현실적인 공존 증명 프로토콜*

은하수†, 임지환†, 오희국†, 김상진‡
†한양대학교 컴퓨터 공학과
‡한국기술교육대학교
e-mail : hseun@infosec.hanyang.ac.kr

Practical Co-Existence Proof Protocol*

Hasoo Eun†, Jihwan Lim†, Heekuck Oh†, Sangjin Kim‡
†Dept. of Computer Science, Han-Yang University
‡Korea University of Technology and Education

요 약

공존 증명(Co-Existence Proof, CEP) 프로토콜은 둘 이상의 태그들이 공존했음을 증명하는 정보를 생성하는 것으로, 2004 년 A. Jules 가 제안한 이래 많은 연구가 진행되어 왔다. 하지만 대부분의 논문이 재전송 공격 방지와 익명화에 초점이 맞추어져 있다. 본 논문에서는 이러한 공존 증명 프로토콜을 인증 프로토콜, 검색프로토콜과 차별화 시킴과 동시에 현실적으로 공존 증명 프로토콜을 사용하고 연산량을 최소화 시킬 수 있는 환경에 대해 논한다.

1. 서론

CEP 는 둘 이상의 태그들이 공존했음을 증명하는 정보를 생성하는 것으로, 2004 년 A. Jules 에 의해 제안되었다. 그 후 J. Saito 등에 의해 재전송 공격에 대한 문제가 제기되면서 공존 증명 프로토콜의 연구는 재전송 공격을 막는 방향으로 진행되었고, 최근에는 익명성을 보장하는 방향으로 연구가 진행되고 있다.

공존이라 함은 시간적, 공간적으로 같은 상황에 있는 것을 말한다. 따라서 이를 증명하기 위해서는 시간적 인접성과 공간적 인접성을 보일 수 있어야 한다.

연구가 진행됨에 따라 프로토콜이 복잡해지고 있다. 이러한 시점에서 프로토콜의 현실성에 대해 분석해볼 필요가 있다. 여기서의 현실성은 프로토콜이 실제로 구현 가능한지에 대해 판단하는 것이다. 본 논문에서는 현실성의 개념을 더욱 확장하여 프로토콜이 실제로 구현한다고 가정했을 때 기존 시스템(인증 프로토콜, 검색 프로토콜)에 비해 얼마나 효율적이고 쓸모 있는지에 대해 판단한다.

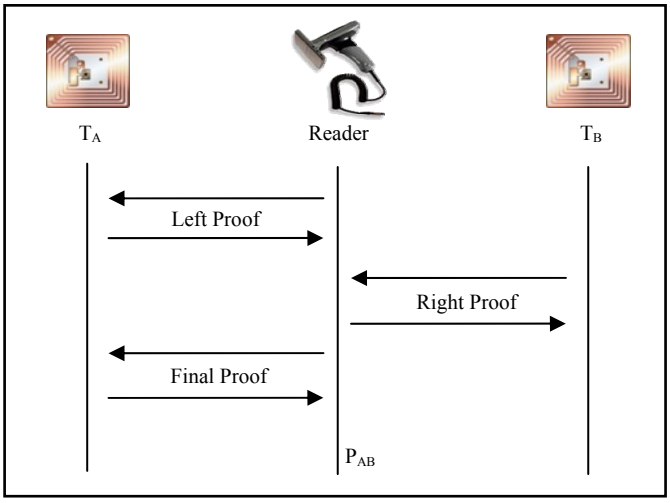
* 본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT 연구센터 지원 사업의 연구결과로 수행되었음 (NIPA-2010-C1090-1011-0010).
* 이 논문은 2010 년도 정부(교육과학기술부)의 재원으로 한국 과학재단의 지원을 받아 수행된 연구임(No. 2010-0000438).

