정형검증을 통한 RFID 보안프로토콜 분석 및 구현
(Analysis and Implementation of RFID Security Protocol using Formal Verification)

김 현 석 † 김 주 배 † 한 근 희 †† 최 진 영 †††
(Hyun-Seok Kim) (Ju-Bae Kim) (Keun-Hee Han) (Jin-Young Choi)

요 약 Radio Frequency Identification(RFID: 무선주파수식별) 기술은 유비쿼터스 구조 기술의 중요한 한 부분을 이루고 있다. 태그를 이용한 모든 제품들이 이러한 서비스의 대상이 되고 있지만 불행히도 다방면에 이용되는 이면에는 사용자의 사생활과 사용자 및 판매자 간의 인증문제를 이용한 서비스 공격의 대상이 되고 있다. 현재 이러한 RFID 시스템의 보안 메커니즘들은 이슈화되고 있으며 본 논문에서는 여러가지 메커니즘들 중 사생활 및 인증문제 해결을 위해 정형검증을 통한 보안프로토콜 분석 및 취약성을 수정한 프로토콜을 제안하고자 한다. 또한 제안된 프로토콜의 실현가능성을 위한 구현가능성을 언급하고자 한다.

키워드 : Radio Frequency Identification(RFID: 무선주파수식별), 보안 프로토콜, 프라이버시, 정형검증

Abstract Radio Frequency Identification (RFID) technology is an important part of infrastructures in ubiquitous computing. Although all products using tags is a target of these services, these products also are a target of attacking on user privacy and services using authentication problem between user and merchant, unfortunately. Presently, it is very important about security mechanism of RFID system and in this paper, we analyze the security protocol among many kinds of mechanisms to solve privacy and authentication problem using formal verification and propose a modified novel protocol. In addition, the possibility of practical implementation for proposed protocol will be discussed.

Key words : Radio Frequency Identification (RFID), Security Protocol, Privacy, Formal Verification

1. 서 론

RFID[1,2]를 위한 컴퓨팅 환경은 일반적인 인터넷 환경과는 달리 많은 제약사항을 갖는다. 이러한 제약사항은 Cellular Phone등을 이용한 무선 인터넷보다 더욱 자원 측면적 한계를 갖는다. 즉 유비쿼터스 컴퓨팅을 위한 RFID환경을 구축하기 위해서 모든 상품이나 사람 등 객체에 설치되는 Tag가격은 5센트 이하로 구현되어야 하며 Reader 장비나 Back End 시스템에서 성능 및 자원 측면에서 열악한 Tag장비의 한계를 극복할 수 있도록 설계 및 운영되어야 한다. 이에 보안 기술 적용에 대한 부분도 운영, 환경 측면을 충분히 고려해야 한다. 본 논문에서는 물리적 레벨의 보호 기법이 아닌 암호 기술을 중심으로 한 RFID에서의 보안 프로토콜을 분석하고자 한다.

보안프로토콜을 구현하기 전에 설계단계에서부터 사용자와 개발자에게 안전성과 신뢰성을 제공하기 위한 기술이 요구되고 있다. 그러한 요구를 만족시키기 위해 진행되는 노력 중 대표적으로 정형 기법[3]이라는 연구가 있으며 이는 정형 명세와 정형 검증의 두 가지 방법으로 구분된다.

정형 명세는 개발하고자 하는 시스템의 동작 및 시스템이 만족해야 하는 특성을 정형적인 표현방법을 이용해 모델링하는 방법이고, 정형 검증은 정형적으로 명세

· 이 논문은 제34회 추계학술대회에서 '정형검증을 통한 RFID 보안프로토콜 분석 및 구현'의 제목으로 발표된 논문을 확장한 것임

† 학생회원 : 고려대학교 컴퓨터학과
hskim@formal.korea.ac.kr
jbkim@formal.korea.ac.kr
†† 정 회 원 : 행정자치부 정보보호정책과
keunhee@mogaha.go.kr
††† 종신회원 : 고려대학교 컴퓨터학과 교수
choi@formal.korea.ac.kr

논문접수 : 2007년 12월 4일
심사완료 : 2008년 4월 29일

정보과학회논문지: 시스템 및 이론 제35권 제7호(2008.8)

된 시스템을 대상으로 그 시스템이 정확한지 혹은 그 시스템의 요구사항으로 주어지는 특성을 만족하는지를 논리적으로 증명하는 방법이다.

그 중 정형 검증은 정리증명과 모델체킹 기법으로 구분되며, 전자는 보안 로직을 이용하여 특정한 논리식으로 시스템을 명세하고 정확한 논리 증명단계로써 정확성을 증명하는 방식이고, 후자는 프로토콜의 인증과정을 유한상태기계의 형식으로 모델링하고 그 모델이 만족해야 하는 요구사항이나 특성이 만족되는지를 검증도구를 이용해 자동으로 증명하는 방식으로 ESTEREL[4], Murphi[5], Failure Divergence Refinement (FDR)[6]과 같은 방법이 있다.

