람하여 개인이 무엇을 소유하고 있는지를 알 수 있다.

이러한 프라이버시 침해를 해결하기 위한 가장 확실한 방법은 RFID 태그를 물리적으로 제거하거나, 태그 기능을 정지 시키는 것이다. 하지만 많은 RFID 응용서비스에서는 개인이 태그 부착 물품을 소유한 이후에도 계속 서비스가 이루어져야 하기 때문에, 태그 기능 정지만이 최상의 프라이버시 보호 방법은 아니다.[7]

3. 위 · 변조 문제

RFID의 사용이 확대되어 대부분의 물건에 태그가 부착이 된다면, RFID 위 · 변조 문제가 심각하게 대두될 것으로 예상된다. 태그 위 · 변조에 대한 대응 방안이 미비하다면, 기업이나 사용자 모두 큰 피해를 입을 수 있다. RFID에 대한 공격자의 공격은 다음과 같다.

① 불법적인 태그 복제: 교통카드나 출입 관리를 위해 사용되는 RFID 태그에 대한 불법적인 복제를 통하여 금전적인 이득을 얻거나 불법적인 물리적 침입을 시도하기도 한다.
② 태그에 저장된 정보 획득: 공격자는 리더를 통해 태그에 저장된 정보를 읽어 이를 기반으로 다른 공격에 사용하거나 개인 정보 획득을 시도한다.
③ 후미 서버로부터 정보 수집: 비인가된 리더를 통하여 후미 서버에 접속하여 RFID 관련 정보를 수집하거나 수집된 태그 정보를 전송하여 공격을 시도 할 수 있다.

표 3.1 RFID 보급과 역기능

구분	특징
수집	• RFID정보주체의 인식여부에 관계 없이 무제한 수집 가능
정보 변화	• RFID태그정보는 RFID태그가 부착된 물품이 사업자간의 교환 및 소비자에게 이동했을 경우 변화 가능
개인 정보화	• RFID태그정보와 개인정보의 융합시 개인 프라이버시 침해가 우려됨
발생 가능한 문제점	• RFID정보를 이용한 개인 신상정보 노출 • RFID정보를 활용한 개인의 물품 보유현황 노출 • RFID인식 기술을 이용한 위치정보 노출 • RFID가 상품에 활용될 때 개인의 구매 패턴 및 선호도 노출 • RFID정보를 원하는 기업이 있는 경우 개인의 의사와 무관하게 개인 정보의 불법적인거래가 이뤄질 수 있음
태그 정보의 침해 유형	• 부적절한 RFID정보의 접근과 수집 • 부적절한 RFID정보 분석 • 부적절한 RFID관련 정보의 이전 • RFID태그 정보를 활용하여 원하지 않는 영업 행위

출처 : 한국정보사회진흥원

◎ 재기록 태그(Re-Write Tag)

EEPROM을 사용하는 태그와 같이 재기록이 가능한 태그의 경우 이 데이터는 안전하지 않을 뿐만 아니라 언제나 다른 사람에 의해서 수정되거나 바뀔 가능성이 존재하게 된다. 가령, EEPROM을 채택한 태그를 사용하는 RFID 시스템으로 운영중인 슈퍼마켓이 있다고 가정하면 어떠한 보안 수단이 존재할 수는 있지만, 이를 무력화 시키고 태그의 정보를 값싼 다른 물건의 정보로 바꿔치기 할 수 있을지도 모른다. 이러한 사실을 깨닫지 못한다면 슈퍼마켓은 맞지 않은 재고 품목으로 물품

확보에 차질을 빚을 것이고, 아무런 문제가 없는 전산시스템의 오류를 의심하여 이를 수정하기 위해 인력과 시간을 낭비할 지도 모른다.

◎ 서비스 거부 공격(Denial of Service)

RFID 시스템에 대한 가장 강력한 공격은 RFID 시스템의 부분 또는 전체를 마비시킬수 있는 서비스 거부 공격이라고 할 수 있다. 무의미한 비트를 무작위로 생성하여 연속적으로 공중에 뿌리는 특수한 태그가 있을 때 리더가 다른 태그를 읽으려고 한다면 리더는 이로 인하여 발생하는 수많은 충돌과 상황에 맞지 않는 데이터에 끝없이 재시도를 할지도 모른다. 물론, 이 공격은 충분히 검색 가능하며 리더는 상황에 맞게 읽기를 포기할 수도 있다. 하지만 이 공격을 검색 가능하다 하더라도 이 태그를 찾아 물리적으로 제거하기 전까지는 쓸 수 없는 상태가 되어버린다.

◎ 스푸핑(Spoofing)

스푸핑 공격은 자격조건이 없는 호스트로부터 올바르지 않은 정보를 수용하거나 사용할 때 일어난다. 위조 공격은 이러한 공격에 대해 취약한 DNS서버에게 치명적일 수 있다. 이런 공격은 인터넷 사이트나 전자우편에 대하여 자격이 없는 호스트로 전달해 주게 된다. 이 문제는 당사자가 사용하는 서버를 위조할 수 없다고 하더라도 다른 호스트를 위조함으로서 당사자에게 직·간접적으로 피해를 입힐 수 있다. 이 공격은 자신이 공격을 당하고 있는지 알지도 못한 채 오랜 시간을 보낼 수 있기 때문에 더욱 위험할 수 있다.

◎ 데이터 베이스 공격(Database Attack)

암호화된 정보를 송출하는 태그의 경우 암호화된 정보가 담긴 데이터베이스를 공격하여 그 내용을 얻어 올 필요는 없다. 또한 암호화되지 않는 정보라고 해도 마찬가지이다. 암호화되거나 되지 않은 정보를 바로 태그의 아이디로 사용하여 독자적인 데이터베이스를 구축할 수도 있기 때문이다. 물론 이를 구축하는 쉬운 일은 아니지만 결코 불가능하지도 않다. 현재의 EPC 네트워크와 같이 상품번호만 제공하는 백앤드 시스템이라면 이러한 공격에 대해서는 태그의 정보를 읽을 수 없도록

원천 봉쇄하거나 사용할 때마나다 다른 키로 암호화하는 방법이외에는 막을 수 있는 방법은 없다.

제 2 절 RFID 포핸드 보안 기술

리더와 태그 사이의 보안 문제는 태그에 탑재될 수 있는 하드웨어 기술으 한계적인데 반해 태그의 정보는 스푸핑, 위조, 서비스 공격, 물리적인 공격등을 비교적 시도하기 쉽기 때문에 발생한다.[8] 이런 문제점을 해결하기 위해 현재 제안된 기술은 인증 및 접근 제어, 도청 방지, 그리고 정보 차단의 3가지 분야로 나누어 연구되고 있다.

1. 도청방지 기술

도청 방지 기술은 고요한 트리워킹(Silent Tree-walking)이나 랜덤 트리워킹(Randomized Tree-walking) 같은 방법이 제안되었다. 또한 비밀키 암호화 방식을 사용하여 암호화된 데이터와 암호화된 정보를 태그에 기록하는 재암호화 방식이 있다.[9] 이 기술은 태그가 리더에게 보내는 데이터는 공격자가 직접 들을 수 없다는 점에 착안하여 리더가 태그의 정보를 부르지만 태그가 리더에게 보낸 마지막 데이터와 리더가 태그에게 보내고 싶은 데이터를 XOR연산하는 방식을 사용한다. 랜덤 트리워킹은 고요한 트리워킹과 유사한 방식으로 태그가 난수를 생성하여 이를 리더에 보내고 이를 기초로 하여 탐색을 하는 방식이다. 재암호화 방식은 Elgamal 공개키 암호화 방식을 사용하여 암호화한 고유 번호를 태그에 덮어쓰는 방식으로 이루어진다.

◎ 고요한 트리워킹(Silent Tree-walking)

트리워킹 알고리즘의 동 RFID 시스템에서 효율적인 다중인식 기술을 위해서는 리더가 태그를 충돌 없이 인식해야 하므로 태그 충돌방지 알고리즘이 요구된다. 현재

RFID 시스템에서는 이진 탐색 방식과 프레임 알람 방식이 태그 충돌 방지 알고리즘으로 널리 쓰이고 있다. 그중 이진 트리 워킹 충돌 방지 알고리즘은 리더가 태그에게 ID의 각 비트를 브로드케스트하기 때문에 전방향 채널에 존재하는 도청공격에 취약하다. MIT가 제안한 고요한 트리워킹 알고리즘은 태그가 리더에게 보내는 데이터를 도청할 수 없다는 점에서 착안하여 리더가 태그에게 보내는 데이터를 XOR 연산을 통해 공격자로부터의 도청공격에 안전하면서 이진 트리 워킹 충돌 방지 알고리즘과 유사한 수행속도를 갖도록 변형한 알고리즘이다.
동작 방식을 설명하면 태그는 리더의 질의에 응답할 수 있는 켜진 상태와 응답할 수 없는 꺼진 상태를 가질 수 있는데, 먼저 리더는 모든 태그를 켜진 상태로 만든다. 그 다음 리더는 첫 번째 비트가 '0'인 태그와 '1'인 태그가 함께 존재한다면 '0'과 '1'을 의미하는 응답이 일어남으로써 충돌이 일어난다. 리더는 첫 번째 비트가 '0'인 태그들이나 '1'인 태그들 중 한 쪽을 선택하여 꺼진 상태로 전환시킨다. 그 다음 켜진 태그들을 상대로 두 번째 비트를 질의하고, 충돌이 일어날 경우 마찬가지로 한 쪽을 선택하여 멈추게 후 세 번째 비트에 대한 질의를 수행한다. 이를 일반화시켜 n번째 비트를 질의한다고 하면, 현재 상태에서 켜진 태그들이 응답을 하게 되는데 이 때 0과 1이 동시에 발생하면 충돌이 일어나게 된다. 응답을 받은 후 리더는 n번째 비트가 '0'인 태그들, 혹은 '1'인 태그들을 지정하여 꺼진 상태로 변화할 것을 명령하고, 다음 단계에서 다시 리더는 n+1번째 비트에 대해 질의한다. 각 각의 태그 안에 들어있는 ID는 유일하기 때문에 이런 알고리즘을 1번째 비트부터 수행함으로써 태그를 적어도 하나씩 분류해 내고, 해당 ID를 읽어낼 수 있다. 하나의 태그를 선별해 낸 다음에는 최근에 멈춘 태그 그룹을 켜진 상태로 변화시킨 후 같은 알고리즘을 적용하고, 켜진 상태로 남겨 놓았던 태그들을 꺼진 상태로 변환시키는 방식으로 위의 알고리즘을 반복함으로써 모든 태그의 ID를 알아 낼 수 있다.