2. 관련 연구

2.1 CEP Overview

CEP 는 둘 이상의 태그들이 공존했음을 증명하는 정보를 생성하는 것으로, 2004 년 A. Jules 에 의해 제안되었다. 최초의 제안은 두 태그에 대한 공존 여부를 증명하는 것이었으며 Minimalist MAC 을 이용한 Hash 기반의 프로토콜이었다. 그 후 J. Saito 등에 의해 재전송 문제가 제기되었으며, 태그 군에 대한 인증으로 확장되었으며 이후로도 많은 연구가 진행되어 왔다. CEP 는 각 태그만이 만들 수 있는 정보를 이용하여 최종 증명 문장을 만드는 것으로 기본적인 형태는 3 단계로 구성된다.

- 1 차 질의(Left proof, LP): CEP 를 수행할 태그 중 첫 번째 태그에게 질의를 하는 과정. 방향에는 큰 의미가 없으나 최초 제안 시 왼쪽 태그부터 인증을 개시했기에 이와 같이 이름 지어졌다고 판단됨
- 2 차 질의(Right proof, RP): LP 의 결과를 이용하여 나머지 태그들에게 질의를 는 과정. RP 에 참여하는 태그가 여럿일 수 있음. 이 과정부터 각 태그의 비밀 값이 이용되기 시작함
- 최종 질의(Final proof, FP): LP 를 수행한 태그에게 RP 의 결과를 보내는 과정. 첫 번째 태그만이 만들 수 있는 정보를 통해 비밀 정보간의 연결성을 강건히 하는 과정으로 이를 마치면 프로토콜이 종료됨



(그림 1) CEP 의 기본적인 형태

위의 세 단계를 거치면 리더는 프로토콜의 진행에서 얻는 정보들을 이용하여 최종 증명문 P_{AB} 를 생성하여 검증자에게 보내게 된다.

2.2 기존의 CEP 프로토콜

2004 년 A. Jules 가 제안한 Yoking-Proof 프로토콜은 두 태그간의 공존을 증명하는 것을 목표로 제안되었다[1]. 하지만 이 프로토콜은 리더를 포획한 공격자가 T_A 와 사전에 통신 후 이를 T_B 에게 보내어 P_{AB} 를 만들어도 정상적으로 작동한다는 문제가 있으나, CEP 를 처음 제안했다는 데에 그 의의가 있다.

2005 년 J. Saito 등은 A. Jules 의 프로토콜에 위와 같은 문제가 있음을 밝히고 타임 스탬프를 이용한 프로토콜을 제안하였다[2]. 또한 A. Jules 가 향후 연구로 제안한 셋 이상의 태그에 대한 공존을 Grouping proof 라는 형태로 제안하였다. 하지만 타임 스탬프가 선형적으로 증가하는 특성을 이용하여 미래의 값을 예측할 수 있다는 문제가 있다.

2006 년 S. Piramuthu 는 J. Saito 등의 프로토콜에 위와 같은 문제가 있음을 밝히고 랜덤 값을 이용한 프로토콜을 제안하였다[3]. 이는 서버에서 생성한 랜덤 값을 이용하여 세션을 관리하는 방법이다.

2.3 환경 정의 및 가정

표기법	설명
T_A	Left proof 를 받는 태그
T_B	Right proof 를 받는 태그
P_{AB}	최종 증명문
n	프로토콜에 참여하는 전체 태그의 수
m	CEP 에 참여하는 태그의 수

<표 1> 표기법

비교를 위한 객관적인 환경을 구축하고 상황을 단순화 하기 위하여 다음과 같이 환경을 정의한다. CEP 를 통해 인증하고자 하는 태그는 T_A, T_B 이며, 편의상 T_A 가 LP 를 받는 태그이며 T_B 는 RP 를 받는 태그라고 가정한다. 공격자의 능력을 제한하기 위하여 리더와 검증자 사이의 채널은 기존 유선 모델과 마찬가지로 Secure 하다고 가정한다. 검증자는 각 태그에 대한 정보를 DB 에 저장하고 있으며, 하나의 레코드는 ID, KEY, EPC, CEP 4 개의 필드로 구성되어있다. ID, KEY, EPC 는 태그의 일반적인 정보들을 저장한 필드이며, CEP 필드는 해당 태그가 공존을 증명할 때 함께 있어야 하는 태그의 아이디를 저장하는 필드이다. CEP 의 효율성을 계산하기 위하여 프로토콜을 다음의 8 단계로 나누고 각 단계에 필요한 질의, 응답 횟수 및 연산량을 계산한다.

