2005년도 대한전자공학회 추계종합학술대회 제28권 제2호

EPC C1G2 상호인증 프로토콜 제안

김건우
한국전자통신연구원

Proposal on EPC C1G2 mutual authentication protocol

Keonwoo Kim
Electronics and Telecommunications Research Institute
E-mail : wootopian@etri.re.kr

Abstract

최근들어 Radio Frequency Identification(RFID) 태그가 다수의 상품에 부착되고 여러 분야에 적용되기 시작했지만, 편리성이나 비용문제로 인해 인증과 암호화 같은 보안기능은 적용되지 않고있다. 보안 기능이 없는 RFID 시스템은 개인정보 노출, 불법 리더의 접근, 위조 태그의 남용과 같은 심각한 부작용을 초래하지만, 태그 자원의 제약으로 인해 보안기능을 적용하기가 쉽지않다. 현재 여러 기술을 따르는 RFID 시스템 중 EPCglobal 의 EPC Class 1 Generation 2(C1G2) 는 산업계의 여러 분야에서 특히, supply-chain 모델에서 사실상 국제표준으로 여겨진다. 본 논문에서는, RFID 보안 프로토콜 중 EPC C1G2 메커니즘의 Inventory 과정에서 태그가 리더를 인증하는 기법을 제안한다. C1G2 시스템에서는 인증되지 않은 리더의 태그 액세스가 가능한데, 이는 태그의 리더 인증으로서 차단될 수 있다. 또한, EPC C1G2 태그-리더 간의 상호인증 기법을 제안한다. 이 과정에서 태그 ID 는 노출되지 않고 전송되며, 태그 인증을 통해 태그 위변조를 방지할 수도 있다. 제안 메커니즘은 태그를 식별하는 절차에서 인증을 위해 프로토콜 패스 수의 증가가 없다. 다만 리더와 태그에서 Inventory 과정의 *ACK* command 와 태그의 reply 구현에 약간의 수정을 필요로 한다.

I. 서 론

최근에 항만, 물류, 유통 등의 여러 분야에서 RFID 가 적용되기 시작하고, 우리나라에서는 이동통신 시스템과 결합한 mobile RFID 서비스를 준비하고 있다. 아직까지는 능동형 RFID 보다 수동형 RFID 를 적용한 분야가 많고 RFID 태그의 범용적인 사용을 위해서는 수동형 태그의 수요가 많을 것으로 예상된다.

수동형 태그는 리더로부터 신호를 받아서 필요한 전력을 공급받고 리더의 명령에 응답한다. 하지만, 제한된 메모리 공간, 적은 게이트 수, 프로토콜 구동에 필요한 전력의 제한 등으로 강한 암호 프리미티브나 보안 요구사항을 만족시키기 어렵다. 즉, 안전한 태그-리더간 통신을 위한 프로토콜 설계에서 전자서명이나 공개키 기반 암호 뿐만 아니라 해쉬나 대칭키 기반 암호까지도 수동형 태그에 구현하기가 쉽지않다. 하지만, 인증, 접근제어, 데이터 암호화와 같은 보안 기능을 제공하지 않는 RFID 시스템에서는 태그 소유자의 위치추적, 프라이버시 침해, 태그 복제, 허가받지 않은 리더의 불법접근 등을 근본적으로 방지하기가 거의 불가능하다. 이런 부작용을 방지하기 위해서는 태그-리더 간의 인터페이스 수행은 물론이고 암호학적 연산 및 보안 프로토콜 수행까지 가능한 마이크로프로세서와 메모리가 태그에 내장되어야 한다[5].

여러가지 기능을 수행할 수 있는 태그와 그 제작 비용과는 상관관계가 있어, 우리는 현재 널리 사용되는 수동형 태그에 연구의 초점을 맞춘다. 이 논문에서 우리는 여러 가지 타입의 수동형 태그 중 특히 EPC 태그의 보안에 관하여 분석한다.

RFID 사용을 위해 EPC 코드를 개발하고 산업계 위주의 표준화을 주도하는 EPCglobal[1]은 수동형 태그와 관련 RFID 시스템을 위해 860MHz ~ 960 MHz Class 1 RFID 태그와 태그-리더 간의 통신 인터페이스를 개발했다[2]. 또한, 최근에는 Class1 의 진화된 버전으로서 Class1 과 동일한 주파수 대역에서 UHF RFID 프로토콜을 적용한 C1G2 태그[3]를 상용화하기 시작했다.

하지만, 태그 자원의 제약으로 인증, 암호 등의 보안 기능은 구현되지 않았다. Kill 정도가 EPC 태그의 보안기능이라고 할수도 있으나, 태그의 기능이 정지되기 때문에 태그로서의 의미는 더이상 없어진다. 그래서, 본 논문에서는 EPC C1G2 를 위한 보안 프로토콜로서,태그의 리더 인증 및 태그-리더의 상호인증 기법을 제안한다.