본 논문에서는 정형검증 도구 중(FDR)이라는 모델체킹 도구를 이용, RFID 보안프로토콜인 해쉬기반 프로토콜들[7]의 취약성을 분석하여 보안 프로토콜의 안전성을 향상시키고자 한다. 기존의 연구[1,7-13]들의 경우 수학적인 수식이나 직관적인 방법으로 분석하였으나 본 논문에서는 정형검증을 통해 좀더 복잡한 프로토콜의 경우에까지 자동화된 도구를 이용하여 분석할 수 있음을 보이고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 해쉬 기반 보안 프로토콜들에 대해서 설명하고 3장에서는 프로토콜을 명세하고 검증하기 위한 A Compiler for the Analysis of Security Protocols(Casper)[14] 및 FDR[6] 도구에 대해 소개하며, 4장에서는 Communicating Sequential Process(CSP)[15], Casper와 FDR을 이용한 해쉬-언락킹 보안프로토콜의 분석 및 결과에 대해 살펴보고, 5장에서는 이러한 보안프로토콜의 취약성을 해결한 새로운 프로토콜을 제안하며 저가의 태그 구현을 위한 타당성을 검증한다. 마지막 6장에서는 결론 및 향후 연구 방향을 제시한다.

2. RFID 시스템, 보안요구사항 및 관련연구

2.1 RFID 시스템 개요

RFID 시스템(그림 1)은 다음 세가지 구성요소로 이루어져 있다.

- RFID 태그(트랜스폰더) : 객체 식별 정보전송
- RFID 리더(트랜시버) : Tag 정보 수집
- Back-end DB(데이타베이스) : 리더에 의해 수집되는 Tag 확인 정보제공

이러한 환경하에서 RFID의 보안 요구사항은 다음과 같이 정리할 수 있다.

2.2 RFID 시스템의 보안 요구사항

위 환경하에서 RFID 시스템의 태그와 리더 등 구성환경에 대해 다음과 같은 사항을 고려해 보안 요구사항을 설정할 수 있다. 특히 여러 가지 보안 요구사항 중 보안 프로토콜과 같은 암호기술적 해결방안의 적용을 위한 요구사항으로서 아래와 같이 4가지를 제시하였다. 아래의 요구사항들은 [1,7,9,10,12]에서 공통적으로 RFID 시스템의 보안 문제를 해결하기 위한 요건으로 제시하고 있다.

A. 인증이 되지 않은 리더에게 정보유출이 되지 않아야 하며, 태그와 그 소유자 사이에 긴 시간 동안의 추적(long-term tracking)이 불가능해야 한다.

B. 태그의 내용은 근제한기법(access control)에 의해 질의채널(interrogation channel)이 안전 하지 않다고 예상되면 암호화되어야 한다.

C. 태그와 리더 사이에는 상호인증(mutual authentication)이 제공되어야 한다.

D. 태그와 리더 모두 재생공격(replay attack) 및 공격자 중간공격(man-in-the-middle attack)에 저항력이 있어야 한다.

2.3 RFID 시스템 보안문제 해결에 관한 관련연구

RFID 시스템에서 사용자 프라이버시의 보호를 위한 많은 연구[1,7-13,16-20]들이 진행되어 오고 있다. 현재까지 진행되어 왔던 연구결과 중, Kill 명령어의 접근법

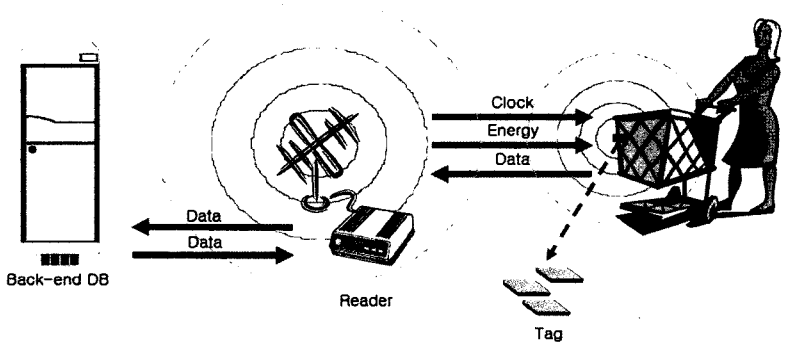

그림 1 RFID 컴포넌트 상호작용도

[16], Blocker 태그 기법[17], 해쉬락(Hash-Lock)기법[1], 랜더마이즈 해쉬락 기법[3], XOR 기반 원타임 패드 기법[8], 외부재암호화 기법[18], 해쉬 체인(Hash-Chain) 기반 기법[9] 등이 있다.