Left proof → 태그 응답→ CEP 에 참여할 태그 확인 → Right proof → 태그 응답→ Final proof → 태그 응답 → 최종 증명문

3. 현실적인 CEP 프로토콜

3.1 인증 프로토콜과 CEP 프로토콜

인증 프로토콜은 리더의 인식범위 내의 태그에게 질의를 하여 주변에 어떤 태그가 있는지 알 수 있다. 본 논문에서는 이를 전역질의(Global Query, GQ)라 하겠다. 인증 프로토콜을 이용하여 CEP 와 같은 공존을 증명할 수도 있다. 이를 이용한 공존의 증명은 다음과 같은 절차로 진행된다.

STEP 1. 리더는 최초의 질의를 자신의 주변 태그들에게 보내게 된다.
STEP 2. 태그들은 각각 자신의 ID 혹은 인증할 수 있는 정보를 리더에게 보낸다.
STEP 3. 리더는 한번에 인증한 태그들 가운데 CEP 가 필요한 태그들을 모아 P_{AB} 를 생성한다.

시간적 인접성 측면에서 인증 프로토콜은 한번의 세션 내에서 공존에 참여하는 태그들을 인식하게 되며, 공간적 인접성 측면에서는 리더의 인식 반경 안에 있는 태그에 대한 인증이므로 공존을 위한 두 가지 요소 모두 만족한다.

이 경우 효율성을 계산하면 리더의 질의 횟수는 GQ 1 회, 각 태그의 응답 횟수는 1 회, 리더가 태그로부터 받는 응답 수는 n 개, 연산을 위한 리더의 연산량은 O(n)이 된다.

3.2 GQ 를 이용한 CEP 프로토콜

인증 프로토콜에서 사용하는 GQ 를 CEP 에서 사용한다고 가정해보자. GQ 를 이용하는 경우 CEP 는 다음과 같은 절차로 진행된다.

STEP 1. 리더는 자신의 범위 안에 있는 모든 태그에게 질의를 보내는 것으로 LP 를 대

신하게 된다.

STEP 2. 질의의 결과로 리더는 n 개의 응답을 받게 된다.

STEP 3. 응답으로 온 n 개의 태그 중 CEP 에 참여할 태그를 검색한다.

STEP 4. CEP 를 해야 하는 태그가 있다면 다시 범위 내의 태그들에게 GQ 를 통해 RP 를 보낸다.

STEP 5. 4 의 과정을 T_A 를 제외한 m 개의 태그들에게 반복적으로 수행한다.

STEP 6. 질의의 결과로 또 다시 n 개의 응답을 받는다.

STEP 7. RP 의 결과를 이용하여 T_A 에게 GQ 를 통하여 FP 를 보낸다.

STEP 8. 리더는 최종적으로 n 개의 응답을 받게 된다

STEP 9. 프로토콜 수행으로 얻은 메시지들을 통해 리더가 P_{AB} 를 생성한다.

이 경우 효율성은 리더의 질의 횟수 m+1 회, 각 태그의 응답횟수 m+1 회, 리더가 태그로부터 받은 응답수는 (m+1)n 개, 연산을 위한 리더의 연산량은 초기 탐색을 통한 인증 비용과 검색 비용을 합하여 O((m+1)n)이 된다. 통신횟수는 인증 프로토콜에 비하여 m+1 배 많으며 연산량 또한 많다.

3.3 지정질의(Designate Query, DQ)를 이용한 CEP 프로토콜

리더의 통신횟수 및 연산량을 줄이기 위하여 리더가 DQ 를 사용한다고 가정하자. DQ 는 검색 프로토콜에 사용되는 것으로 리더가 특정 태그를 지목하고 질의를 하면 해당 태그만 응답을 하게 된다. DQ 를 사용하는 경우 CEP 프로토콜은 다음과 같은 절차로 진행된다.