본 논문은 다음과 같이 구성된다. 2 장에서는 RFID 시스템과 EPCglobal 시스템에 관하여 간략히 살펴본다. 3 장에서는 EPC C1G2 프로토콜을 설명한 후 태그의 리더 인증 기법을 제시한다. 4 장에서는 EPC C1G2 태그-리더 사이의 상호인증 메커니즘을 제안한다. 마지막으로 5 장에서 결론을 내린다.

II. RFID 와 EPCglobal 시스템

RFID 시스템은 태그, 리더, 백엔드 데이터베이스, 그리고 다른 부가적인 시스템으로 구성된다[5].

리더는 Transceiver 혹은 Interrogator 라고도 하며, RF 인터페이스를 통하여 태그에 데이터를 보내거나 태그로부터 데이터를 읽는 장치이다. 리더는 태그와의 물리적인 접촉없이 다른 태그와의 충돌을 피하여 태그로부터 고유의 식별코드를 인식할 수 있어야 하고 태그에 정보를 기록할 수 있어야 한다. 또한, 태그의 로깅 정보나 암호키와 같은 데이터를 저장하는 백앤드 데이터베이스와 연결될 수 있다.

Transponder 라고도 불리는 태그는 프로토콜 수행과 암호연산 등을 수행하기 위한 프로세서, 데이터를 저장하는 메모리, 그리고 RF 안테나로 구성된다. 태그는 각각 고유의 식별수단인 ID 를 가지고 있고, RF 동작 범위 내에서 리더에게 ID 를 전송한다. 태그는 두가지 종류가 있는데, 자체 밧데리를 내장한 능동형 태그와 자체 밧데리를 내장하지 않는 수동형 태그가 그것이다. 수동형 태그는 리더로부터 RF 신호를 받아서 태그 동작에 필요한 전력을 공급받는데, 능동형 태그에 비해 비싸지 않게 제조가 가능하고 현재 많은 응용분야에서 사용된다. 한편, 태그의 제한된 전력으로 인하여 태그-리더 사이의 하향 채널 통신거리는 리더-태그 사의의 상향 채널 통신거리 보다 훨씬 짧다. 또한, 수동형 태그에 tamper-resistant 메모리를 적용하기에는 비용이 많이 들어 태그 내부 데이터가 차분 전력 공격, EM 공격과 같은 물리적 공격에 의하여 쉽게 노출될 수 있다.

RFID 시스템을 구성하는 요소로서 태그, 리더 외에 리더와 연결된 정보 서버, 미들웨어 시스템, 추적 관리 개체 등이 있다.

한편, EPCglobal 은 supply-chain 모델에서 태그 정보의 실시간, 자동식별을 위한 국제 표준을 사실상 주도하는 기관으로, EPCglobal 네트워크를 구성하고 지원한다. EPCglobal 네트워크는 다음과 같은 요소로 구성된다 [1].

- EPC(Electronic Product Code)
  - 각각의 태그를 식별하는 고유의 식별번호
  - 여러 코드 체계 중 태그 식별을 위한 사실상 표준
- ID System
  - EPC 태그와 EPC 리더로 구성
  - EPC 는 EPC 태그에 저장되고, EPC 태그는 케이스, 팔레트, 컨테이너 등의 아이템에 부착됨
  - EPC 리더는 EPC Middleware 를 사용하여 정보를 관리하고 EPC 태그와 통신
- EPC Middleware
  - EPC IS 와의 통신을 위해서 실시간 read 이벤트와 정보를 관리하고 alert 를 제공하며, EPC 리더와의 통신을 위해 기본적인 read 정보를 관리함
- EPC IS(EPC Information Service or Server)
  - EPC 와 관련된 데이터를 EPCglobal 네트워트를 통하여 상대방과 교환가능하도록 하는 서비스
- EPC Discovery Service
  - 특정 EPC 와 관련된 데이터를 찾을수 있게하고 그 데이터에 접근을 요청하도록 하는 서비스
  - Object Naming Service (ONS)

본 논문에서는 수동형 태그와 이의 보안 시스템에 관심이 있기 때문에, EPCglobal 의 EPC Class 0~5 중에서 EPC C1G2 이거나 Class 2 정도의 수준에 초점을 맞춘다.

III. EPC C1G2 동작 프로토콜 및 리더인증 메커니즘 제안

3 장에서는 EPC C1G2 프로토콜을 간단히 설명하고, 태그가 리더를 인증하는 방법을 제시한다.