RFID 네트워크 시스템의 보안을 위한 공개키 알고리즘의 하드웨어적인 구현은 2004년 Rabin, NTRU, ECC등의 공개키 알고리즘에 대한 구현 결과 제시에 의해 NTRU의 경우 20μW의 저전력에 3000개의 게이트만 필요하며, 경량화된 센서 노드에 탑재 가능한 것으로 알려져 있다[19]. 그러나 현재는 기존의 알고리즘을 개선하여 사용하고 있으며 향후에는 새로운 알고리즘 개발이 요구된다. 그룹키 관리를 위해 대칭키 방식 적용시 리더와 태그간의 키를 공유해야 하며, 각 태그마다의 유일한 키를 관리하는 등의 많은 계산량 때문에 사용하기 어려우며, 키의 유출에 의한 태그 무력화, 장기간의 사용에 따른 노출 가능성 등의 문제가 있다. 또한, 태그에 암호키를 탑재하는 방식은 에칭 (etching), 탐침 등의 물리적공격에 취약하며 암호키의 노출 가능성이 있다. 버클리 대학의 SmartDust 프로 젝트에서 채택한 센서 네트워크의 보안 프로토콜 인 Security Protocol for Sensor Network(SPINS)[20]은 μTESLA와 SNEP로 구성되어 있으며 메시지 인증, 무결성, 기밀성, 적시성 등의 서비스를 제공하고 있다. 랜덤키 사전 분배방식은 키 DB을 선택하고 무작위로 키를 선택하여 센서 노드에 할당하며, 두 개의 노드는 자신의 키 DB를 탐색하여 상대방이 같은 공통키를 소유하고 있으면 이 키를 세션키로 사용하는 방식이다. 본 논문에서는 물리적 레벨의 보호 기법이 아닌 암호 기술을 중심으로 한 RFID에서의 보안 프로토콜을 분석 한다.

3. Casper와 FDR 도구

3.1 Communicating Sequential Process(CSP)

CSP[15]는 프로세스 알제브라 언어로서, 병렬성을 갖는 통신프로토콜의 동작을 효율적으로 명세하기 위한 언어이다. 최초 일반 통신 프로토콜 및 제어 시스템의 명세를 위해 사용되어졌으나, 점차 보안 프로토콜의 명세를 위한 영역으로 확대되어 가고 있다. CSP에서 제공하는 pure synchronization(|||)과 Interleaving par-all-elism(||) 개념을 사용하여 분산 시스템 환경하에서 동작하는 클라이언트 서버와 공격자 모델을 정형 적으로 표현할 수 있는 장점을 갖고 있다. 예를 들어, 분산 시스템 환경하에서 동작하는 보안 시스템은 다음과 같이 간략히 표현할 수 있다.

SYSTEM = CLIENT1 ||| CLIENT2||| SERVER || INTRUDER

3.2 A Compiler for the Analysis of Security Protocols (Casper)

CSP[15] 언어를 이용하여 보안프로토콜 행위를 명세하고 FDR[6] 정형검증 도구를 이용하여 보안속성을 검증하는 연구가 진행되었다. 하지만, CSP 언어를 이용한 정형명세과정은 정형적 설계 방법에 익숙치 않은 보안 프로토콜 설계자에게는 매우 복잡한 명세언어라는 단점을 갖고 있었다. 이에 따라, 보안 프로토콜의 행위를 간략히 명세할 수 있도록 Casper[14]도구가 개발되었다. Casper도구로 보안프로토콜의 행위와 검증속성을 명세하게 되며, 자동변환기능을 이용해 CSP 명세코드를 생성하고, FDR 정형검증도구에 입력하여 보안프로토콜을 검증하게 된다.

3.3 Failure Divergence Refinement (FDR)

FDR[6]도구는 CSP 명세언어를 입력으로 받아들이는 모델체킹 도구로서, CSP 명세언어로 기술된 보안프로토콜 모델이 보안성 및 인증속성과 같은 보안속성들을 만족하는지 검증하게 되며, 만일 만족하지 않을 경우에는 CSP 이벤트로 기술된 반례(counterexample)을 보여주어 보안상 취약점 분석을 용이하게 한다.

4. Casper/FDR을 이용한 해쉬-언락킹 프로토콜분석 및 수정된 프로토콜 제안

이 장에서는 해쉬-언락킹 프로토콜과 랜더마이즈 해쉬-언락킹 프로토콜의 취약성을 정형검증을 통해 분석하고자 한다.

4.1 해쉬 언락킹 프로토콜 분석 및 결과

태그는 해쉬 메커니즘을 처리할 수 있는 H/W 기반의 암호화 모듈로서 보안요구사항을 처리할 수 있다. Tag에는 metaID 정보만을 보관할 수 있는 저장 공간을 보유하고 있어야 하며 Lock과 Unlock 처리기능만 동작하면 된다. Unlock이 된 Tag만이 Reader장비와 운영이 가능하다.

해쉬락 스킴은 해쉬-락과 해쉬-언락킹 프로토콜로 이루어져 있으며 태그가 부착된 모든 사물에 metaID 값을 이용하여 간단하게 태그의 정보를 리더가 수집하게 하는데 목적이 있다.

표 1 해쉬-락 스킴의 표현법

T	RF 태그의 식별자
R	RF 리더의 식별자
DB	백엔드 데이타 베이스의 식별자
Xkey	통신참여자 X의 세션키
metaID	키를 해쉬값으로 처리한 값
ID	태그의 정보값
Xn	통신참여자 X에 의한 난수값
H	해쉬함수

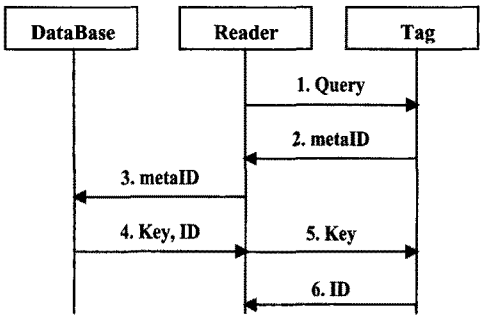

그림 2 해쉬-언락킹 프로토콜

해쉬-락킹 프로토콜의 경우, 리더에 의해 선택된 키 값을 metaID 형태로 태그에 기록하고 이때 키값을 가진 정상적인 리더만이 언락이 가능하도록 잠긴 상태로 들어가게 되며, 해쉬-언락킹 프로토콜은 리더에 의해 락킹에서 보유하고 있던 키를 이용하여 태그의 잠긴 상태를 해제함으로써 태그 정보를 받게 된다.

