RFID 시스템에서 다중 객체를 지원하기 위한 경량화된 SEED 기반의 암호화 알고리즘†

(An Encryption Algorithm Based on Light-Weight SEED for Accessing Multiple Objects in RFID System)

김 지 연*, 정 종 진**
(Ji-Yeon Kim and Jong-Jin Jung)

요 약 최근 RFID 시스템은 응용 범위를 넓혀가고 있으나 무선 통신의 불안전한 특성으로 인한 정보 보안 문제 및 응용 객체별 별도 태그 사용에 의한 불편함 등의 문제점들을 내포하고 있다. 따라서 RFID 기술을 응용한 시스템의 확산을 위해서는 안정적인 정보 보호를 바탕으로 한 여러 응용 객체들 간의 효율적인 정보 공유에 대한 연구가 마련되어야 한다. 본 논문에서는 하나의 태그로 다수 개의 RFID 응용 객체들에 접근할 수 있도록 하는 다중 객체 접근을 위한 RFID 태그 구조를 설계하고, 이러한 태그 내에 저장되는 각 응용 객체에 대한 정보를 보호하기 위한 암호화 알고리즘을 제안한다. 제안하는 알고리즘은 기존의 암호화 기법인 SEED 알고리즘을 RFID 시스템의 특성에 맞도록 경량화한 방법이다. 태그 정보를 보호하기 위한 SEED 변형 알고리즘의 성능은 암 복호화의 속도를 측정하여 기존의 SEED 알고리즘과 비교하여 평가한다.

핵심주제어 : RFID 태그, 다중 객체, 암호화

Abstract Recently, RFID systems are spreading in various industrial areas faster but cause some serious problems of information security because of its unstable wireless communication. Moreover, traditional RFID systems have a restriction that one tag per each application object. This restriction deteriorates their usability because it is difficult to distinguish many tags without some kind of effort. Therefore, efficient information sharing of objects based on information security has to be studied for more spreading of RFID technologies. In this paper, we design a new RFID tag structure for supporting multiple objects which can be shared by many different RFID applications. We also design an encryption/decryption algorithm to protect the identifying information of objects stored in our tag structure. This algorithm is a light revision of the existing SEED algorithm which can be operated in RFID tag environment. To evaluate the performance of our algorithm, we measure the encryption and decryption times of this algorithm and compare the results with those of the original SEED algorithm.

Key Words : RFID tag, Multiple objects, Encryption

1. 서 론

유비쿼터스 환경이 도래하면서 RFID 기술이 적용된 다양한 응용 솔루션으로 개발되고 있다. 또한 최근에는 RFID 하드웨어에 대한 기술 개발과 구축비용이 하락함에 따라 기업들이 RFID 기술을 기존의 비즈니스

† 이 논문은 2010학년도 대진대학교 학술연구비 지원에 의한 것임.
* University of North Texas, 제1저자
** 대진대학교 컴퓨터공학과, 교신저자

들과 결합해 통합 응용시킴으로써 그 가치가 배가되고 있는 상황이다[1,2]. 그러나 현재 RFID 태그에는 하나의 응용 객체에 대한 식별 정보만을 저장하고 있기 때문에 RFID 응용들이 광범위해짐에 따라 각종 물건들에 점점 더 많은 태그를 부착하게 되는데, 이는 RFID 기술의 적용 및 확대에 있어서 저해요소가 될 수 있다. 예를 들어, RFID 기술이 생활 전반에 적용되면 회사의 출입증이나 자동차 키 등과 같이 개인의 생활환경에 필요한 여러 물건들에 각기 다른 목적으로 태그가 부착되고, 사용자는 여러 개의 태그를 소유하게 된다. 그러면 사용자의 입장에서는 점점 더 많은 태그를 소지하게 되어 많은 태그들을 용도에 따라 구별하여 사용한다는 것이 불편하고 어려워질 수 있다. 더구나 RFID는 무선 통신을 이용한 기술이므로 그에 대한 공격이 일반 네트워크 환경에 비해 용이하다. 기존의 바코드 기술에 비해 편리성은 향상되었지만 태그의 정보를 누구든지 항상 읽을 수 있다는 점 때문에 정당하지 않은 사용자에 의한 불법적인 접근이 가능하다. 따라서 많은 태그들에 저장된 개인 정보의 보안 문제를 발생시킬 가능성이 있다[3,4]. 이를 해결하기 위한 기반 기술로서 암호화 알고리즘과 같은 정보보안에 대한 기술 개발이 필수적이다. 그러나 RFID 기술의 상용화를 위해서는 태그의 가격이 낮아질 수밖에 없고, 이로 인해 태그의 IC 또한 적은 수의 게이트를 가질 수밖에 없는 실정이다. 따라서 저가의 태그에서 기존의 DES, AES, SEED와 같은 암호화 알고리즘들의 사용 가능 여부는 여전히 불투명하다. 이를 해결하기 위해서는 낮은 가격에 좋은 성능을 가지는 IC를 개발하는 것도 중요하지만, 자원의 소모가 적으면서도 안전한 암호 알고리즘의 개발이 필수적이다[5].

