제25회 한국정보처리학회 춘계학술발표대회 논문집 제13권 제1호 (2006. 5)

모델 체킹을 이용한 RFID 인증 프로토콜 안전성 분석

오정현, 최진영
고려대학교 컴퓨터학과

e-mail : {jhoh,choi}@formal.korea.ac.kr

Safety Analysis of the RFID Authentication Protocol using Model Checking

Jung-Hyun Oh, Jin-Young Choi
Dept. of Computer Science and Engineering , Korea University

요 약

RFID 시스템의 보안적 취약점을 보안하기 위해 많은 프로토콜들이 제안되었지만, 아직까지 완벽하게 보안성과 경제성을 모두 만족시키지 못하였다. 본 논문에서는 RFID 시스템의 보안적 취약점을 보안하기 위해 제안된 보안 프로토콜의 안전성 만족 여부를 정형기법을 이용하여 검증을 실시하고, 발전방향에 대해 제시하였다.

1. 서론

RFID 시스템은 Radio Frequency 를 사용하여 물리적인 접촉이 필요 없이 물품의 정보를 자동적으로 읽어들이는 유비쿼터스 기술 중에 하나이다.

하지만 시스템을 구성하는 요소간의 통신 채널이 무선환경이라는 특수성 때문에 도청을 통한 위조 또는 추적 등 보안적인 취약점을 내포하고 있다. 더하여 RFID 태그의 가격을 5 센트 이하로 낮추려는 움직임에 의해 태그를 구성하는 하드웨어의 제약을 가져다주게 되어, 기존의 유선 통신 환경에서 사용되던 암호화 방법들을 그대로 사용할 수 없게 되었고, 이 때문에 보안성과 경제성을 모두 만족시키는 인증 프로토콜의 개발이 필요하게 되었다.

본 논문에서는 수동형(Passive)태그를 기반하는 하는 RFID 시스템의 보안적 취약점을 분석하고, 기존에 제시된 프로토콜들을 정형기법을 사용하여 그 보안성 만족 여부를 검증하였다.

정형기법[1][2]은 수학적 논리나 이론을 바탕으로 하여 하드웨어나 소프트웨어 시스템이 요구사항에 맞게 설계되었고, 안전하게 개발되었는지 확인 및 검증하는 방법론으로, 일반적으로 시스템의 동작 및 특성을 정형적 방법에 의해 명세하는 정형명세와 정형적으로 명세된 시스템이 주어진 요구사항을 만족하는지 정형적으로 검증하는 방법인 정형검증으로 나뉜다.

정형검증은 다시 논리증명과 정형검증으로 나뉘는데, 논리증명이란 수학적 논리를 통해 시스템이 요구사항을 만족하는지 논리적으로 유도해 내는 방법이고, 모델체킹은 정형적으로 명세된 시스템을 모델체킹 도구를 사용하여 요구사항을 위반하는 사례가 있는지 자동적으로 찾아 보는 방법이다.

보안 프로토콜을 정형검증 하는 방법은 BAN, GNY Logic[3][4] 등을 통한 정리증명 또는, SPIN[5], SMV[6] 등을 이용한 모델체킹 등 다양한 방법들이 있으나, 본 논문에서는 CASPER 를 사용하여 프로토콜을 CSP 명세언어로 명세하고 FDR 모델체킹 도구를 통해 보안성을 검증하였다.

본 논문은 2 장에서 정형기법을 통한 검증의 방법에 대해 소개하고, 3 장에서 기존에 제시된 인증 프로토콜들을 분석한 뒤 4 장에서 정형적인 방법에 의한 검증 실시와 결과를 제시하고, 마지막으로 5 장에 결론 및 미래 연구하고자 하는 분야에 대해 제시하는 것으로 구성되어 있다.