EPC C1G2 리더가 태그를 식별하고 특정 태그에게 명령을 보내기 위해서는 Select, Inventory, Access 과정을

거친다. Select 는 Inventory 와 Access 를 위해 리더가 특정 태그 집단을 선택하는 단계로, 리더의 *Select* command 에 의해 태그의 SL(Selected Flag) 설정값이 달라진다. Inventory 는 태그를 식별하는 단계로, 리더는 태그가 가지는 네가지 세션 중 하나를 선택해서 *Query* command 를 보냄으로서 Inventory 라운드를 시작하고, 하나 이상의 태그가 응답한다. 리더는 특정 태그의 응답을 발견하고, 그 태그에게 PC, EPC, CRC-16 을 요구한다. Inventory 는 *Query, QueryAdjust, QueryRep, ACK, NAK* command 로 구성되며, Inventory 라운드는 한번에 하나의 세션에서만 동작한다. 그리고, Access 는 태그에게 읽기/쓰기 등의 명령을 내리는 단계로서, *Req_RN, Read, Write, Kill, Lock, Access, BlockWrite, BlockErase* command 로 구성된다[3].

그리고, EPC C1G2 태그 메모리는 논리적으로 4 개의 뱅크, 즉, Reserved memory, EPC memory, TID memory, User memory 로 구분되며, 각 메모리 맵에 의해 Kill password, Access password, EPC, PC, CRC, TID, user-specific data 등의 정보를 저장한다. 또한, C1G2 태그는 Ready state, Arbitrate state, Reply state, acknowledged state, Open state, Secured state, Killed state 와 Slot Counter 를 구현한다. 각 state 에 따라 리더 command 에 대한 태그 reply 가 결정되고, 다른 state 로의 천이가 달라진다.

그림 1 은 리더가 태그를 Inventory 하고 Access 하는 절차와, 태그가 리더를 인증하는 과정을 나타낸 것이다.

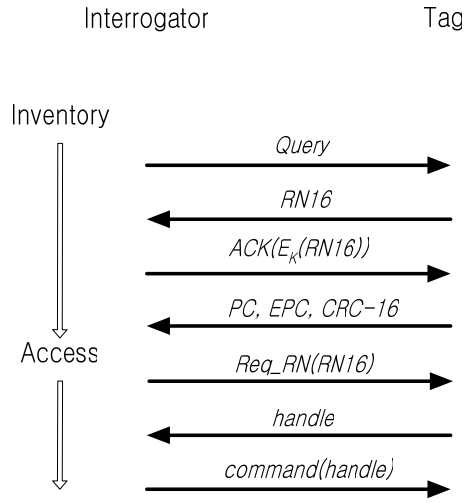

그림 1. 태그의 Inventory, Access 및 리더 인증 절차

① 리더는 *Query* command 를 보냄으로서, Inventory 라운드를 초기화한다.

② *Query* command 를 수신한 태그는 SL, 세션 파라미터 등을 확인하고 랜덤수를 생성해서 리더에게 backscattering 한다.

③ 리더는 태그를 식별하기 위해 *ACK* command 를 보내는데, 이때 이전 단계에서 태그로부터 수신한 랜덤값 RN16 을 리더-태그 사이에서 미리 공유한 키로 암호화 해서 보낸다.

④ *Ack* command 를 수신한 태그는 자신의 PC, EPC, CRC-16 을 리더에게 보낸다. 이로서 Inventory 절차가 마무리된다.

⑤ Access 절차를 위해 러더는 새로운 랜덤값을 태그에게 요구하면 태그는 이에 응답하고, 이 랜덤값 handle 이 *Read, Write, Kill* 같은 *Access* command 에 사용된다.

Inventory 과정의 ②와 ③에서 리더가 태그로부터 수신한 랜덤수를 암호해서 보내고 이를 태그가 확인함으로서 태그는 리더를 인증할 수 있다. 만약 리더가 동일한 비밀키를 가지고 있지 않다면, ③ 과정에서 태그는 리더의 command 를 무시하고 인증은 실패한다. 제안하는 방법은 태그가 리더를 인증한 후에 태그가 리더에게 인식되기 때문에 불법리더의 접근을 제한할 수 있다.

IV. EPC C1G2 상호인증 프로토콜 제안

4 장에서는 3 장에서 설명한 태그-리더 사이의 프로토콜을 기반으로 태그-리더 간의 상호인증 프로토콜을 제안한다.

태그-리더 상호인증 역시 Inventory 과정에서 이루어지는데, 태그의 리더 인증은 3 장에서 설명한 방식과 동일하다. 러더의 태그 인증은 리더가 랜덤수를 생성해서 이를 암호해서 태그에게 보내고 이를 다시 태그로부터 확인하는 과정을 통해 가능하다. 이것은 ISO/IEC 9798-2[4] 에서 정의된 대칭키 기반 시도-응답 프로토콜을 적용한 것이다. 이를 통해 불법 리더의 접근을 막을 수 있을 뿐만 아니라, 태그의 ID 도 노출되지 않게 할수 있고 태그인증을 통해 태그 위변조를 방지할 수도 있다. 그림 2 는 EPC C1G2 의 상호인증 프로토콜을 나타낸 것이다.