STEP 1. 이 경우에도 CEP 는 자신의 주변에 어떤 태그가 CEP 를 필요로 하는지 알 수 없기 때문에 GQ 를 통해 LP 를 진행해야 한다.

STEP 2. 리더는 LP 의 결과로 n 개의 태그에 응답을 받는다.

STEP 3. 응답으로 받은 n 개의 메시지 중 CEP 를 해야 하는 태그를 발견하면 이를 T_A 라 인식하고 DB 에서 T_B 가 될 태그를 확인한다.

STEP 4. 위의 결과를 토대로 T_B 에게 DQ 로 RP 를 보낸다.

STEP 5. 이에 대하여 T_B 만 응답하게 되고 리더는 1 개의 응답을 받게 된다.

STEP 6. 함께 CEP 해야 하는 태그가 다수인 경우 4 ~ 5 과정을 반복 수행한다.

STEP 7. 리더는 RP 의 결과를 T_A 에게 FP 로 보낸다.

STEP 8. 리더는 T_A 로 부터 FP 의 결과로 하나의 응답을 받는다.

STEP 9. 프로토콜 수행으로 얻은 메시지들을 통해 리더가 P_{AB} 를 생성한다.

이 경우 효율성을 계산하면 다음과 같다. 리더의 질의횟수는 GQ 1 회 + DQ m 회, 각 태그의 응답횟수는 CEP 에 참여하는 태그는 2 회, 미 참여 태그는 1 회이다. 리더의 연산량은 초기 탐색을 통한 인증 비용과 검색 비용을 합하여 O((m+1)n)이다. 3.1 의 경우에 비하여 메시지 전송 횟수(태그의 응답 횟수)가 감소하였으나 리더의 연산량은 동일하다.

3.4 현실적인 CEP 프로토콜

3.3 에서 보인 바와 같이 DQ 를 사용해도 연산량은 줄어들지 않는다. 연산 자체가 프로토콜의 초기 탐색 비용이기 때문에 DQ 로는 응답의 수를 줄이는 효과밖에 주지 못한다. 따라서 이를 더욱 효율적으로 사용할 수 있는 다음과 같은 상황을 고려해보자.

프로토콜의 목표 범위를 좁혀서 T_A 와 T_B 의 공존여부만 판단한다. 이때 리더가 인식해야 하는 태그는 T_A 와 T_B 이다. 리더는 이 둘에 대한 리스트만을 유지하고 있다. 따라서 초기 탐색비용이 필요 없으며 프로토콜 시작 시 T_A 또는 T_B 가 있는지 확인하면 된다. 공존을 위해 리스트에 있는 모든 태그들이 존재해야 하므로 응답이 없을 시 프로토콜은 실패하며 공존하지 않는 것으로 간주한다. 현실적인 예로 안전화와 안전모를 동시에 착용해야 공사현장에 진입할 수 있도록 하는 출입문을 생각해보자. 출입문에 설치된 인식기는 안전화와 안전모만 인식할 수 있으면 되며, 리더 역시 이들에 대한 정보만 유지하면 된다.

STEP 1. LP 는 리더가 유지하고 있는 리스트의 태그 중 하나에게 DQ 를 보내는 것으로 시작한다.

STEP 2. DQ 의 특성상 LP 를 받은 T_A 만 응답하게 되므로 리더는 1 개의 응답만 받게 된다.

STEP 3. LP 의 결과를 T_B 에게 보내어 RP 를 수행한다.

STEP 4. RP 의 결과도 LP 와 마찬가지로 1 개의 응답만 돌아온다.

STEP 5. 함께 CEP 해야 하는 태그가 다수인 경우 3 ~ 4 과정을 반복 수행 한다.

STEP 6. 이를 다시 T_A 에게 보내어 FP 를 수행한다.

STEP 7. FP 를 받은 T_A 는 리더에게 응답하게 되고, 리더는 1 개의 응답을 받는다.