본 논문에서는 하나의 태그를 이용하여 다수의 RFID 응용 객체에 접근하기 위한 다중 객체 접근 방식의 RFID 시스템에서 객체의 식별 정보를 보호하기 위한 암·복호화 알고리즘을 제안한다. 태그의 식별정보를 암호화하기 위해 기존의 SEED 알고리즘을 RFID 응용 환경에 적합하도록 수정하여 암호화 속도면에서의 성능을 향상시킨 SEED 변형 알고리즘을 설계하였다.

2. 관련 연구

RFID 태그 식별 정보와 같이 고정된 크기의 값에 대한 암호화 방식으로는 블록 암호화 알고리즘이 가장 적합하다. 대표적인 블록 암호화 알고리즘으로는 DES, AES 및 SEED 알고리즘 등이 있다.

DES(Data Encryption Standard) 알고리즘은 56비트 비밀 키 하에서 변환(Permutation)과 치환(Substitution)을 사용하여 64비트의 입력 블록을 수행하는 블록 암호화 알고리즘으로 Feistel 구조로 이루어져 있다[6]. DES 알고리즘에서는 64비트의 평문을 64비트의 암호문으로 만드는데, 64비트의 키를 사용한다. 이 중 56비트는 비밀 키가 되고, 나머지 8비트는 검사용 비트로 사용된다. 또한, 안전성을 증가시키기 위해 키의 길이를 두 배인 128비트를 키로 하는 변형된 알고리즘도 있다. DES 알고리즘은 16라운드의 반복적인 암호화 과정을 가지고 있으며, 각 라운드마다 변환 및 치환의 과정을 거친 평문과 56비트의 비밀 키에서 나온 48비트의 키가 섞여 암호문을 만든다. 복호화는 암호화 과정과 동일하나, 사용하는 키만 역순으로 작용한다. DES 알고리즘에서 수행되는 유일한 산술은 비트 문자열의 XOR이기 때문에, 하드웨어적으로 또는 소프트웨어적으로 매우 효율적으로 수행할 수 있다.

컴퓨터 시스템의 발달에 따른 계산능력 향상으로 DES 알고리즘의 안전성을 보장받을 수 없게 되자, NIST(National Institute of Standard and Technology)에서는 1997년에 이를 대신할 차세대 블록 암호알고리즘(Advanced Encryption Standard, AES)[7]을 공모하였고, Rijmen과 Daemen이 만든 Rijndael 알고리즘을 선정하게 되었다. 많은 후보들 중 AES로 선정된 Rijndael 알고리즘은 안전성, 성능, 구현의 간단함 그리고 유연성의 결합이 장점이다. 특히, 이 알고리즘은 광범위한 컴퓨팅 환경에서 하드웨어와 소프트웨어에서 일정하게 매우 좋은 성능을 보인다. 또한, 키 설정 시간의 우수성과 낮은 메모리 요구는 제한된 환경에서 매우 잘 적응할 수 있기 때문에, 메모리 용량이 극히 제한적인 RFID 태그에 AES 알고리즘을 구현하기 위한 연구가 시도되기도 하였다[8]. AES 알고리즘은 가변 블록 길이와 가변 키 길이를 갖는 반복 구조의 블록 암호 방식이며, 128/192/256비트 크기의 블록과 키를 독립적으로 지정할 수 있다[9]. 라운드 수는 블록과 키 크기에 따라 결정된다. AES 알고리즘의 암호화 과정은 BS(ByteSubstitution), SR(ShiftRow), MC(MixColumn), ARK(AddRoundKey) 함수들의 연산으