2. CSP, CASPER 그리고 FDR

2.1 CSP(Communicating sequential Process)

CSP[7]는 프로세스 알지브라 언어로 시스템을 정형적으로 명세하는데 쓰이는 언어 중의 하나이다. CSP는 병렬성을 갖는 통신 프로토콜을 효율적으로 명세할 수 있어 통신프로토콜 및 제어 시스템을 명세하는데 많이 사용되었고, 분산 시스템 환경하에서 동작하는 클라이언트 서버와 공격자 모델을 정형적으로 명세할 수 있는 장점으로 보안 프로토콜을 명세하는 분야까지 확대되어 사용되고 있다. 분산시스템 환경하에서 동작하는 보안 시스템은 다음과 같이 표현할 수 있다.

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

2.2 CASPER(A Compiler for the Analysis of Security Protocols)

CSP 는 통신 프로토콜을 효율적으로 명세할 수 있지만, 명세가 매우 어렵고 복잡하여 숙련된 프로그래머에 대한 의존도가 매우 높다. 이에 따라 프로그래머에 의한 오류코드가 포함될 수 있는 문제점이 존재한다.

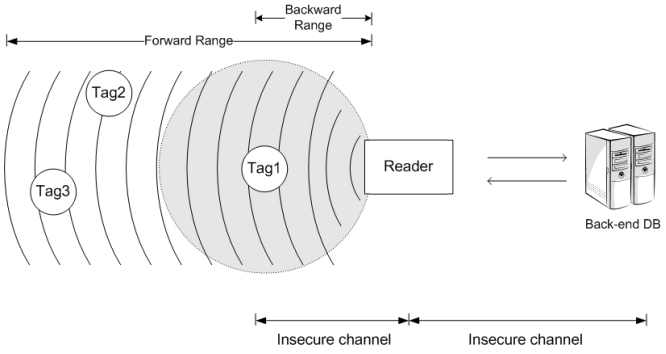
CASPER[8]는 보안 프로토콜을 명세하는데 사용되는 도구로써, 프로토콜을 명세하는 방법이 매우 명료하고 간단하다. 그리고 CASPER 는 간단한 문법을 통해 명세된 코드를 CSP 언어로 자동적으로 변환해주기 때문에 위에서 언급한 프로그래머에 의해 발생될 수 있는 문제점을 제거시켜준다.

2.3 FDR(Failure Divergence Refinement)

FDR[9]은 CSP 를 입력언어로 받는 도델체킹 도구로서, CSP 로 명세된 프로토콜이나 보안 시스템이 그들이 제시하는 비밀성 또는 인증과 같은 보안속성을 만족 하는지 자동적으로 확인해준다. 이를 통해 해당 속성을 만족시키지 못할 경우에는 반례를 제시하여, 가능한 공격 시나리오 분석을 도와준다. 보안 프로토콜의 경우, 반드시 갖추어야 하는 비밀성, 무결성, 인증성, 부인방지와 같은 보안속성의 만족여부를 검증해주고, 만족시키지 못할 경우 반례를 제시해준다는 것이다.

3. RFID 인증 프로토콜 분석

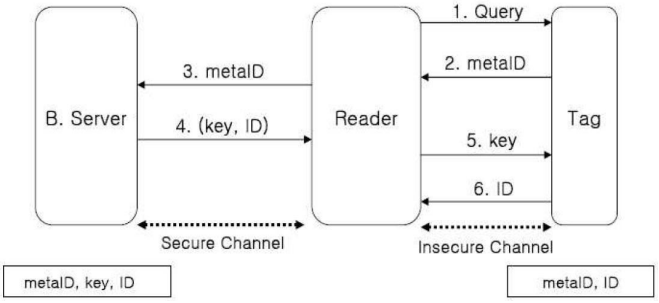
일반적으로 RFID 시스템은 태그와 리더 그리고 데이터베이스 서버로 구성되어 있다 (그림 1).



(그림 1) RFID 시스템

RFID 시스템은 통신환경의 특수성 때문에 도청이 매우 용의하여 RFID 시스템은 정보 누출(Data Leakage)과 정보추적(Traceability)이라는 취약점을 갖고 있다.

위와 같은 보안적 취약점을 보안하기 위해서 제안된 프로토콜의 대부분이 해쉬기반 인증 프로토콜이다.