◎ 랜덤 트리워킹(Random Tree-walking)

랜덤화된 트리워킹 방식은 태그가 두 개의 ID를 보유한다. 하나는 실제 태그 ID

이고, 다른 하나는 태그 제조 회사 또는 태그 자신이 초기화시 랜덤하게 부여한 임의 ID이다. 일반적으로 멀리 존재하는 공격자가 정보를 수집할 수 있는 것은 리더의 정보 전달 거리가 멀기 때문이다. 리더는 태그의 ID를 찾기 위형 트리워킹 방식을 적용하는데, 이때 적용하는 ID는 태그의 실제 ID가 아닌 임의 ID가 대상이 된다. 이 과정을 통하여 최종적으로 한 태그가 선별되면, 이 태그는 백워드 채널로 자신의 실제 ID를 리더에게 전달하여 태그를 인식할 수 있도록 한다. 공격자는 원거리에서 리더부터 태그들의 포워드 채널상의 신호만 인식하므로, 공격자가 인식하는 ID는 랜덤하게 생성된 임의의 ID이므로 실제 태그의 ID는 안전하다. 이 방식은 도청자가 포워드 채널 영역내의 원거리에 있고, 백워드 채널 영역내에 없을 경우에는 효과적으로 태그 정보의 보호가 가능하다. 하지만 랜덤화된 트리워킹 알고리즘의 문제점은 공격자가 태그와 매우 가까운 백워드 채널에 있게 되면 실제 ID 정보 유출이 가능하게 된다는 문제점을 가지고 있다.

◎ 재암호화(Re-encryption)

재암호화은 RSA 연구소에서 유로에 내장하기 위하여 개발한 보안기술로, 노출되는 태그의 정보를 보호하면서 권한이 있는 정부기관이 지폐를 추적할 수 있는 기술이다. 하지만 공개키 방식의 암호화 알고리즘이 태그에 삽입되어야 하기 때문에 태그의 가격이 상승된다는 문제점을 가지고 있다.

2. 정보차단 기술

정보 차단을 위하여 물리적인 방법으로 킬 태그(Kill tag) 방법이나 패러데이 우리(Faraday Cage), 방해전파 방법(Active Jamming), 차단 태그(Blocker Tag)를 사용하는 방법들이 제안되었다.[8] 킬 태그 방법은 사용이 불필요한 분야에서 사용이 가능하며, 태그가 8비트의 패스워드와 킬 명령을 받을 경우 그 태그가 비활성화되게 된다. 패러데이 우리방식은 무선 주파수가 침투하지 못하도록 금속성의 그물

이나 박막을 입히는 방법이며, 방해전파 방법은 리더가 태그의 정보를 읽지 못하도록 방해신호를 보내는 물건을 소비자가 소지하는 방식이다. 차단자 태그는 리더의 모든 질의에 '그렇다' 혹은 '아니다'라는 일관적인 응답만 하는 태그들로, 보안 구역 정할 수 있다.[10]

◎ 방해전파 방법

방해전파 방법은 리더의 기능을 막거나 방해하는 신호를 브로드캐스트하는 장치를 이용하는 방법이다. 그래서 태그의 정보를 악의적인 목적을 가진 리더로부터 차단해 보호할 수 있다. 하지만 이 방법을 사용하면 주변의 다른 정당한 리더에게도 영향을 미칠 수 있고, 이 방법 자체가 공격방법으로 사용될 수 있으므로 특별한 경우가 아니면 사용할 수 없는 매우 강력한 해결책이다.[11]

◎ 킬 태그

이 방법은 MIT의 Auto-ID 센터에서 제안하는 방법으로 태그의 설계에 8비트의 패스워드를 포함하고 태그가 이 패스워드와 킬 명령을 받을 경우 태그가 비활성화되는 방식이다. 태그는 내부에 단락회로가 있기 때문에 이를 끊음으로서 자살 명령을 실행하게 되는데 이렇기 때문에 한 번 죽은 태그는 다시 살릴 수 없다. 이런 경우 태그를 재사용할 필요가 있는 분야에는 사용이 불가능하다.

물론 읽기 쓰기로 설계된 태그의 경우 플래그 비트를 이용하여 태그를 죽였다 다시 살릴 수도 있지만 이 경우 여전히 태그에 사용되는 8비트의 암호에 대한 문제가 남는다. 수많은 제품에 사용되는 태그라는 것을 감안하고 보안을 생각하다면 128비트 이상을 암호로 사용해야만 하지만 이는 태그에 상당한 부담이 된다. 태그마다 다른 암호를 사용하다면 이를 저장하는 것도 문제이다. 그리고 패스워드가 8비트 밖에 되지 않기 때문에 악의적인 사용자에 의해서 공격을 받을 우려가 있기 때문에 EPC global에서는 패스워드를 32비트로 규정하고 있다.[11]

◎ 패러데이 우리

페러데이 우리는 “전파는 금속을 투과할 수 없다.”라는 성질을 이용한 방법이다. 실제로 2005년 유로화의 RFID 시스템의 도입에 대비하여 돈 봉투에 금속성 그물을 입힌 상품이 나오기도 하였다. 그러나 이 경우 사용범위가 극히 제한적이고, 침묵하지 말아야할 태그를 침묵시키는 역효과가 나타날 수 있다. 또한 상품에 삽입된 태그의 위치를 알게 되면 태그 크기의 박막을 붙이는 것만으로도 이 상품은 도난을 방지할 수 없게 된다.[11]

◎ 차단 태그

차단 태그는 태그위에 붙이는 것으로 태그 탐색을 위한 리더의 모든 질문에 ‘그렇다’로 응답하는 태그를 말한다. 그래서 이 방식은 이진 트리 워킹 이나 ALOHA 충돌 방지 알고리즘을 사용하는 태그에서 사용할 수 있다. 모든 질문 메시지에 응답을 하기 때문에 이진 탐색 기법을 사용하여 태그를 읽어 들이는 방식에는 탐색의 모든 영역을 검색하게 되는 결과를 가져오고 ALOHA 기법을 사용하는 태그에서는 태그를 전혀 읽을 수 없도록 만든다. 태그의 고유 번호 길이가 길어지면 길어질수록 리더는 리더의 용량을 초과하는 태그를 찾기 위해 시도할 것이고 이는 리더에게 치명적인 결과를 가져올 것이다. 이를 조금만 응용한다면 리더에 대한 서비스 거부 공격을 실행할 수 있다는 것을 쉽게 알 수 있다. 이를 해결하기 위해 리더 친화적인 방법이 강구되기도 했는데 우선 리더는 자신이 읽으려는 태그 그룹에게 차단 태그가 있는지 물어보고 차단 태그는 이에 대해 응답을 한다. 이런 차단 태그는 보호 구역을 만들어서 차단 태그와 동일한 시작비트를 갖는 태그들을 보호할 수 있다. 하지만 악의적인 목적을 가진 사용자에게 리더로의 DoS공격을 가능하게 하는 툴로써 이용될 수 있는 단점이 있다.[11]

3. 인증 및 접근 제어 기술

인증 및 접근 제어 기술은 해쉬 기반 접근 제어나 랜덤 접근 제어 방법론이 있다.[12] 해쉬 기반 접근 제어 방법은 태그를 잠그고 풀기 위하여, 리더가 랜덤한

키를 해쉬하여 데이터베이스에 저장하고 이를 태그의 Matal ID로 사용한다. 랜덤 접근 제어 방법에서 태그는 Matal ID뿐만 아니라 난수를 사용하여 보안성을 높일 수 있지만, 리더가 연산해야 할 태그가 많을수록 인증하는데 걸리는 시간이 늘어난다는 단점을 가진다.

◎ 해쉬-락

해쉬 기반 접근 제어 은 태그에 대한 접근 제어 기술이다. 접근 제어는 허가된 사용자만이 태그를 접근할 수 있도록 하기 위한 방법을 말한다. 태그를 잠그기 위하여, 리더는 난수 형태의 키를 해쉬하여 데이터베이스에 저장하고 이를 태그의 Matal ID로 사용한다. 그리고 리더는 태그에게 Matal ID를 보내고 태그는 이를 저장하고 잠김 상태가 된다.

태그를 풀기 위하여 리더는 태그에게 잠김을 풀려고 한다는 메시지를 보낸다. 태그는 자신이 저장했던 Matal ID를 리더에게 보내고 리더는 데이터베이스에서 Matal ID에 해당하는 ID와 자신이 생성했던 키를 가져온다. 리더는 태그에게 키를 보내고 태그는 이 키를 해쉬해서 나온 값이 자신이 Matal ID와 동일하다면 태그는 잠김 상태에서 빠져 나와 주위의 리더에게 반응하게 된다.[13]

제 4 장 토큰키와 해쉬 함수를 이용한 인증 방법 설계

제 1 절 기존의 해쉬 인증방법

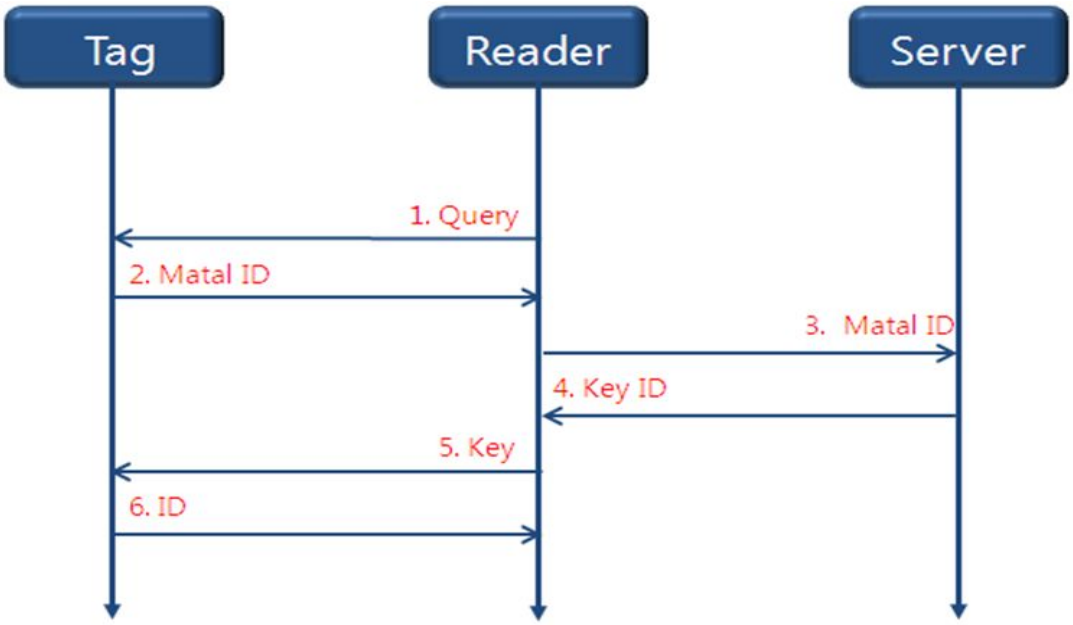

그림 4.1 MIT 해쉬 함수

그림 4.1은 MIT 해쉬함수이며, 인증된 리더만이 태그의 정보를 수신할 수 있는 방법으로, Weis[11]가 제안하였다. 인증과정은 다음과 같다.[13,14]

① Query

리더는 태그에게 정보전송을 요청한다.

② Matal ID

태그는 자신의 키에 대한 Hash 값인 Metal ID값을 계산한 뒤, 리더가 제시하

는 Query에 대해 응답한다.

Metal ID = Hash(key)

③ Matal ID

태그의 Lock 상태를 풀기 위해 리더는 Metal ID값을 서버에 Query로 제시한다.

④ Key ID

서버는 저장된 ID에 대한 키를 찾아 리더에게 전달한다.