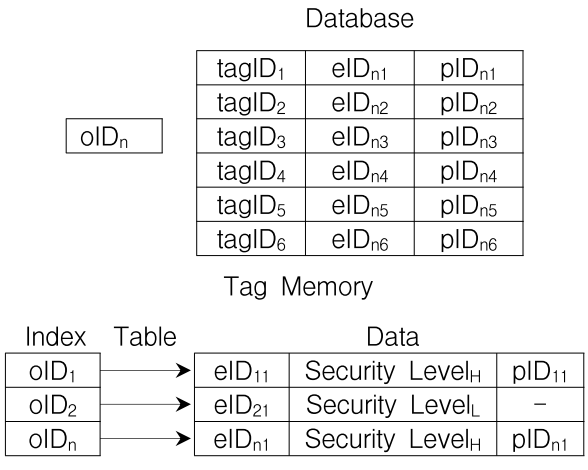
로 구성된다. AES 알고리즘의 복호화 과정은 암호화 연산의 역 연산을 사용하는 non-Feistel 구조를 바탕으로 하고 있다. SEED 알고리즘은 전자상거래, 금융, 무선통신 등의 분야에서 전송되는 개인정보와 같은 중요한 정보를 보호하기 위해 한국정보보호진흥원과 국내 암호전문가들이 순수 국내기술로 개발한 블록 암호알고리즘이다[10]. SEED는 대칭키 블록 암호화 알고리즘으로서, 알고리즘의 전체 구조는 대부분의 블록 암호화 알고리즘과 같이 Feistel 구조이다. SEED 알고리즘에서는 128비트의 평문 블록이 128비트의 키로부터 생성된 64비트의 라운드 키 16개를 입력받아 총 16번의 라운드를 거치면서, 128비트의 암호문 블록을 출력하게 된다. 또한, 전체 알고리즘의 라운드 수는 요구되는 보안 강도와 수행 효율성의 상호 절충적 관계에서 결정된다. 보통 Feistel 구조는 3라운드 이상이며, 짝수 라운드로 구성된다. Feistel 구조의 장점은 라운드 함수에 관계없이 역변환이 가능하며, 알고리즘의 수행속도가 빠르다는 것이다. SEED 알고리즘의 속도는 DES와 비슷하나, 키 생성 알고리즘은 상당히 빠른 편이며 효율적이다. 또한, 안전성과 성능에 대한 다양한 검사와 분석을 통해 SEED 알고리즘이 암호화 과정의 효율성을 지원하면서도 데이터의 안전도를 충분히 지원하고 있음을 증명하고 있다.

3. 다중 객체 기반의 태그 및 암호화 메커니즘

본 논문에서는 RFID 태그의 사용 편의성이라는 목적을 위해 하나의 태그 내에 사용분야와 목적에 따라 정의되는 다수 개의 ID를 가지는 태그 구조와 태그 내에 저장되는 정보의 암호화 메커니즘을 설계하였다. 다음 장에서는 이에 대한 구체적인 설명을 하도록 한다.