(그림 2) 해쉬 락 프로토콜

대표적인 해쉬기반 프로토콜에는 해쉬 락 기법[10]을 들을 수 있다(그림 2). 이 프로토콜은 리더에게 의해 태그의 ID 가 정해지고, 리더는 이 값을 태그에게 보낸다. 태그는 리더로부터 ID 값을 받은 후에 자신을 잠금 상태로 전환시킨다. 후에 리더가 태그에게 정보를 요구하게 될 경우 리더는 태그로부터 ID 값을 받아 서버에게 보내어 태그를 잠금 상태를 해제할 수 있는 키값을 받고, 이 값은 태그에게 보낸다. 태그는 리더에게 키값을 받게 되면 자신의 상태를 해제하고 리더의 요구에 응답한다. 하지만 이 프로토콜은 태그의 잠금 상태를 해제하는 키값이 공격자에게 노출되어 있고, 태그, 리더 및 서버간 주고 받는 데이터의 값이 고정 값이기 때문에 추적이 이것을 통해 추적이 가능하다는 문제점이 있다.

위에서 언급한 태그의 추적 가능성을 제거하기 위해 Randomized 해쉬 락 기법[11]이 제안 되었으나, 리더가 서버로부터 받은 ID 값 중에서 태그가 보내온 ID 값과 매칭되는 것을 찾기 위해 많은 자료를 처리해야 하는 부담을 갖게 되는 큰 단점 있고, 리더가 매칭되는 ID 값을 태그에게 다시 보내는 과정에서 ID 값이 노출되어 익명성 또한 보장해주지 못한다.

이에 태그의 익명성을 보장하기 위해 태그와 서버의 성공적인 인증 종료되면 태그와 서버가 해쉬함수를 사용하여 태그의 ID 값을 바꾸도록 하는 해쉬 체인 기법[12]이 제안 되었으나, 단위 태그당 서버가 계산해야 하는 해쉬값 계산이 너무 많다는 문제점을 갖고 있고, 한 세션 내에 인증이 끝나지 않을 경우 주고받는 값들이 고정 값이기 때문에 이 방법도 태그의 익명성을 보장해주지 못한다.

마지막으로 RFID 시스템을 구성하는 요소들이 임의의 값을 사용하게끔 하는 해쉬 기반 ID Variation 기법[13]이 제시되었다. 하지만 태그가 해쉬 함수를 사용하여 태그의 ID 값을 암호화한 값을 리더에게 보내게 되는데, 인증 세션이 종료되지 않는 상태일 때에는 이 값이 고정 값이기 때문에 부분적인 태그의 추적이 가능하다. 또한 서버가 분산되어 있는 환경에서 사용될 경우 태그의 정보가 분산된 서버간에 실시간적으로

갱신이 되지 않기 때문에, 서버간의 데이터 베이스 정보의 차이로 문제가 발생될 수 있다.

4. 모델체킹을 이용한 인증 프로토콜의 정형적 검증

4.1 CASPER 를 이용한 프로토콜 명세

CASPER 를 사용하여 프로토콜을 명세할 때에는 크게 프로토콜의 세부사항을 명세 하는 부분과 프로토콜이 사용되는 시스템을 명세 하는 부분을 나뉘고 세부적으로는 다음과 같이 나뉜다.

- Free variables
- Processes
- Protocol description
- Specification
- Actual variables
- Functions
- System
- Intruder Information

다음은 CASPER 를 이용하여 해쉬 락 기법의 언락(잠금해제) 프로토콜을 명세한 것이다.

```
-- Hash Lock 의  Unlock Protocol

#Free variables
R, T : Agent
key : SessionKey
Id : Text
DB: Server
InverseKeys = (key, key)
H : HashFunction

#Processes
INITIATOR(T, R, key, Id)
RESPONDER(R, DB)
SERVER(DB, T, key, Id)

#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

#System

INITIATOR(Tag, Reader, Key, ID)
RESPONDER(Reader, DataBase)
SERVER(DataBase, Tag, Key, ID)
```