• 해쉬-락의 locking 프로토콜
  ① 리더는 랜덤한 키 key를 선택하고, metaID값으로 hash(key)를 계산한다.
  ② 리더는 metaID를 태그에 기록한다.
  ③ 태그는 잠긴 상태(locked state)에 들어간다.
  ④ 리더는 metaID, key를 저장한다.
• 해쉬-락의 unlocking 프로토콜(그림 2. 참조)
  ① 리더는 태그에게 태그의 metaID를 질의한다.
  ② 리더는 데이타베이스에서 metaID와 key를 조사한다.
  ③ 리더는 태그에게 key를 전송한다.
  ④ 만일 hash(key)와 metaID가 일치하면, 태그는 잠긴 상태에서 빠져 나온다(unlock).

그림 3은 해쉬-언락킹 프로토콜을 Casper도구를 이용하여 모델링한 것이며, 8가지 항목 중 자유변수 영역과 프로토콜 기술영역, 침입자 영역에 대한 표현이다.

먼저 자유변수 영역에서, R은 리더, T는 태그로서 각각 Agent로 나타내고, DB는 백엔드 서버의 역할을 한다. key는 Session 키, Id는 Tag의 정보를 표현하고, InverseKeys는 Session 키에 대한 암호화 및 복호화를 표현하며, H는 해쉬함수를 뜻한다. 다음으로 프로토콜 기술 영역은 해쉬-언락킹 프로토콜을 명세한 부분으로 여기서 % 표현은 메세지 1에서 T가 H(key) 값을 metaID로서 수신자인 R에게 복호화의 목적이 아닌 단지 다른 수신자 DB에게 전달하는 목적을 지니고 있다. 따라서 메세지 2에서 이 메세지가 DB에게 전달되어 복호화된다. 마지막으로 침입자 영역에 대한 정보가 제시되어 있다.

```
#Free variables
R, T : Agent
DB   : Server
key  : SessionKey
Id   : Text
H    : HashFunction

InverseKeys = (key, key)

#Protocol description
0.      -> T  : R
1.  T   -> R  : (H(key)) % metaID
2.  R   -> DB : metaID % (H(key))
3.  DB -> R   : key, Id
4.  R   -> T  : key
5.  T   -> R  : Id

#Intruder Information
Intruder = Mallory
IntruderKnowledge = {Tag, Reader, DataBase}
```

그림 3 Casper를 이용한 해쉬-언락킹 프로토콜 명세

4.1.1 해쉬-언락킹 프로토콜 검증 결과

해쉬-언락킹 프로토콜에서는 metaID의 값을 중간자 공격 및 재생 공격에 이용함에 따라 태그 정보의 노출 및 추적을 가능하게 하고 리더기와 태그의 인증에 실패하는 결과를 초래하였다. 이를 Casper script로 명세하기 위해 해쉬-언락킹 프로토콜의 두 개체간 사용된 정보에 대한 비밀성과 개체간 상호 ID에 대한 인증을 만족해야 하며 이는 다음과 같이 표현할 수 있다.

```
Secret(R, key, [T])
Secret(R, Id, [T])
Agreement(T, R, [Id, key])
```

첫번째 표현은 "R은 key 정보를 오직 T와만 알고 있다"라고 풀이할 수 있고 두번째 표현은 "R은 Id 정보를 오직 T와만 알고 있다"로 풀이할 수 있다. 세번째 표현은 "T는 Id, key 정보를 통해 R로부터 자신의 개체를 인증받는다"라고 풀이할 수 있다.

모델 체커를 이용해 비밀성과 개체인증 속성의 만족 여부를 확인한 결과, 첫번째 표현에서 R이 전달하는 key에 대해 T와의 비밀성 속성을 만족하지 않았고 이에 따라 결국 두 개체간의 데이타가 누설되었다. 또한 Id의 정보도 비밀성 속성을 만족하지 않았으며, 마지막 속성인 개체 인증에서도 Id, key의 정보를 이용해 두 개체간의 인증에 실패했다.

위 비밀성 요구사항의 반례에 대해 FDR의 interpret 기능을 통해 분석한 결과는 그림 4와 같다.

2.2에서 제시된 보안 요구사항에 대해 위 분석 결과를

```
0.              ->   Tag       : Reader
1.   Tag       ->   I_Reader   : H(Key)
2.   I_Tag     ->   DataBase   : H(Key)
1.   I_Tag     ->   Reader     : H(Key)
3. DataBase    ->   I_Tag      : Key, ID
2.   Reader    ->   I_DataBase : H(Key)
Reader believes ID is a secret shared with Tag
The intruder knows ID
```

그림 4 FDR을 이용한 반례의 분석결과

토대로 다음과 같이 정리할 수 있다.