⑤ Key

찾은 키를 리더가 태그에게 전송하면 태그는 그 키를 Hash 함수로 연산하여 Metal ID값과 일치한지 검사한다.

⑥ ID

일치하면 Lock 상태가 풀고, 근처의 리더에게 자신의 ID 및 모든 기능을 제공한다.

해쉬락 기법에서는 리더와 태그가 모두 동일한 해쉬 함수를 미리 가지고 있으며, 인증 정보를 생성하고 태그를 잠금 상태로 만드는 과정과 실제 태그-리더간 인증을 수행하는 풀림 과정으로 구성된다. 해쉬락 기법의 경우 스푸핑 공격에 대해 안전하지 않다. 스푸핑 공격은 공격자가 태그에게는 리더로 가장하고, 리더에게는 태그로 가장하는 공격 기법으로, 공격자는 메타 ID와 키를 모두 알아낼 수 있다. 스푸핑 공격을 방지하기 위해 태그가 잠금을 풀기 전에 한번 더 ID를 서버에서 검색하는 방법이 제안되었다. 그러나, 이 방법은 마지막에 보내는 ID가 여전히 식별자의 역할을 수행하기 때문에 사용자에 대한 추적이 가능하다.

제 2 절 토큰키와 해쉬 함수를 이용한 인증방법

1. 키 분배 기법

RFID 시스템은 브로드캐스트 방식을 사용하기 때문에 여러 가지 보안문제를 가지고 있다. 이를 보안하기 위해 사용되는 기술들 중에서 인증 및 접근 제어 기술의 해쉬-락 기법은 정보가 노출된다는 문제점을 가지고 있다. 또한 기존의 RFID 인증 시스템은 해쉬-락 기반으로 하는 방식이며, 키 분배도 없다. 그렇기 때문에 태그의 정보가 쉽게 누출이 되며, 또한 태그의 복제가 가능하다.[15]

본 논문에서 제안하는 키 분배 방식은 키 노출의 문제점을 보안한 방식이다. 해쉬-락 기법에서 키 분배 방식은 리더가 무선통신을 이용하여 태그에게 키를 전송하는 방식이다. 이 방식은 키가 아무런 보호를 받지 않기 때문에 도청공격이나 재생공격과 같은 여러 공격에 매우 취약하다.

제안하는 인증방법은 포핸드인증 방식으로 태그와 리더간의 인증과정은 키 대칭과정을 의해 이루어진다. 이 방법은 키가 노출될 수 있기 때문에 아주 위험한 방법이다. 그래서 제안하는 방식은 분배된 키를 이용해 새로운 키를 유출해 내는 방식이다. 제안하는 인증방법에서 키 분배방식은 리더가 전송한 키와 태그의 아이디 넘버를 기반으로 쉬프트연산을 통해 새로운 키를 생성한다. 이 키는 외부로 전송되거나 노출되지 않았기 때문에 도청공격이나 재생공격과 같은 공격에 강하다. 제안하는 키 분배 방식에서 간단한 쉬프트 방식을 사용하는 이유는 태그의 제한된 환경적 요소인 게이트의 수 때문에 사용하였다. 토큰키 생성 방법은 다음과 같은 절차에 의해 이루어진다.

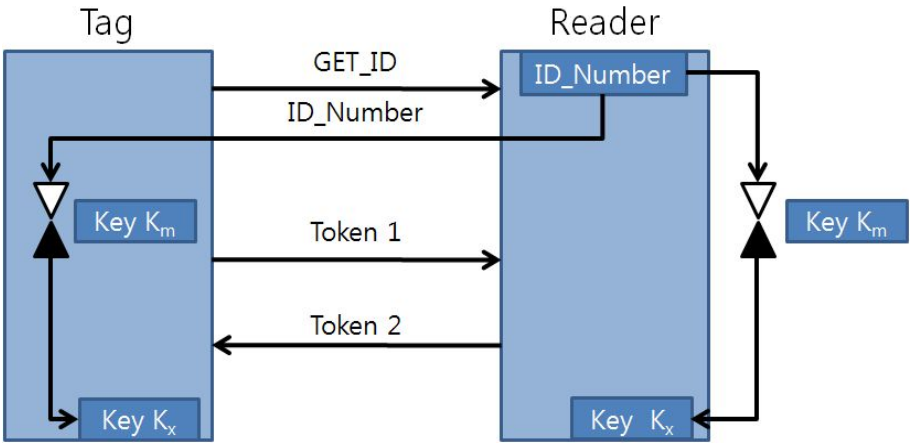

그림 4.2 암호화 알고리즘 상호인증 키 분배

① GET_ID

태그가 리더에게 자신의 고유 아이디를 전송한다.

② ID_Number

태그에서 전송받은 ID를 기반으로 데이터베이스에 저장된 정보를 태그에게 전송한다.

③ Create Key

Key K_m=

(Key K_x ⊕ ID_Number)

리더와 태그는 ID_Number와 K_x를 조합하여 새로운 키 K_m을 생성한다. 조합 방식은 키 쉬프트 방식이며, ID_Number만큼 K_m를 우측으로 쉬프트 한다.

즉, ID_Number가 1일 경우 우측으로 한 칸 쉬프트 되며 0일 경우 쉬프트 되지 않는다.

④ Compress Token Key

생성된 키인 K_m을 토큰키로 사용하여 서로 맞는 지 확인한다.

Token 1 = EK_m(ID ‖ Key K_x)

Token 2 = EK_m(ID ‖ ID_Number ‖ Key K_x)

제안하는 키 분배 방법에서 3번 과정은 생성된 키 K_m은 분배된 키 K_x와 ID_Number를 쉬프트 연산을 통해 얻어지는 새로운 키이므로 외부에서 도청 공격하는 공격자는 분배되는 키 K_x을 획득을 할 수 있지만 생성된 키 K_m을 획득할 수 없다. 4번 과정에서는 내부 연산을 통해 생성된 키 K_m을 암호키로 하여 대칭 암호화한 두 개의 토큰키를 생성한다. 두 개의 토큰키는 리더와 태그사이 접근키로 사용되며 이 값이 일치 하지 않으면 리더와 태그는 서로 통신을 할 수 없다. 만약 공격자가 토큰키를 획득하더라도 복호키를 알 수 없기 때문에 안에 저장된 내용을 알 수 없다. 그러므로 인증과정에서도 사용되는 해쉬함수도 안전하게 된다.

2. 인증 방법

기존의 해쉬함수를 이용하는 인증방법은 태그와 리더사이의 핸드쉐이크 과정이 4번으로 되어있지만 제안하는 방법은 3번으로 줄였다. 과정에 줄었기 때문에 효율성 측면에서는 기존의 인증 방법보다 높다는 것을 알 수 있으며, 또한 외부로 노출되는 빈도를 줄였기 때문에 공격자에 노출되는 과정이 줄게 되었다. 또한 제안하는 인증 방법은 2차 해쉬함수를 이용하여 이루어진 방법이다.

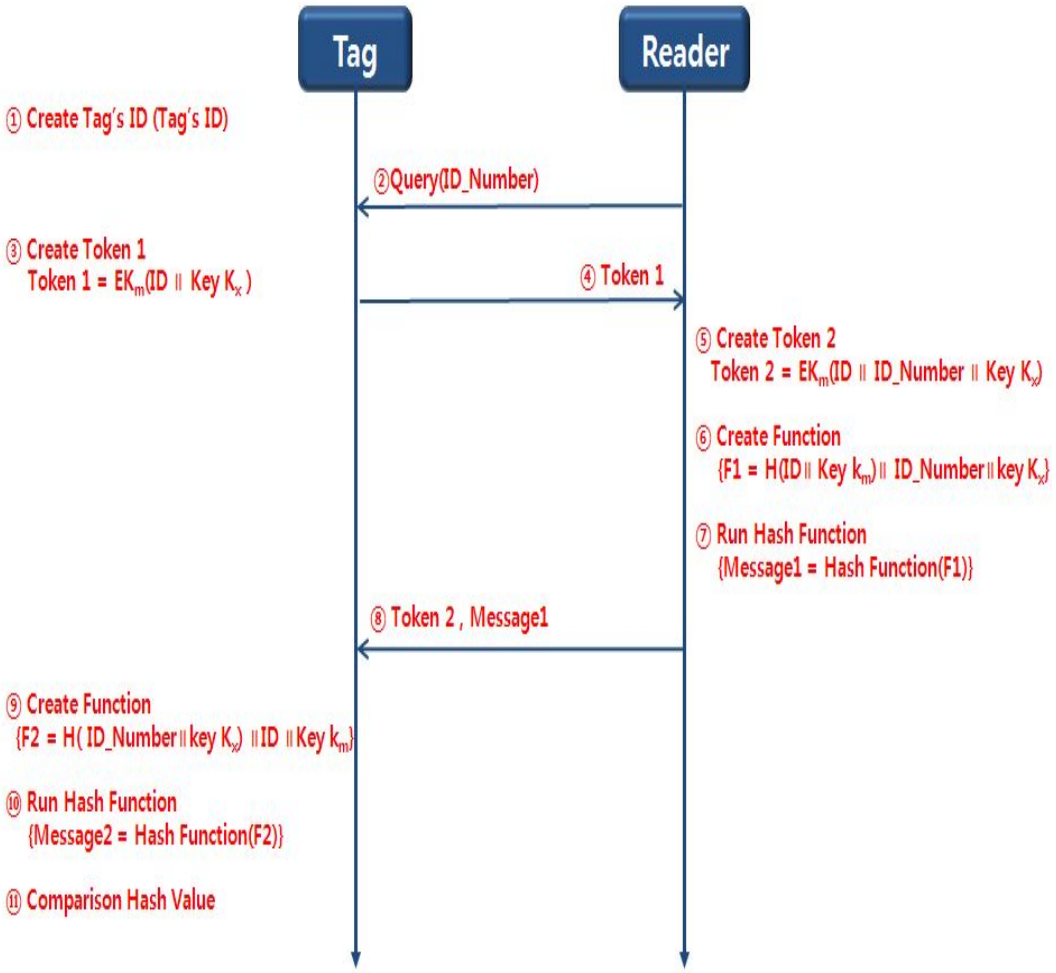

그림 4.3 태그와 리더간의 인증과정

① Create Tag's ID

태그는 ID를 생성한다.

② Query(ID_Number)

리더는 Query를 태그에게 전송한다.

③ Send Token 1

Token 1 = EK_m(ID ‖ Key K_x)

태그는 리더에게 토큰키를 전송한다.

1, 2, 3번 과정은 키 생성과정에서 이루어지는 부분이다.

④ Create Hash Function

{f1= H(ID ‖ Key K_x) ‖ ID_Number ‖ Key K_m}

리더는 키 생성을 통해 얻은 K_m과 태그의 ID를 해쉬 연산한 다음 ID_Number와 토큰에 얻은 K_m으로 조합한다.

⑤ Run to Hash Function

{Message1= Hash Function(f1)}

4번 과정에서 조합된 값을 해쉬함수로 연산한다.

⑥ Send Token2, Message

Token 2 = EK_m(ID ‖ ID_Number ‖ Key K_x)

Message

리더는 태그에게 토큰키와 5번 과정에서 생성된 값을 전송한다.