위에서 #Free variables 은 프로토콜에서 사용되는 데이터의 유형 및 함수 등을 정의하는 부분이고, #Processes 는 RFID 시스템을 구성하는 요소들이 프로토콜에서 갖는 위치와 그들이 알고 있는 정보에 대해서 정의하는 부분이다. 그리고 RFID 시스템 요소간 어떤 절차에 의해 통신이 이루어지는지 정형적으로 명세하는 부분은 #Protocol description 부분이다. 또한 이러한 절차를 통해 프로토콜이 만족시키고자 하는 요소에 대해 명세하는 부분은 #Specification 이다.

다음은 해쉬 락 기법을 사용한 프로토콜이 만족시키고자 했던 태그의 ID 와 해제에 사용되는 키값의 비밀성 그리고 이 두 개 값을 사용하여 태그와 리더가 정상적인 상호인증을 할 수 있음을 명세한 것이다.

```
#Specification

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

#System 은 이 프로토콜을 사용하는 RFID 시스템 구성원들의 기본적 사항에 대해서 정의하는 곳으로, 프로토콜을 세부사항을 명세하는 부분에서 #Processes 와 대칭된다.

프로토콜을 명세하는데 가장 중요한 것은 공격자 모델을 명세하는 것이다. 다음은 공격자가 누구이고, 그가 알고 있는 지식이 무엇인지 CASPER 를 통해 명세하는 방법을 보여주고 있다.

```
#Intruder Information

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

위의 코드에서는 Mallory 가 공격자의 모델이 된다는 것과 Mallory 는 태그, 리더, 그리고 서버가 누구인지 알고 있음을 명세한 것이다.

4.2 검증 결과

FDR 은 CASPER 명세코드 중에서 #Specification 에 명세된 보안 요소들이 주어진 프로토콜에서 만족되는지 검증한다. 해쉬 락 기법을 통한 언 락 프로토콜에서 만족시키고자 하는 세가지 부분을 FDR 을 통해 검증한 결과, 3 가지 모두 만족시키지 못한다는 것을 확인하였다.

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

위의 두 조건은 리더(R)은 태그(T)와 key 와 Id 는 비밀값으로, 공격자는 이를 알 수 없음을 명세한 것이다. 하지만, 공격자가 리더로 위장하여 태그에게 H(key)값을 받아 서버에게 보내게 되면, 서버는 리더로 위장한 공격자에게 태그의 잠금 상태를 해제할 수 있는 key 값과 ID 값을 줄 수 있기 때문에, 위와 같은 조건을 만족시키지 못한다.

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

위의 조건은 태그와 리더는 Id 와 key 를 갖고서 상호 인증한다는 것을 의미한다. 하지만 리더로 위장한 공격자가 key 와 Id 값을 정상적으로 획득할 수 있기

때문에 key 와 Id 를 통한 상호인증은 의미가 없어서 위와 같은 조건을 만족시키지 못한다.

다음은 FDR 이 해쉬 락 기법의 언락 프로토콜이 만족시키고자 하는 조건 중에서 리더와 태그는 Id 를 비밀값으로 갖고 있고, 공격자는 이를 알 수 없다는 조건을 어떻게 위배하는지 보여주는 반례 코드이다.

- Secret(R, Id, [T])에 대한 반례를 보여주는 CSP 이벤트

```
env.Tag.(Env0,Reader,<>)
send.Tag.Reader.(Msg1,Hash.(H,<Key>),<>)
receive.Mallory.DataBase.(Msg2,Hash.(H,<Key>),<>)
send.DataBase.Mallory.(Msg3,Sq.<Key,ID>,<>)
receive.Tag.Reader.(Msg1,Garbage,<>)
send.Reader.DataBase.(Msg2,Garbage,<>)
receive.DataBase.Reader.(Msg3,Sq.<Key,ID>,<>)
send.Reader.Tag.(Msg4,Key,<>)
signal.Claim_Secret.Reader.ID.{Tag}
leak.ID
```

위의 CSP 이벤트는 CSP 전문가가 아니면 분석하기 어렵다. 하지만 CASPER 의 interpret 명령어 기능을 사용하면 다음과 같이 비밀속성을 위배하는 공격 시나리오를 확인할 수 있다.