3.1 다중 객체 접근을 위한 태그 구조

일반적으로 RFID 시스템은 여러 개의 태그들을 서로 구별하기 위한 고유한 식별자 값으로 *tagID*를 이용한다. RFID 응용 시스템들을 그 용도에 따라 분류하여 객체의 형태로 정의할 수 있는데, 본 논문에서는 각 RFID 응용 시스템 객체에 대해 서로 구분하기 위한 객체 번호로 *oID*(object ID)를 정의한다. 제안하는 다중 객체 접근을 위한 RFID 태그는 이러한 객체를 분류하기 위한 *oID*를 메모리에 저장한다. 또한, *oID*와 *tagID*는 3.2절에서 설명할 암호화 과정에 사용되는데, 암호화 과정은 RFID 서버에서 시스템 설정 과정으로 수행된다. 암호화된 결과값을 본 논문에서는 *eID*(encrypted ID)로 정의하고, 태그의 메모리에 *oID*와 함께 저장한다. *oID*와 *eID*는 [11]에서 제안한 RFID 인증 프로토콜 수행 시 응용 객체별로 각 태그를 구별할 수 있는 식별자로서 이용된다. 특정 리더로부터의 질의에 대해 태그에서는 태그 메모리에 저장되어 있는 *oID*의 값을 이용하여 인덱스 테이블로부터 *eID*를 검색한 후, 인증 프로토콜에 따라 리더에게 적절한 응답을 하게 된다. 제안하는 RFID 태그 구조는 하나의 태그 내에 여러 개의 응용 객체별 식별 정보를 포함하지만, 식별 정보 *eID*를 검색하기 위한 인덱스 구조를 이용하면 리더의 질의에 대한 태그의 응답 시간을 최소화할 수 있으며 효율적인 동작을 가능하게 한다. 그리고 인증 프로토콜의 효율적인 동작을 위해 각 응용 객체의 특성에 따라 보안 레벨(Security Level)을 정하고, 태그에 객체별로 저장한다. 각각의 저장되는 정보에 대한 정의는 [11]에 설명되어 있다. 제안하는 다중 객체 접근을 위한 RFID 서버의 데이터베이스와 태그의 저장 구조는 <그림 1>과 같다.



<그림 1> 제안하는 태그와 서버의 저장 구조

<그림 1>에 나타나는 *pID*(partial ID)는 보안 수준에 따라 다르게 동작하는 인증 프로토콜에서 매 인증 세션마다 응답을 달리하기 위해 태그에서 생성하는

랜덤 값으로, 고수준의 보안을 요구하는 응용 객체의 인증 프로토콜에서만 이용한다. 이러한 태그 구조는 하나의 태그 내에 다수 개의 식별 정보를 저장하도록 함으로써, 여러 종류의 RFID 응용 객체에 접근할 수 있다는 장점이 있다.

3.2 암호화 알고리즘

본 절에서는 다중 객체 접근을 위한 RFID 태그와 서버의 데이터베이스에 저장되는 식별 정보를 암호화하기 위한 암호화 알고리즘에 대해 설명한다. 태그에 부여되는 유일한 식별 정보는 위조가 불가능해야 한다. 그러나 RFID 시스템에서 사용하는 무선 통신의 특성상 태그 정보는 보안에 취약할 수밖에 없으며, 이로 인하여 태그 정보의 유출과 유통 가능성, 위변조 및 오동작과 같은 문제점이 발생하게 된다. 이러한 문제점들을 해결하기 위해 본 논문에서 제안하는 RFID 시스템에서는 태그에 저장되는 데이터의 암호화/복호화를 통한 보안 모듈을 적용한다. 그러나 태그의 물리적인 특성으로 인해 RFID 태그에 적용 가능한 보안 모듈은 저전력과 고속의 성능을 가지도록 해야 한다 [12]. 현재 국내에서 널리 사용되고 있는 SEED 알고리즘은 암호 사용을 촉진하기 위해 개발된 암호화 알고리즘이며, 본 논문에서는 이러한 SEED 알고리즘을 RFID 시스템과 같이 경량화를 유지하여야 하는 응용들에 적용하고자 하는 목적으로 경량화된 SEED 알고리즘을 설계하였다. 이를 위해 키의 크기와 라운드 횟수를 조정하는 방법을 사용하였는데, 키의 크기를 이용한 방식을 K-SEED 알고리즘이라 정의하고, 라운드 횟수를 이용한 방식을 R-SEED 알고리즘이라 정의한다. K-SEED 알고리즘에서는 키의 크기를 확장하여 암호화 과정에 소요되는 전체 블록의 개수를 감소시킴으로써 암호화와 복호화 속도를 향상시키도록 한다. 그리고 R-SEED 알고리즘에서는 암호화/복호화 속도를 향상시키기 위해 라운딩 횟수를 감소시킨다. SEED 변형 알고리즘의 설계 기준으로 사용된 RFID 태그는 읽기/쓰기가 가능한 Class 2 타입의 수동형 태그이며, 메모리 블록의 크기는 128비트로 한다. 암호화 알고리즘의 입력으로는 3.1절에서 정의한 *tagID* 값과 *oID*의 값을 이용한다. 다음은 제안하는 암호화 알고리즘에서 사용되는 표기법이다.