⑦ Create Hash Function

{f1= H(ID_Number ‖ Key K_m) ‖ ID ‖ Key K_x}

태그는 토큰에서 얻은 K_m과 ID_Number를 해쉬 연산한 다음 ID와 키 생성을 통해 얻은 K_x으로 조합한다.

⑧ Run to Hash Function

{Message2= Hash Function(f2)}

7번 과정에서 조합된 값을 해쉬함수로 연산한다.

⑨ Compress Hash Value

5번 과정에서 리더로부터 전송받은 해쉬함수 값과 8번 과정에서 생성된 해쉬함수 값을 비교한다. 이 과정에서 두 해쉬함수 값이 같다면 태그와 리더간의 인증은 성공적으로 이루어지고 리더는 태그에 접근이 가능하게 되며, 태그에 저장된 데이터를 읽을 수 있게 된다.

제 3 절 토큰키에 난수를 추가한 인증 방법

1. 키 분배 기법

첫 번째 제안하는 인증 방법인 토큰키와 해쉬함수를 이용한 인증 방법은 저가형 태그를 위해 설계한 인증 방법이다. 저가형 태그는 가지고 있는 환경요소가 극히 적기 때문에 많은 연산을 하지 못한다. 그래서 첫 번째 제안하는 인증 방법은 많은 제약을 가지고 설계를 하였으며, 두 번째 제안하는 인증 방법은 좀 더 높은 연산능력을 가지는 태그를 위해 설계를 하였다. 첫 번째 제안하는 인증 방법은 매번 접속할 때 사용하는 토큰키가 같다는 취약점을 가지고 있지만 두 번째 제안하는 인증 방법은 좀 더 많은 연산이 가능하므로 랜덤 함수를 추출할 수 있는 기능을 가진 인증 방법을 설계하였다. 랜덤 함수는 리더에서 전송받은 태그의 ID_Number를 기초로 추출하며, 이는 매번 접속할 때 사용하는 토큰키의 일회성 성질을 부여하게 된다. 토큰키 생성과정은 다음과 같은 절차에 의해 이루어진다.

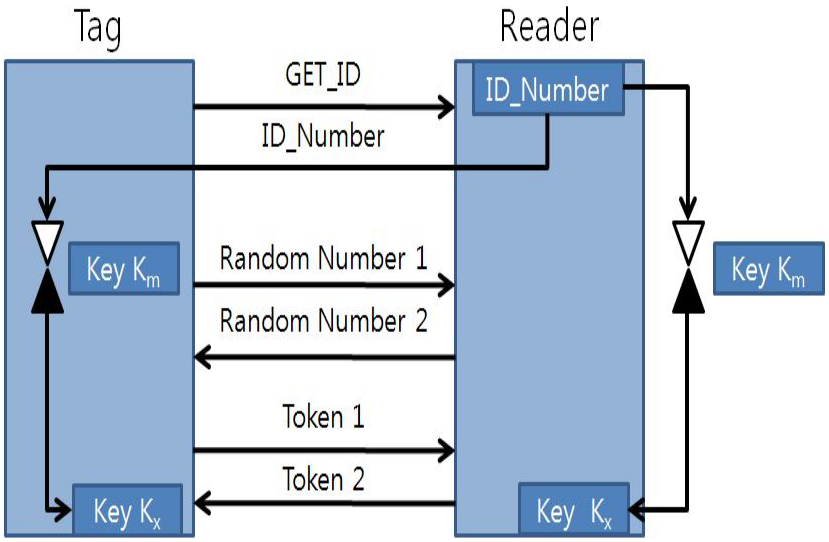

그림 4.4 난수 발생 키 분배

① GET_ID

태그는 ID를 리더에게 전송한다.

② ID_Number

태그에서 전송받은 ID를 기반으로 데이터베이스에 저장된 정보를 태그에게 전송한다.

③ Random Number 1

태그는 전송받은 ID_Number를 기반으로 3자리 숫자를 추출하여 리더에게 전송한다.

④ Random Number 2

리더는 ID_Number를 기반으로 3자리 숫자를 추출하여 태그에게 전송한다.

⑤ Create Key

Key K_m=

(Key k_x ⊕ ID_Number)

리더와 태그는 전송받은 데이터를 가지고 새로운 키를 생성한다.

⑥ Token 1

Token 1=

Ek_m(Random Number1 ‖ Tag's ID)

생성된 키를 기반으로 대칭키 암호화 방식을 사용하여 ID와 Random Number1을 암호화하여 리더에 전송한다. 이는 태그와 리더간의 인증과정에서 토큰키로 사용된다.

⑦ Token 2

Token 2=

Ek_m(Random Number2 ‖ ID_Number ‖ Tag's ID)

생성된 키를 기반으로 대칭키 암호화 방식을 사용하여 Random Number2, ID_Number, ID를 암호화하여 태그에 전송하다. 이는 마찬가지로 태그와 리더간의 인증 과정에서 토큰키로 사용된다.

제안하는 인증 방법의 토큰키 생성과정 첫 번째 인증 방법의 토큰키 생성과정과 비슷하다. 5번 과정에서 생성되는 키 K_m은 분배된 키 K_x와 ID_Number를 쉬프트 연산을 통해 얻어지며 이 키는 외부에 노출되지 않는다. 3번 과정과 4번 과정에서 얻어지는 Random Number는 태그의 ID_Number에서 무작위로 순서에 관계없이 추출되는 방법을 사용하였다. 그리고 6번 과정과 7번 과정에서 생성되는 토큰키는 5번 과정에 생성된 키를 기반으로 암호화 하였으며, 안에 랜덤함수를 첨부하였기 때문에 매번 접속할 때 토큰키가 변하는 성질을 가지게 된다. 공격자가 토큰키를 복사해 그대로 사용하더라도 매번 토큰키가 변하기 때문에 공격자는 통신에 참여를 할 수 없게 된다. 그러므로 인증과정에서 사용되는 해쉬함수도 안전하게 되는 것이다.

2. 인증 방법

인증 방법은 첫 번째 제안하는 인증 방법과 같이 2차 해쉬함수를 기반으로 설계하였다. 첫 번째 제안한 인증방법과 다른 점이 있다면 해쉬함수에 들어가는 변수가 랜덤함수가 들어가므로 매번 인증할 때 사용되는 해쉬함수의 결과 값이 일회성의 성질을 가지고 있다. 그러므로 인증 값을 도중에 가로채서 공격하는 공격방법에도 취약하지 않게 될 것이다.

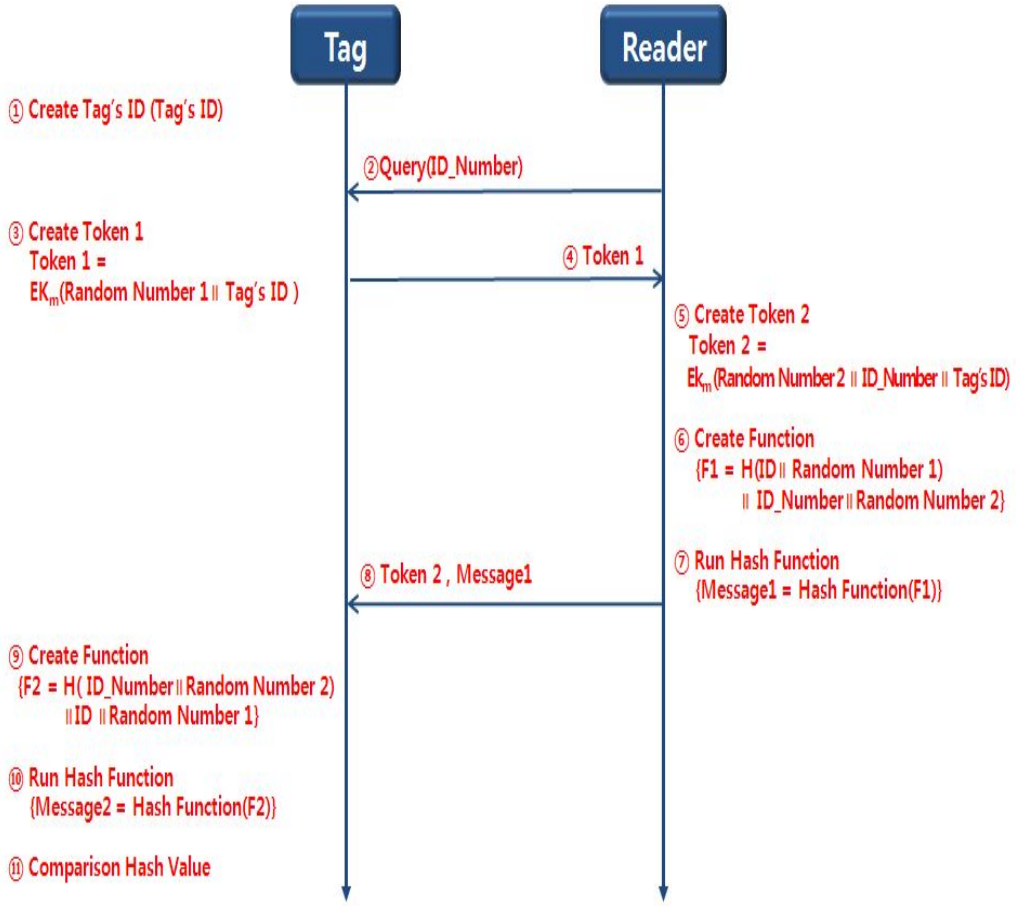

그림 4.5 태그와 리더간의 인증과정

① Create Tag Key(Tag's ID, Random Number1)

태그는 아이디와 Random Number1을 생성한다. 이 과정은 키 분배과정을 통해 이루어진다.

② Create Token Key

{Ek_m(Random Number1 ‖ Tag's ID}

태그는 리더에 접근을 하기 위한 토큰키를 생성하며 이 과정 또한 키 분배과정에서 생성된 토큰키를 사용된다.

③ Send Message {Message1=(Token Key1)}

생성된 토큰키를 전송하여 이를 기반으로 리더에 접근이 가능하게 된다. 새로 생성된 키로 암호화를 하였기 때문에 아이디는 유출되지 않는다.

④ Create Token Key

Ek_m(Random Number2 ‖ ID_Number ‖ Tag's ID)

리더는 태그에 접근을 하기 위한 토큰키를 생성하며 이 과정 또한 키 분배과정에서 생성되는 토큰키를 사용한다.

⑤ Create Hash Function

{f1 = H(Tag's ID ‖ Random Number1) ‖ ID_Number ‖ Random Number2}

태그의 ID와 태그에서 생성된 Random Number를 기반으로 해쉬함수를 연산하도록 하며 리더에 전송되는 ID_Number와 Random Number를 해쉬함수와 같이 연산하는 방정식을 생성한다.

⑥ Run to Hash Function

{Hash Value1=Hash Function(f1)}

5번 과정에서 생성된 해쉬함수를 연산을 한다.

⑦ Send Message

{Message2=(Hash Value1, Token Key2)}

연산되어 생성된 값과 리더에서 생성된 토큰키를 태그에게 전송한다.