A. 인증을 통한 long-term tracking 방지 : R입장에서 T로부터 정상적인 데이타를 전송받았다고 간주했으나, I_Tag나 I_DataBase와 같이 악의적인 개입이 가능했으며, metaID 정보는 tracking에 사용될 수 있다.

B. 암호화를 통한 채널 안전성 확보 : 위 반례에서 알 수 있듯이, 세번째 메시지의 Key와 ID는 암호화되지 않은 채 전송되어 노출되었다.

C. 상호 인증 : R입장에서 T로부터 정상적인 데이타를 전송받았다고 간주했으나, I_Tag나 I_DataBase와 같은 악의적인 개입이 가능했다.

D. Tag 정보의 유출방지를 통한 재생공격 및 중간자 공격 방지 : T가 R에게 정상적인 데이타 전송을 했다고 간주했으나 I_Reader에 의해 H(key)정보가 노출되었다. 결과적으로 T의 metaID 정보는 중간자 공격에 이용되었다.

4.2 랜더마이즈 해쉬-락 스킴 분석 및 결과

랜더마이즈 해쉬-락 스킴 또한 해쉬-언락 스킴의 확장 프로토콜로서, 태그가 부착된 모든 사물에서 수집된 값을 리더가 데이타베이스로부터 확인한 값과 비교하여 일치하는 값을 태그에게 보내어 일치할 경우 그 값을 취하게 된다.

해쉬-언락킹 프로토콜은 태그가 기존의 해쉬락 스킴의 락킹 프로토콜과 같이 잠긴 상태에서 ID값과 난수값을 조합한 값을 리더에게 보내고, 리더는 데이타베이스로부터 모든 ID값들을 요청 및 이 값들을 보내준 난수값과 함께 각각 해쉬하여 그 값들을 비교한 후 일치한 값을 태그에게 재전송하여 확인하게 된다. 즉 해쉬-락기법에서는, 직접 리더가 데이타베이스와 통신한다는 점에서 사용자 추적을 방지하기 위한 방식이다. 이 기법에서는 태그에 일방향 해시 함수와 난수발생기(PRNG)가 구축되어있어야 한다.

• 랜더마이즈 해쉬-락의 언락킹(unlocking) 프로토콜(그림 5 참조)

① 리더는 태그에게 질의를 보낸다.

② 태그는 랜덤한 난수값을 생성하고, hash(ID || R) 값을 계산한다.

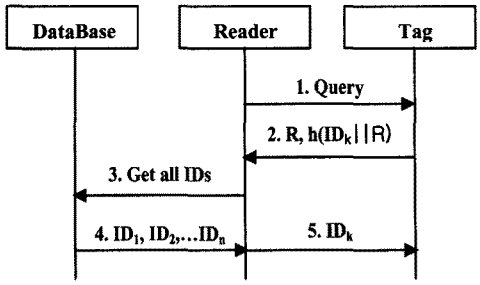

그림 5 랜더마이즈 해쉬-언락킹 프로토콜

③ 태그는 리더에게 (R, hash(IDk || R))을 전송한다.

④ 리더는 모든 알려진 ID값들에 대해 hash(IDk || R)을 계산한다.

⑤ 만약 hash(IDk || R)의 만족하는 IDk를 찾는다면, 리더는 태그에게 IDk를 전송한다.

⑥ 만약 IDk와 ID가 일치한다면, T는 잠긴 상태에서 빠져나온다.

이 방식은 초당 100~200개의 태그를 읽어야 하는 많은 개수의 태그를 소유한 환경에서는 비현실적이다. 그러나 상대적으로 적은 수의 태그 사용자를 갖는 환경에서 가능한 방식이다.

그림 6은 랜더마이즈 해쉬-락의 언락킹 프로토콜을 Casper표현방식으로 모델링한 것으로 8가지 항목 중 자유변수 영역과 프로토콜 기술영역, 침입자 영역에 대한 표현이다.

먼저 자유변수 영역에서, R은 리더, T는 태그로서 각각 Agent로 나타내고, DB는 백엔드 서버의 역할을 한다. n은 태그가 생성한 임의의 난수이며 Id1부터 Id7까

```
#Free variables
R, T : Agent
DB: Server
n : Nonce
Id1, Id2, Id3, Id4, Id5, Id6, Id7 : Text
InverseKeys = (n, n)
H : HashFunction

#Protocol description

0.       -> T : R
1a. T  -> R : n, H(Id5, n)
1b. DB -> R : Id1, Id2, Id3, Id4, Id5, Id6, Id7
2.    R  -> T : Id5

#Intruder Information
Intruder = Mallory
IntruderKnowledge = {Tag, Reader, DataBase}
```

그림 6 Casper를 이용한 랜더마이즈 해쉬-언락킹 프로토콜 명세

지는 리더가 데이타베이스에게 요청한 태그들에 대한 정보를 의미한다. InverseKeys는 Session 키에 대한 암호화 복호화를 표현하며, H는 해쉬함수를 뜻한다. 다음으로 프로토콜 기술 영역은 해쉬-언락킹 프로토콜을 명세한 부분으로, 메시지 1a는 R이 T 로부터 난수값과 해당 T의 정보가 함께 해쉬된 값 및 난수값을 받은 후 메시지 1b에서 R이 DB에게 Id값들을 요청하여 수신한 후 메시지 2에서 T에게 정확한 Id 값을 송신하여 언락 상태를 만들게 된다. 마지막으로 침입자 영역에 대한 정보가 제시되어 있다.