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

I_Mallory, I_Reader 그리고 I_DataBase 는 리더 또는 서버로 위장하여 메시지를 가로채는 공격자를 나타낸다. 위의 시나리오를 보면 중요 데이터인 태그의 ID 와 잠금 해제용 키값이 공격자에게 노출되는 것을 알 수 있고, 그 결과 해쉬 락 기법을 사용한 언락 프로토콜은 비밀성을 만족시키지 못한다는 사실을 확인 할 수 있다.

5. 결론

RFID 기술은 매우 편리한 기술이지만, 기술이 지니고 있는 보안적 취약점을 제거하지 않는다면 큰 문제점을 발생시킬 수 있는 기술이다.

본 논문에서는 RFID 시스템의 보안적 취약점을 제거하기 위해 제안된 프로토콜들을 정형기법을 사용하여 설계단계에서 검증하는 방법을 제시하였다. 그리고 모델체킹 방법 중 널리 사용되고 있는 CASPER/CSP 및 FDR 도구를 이용하여 RFID 해쉬 락 프로토콜의 보안성을 검증한 결과, 비밀성과 인증성을 만족시키지 못한다는 것을 보여주었다.

향후 정형기법을 사용하여 Challenge-Response 인증 기법을 분석하고, 이 기법을 사용하여 제안된 인증 프로토콜들을 정형적으로 검증하여, 보다 나은 RFID 인증 프로토콜을 설계하고자 한다.

참고문헌

[1] E. M. Clarke and J. M. Wing, "Formal Methods: State of the Art and Future Directions", ACM Computing Surveys, vol. 28, No. 4, pp.626-643, 1996

[2] D. H. Craigen, S. L. Gerhart and T. J. Ralston, "An International Survey of Industrial Applications of Formal Methods", NRL/FR/5546 —93-9581, Vol. 1, 1993

[3] M. Abaid, M. Burrow, and R. Needham.“ A Logic of Authentication”, Proceedings of the Royal Society, Series A,426,1871,pp.233-271, December 1989

[4] L. Gong, R. Needham. R, Yahalom, “Reasoning about Belief in Cryptographic Protocols”, IEEE, 1990

[5] G. J. Holzmann, “The Model Checker – SPIN", IEEE Transactions on Software Engineering, 1997

[6] Kenneth. L. McMillan. “The SMV system, symbolic model checking-an approach”, Technical Report CMU-CS-92-131, Carnegie Mellon University, Pittsburgh, 1992

[7] Stephen Brookes, C. A. R. Hoare, and A. W. Roscoe, "A Theory of Communicating Sequential Processes", Journal of the ACM, vol. 31, no. 3, pp.560-599, Jun 1984

[8] Gavin Lowe, "Casper: A Compiler for the Analysis of Security Protocols", In Proceedings of The 10th Computer Security Foundations Workshop, 1998

[9] Gavin Lowe, “Breaking and Fixing the Needham Schroeder Public Key Protocol using FDR”

[10] S.E. Sarma. Weis, and D.W. Engels, “RFID systems, Seucrity and Privacy Implications”, White Paper MIT-AUTOID-WH-014, AUTO-ID CENTER, 2002

[11] S.A. Weis, “Security and Privacy in Radio Frequency Identification Devices”, MS Thesis, MIT, May 2003

[12] M. Ohkubo, K Suzuki, and S, Kinochita, “Hash-Chain Based Forward-Secure Privacy Protection Scheme for Low-Cost RFID”, Proceedings of the SCIS 2004, pp.719-724, 2004

[13] D. Henrici, P. Muller, “Hash-based Enhancement of Location Privacy for Radio-Frequency Identification Devices using Varying Identifiers”, Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops(PERCOMW’04), pp.149-153, IEEE, 2004