STEP 8. 프로토콜 수행으로 얻은 메시지들을 통해 리더가 P_{AB} 를 생성한다.

STEP 1.에서 응답이 없다면 위에서 정의한 바와 같이 프로토콜 실패로 간주되며 공존을 증명하기 위해서는 CEP 에 참여하는 모든 태그가 존재해야 한다.

따라서 질의는 CEP 에 참여하는 태그가 있다는 가정 하에 시작하며 초기 탐색이 필요 없다. 이 경우 효율성은 리더의 DQ m+1 회, T_A 의 응답 2 회, T_B 에 해당하는 태그들의 응답 횟수 각 1 회, 연산 비용은 O(m)이다. 만일 m < n 이라면 인증프로토콜의 연산비용보다 CEP 의 연산 비용이 적어진다. 즉, 리더의 인식범위 내에 CEP 에 참여하는 태그 외의 다른 태그들이 다수 존재할 때 인증 프로토콜에서 인증을 통해 공존을 증명하는 것보다 CEP 을 통해 공존을 증명하는 것이 효율적이다.

3.5 검색 프로토콜과 CEP 프로토콜

3.4 에서는 검색 프로토콜에 사용하는 DQ 를 이용하여 연산비용을 줄였다. 그렇다면 검색 프로토콜로 CEP 를 대체할 수 있는지에 대해 살펴보자. 이 경우 프로토콜의 진행 과정은 다음과 같다.

STEP 1. 리더는 DQ 를 통해 자신이 유지하고 있는 리스트 안의 태그들에게 단발적인 질의를 보낸다.
STEP 2. 리더는 자신이 질의한 각 태그들로부터 응답을 받는다.
STEP 3. 프로토콜 수행으로 얻은 메시지들을 통해 리더가 P_{AB} 를 생성한다.

이때의 연산량은 리더의 DQ m 회, 각 태그의 응답 횟수는 1 회이다. 연산량은 O(m)으로 3.4 과 같다. 메시지의 수가 줄어들어 더 이득이 될 것 같지만, 이렇게 생성된 정보로는 공존을 증명할 수 없다. 앞서 이야기한 공존을 증명하기 위한 시간적 인접성은 매크로화된 질의를 통해 질의 간의 시간을 줄임으로써 가능하게 한다 하더라도 독립된 세션을 갖는 m 개의 메시지가 공간적으로 인접했다는 것을 증명할 수는 없다. 따라서 검색 프로토콜 만으로는 CEP 프로토콜을 대체할 수 없다.

4. 결론

본 논문에서는 CEP 프로토콜의 기본적인 구조와 특징, 인증프로토콜 및 검색 프로토콜과의 차별성에 대해 논하였다. 또한 CEP 프로토콜이 최대의 효율을 발휘할 수 있는 환경과 상황을 모색하였다. 기존의 CEP 프로토콜은 차후에 제 3 의 검증자가 사용하기 위해 검증정보를 생성하는 과정이지만, 실제로는 이들을 이용하여 정보 생성과 동시에 검증을 할 수 있다.

5. 향후 연구

본 논문에서는 CEP 프로토콜에서 리더의 질의와 환경에 따른 연산 비용을 계산하였다. 하지만 여기서 사용된 환경이 매우 제한적이기에 여러 개의 CEP 를 요구하는 태그와 동일 제품들이 뒤섞여 있는 환경 등 더욱 일상적인 상황에 대한 모색이 필요하다.

참고문헌

[1] A. Jules, “Yoking-Proofs for RFID Tags,” Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops, 2004, pp.138-143, 2004.
[2] J. Saito, K. Sakurai, “Grouping proof for RFID tags,” 19th International Conference on Advanced Information Networking and Applications, 2005. AINA 2005, Vol.2, pp.621-624, 2005.
[3] S. Piramuthu, “On Existence Proofs for Multiple RFID tags,” 2006, ACS/IEEE International Conference on Pervasive Services, pp.317-320, 2006.