4.2.1 랜더마이즈 해쉬-언락킹 프로토콜 검증 결과

앞서 살펴본 4.1.1의 해쉬-언락킹 프로토콜과 같이 Casper와 FDR을 통해 검증한 결과, 다음과 같은 결과를 얻을 수 있다.

A. 인증 통한 long-term tracking 방지 : T가 R의 식별자 정보를 해쉬하여 R에게 보냄으로써 T는 R을 신뢰하고 IDk를 전송하고 있으나 IDk가 고성능 리더기에 의해 그 값을 식별할 경우 tracking 가능하다.

B. 암호화를 통한 채널 안전성 확보 : DB로부터 R에게 전송되는 ID들은 암호화한 값들이 아니며, 이들의 값은 Tag의 분석에 이용 가능하다.

C. 상호 인증 : R입장에서 T로부터 정상적인 데이타를 전송받았다고 간주했으나, I_Tag나 I_Data-Base와 같은 악의적인 개입이 가능하다.

D. Tag 정보의 유출방지를 통한 재생공격 및 중간자 공격 방지 : T가 R에게 정상적인 데이타 전송을 했다고 간주했으나, I_Reader에 의해 H(key)정보가 노출되었다. 결과적으로 T의 metaID 정보는 중간자 공격에 이용가능하고, R 입장에서 허위 IDk에 의해 재생 공격이 가능하다.

5. 해쉬기반 프로토콜의 취약성을 수정한 제안프로토콜

5.1 제안프로토콜 및 보안분석

이러한 해쉬-언락킹 프로토콜의 문제점으로 분석되었던 부분은 태그 내에 저장된 정보를 해쉬 기반기법으로 태깅할 수 있게 함으로써 발생되었으며, 이는 태그의 정보를 인증된 리더기에 의해 데이타베이스에 접근함으로써 태그정보를 가져갈 수 있도록 하여 중간자 공격과 재생공격을 방지할 수 있다. 즉 제안프로토콜은 다음(그림 7)과 같은 절차로 인증이 이루어지며, 앞서 언급된 취약성은 인증과정에서 도입된 3가지 기술에 의해 극복할 수 있다.

A. 첫번째로 태그의 난수와 데이타 베이스의 난수가 데이타 프라이버시와 태그 응답시의 악의적인 리더로부터의 재생공격을 막기 위해 도입되며, 장소 추적을 막기 위해서 적어도 하나는 필요하다.

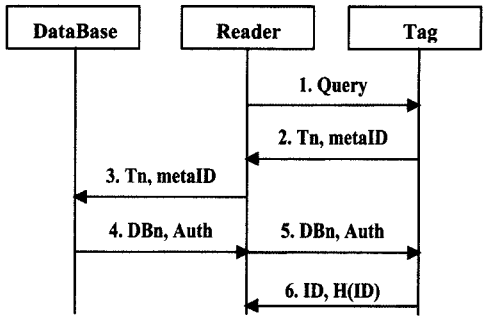

그림 7 해쉬기반 제안프로토콜

B. 통신참여자들간의 기밀성을 보장하기 위해 배타적 합(Exclusive-or) 기술이 도입되었다.

C. 리더와 태그, 데이타 베이스와 리더간의 안전한 채널을 구축하기 위해, 또 다른 metaID와 같은 타입의 데이타 베이스의 난수값과 리더의 원래 키값으로 이루어진 Auth라는 값이 사용되었다.

위의 3가지 기술을 통해 그림 7의 절차는 다음과 같이 설명될 수 있다.

메시지 2. 재생공격 방지를 위한 태그의 난수값(Tn), 리더의 키값과 태그의 난수값을 배타적 합으로 처리하고 이 값의 재전송을 위해서 해쉬처리한 metaID값 (= H(key)(+)Tn)을 리더에게 전송한다.

메시지 3. 태그로부터 전송 받은 값이 메타값으로 처리되어 있으므로 이를 데이타베이스로 재전송만 가능하다. 즉, 데이타베이스가 리더를 인증하기 위한 절차이다.

메시지 4. 데이타베이스는 리더로부터 전송받은 태그의난수값과 metaID내의 리더의 키값 및 태그의 난수값의 일치 여부를 비교하고 일치할 경우 리더가 태그를 인증하도록 데이타베이스의 난수값(DBn)과 리더의 키값 및 데이타베이스의 난수값으로 조합되어, 해쉬처리된 Auth값(= H(key(+)DBn))을 리더에게 전송한다.

메시지 5. 메시지 2, 3의 목적과 동일하게 재전송한다.

메시지 6. 메시지 5를 수신한 후 이를 인증할 경우, 태그가 리더에게 전송하고자 하는 값(Id)을 해쉬 처리하여 전송한다.

또한 위의 프로토콜을 Casper로 명세하면 다음 그림 8과 같다.