- $X^{<<s}$: X를 s 비트 만큼 왼쪽으로 순환 이동하는 연산
- $X^{>>s}$: X를 s 비트 만큼 오른쪽으로 순환 이동하는 연산
- L_i : i라운드에서 출력된 왼쪽 메시지 블록
- R_i : i라운드에서 출력된 오른쪽 메시지 블록
- $K_i = (K_{i,0}, K_{i,1})$: i라운드의 라운드키
- $K_{i,0}$: i라운드 F 함수의 오른쪽 입력키
- $K_{i,1}$: i라운드의 F 함수의 왼쪽 입력키
- $X = (X_3 \| X_2 \| X_1 \| X_0)$: G 함수의 입력 값
- $Y = (Y_3 \| Y_2 \| Y_1 \| Y_0)$: G 함수에서 S-box(S_1,S_2)의 출력값
- $Z = (Z_3 \| Z_2 \| Z_1 \| Z_0)$: G함수의 출력 값
- m_i : 상수
- KC_i : 라운드 키 생성 과정에서 사용되는 i+1 라운드 상수

3.2.1 키 크기 변경 알고리즘(K-SEED)

암호화 처리 속도를 높이기 위한 목적으로 SEED 알고리즘의 키 크기를 변경하여 RFID 시스템의 특성에 맞게 변형시킨 암호화 방식을 본 논문에서는 K-SEED 알고리즘으로 정의한다.

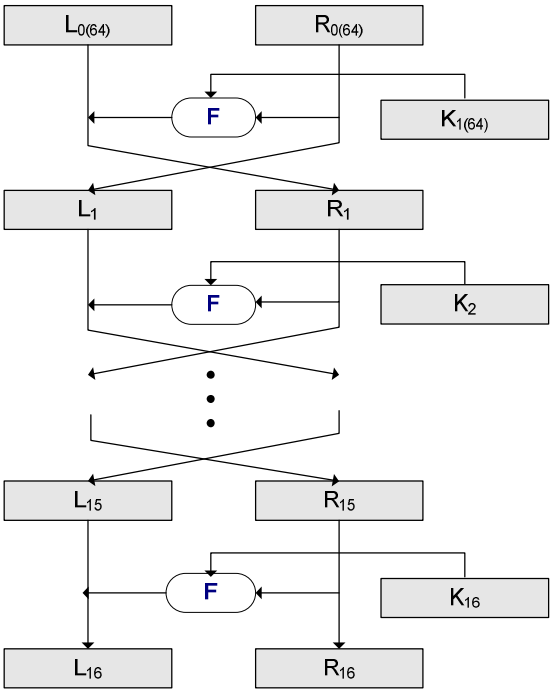
가. K-SEED 알고리즘의 구조

K-SEED 알고리즘의 구조는 <그림 2>로 나타낼 수 있다.