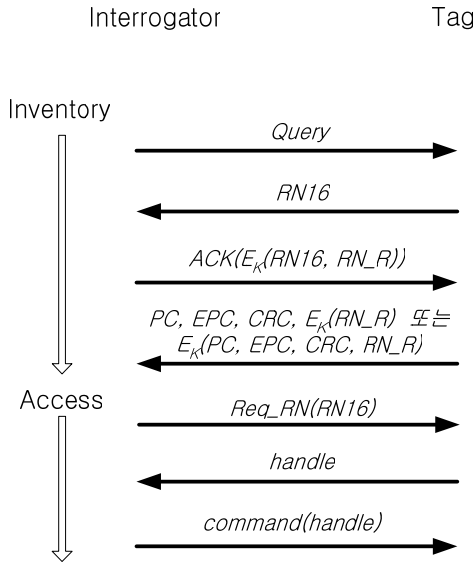

그림 2. EPC C1G2 상호인증 프로토콜

① 리더는 *Query* command 를 보냄으로서, Inventory 라운드를 초기화한다.

② *Query* command 를 수신한 태그는 랜덤수 RN16 을 리더에게 backscattering 한다.

③ 리더는 태그를 식별하기 위해 *ACK* command 를 보내는데, 태그로부터 수신한 랜덤값 RN16 과 자신이 생성한 랜덤값 RN_R 을 태그와의 공유 비밀키로 암호화 해서 보낸다. RN_R 의 암호전송은 태그 인증을 위해 필요하다.

④ 태그는 복호한 RN16 과 자신이 보낸 RN16 을 비교함으로서 리더를 인증한다. 리더 인증이 성공하면, 태그는 자신의 PC, EPC, CRC-16 을 리더에게 보낸다. 이때, 리더로부터 수신한 RN_R 을 비밀키로 암호해서 함께 보낸다. 만약 PC, EPC, CRC-16, RN_R 전부에 암호를 적용하면 태그 Identity 노출을 막을 수 있다.

⑤ 리더는 태그로부터 수신한 RN_R 을 복호해서 이를 자신이 보낸값과 비교함으로서 태그를 인증한다. 태그 인증이 성공하면 Inventory 절차가 종료된다.

⑥ 태그-리더 간의 상호인증이 성공하면, 태그-리더 통신을 위해 Access 절차가 시작된다.

제안 프로토콜의 태그를 식별하는 절차에서 인증을 위해 프로토콜 패스 수의 증가가 없다. 다만 리더와 태그 Inventory 과정의 *ACK* command 구현에서 다음과 같은 수정이 필요한다.

	Command	RN16	RN_R
비트 수	2	16	16
설명	01	(RN16 ‖ RN_R) 의 암호화	

원래의 C1G2 *ACK* command 와 비교해서 16 비트 RN_R 이 추가되었고 RN16 ‖ RN_R 의 암복호 과정이 필요하다. 또한, 태그의 reply 구현에도 16 비트 $E_K(RN\_R)$ 추가가 필요하다. 리더가 랜덤수를 생성하고 암복호하는 것은 리더 성능에 큰 영향을 주지 않지만, 제한된 환경을 가진 EPC C1G2 태그에서 암복호 기능을 구현하는 것은 쉽지 않다. 이 논문에서는 사용되는 암호 알고리즘, 사전 공유 비밀키 분배, 비밀키 길이 등은 논하지 않는다.

V. 결 론

본 논문에서는 EPC C1G2 프로토콜에서 태그의 리더 단향방 인증 및 태그-리더 상호인증 프로토콜을 제안하였다. 태그의 제한된 환경을 고려할 때, 기존 메커니즘에서 어떠한 변경없이 일부 command, reply 포맷의 수정으로만 인증이 가능하게 함으로서 EPCglobal 표준 호환성과 문제가 없게 하였다. 향후 EPC C1G2 같은 수동형 태그에 적합한 lightweight 암호 알고리즘을 설계, 구현하는 과제가 남아있다.

참고문헌

[1] http://www.epcglobalinc.com

[2] EPCglobal, “860MHz - 930 MHz Class 1 Radio Frequency Identification Tag Radio Frequency & Logical Communication Interface Specification”, Technical Report, Candidate Recommendation, Version 1.0.1, 2002.

[3] EPCglobal, “Class-1 Generation-2 UHF RFID Protocol for Communications at 860MHz – 960MHz”, Version 1.0.9, 2005.

[4] ISO/IEC 9798-2, “Information Technology -Security Techniques- Entity Authentication Mechanisms Part 2 : Entity Authentication using symmetric techniques”, 1993.

[5] K. Finkenzeller, “RFID Handbook”, John Wiley & Sons. 2003.