먼저 자유변수 영역에서, 추가적인 부분은 태그와 데이타베이스 난수(tn, dbn)이며, 프로토콜 기술 영역은 제안 프로토콜을 명세한 부분으로 여기서 % 표현은 해쉬 언락킹 프로토콜과 같이 복호화의 목적이 아닌 단지 다른 수신자에게 전달하는 목적을 지니고 있다. 따라서

```
#Free variables
R, T : Agent
key: Nonce
tn, dbn : Nonce
Id  : Text
DB: Server
InverseKeys = (key, key), (tn, tn), (dbn, dbn)
H : HashFunction

#Protocol description
0.      -> T  : R
1 . T   -> R  : tn, H(key (+) tn)%metaID
2.  R   -> DB : tn, metaID%H(key (+) tn)
3.  DB -> R   : dbn, H(key (+) dbn)%Auth
4.  R   -> T  : dbn, Auth%H(key (+) dbn)
5.  T   -> R  : Id, H(Id)

#Intruder Information
Intruder = Mallory
IntruderKnowledge = {Tag, Reader, MKey, MN}
```

그림 8 Casper를 이용한 제안 프로토콜 명세

메시지1, 3에서 이 메시지가, 각각 메시지 2, 4 에서 수신자에게 전달되어 복호화 된다. 마지막으로 침입자 영역에 대한 정보가 제시되어 있다.

명세된 위 프로토콜은 FDR을 통해 검증 결과, 2.2의4가지 속성을 모두 만족하였다.

5.2 제안 프로토콜 구현

현재 수동형태그에 대한 이슈는 단가가 낮은 보안 태그의 구현에 초점을 두고 있다. 따라서 공개키 기반 암호화 기술이나 대칭키 기반 암호화 기술이 적용되기도 하지만, 최소 5센트 미만이라는 요구조건을 충족시키기에는 적합한 비용이 아니다[9]. 따라서 본 논문에서는 태그의 구현에 필요한 최소 게이트를 통한 보안 목표 달성을 하고자 한다. 제안된 프로토콜이 실질적으로 구현될 수 있는지를 검증하기 위해 ASIC 구현을 통해 총 게이트 수를 실험하였다.

아래의 표 2와 3은 이에 대한 결과를 나타내는 것으로서 표 2는 제안프로토콜을 구현하기 위한 총 게이트 개수를 각 비트별로 실험한 내용이다. 4번째 행은 128비트 기준의 클락일 경우의 각 비트에 대한 클락 수를 나타낸 것이다.

실험을 통해 태그의 시퀀스 데이타 부분의 32비트일 경우, 128비트 기준 4 클락으로 5,208 게이트만으로 구현이 가능함을 알 수 있다.

표 2 제안 프로토콜을 이용한 총 게이트 수 구현

데이타 비트 길이	32	64	128
XOR모듈 게이트 수	4,280	8,560	17,120
레지스터 게이트 수	928	1,875	3,713
총게이트 수	5,208	10,335	20,833
128비트 클락	4	2	1

표 3 보안 모듈에 필요한 최소 게이트 수

종류	형태	게이트 수
해쉬 함수	SHA-256	10,800
	SHA-1	8,120
	MD5	8,400
	MD4	7,350
대칭형 암호화	AES	25,000
수정된 해쉬기반 배타적합 기법		5,208

위의 표 3에서는 실험을 통해 도출된 해쉬 및 대칭형 암호화 기법들의 게이트 수에 비해 제안 프로토콜의 해쉬 기반 배타적 합 기법이 경량화된 게이트 수로 구현될 수 있음을 보이고 있다.

즉 기존의 암호화 기술들이 최소 7,350개(MD4 기준)의 게이트 수가 소요되나, 5,208개의 게이트로 구현될 수 있음을 알 수 있다.

6. 결론 및 향후 연구방향

RFID 기반의 유비쿼터스 컴퓨팅 환경은 네트워크를 기반으로한 장치 간의 연결을 기본으로 하고 있다. 특히 무선중심의 근거리 통신기술이 발달함에 따라 유비쿼터스 컴퓨팅 환경을 이루는 무수히 많은 개체들은 유기적으로 연결되어, 서로 데이타를 주고받고, 이를 통해 서비스를 제공하게 된다. 이러한 개체간의 연결은 보안 취약점을 발생시키며, 이에 따른 정보보호 관점의 보안 요구사항이 도출된다. 본 논문에서는 암호기술적 관점에서 해쉬 기반의RFID보안 프로토콜들 중, 대표적으로 해쉬 언락 및 랜더마이즈 해쉬 언락-스킴을 보안 프로토콜 분석용 자동화 도구를 이용하여 취약성을 분석하고 수정된 보안 프로토콜을 제시하였다. 또한 최근의 RFID를 위해 제안되고 있는 보안프로토콜들은 취약한 보안성을 보완하기 위하여 대칭 및 비대칭의 중량의 암호화 기법을 사용함에 따라 구현시 리더에 계산적 능력이 더욱 요구되는 경향이 높아지고 있으며 이에 따라 태그의 가격 또한 현실적이지 못할 가능성이 높아지고 있다. 본 논문에서는 구현시 태그 가격에 실질적인 영향요소인 게이트 수에 초점을 맞추어 구현가능성을 검증하였다.

