RFID시스템을위한의사랜덤함수TreeWalking알고리즘방식

배효준, 김재홍, 한치문

한국외국어대학교 전자정보공학과

kimjaehong@hufs.ac.kr

Method and Its characteristics of the Pseudo-Random Function Tree Walking Algorithm for Secure RFID system

Bae Hyo Chun, Kim Jae Hong, and Han Chi Moon

Department of Electronics and Information Engineering Hankuk University of Foreign Studies

요 약

본 논문에서는 RFID 시스템은 같은 시간에 많은 제품의 관리를 정확하게 할 수 있는 장점을 제공하고 있다. 하지만, 개인적인 프라이버시 문제가 발생하고 있다. 이러한 프라이버시 문제를 해결하기 위해, 태그 ID의 도청을 막는 Tree Walking 알고리즘 방식들이 연구되고 있다. 본 논문에서는 프라이버시 문제를 해결하기 위해 연구되고 있는 방식들의 문제점들을 파악하고 분석한다. 리더와 태그 사이의 데이터 전송에 사용되는 보안 알고리즘인 Tree Walking 알고리즘 방식에서 반복회수, 보안강도를 비교 한다. 그리고 Randomized Tree Walking 알고리즘 방식에서 가상 ID 의 길이와 Threshold의 최적 값을 찾아서 Traverse알고리즘 실행 시간을 단축시키는 방법을 연구한다. 마지막으로, 리더와 태그 간의 인증에 쓰이는 Randomized PRF Tree Walking 알고리즘의 특성을 분석하고 Randomized Tree Walking 알고리즘 방식과 동작시간, 보안강도에 대하여 비교 확인하였다.

1. 서 론

RFID은 같은 시간에 많은 제품의 관리를 정확하게 할 수 있는 장점을 제공하고 있지만, 상품 ID를 도청하여 상품의 정보를 알아내거나, 소비자가 구매한 상품을 알아내어 어떤 취향을 가지고 있는지 등을 알아 낼 수 있다. 이러한 개인적인 프라이버시 문제와 보안 문제를 해결하기 위해, 리더와 태그 사이의 무선 통신 중 태그 ID의 도청을 방지 할 수 있는 Tree Walking 알고리즘을 이용한 방식들이연구 되고 있다. [1]가장 단순한 형태인 Tree Walking 알고리즘과 특정값(S)를 이용하여 보안 강도를 보완한 Silent Tree Walking 알고리즘, 가상 ID를 사용하여 공격자에게서 상품의 ID를 보호하는 Randomized Tree Walking 알고리즘 그리고 [2]리더와 태그 사이에 인증과정을 추가한 Randomized PRF Tree Walking 알고리즘 방식도 연구되고 있다.

본 논문에서는 이런 알고리즘의 문제점들을 파악 및 분석하고자 한다. Tree Walking 알고리즘을 이용한 방식들에서 각 방식의 리더와 태그 사이의 동작을 검토하고 파라미터들을 이용하여 각 방식들의 특성을 비교하여 장·단점을 파악하고 Randomized Tree Walking 알고리즘 방식에서 가상 ID의 최적 길이와 Threshold의 최적 값을 찾아서 Randomized Tree Walking 알고리즘의 Traverse 알고리즘 실행시간을 단축시키는 방법을 연구한다. 또한 Randomized PRF Tree Walking 알고리즘 방식의 특성을 분석하고 Randomized Tree Walking 알고리즘 방식과 동작시간, 보안강도에 대하여 비교한다.

2. Randomized Tree Walking 알고리즘

2.1 알고리즘 원리

Randomized Tree Walking 알고리즘은 임시적인 Random Number를 생성하여 Tree Walking 알고리즘을 수행하는 동안에 쓰는 방식이다. [3][4] Random Number가 생성되면, 리더는 그림 1 과 같은 Traverse 알고리즘을 통하여 일차적으로 태그를 선택하고, 조건이 만족되면 Tree Walking 알고리즘을 수행한다. 최종적으로 하나의 태그가 선택되면, 태그는 리더에게 자신의 진짜 ID를 전송한다.

```
Traverse (i, count)
bi = read random bit i from all active tags.
if collision on bi detected:
    Suspend all tags with bi == 1.
Each suspended tag stores i.
Traverse (i+1, 0).
    Wake up all tags suspended on bit i.
Traverse (i+1, 0).
else if no collision on bi is detected:
if (count > threshold) tree-walk remaining tags.
else Traverse (i+1, count+1).
```

그림 1. Traverse 알고리즘