⑧ Create Hash Function

{f2 = H(ID_Number ‖ Random Number2) ‖ Tag's ID ‖ Random Number1 }

리더에서 전송한 ID_Number와 Random Number를 가지고 해쉬함수를 연산하도록 하며 태그에서 생성된 ID와 Random Number를 해쉬함수와 같이 연산하는 방정식을 생성한다.

⑨ Run to Hash Function

{Hash Value1=Hash Function(f2)}

8번 과정에서 생성된 해쉬함수를 연산을 한다.

⑩ Compress Hash Value

6번 과정에서 생성된 해쉬값과 9번 과정에서 생성된 해쉬값을 비교하여 일치하면 인증이 되며 접근이 가능하게 된다.

제 5 장 실험 및 고찰

제안하는 인증 방법과 기존의 해쉬-락 인증을 서비스 거부 공격, 재생 공격, 그리고 도청 공격과 같은 3가지 형태의 공격을 통해 인증 방법의 안정성 측면에서 비교 분석을 한다. 또한 RFID 시스템의 전송 특성상 무선이라는 점을 감안하여 높은 보안성을 가진 인증 방법이라고 해도 전송 효율성이 떨어진다면 사용자 입장에서는 사용을 할 수 없기 때문에 전송 효율성 측면에서도 비교 분석을 한다.

제 1 절 시뮬레이션 구현

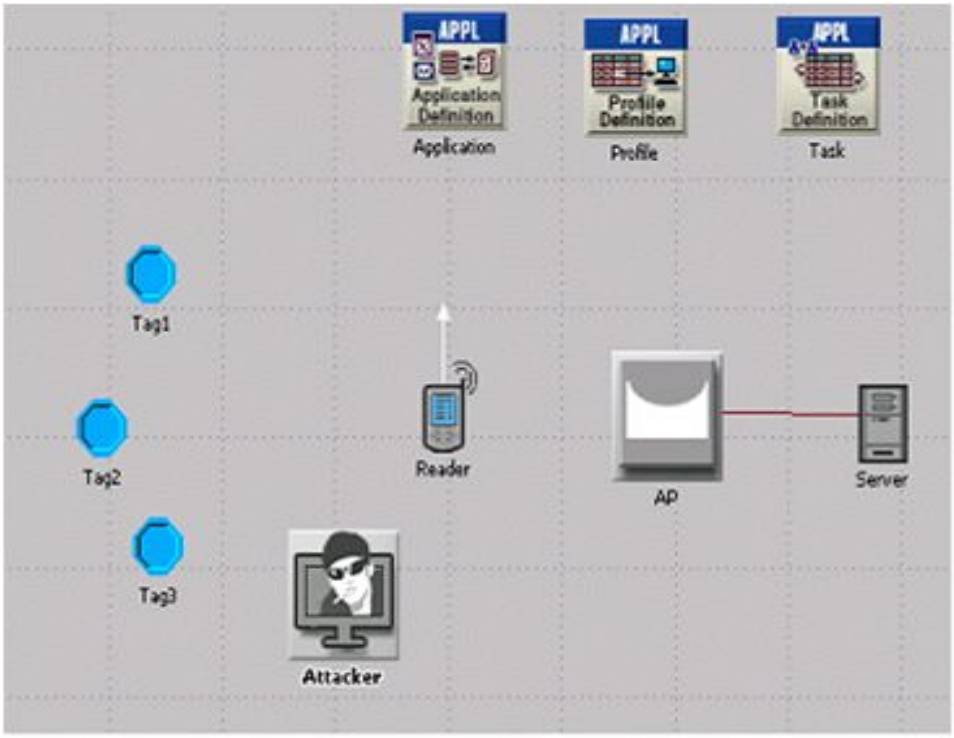

그림 5.1 시뮬레이션 환경

표 5.1 시뮬레이션 파라미터

Statistics	Value
Scenario Size	100m X 100m
Transmission Range (Tag → Reader)	< 5 meter
Transmission Range (Reader → Server)	< 100 meter
Tag Transmit Power	0.001W
Reader Transmit Power	0.005W
Tag Type	Semi-passive
Tag Processer	54 bit
Tag Date Rate	1 Mbps
Reader Data Rate	11 Mbps
Simulation Time	1 hour

시뮬레이션은 OPNET14.5에서 구현하였다. 기본적인 RFID 시스템에서 하였으며, 태그 3개, 유동성 리더 1개, AP 1개, 서버 1개, 그리고 공격자 1개로 구성되었다. 태그는 능-수동방식의 태그를 사용하였다. 능-수동방식은 능동형 방식과 수동형 방식을 결합해 놓은 하이브리드형 태그이다. 배터리는 내장되어 있으며 이 배터리는 내부적인 프로세싱에 사용된다. 그림 5.1은 시뮬레이션을 OPENT 14.5에서 구현한 화면이다. 공격자는 태그와 리더간의 통신을 공격하며 이 경우 포핸드 인증방식이 공격을 받게 된다. 공격방법은 서비스 거부 공격, 도청공격, 재생공격으로 총 3가지의 방법으로 하였다. 서비스 거부 공격은 RFID 시스템을 악의적으로 공격하여 태그와 리더의 자원을 부족하게 하여 원활한 통신을 하지 못하게 하는 공격이다. 리더에게 수많은 접속 신호를 만들어 태그가 정상적으로 접속을 하지 못하게

하거나, 리더의 TCP 연결을 바닥내는 등의 공격을 한다. 공격자가 사용하는 공격방법들 중 도청 공격과 재생 공격의 특성은 거의 똑같다. 도청공격은 인증되지 않은 리더가 태그에 접근하여 태그의 출력을 얻어 이를 기반으로 태그의 ID와 정보를 획득하는 공격방법이다. 재생공격은 리더와 태그사이의 인증과정에서도 발생될 수 있는 공격방법으로 통신과정에서 발생하는 신호를 그대로 재생하여 보내는 공격방법이다. 재생공격과 도청 공격은 리더와 태그사이의 통신을 획득한 정보를 토대로 이루어지는 공격방법이다.

만약 제안하는 인증 방법이 3가지 공격에 취약하다면 공격자 리더나 태그에 전송하는 정보를 처리하게 된다. 이에 따라 리더와 태그가 처리해야하는 정보가 증가하게 되는 현상이 발생하게 되며, 이 현상은 리더와 태그사이에 원활한 통신을 할 수 없게 만들며 그로 인해 응답시간이 지연되게 된다. 또한 공격에 취약한 인증방법일 경우 공격자가 전송한 데이터를 응답하게 되어 태그와 리더가 가진 정보를 공격자에게 전송하게 되는 문제점을 발생하게 될 것이다.

제 2 절 시뮬레이션 결과 분석

1. 토큰키와 해쉬함수를 이용한 인증 방법

▶ 재생 공격(Replay Attack)에 대한 반응

재생 공격은 공격자가 리더와 태그사이에서 일어나는 통신 신호를 획득하여 그대로 다시 전송하는 공격방법이다. 그림 5.2에서 재생 공격에 대한 응답을 보면 기존의 해쉬-락 인증 방법은 37[Sec]로 나왔으며, 이는 재생 공격에 취약하다는 결과값을 보여준다. 또한 제안하는 인증 방법은 27[Sec]로 나왔으며, 마찬가지로 재생 공격에 취약하다는 결과값을 보여준다. 이는 리더와 태그사이 통신에 사용되는 접근키인 토큰키가 그대로 복사되어 접속을 하였기 때문에 재생 공격에는 취약하다는 결과를 보여준다.

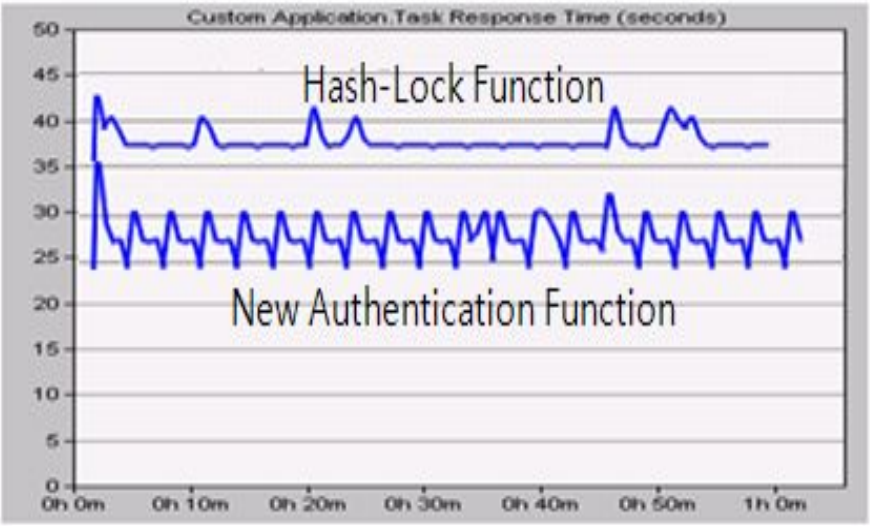

그림 5.2 재생 공격에 대한 응답

표 5.2 재생 공격에 대한 응답

	해쉬-락 기법	제안하는 인증기법	차이
응답시간	37Sec	27Sec	10Sec

▶ 서비스 거부 공격(Denial of Service Attack)에 대한 반응

서비스 거부 공격은 리더나 태그에게 수많은 접속 신호를 보내는 공격방법이다. 그림 5.3에서 서비스 거부 공격에 대한 응답을 보면 기존의 해쉬-락 인증 방법은 37[Sec]로 나왔으며 이는 서비스 거부 공격에 취약하다는 결과값을 보여준다. 기존의 해쉬-락 인증 방법은 계속적으로 접속을 요청하는 공격자의 정보를 무시하지 않고 받아드린다는 것이다. 하지만 서비스 거부 공격에 대한 제안하는 인증 방법의 응답은 3[Sec]로 나왔다. 제안하는 인증 방법은 토큰키를 사용하는 방법이다. 토큰키는 암호화를 통해 만들어지며 암호화 키 또한 내부적으로 생성되는 키이기 때문에 공격자는 토큰키를 생성하지 못하게 된다. 그러므로 토큰키가 없는 공격자는 리더나 태그에게 접속 요청을 해도 리더나 태그는 그 정보를 받아들이지 않게 된다.

태그는 제한적인 환경요소를 가진 매체이다. 제한적 요소부분이 한계점에 이르면 태그의 수명은 짧아지게 된다. 기존의 해쉬-락 인증 방법은 서비스 거부 공격에 취약하기 때문에 공격자가 보낸 정보를 처리하거나 응답을 하는데 제한적 요소가 소모되며, 그 결과 태그의 수명이 짧아져 유지 보수 비용이 추가로 들어가는 문제점이 발생된다. 하지만 제안한 인증 방법 공격자의 서비스 거부 공격에 취약하지 않다. 토큰키가 없는 공격자가 보내는 정보를 태그가 처리하지 않아도 되기 때문에 태그의 제한적 요소를 소모되는 현상이 일어나지 않아 유지 보수 비용이 추가로 들어가지 않은 장점을 가지고 있다.