참 고 문 헌

[1] Sarma, S., Weis, S., and Engels, D., "RFID Systems and Security and Privacy Implications," In Workshop on Cryptographic Hardware and Embedded Systems (CHES) 2002, LNCS No. 2523, pp. 454-469, 2003.

[2] EPCGLOBAL INC.: http://www.epcglobalinc.org.
[3] Clarke, E.M. and Wing, J.M., "Formal Methods: State of the Art and Future Directions," ACM Computing Surveys (CSUR), Volume 28, Issue 4, pp. 626-643, 1996.
[4] Boussinot, F. and de Simone, R., "The ESTEREL language," In Proc.of the IEEE, Volume 79, Issue 9, pp. 1293-1304, 1991.
[5] Ulrich S. and David L. Dill, "Parallelizing the Murϕ Verifier," Formal Methods in System Design, Volume 18, No.2, pp. 117-129, 2001.
[6] Formal Systems Ltd. FDR2 User Manual, Aug. 1999.
[7] Weis, S., Sarma, S., Rivest, R. and Engels, D., "Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems," In 1st Intern. Conference on Security in Pervasive Computing (SPC), pp. 201-212, 2003.
[8] Juels, A., "Privacy and Authentication in Low-cost RFID tags," In submission, Available at http://www.rsasecurity.com/rsalabs/staff/bios/ajuels/
[9] Weis, S., "Security and privacy in radiofrequency identification devices," Massachusetts Institute of Technology (MIT). Massachusetts, USA, 2003.
[10] Chien, H.Y. and Chen, C.H., "Mutual Authentication Protocol for RFID Conforming to EPC Class 1 Generation 2 Standards," Computers Standards & Interfaces, Volume 29, Issue 2, pp. 254-259, 2007.
[11] Kosta, M. E., Hansen, M., and Gasson, M., "An Analysis of Security and Privacy Issues Relating to RFID Enabled ePassports," IFIP SEC, pp. 467-472, 2007.
[12] Peris-Lopez, P., Hernandez-Castro, J. C., Estevez-Tapiador, J. M., and Ribagorda, A., "EMAP: An Efficient Mutual Authentication Protocol for Low-cost RFID Tags," OTM Federated Conferences and Workshop: IS Workshop, pp. 352-361, 2006.
[13] Defend, B. and Juels, A., "Cryptanalysis of Two Lightweight RFID Authentication Schemes," International Workshop on Pervasive Computing and Communication Security PerSec (2007).
[14] Lowe, G., "Casper: A compiler for the analysis of security protocols," the 1997 IEEE Computer Security Foundations Workshop X, IEEE Computer Society, Silver Spring, MD, pp. 18-30, 1997.
[15] Hoare, C.A.R., Communicating Sequential Processes, Prentice-Hall, 1985.
[16] Juels, A., "Minimalist Cryptography for RFID Tags," In Proc. of Security in Comm. Networks (SCN), LNCS, pp. 149-164, 2004.
[17] Juels, A., Rivest, R.L., and Szydlo, M., "The Blocker Tag: Selective Blocking of RFID tags for Consumer Privacy," In Proc. of the 10th ACM conference on Computer and communications security, pp. 103-111, 2003.
[18] Golle, P., Jakobsson, M., Juels, A., and Syverson, P., "Universal Reencryption for Mixnets," RSA Conference Cryptographers Track (CT-RSA 2004), LNCS 2964, pp. 163-178, 2004.
[19] Gaubatz, G., Kaps, J., and Sunar, B., "Public Keys Cryptography in Sensor Networks Revisited," the 1st European Workshop on Security in Ad-Hoc and Sensor Networks(ESAS 2004), pp. 2-18, 2005.
[20] Perrig, A., et al., "SPINS : Security Protocols for Sensor Networks," Mobile Computing and Networking 2001.

김 현 석
육군사관학교 경제·경영학과 졸업. 고려대학교 컴퓨터학과 석사 졸업. 2006년~현재 고려대학교 컴퓨터학과 박사과정 2007년~현재 육군사관학교 전자·정보학과 전임강사. 주관심분야는 정형기법, 네트워크 보안, 전자상거래 보안, RFID 보안프로토콜 설계, 스마트 카드 보안 설계

김 주 배
공군사관학교 외국어학과 졸업. 현재 고려대학교 컴퓨터학과 석사과정. 주관심분야는 정형기법, RFID 프로토콜, 스마트 카드 보안, 보안 프로토콜 검증

한 근 희
서울산업대학교 컴퓨터학과 학사. 한양대학교공과대학원 컴퓨터학과 석사졸업. 고려대학교 컴퓨터학과 박사. 2006년~현재 행정안전부 정보보호정책과 근무. 2004년~현재 건국대학교 정보통신대학원 겸임교수 주관심분야는 통합보안관리와 인터넷 보안, 모바일 보안, 차세대 인터넷

최 진 영
서울대학교 컴퓨터 공학과 졸업. Dept. of Mathematics and Computer Science, Drexel Univ. 석사 졸업. Dept. of Computer and Information Science, University of Pennsylvania 박사 졸업 1996년~현재 고려대학교 컴퓨터·통신공학부 교수. 주관심분야는 정형기법, 임베디드 실시간 시스템, 프로그래밍 언어, 프로세스 대수, 소프트웨어 공학