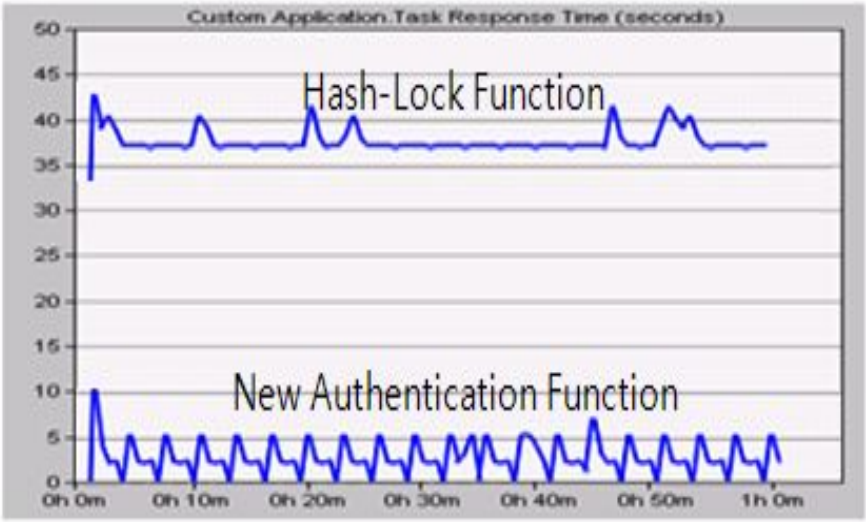

그림 5.3 서비스 거부 공격에 대한 응답

표 5.3 서비스 거부 공격에 대한 응답

	해쉬-락 기법	제안하는 인증기법	차이
응답시간	37Sec	3Sec	34Sec

▶ 도청 공격(Eavesdropping Attack)에 대한 반응

도청 공격은 리더와 태그사이에서 발생하는 통신신호를 해석하여 그 정보를 획득하는 공격 방법이다. 이는 공격자가 리더나 태그사이에 전송되는 아이디나 패스워드를 획득하고 이를 기반으로 정보를 전송하여, 수신하는 리더나 태그 입장에서는 전송받은 정보가 아무런 문제가 없기 때문에 공격자를 인증된 장치로 인식하여 응답을 하게 된다. 그림 5.4를 보면 기존의 해쉬-락 인증 방법의 응답이 4.7[Sec]라는 것을 볼 수 있다. 이는 공격자가 획득한 아이디나 패스워드를 기반으로 전송한 정보가 맞다는 것을 의미하며, 이는 공격자에게 리더나 태그의 아이디나 패스워드가 노출되었다는 것을 의미한다. 그림 5.4를 보면 제안하는 인증 방법의 응답이 0.2[Sec]라는 것을 볼 수 있다. 이는 공격자가 리더와 태그사이에서 발생한 신호를 해석을 하지 못하였다는 것을 의미한다. 제안하는 인증 방법은 내부적으로 쉬프트함수를 이용하여 새로운 키를 생성하여 이를 기반으로 토큰키를 생성하는 방법을 사용한다. 만약 공격자가 토큰키를 획득하여도 이를 복호화하는 키를 알지 못하기 때문에 키 안에 들어있는 정보를 획득할 수 없게 된다. 그러므로 제안하는 인증 방법은 도청공격에 강하다는 것을 알 수 있다.

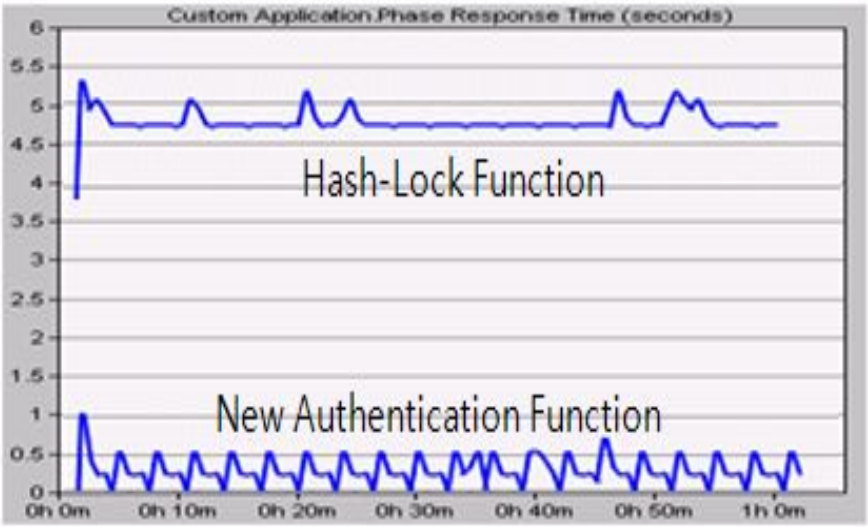

그림 5.4 도청 공격에 대한 응답

표 5.4 도청 공격에 대한 응답

	해쉬-락 기법	제안하는 인증기법	차이
응답시간	4.7Sec	0.2Sec	4.5Sec

2. 토큰키에 난수를 추가한 인증방법

▶ 재생 공격(Replay Attack)에 대한 반응

재생 공격은 첫 번째 제안하는 인증과 같이 리더와 태그사이에 발생하는 통신 정보를 획득하여 그대로 다시 전송하는 공격방법이다. 그림 5.5에서 재생 공격에 대한 응답을 보면 기존의 해쉬-락 인증 방법은 37[Sec]로 재생 공격에 취약하다는 것을 보여준다. 하지만 두 번째로 제안하는 인증 방법의 응답은 3[Sec]로 재생 공격에 강하다는 것을 보여준다.

다시 정리하면 재생공격에 대한 응답은 기존의 해쉬-락 인증 방법이 37[Sec], 첫 번째 제안하는 인증 방법이 27[Sec], 두 번째 로 제안하는 인증 방법이 3[Sec]이다. 기존의 해쉬-락 인증 방법의 경우 리더와 태그사이에 발생하는 통신 정보는 아무런 암호화 과정도 없고, 이를 다시 전송하더라도 리더나 태그는 공격자라는 것을 확인할 방법이 없다. 그리고 첫 번째로 제안하는 인증 방법은 매번 접속할 때 사용되는 토큰키가 똑같기 때문에 공격자 토큰키를 그대로 재생하여 사용한다면 기존의 해쉬-락 인증 방법과 같이 리더나 태그는 공격자라는 것을 확인할 방법이 없다. 하지만 두 번째로 제안하는 인증 방법은 랜덤함수를 사용한다는 특성을 가지고 있다. 랜덤함수는 매번 접속할 때 사용되는 토큰키가 변한다는 성질을 부여한다. 이는 토큰키가 일회성의 특성을 가지고 있기 때문에 한번 사용된 키는 재생하여 다시 사용할 수 없게 된다. 그러므로 다른 인증방법과 달리 재생 공격에 응답을 하지 않게 되는 장점을 가지고 있다.

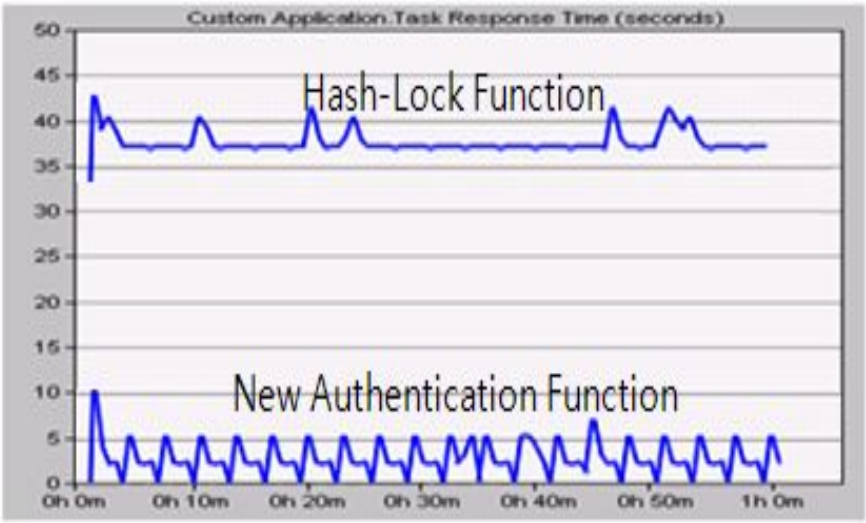

그림 5.5 재생 공격에 대한 응답

표 5.5 재생 공격에 대한 응답

	해쉬-락 기법	제안하는 인증기법	차이
응답시간	37Sec	3Sec	34Sec

▶ 서비스 거부 공격(Denial of Service Attack)에 대한 반응

서비스 거부 공격은 첫 번째 제안하는 인증 방법과 같은 공격방법이다. 그림 5.6에서 서비스 거부 공격에 대한 응답을 보면 기존의 해쉬-락 인증 방법은 37[Sec]로 서비스 거부 공격에 취약하다는 것을 보여준다. 하지만 두 번째로 제안하는 인증 방법의 응답은 3[Sec]로 서비스 거부 공격에 강하다는 것을 보여준다. 이는 두 번째로 제안하는 인증 방법이 첫 번째와 동일하게 토큰키를 사용하기 때문에 토큰키가 없는 공격자가 접근 요청을 하더라도 리더나 태그는 그 요청을 받아들이지 않게 된다. 그러므로 공격자의 접속요청에 의해 태그의 제한적 환경요소는 소모되지 않게 되어 공격에 의한 유지 보수 비용이 증가하지 않게 된다.

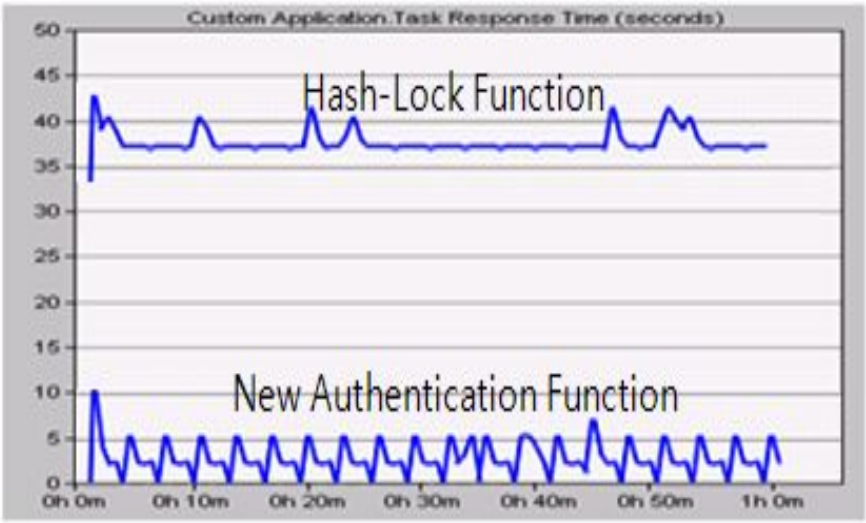

그림 5.6 서비스 거부 공격에 대한 응답

표 5.6 서비스 거부 공격에 대한 응답

	해쉬-락 기법	제안하는 인증기법	차이
응답시간	37Sec	3Sec	34Sec

▶ 도청 공격(Eavesdropping Attack)에 대한 반응

도청 공격은 첫 번째 제안하는 인증과 같이 리더와 태그사이에서 발생하는 통신 정보를 해석하여 그 정보를 획득하는 공격방법이다. 도청 공격은 리더와 태그사이의 정보를 습득하는 것으로 시작 한다. 그리고 습득한 정보를 기반으로 태그나 리더로 둔갑하여 네트워크에 참여하는 공격방법이기 때문에 다른 공격방법 보다 응답시간이 짧다. 그림 5.7에서 기존의 해쉬-락 인증 방법의 재생공격에 대한 응답은 4.7[Sec]이며 이 결과는 도청공격에 취약하다는 것을 보여준다. 하지만 두 번째로 제안하는 인증 방법의 응답은 0.2[Sec]로 도청 공격에 강하다는 것을 보여준다. 두 번째 인증 방법은 첫 번째 인증 방법과 같이 내부에서 생성되는 키를 가지

고 암호화한 토큰키를 사용한다. 이 키를 알지 못하는 공격자는 통신 정보를 획득할 수 없기 때문에 도청 공격에 강하다.

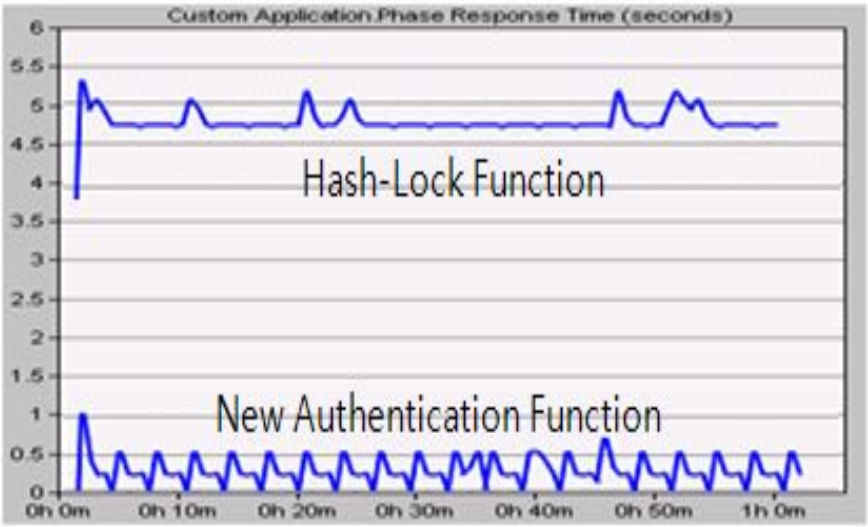

그림 5.7 도청 공격에 대한 응답

표 5.7 도청 공격에 대한 응답

	해쉬-락 기법	제안하는 인증기법	차이
응답시간	4.7Sec	0.2Sec	4.5Sec

3. 인증 기법 비교

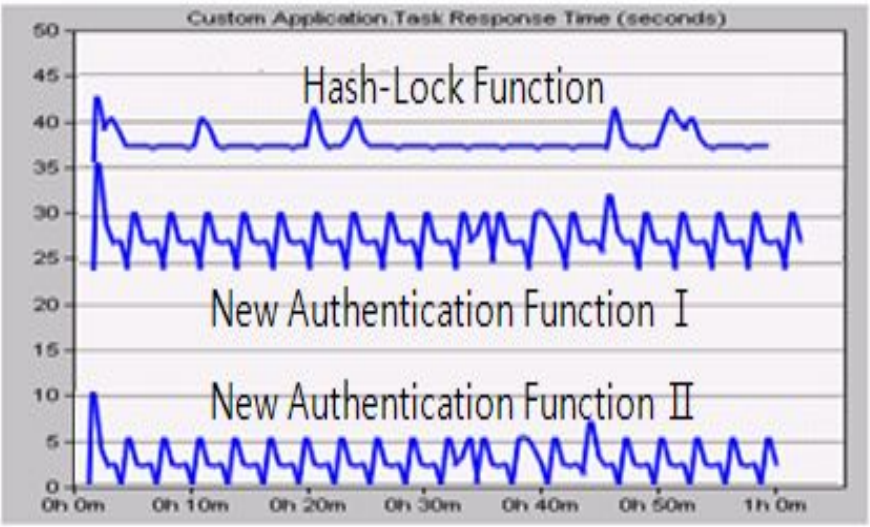

그림 5.8 재생 공격에 대한 응답

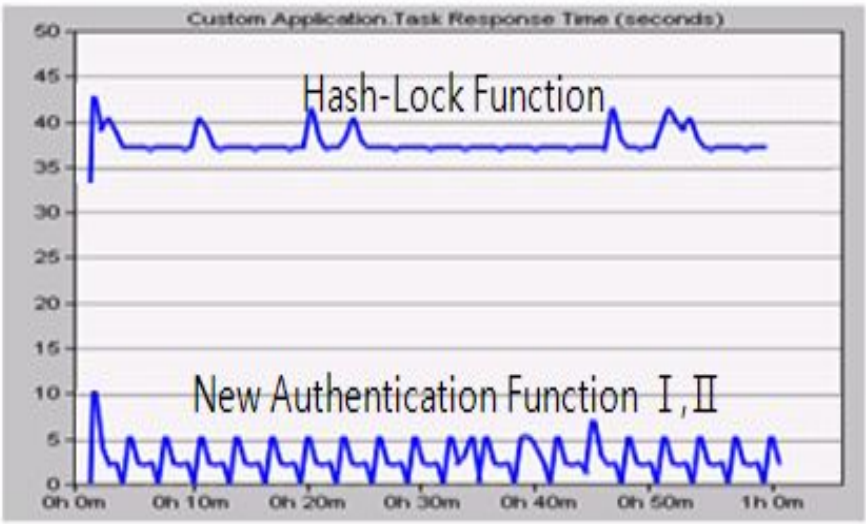

그림 5.9 서비스 거부 공격에 대한 응답

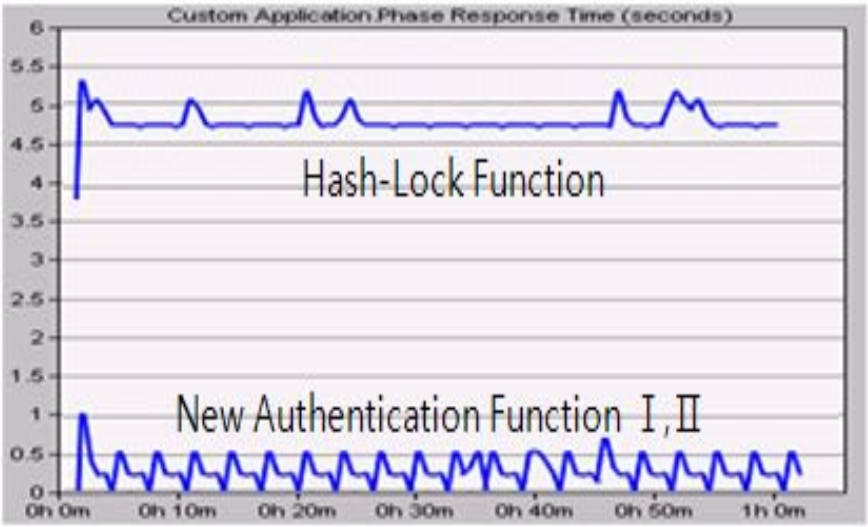

그림 5.10 도청 공격에 대한 응답

표 5.8 각 공격에 대한 응답시간 비교

	해쉬-락 기법	토큰키와 해쉬함수를 이용한 인증방법	토큰키에 난수를 추가한 인증방법
재생공격에 대한 응답 시간	37Sec	27Sec	3Sec
서비스 거부 공격에 대한 응답 시간	37Sec	3Sec	3Sec
도청공격에 대한 응답 시간	4.7Sec	0.2Sec	0.2Sec

재생 공격, 서비스 거부 공격 그리고 도청 공격에 대한 응답시간을 제안한 두가지 인증 기법인 토큰키와 해쉬함수를 이용한 인증방법과 토큰키에 난수를 추가한 인증방법에 대한 재생 공격, 서비스 거부 공격 그리고 도청 공격에 대한 응답시간을 표 5.8에 표시하엿다. 표 5.8에서 볼 수 있듯이 두 인증 방법은 서비스 거부 공격과 도청 공격에는 강한 방어력을 보인다. 하지만 재생 공격에서 다른 결과값을 보이는데 그 이유는 토큰키와 해쉬함수를 이용한 인증 방법은 매번 같은 토큰키를

이용하기 때문에 토큰키를 재생하여 공격하는 방법에는 취약하다. 그러나 토큰키에 난수를 추가한 인증방법은 매번 인증때마다 토큰키가 달라지므로 토큰키를 재생하여 공격하는 방법에는 강한 방어력을 가진다.

제 3 절 효율성 분석

1. 효율성 측면

시뮬레이션에서 통신 환경은 기본적인 RFID시스템 환경이며 공격자가 통신에 간섭하는 가운데 기존의 해쉬-락 인증 방법과 제안하는 인증 방법을 사용하여 전송량과 지연시간을 비교하였다.

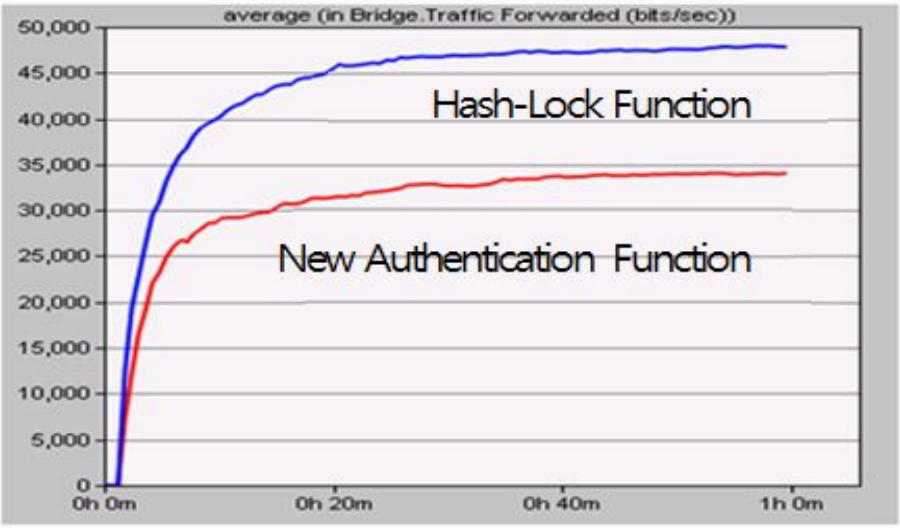

그림 5.11 도청 공격 중 태그와 리더간의 전송량[bits/sec]

위 그림 5.8는 기존의 해쉬-락 인증 방법과 제안하는 인증 방법의 전송량을 비교한 것이다. 기존의 해쉬-락의 평균 전송량은 45,000[bit/sec]이고, 제안하는 인증 방법은 30,000[bit/sec]로 차이가 15,000[bit/sec]가 나는 것으로 측정되었다.

공격자가 도청공격을 할 경우 기존의 해쉬-락 인증 방법은 공격자를 태그나 리더로 인식하여 공격자가 전송한 정보에 응답을 하기 때문에 리더와 태그 사이의 전송량이 높게 측정되었다. 하지만 기존의 인증방법은 리더나 태그는 공격자가 전송한 정보에 대한 인증이 이루어지지 않았기 때문에 공격자가 리더와 태그사이의 통신에 참여 할 수 없게 된 것이다. 다시 말하면 제안한 인증 방법은 공격자를 제외한 인증을 받은 리더와 태그 사이에 이루어진 통신량이 측정이 된 것이며, 기존의 해쉬-락 인증 방법은 인증을 받은 리더와 태그 그리고 공격자의 통신량이 모두 측정이 된 것이다. 이 차이는 공격자가 통신에 참여를 못하였다는 것을 의미하며, 그 의미는 태그나 리더의 정보 유출을 막았다고 볼 수 있다. 공격자가 보낸 정보를 처리하지 않아 태그와 리더는 원활한 통신을 할 수 있다는 것을 그림 5.9의 지연시간 비교를 통해서도 알 수 있다.

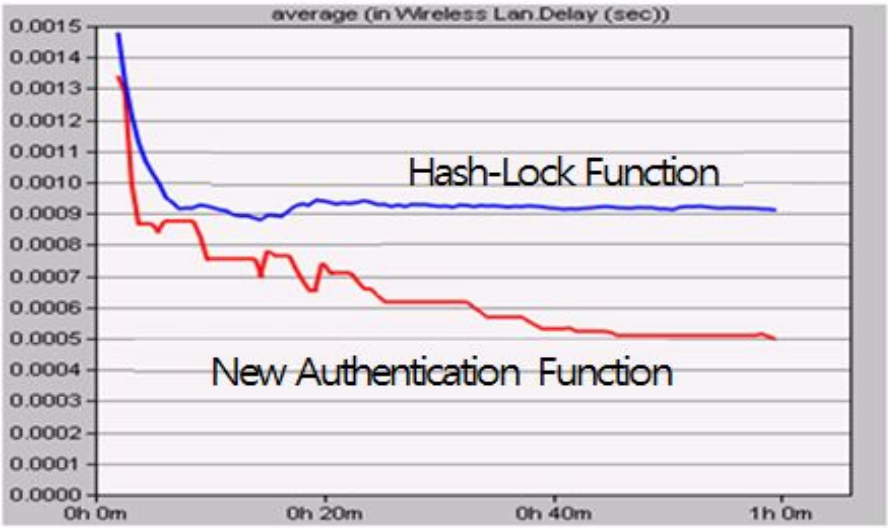

그림 5.12 도청 공격 중 태그와 리더간의 지연시간[sec]

그림 5.9는 기존의 해쉬-락 인증 방법과 제안하는 인증 방법의 지연시간을 비교한 것이다. 기존의 해쉬-락 인증 방법의 평균 지연시간은 0.009[Sec]이고, 제안하는 인증 방법의 평균 지연시간은 0.006[Sec]이다. 지연시간의 차이가 생기는 이유는 공격 정보를 전송하여 생기는 통신 지연시간으로 기존의 해쉬-락 인증 방법보다

제안하는 인증 방법이 공격에 취약하지 않기 때문에 통신의 지연시간이 감소한 것을 알 수 있다. 또한 두 개의 결과값으로 제안하는 방식이 기존의 인증 방식보다 보다 안전하다는 것을 입증하게 된다. 공격자가 전송하는 데이터는 인증이 되지 않는 데이터이므로 리더와 태그는 그에 대한 응답을 전혀 하지 않으며 그로 인해 공격자의 공격으로 인해 생기는 딜레이가 점점 줄어들어 태그와 리더간의 통신이 공격자의 간섭이 없이 통신을 할 수 있다는 것을 보여준다.

제 6 장 결 론

RFID는 매우 유용한 시스템이지만 리더의 제한된 환경적 요소로 인해 보안이 매우 취약하다. 보안적 문제를 해결하기 위해 해쉬-락 인증 방법이 많이 사용되고 있지만 이 방법은 아이디 및 패스워드 유출이라는 문제점을 가지고 있다. 그래서 본 논문에서는 해쉬-락 인증 방법을 변형시킨 두가지 인증 방법을 제안하고 있다.

본 논문에서는 두가지 인증 방법의 특성 분석을 통해 다음과 같은 결론을 얻었다.

1. 토큰키와 해쉬 함수를 이용한 인증 방법
   - 키 분배를 통해 새로운 키를 얻어내는 과정을 추가하여 키 유출을 막았다.
   - 2차 해쉬함수 연산을 통해 보다 안정적인 해쉬 인증 방법을 구현하였다.
   - 기존의 해쉬 함수에 비해 적은 핸드쉐이크 과정으로 외부 유출을 최소화 하였다.
   - 서비스 거부 공격, 도청 공격에는 강한 방어력을 가진다.
   - 토큰키를 기반으로 하는 재생 공격에는 취약하다.

2. 토큰키에 난수를 추가한 인증 방법
   - 키 분배를 통해 새로운 키를 얻어내는 과정을 추가하여 키 유출을 막았으며, 난수를 이용하여 토큰키의 일회성 성질을 부여하였다.
   - 2차 해쉬함수 연산을 통해 보다 안정적인 해쉬 인증 방법을 구현하였으며, 해쉬함수에 들어가는 정수는 매번 달라져 패턴인식 공격에 강한 방어력을 가진다.
   - 기존의 해쉬 함수에 비해 적은 핸드쉐이크 과정으로 외부 유출을 최소화 하였다.
   - 서비스 거부 공격, 도청 공격, 재생 공격에 강한 방어력을 가진다.
   - 랜덤 함수 알고리즘으로 인해 기존의 저가형 태그보다 더 많은 연산능력이 필요한 태그에만 적용이 가능하다.

앞으로 연구되어야 할 방향은 모든 공격에 강한 방어력을 가진 저가형 태그를 위한 인증 방법이 연구되어야 할 것이다. 이는 제한된 환경에 보다 우수한 인증 및 암호화를 통해 강한 보안성을 지니는 보안 기술이 필요한 지금 매우 절실히 필요한 연구 분야이다. 이 기술이 연구되어 발전되어진다면 기존의 인터페이스를 그대로 사용하면서 강한 보안성을 지닌 물류 시스템의 구축이 가능해 진다고 예상한다. 또한 다른 연구 방향은 태그의 제한된 환경요소의 확장이다. 지금 현재 사용중인 저가형 태그보다 더 많은 연산 능력과 저장 능력이 가능한 태그가 개발이 되어 같은 가격에 시장에 나온다면 물류 시스템에서 RFID 기술을 사용하는 시점을 앞당길 수 있으리라 예상한다.

현재 RFID 기술의 사용은 미미하다. 하지만 앞으로 소프트웨어적으로나 하드웨어적으로 더욱 연구되어 나간다면 RFID 기술은 우리 일상생활에서 없어서는 안되는 기술이 될 것이다. 또한 RFID 기술이 발전해 나가면서 보안적 기술도 같이 접목하여 나가야 할 것이다.

참 고 문 헌

[1] 나종민, “트리-워킹 방식 기반의 RFID 시스템 스푸핑 방어 알고리즘.”, 광운대학교 대학원 컴퓨터과학과, 2007

[2] Miyako Ohkudo. Koutatou Suzuki and Shingo Kinoshita, "Cryptographic Approach to Privacy-Friendly Tags.", Submitted 2003.

[3] CRYPTOREC reports, published 2002 in Japen

[4] David J. Wheeler and Robert M. Needham. TEA, a Tiny Encryption Algorithm. Technical report. Computer Laboratory. University of Cambridge. 1995

[5] NTRU. GenuID. http://www.ntru.com/products/genuid.htm.

[6] 주학수, “RFID 시스템의 보안 및 프라이버시 보호를 위한 기술 분석”, 전자정보센터 IT리포트

[7] ISO 18185-4, "Freight containers - Identification and Communication, Electronic seals - part 4: Data Protection", ISO Working Draft, 2004.

[8] 강전일, 박주성, 양대헌, “RFID 시스템에서의 프라이버시 보호 기술.”, 한국정보보호학회 학회지 제 14권 6호, pp.28-36, 2004

[9] Ari Juels and Ravikanth Pappu, "Squealing Euros; Privacy Protection in RFID Enalbed Banknotes", Financial Cryptography 2003 Springer Verlag 2003

[10] Juels, A et al. "The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy, 10th ACM Conference on Computer and Communication Security, 2003

[11] S.A. Weis, S.E. Sarma, R.L. Rivest, and D.W. Engels. "Sercurity and Privacy Aspects of Low-Cost Radio Frequency Identification System.", In Security in Pervasive Comp., Vol. 2802 of LNCS, pp. 201-212. 2004

[12] 박규진, 한경헌, 성종엽, 유도경, 최동유, 한승조. " RFID Tag의 인증을 통한 보안성 향상", 정보보호학회 하계학술대회, 2009.

[13] H.Y. Chien, "Secure Access Control Schemes for RFID System with Anonymity", In Proceedings of 1005 national Workshop on Future Mobile and Ubiquitous Information Technologies. 2006

[14] J. Yang, J. Park, and K. Kim "Security and Privacy on Authentication Protocol for Low-Cost Radio", In The 2005 Symposium on Cryptography and Information Security. 2005

[15] K. Yuksel, "Universal Hashing for Ultra-Low-Power Cryptograaphic Hardware Applications", Master's Thesis, Dept, of Electroniacal Engineering, WPI, 2004

저작물 이용 허락서

학 과	정보통신공학과	학 번	20087497	과정	석사
성 명	한글 한 경 헌 한문 한 경 헌 영문 Kyeoung-heon Han				
주 소	전라남도 여수시 소호동 주은금호 아파트 206동 1402호				
연락처	E-mail : newromeo12 @ naver.com				
논문 제목	한글 :토큰키와 해쉬함수를 이용한 RFID 인증 프로토콜 설계				
	영문 : Design of an RFID Authentication Protocol Using Token Key and Hash Function				

본인이 저작한 위의 저작물에 대하여 다음과 같은 조건 아래 조선대학교가 저작물을 이용할 수 있도록 허락하고 동의합니다.

- 다 음 -

1. 저작물의 DB구축 및 인터넷을 포함한 정보통신망에의 공개를 위한 저작물의 복제, 기억장치에의 저장, 전송 등을 허락함.
2. 위의 목적을 위하여 필요한 범위 내에서의 편집과 형식상의 변경을 허락함. 다만, 저작물의 내용변경은 금지함.
3. 배포·전송된 저작물의 영리적 목적을 위한 복제, 저장, 전송 등은 금지함.
4. 저작물에 대한 이용기간은 5년으로 하고, 기간종료 3개월 이내에 별도의 의사 표시가 없을 경우에는 저작물의 이용기간을 계속 연장함.
5. 해당 저작물의 저작권을 타인에게 양도하거나 출판을 허락을 하였을 경우에는 1개월 이내에 대학에 이를 통보함.
6. 조선대학교는 저작물 이용의 허락 이후 해당 저작물로 인하여 발생하는 타인에 의한 권리침해에 대하여 일체의 법적 책임을 지지 않음.
7. 소속 대학의 협정기관에 저작물의 제공 및 인터넷 등 정보통신망을 이용한 저작물의 전송·출력을 허락함.

동의여부 : 동의(○) 반대( )

2010년 8월 25일

저작자 : 한 경 헌 (인)

조선대학교 총장 귀하