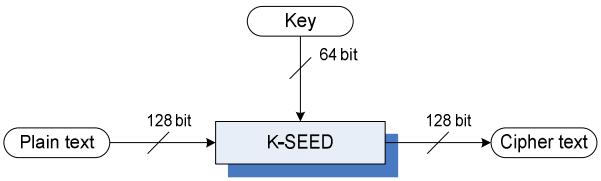
K-SEED 알고리즘의 전체 구조는 Feistel 구조로 이루어져 있으며, 다중 객체 접근 태그의 *tagID*와 *oID*를 각각 평문과 라운드 키로 하여 총 16라운드를 반복 수행하여 암호문 블록을 생성하게 된다. 태그 메모리 블록의 크기가 128비트인 경우 *tagID*는 128비트이고 *oID*는 64비트가 되므로 128비트 평문으로부터 생성된 64비트의 라운드 키를 입력으로 받아 총 16라운드를 거쳐 128비트의 eID 값을 출력하게 된다. 즉, 태그 메모리 블록의 크기가 128비트인 경우, K-SEED 알고리즘은 <그림 3>과 같이 128비트의 태그 정보를 하나의 블록 단위로 인식하고, 64비트 크기의 키를 이용하여 128비트의 암호화 문서를 생성한다.

나. F 함수

Feistel 구조의 블록 암호화 알고리즘은 F 함수의 특성에 따라 구분할 수 있는데, 본 논문에서 제안하는



<그림 2> K-SEED 알고리즘의 구조



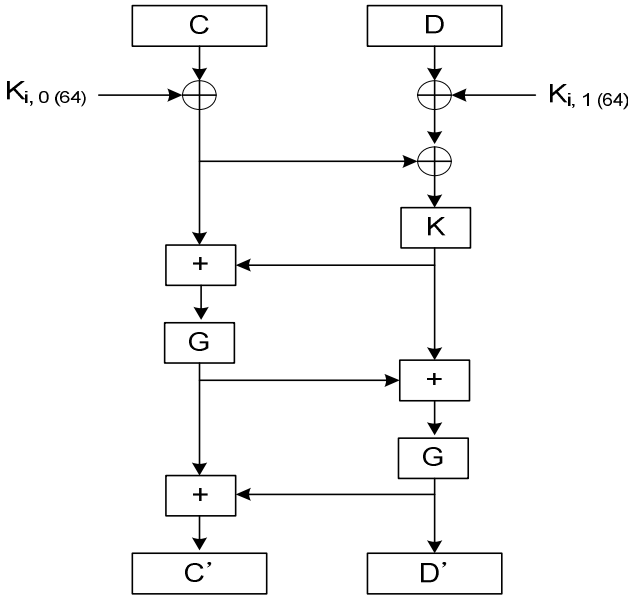
<그림 3> K-SEED 알고리즘의 입출력 블록 구조

K-SEED 알고리즘의 F 함수는 태그 메모리 블록의 크기를 128비트로 기준하였으므로 <그림 4>와 같이 64비트 크기의 블록을 처리하도록 한다.

<그림 4>에서 64비트 크기의 블록 두 개 (C, D)를 입력으로 받아 64비트 크기의 블록 두 개 (C′, D′)를 출력하는데, 암호화 과정에서 C, D와 키 값으로 K_i를 F 함수의 입력으로 처리한다.

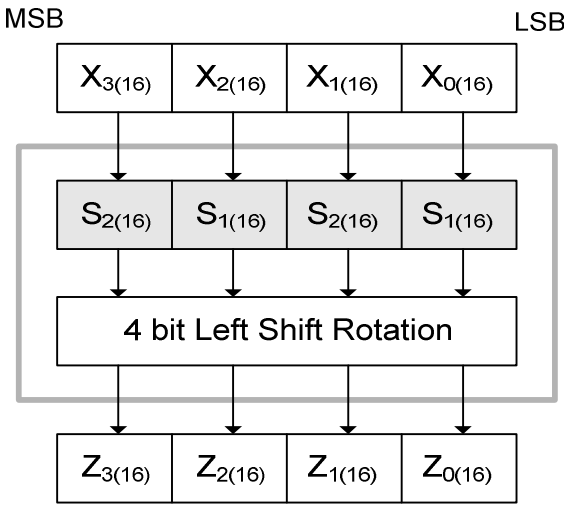
다. G 함수

G 함수는 기존의 SEED 알고리즘에서 사용하는 두 개의 16비트 S-box(S_1,S_2)를 이용하여 입력의 각 비트를 비선형 변환한 후, 그 결과인 64비트를 4비트 왼쪽으로 회전 이동한 후 출력한다. 즉, G 함수의 입력 값(64비트)을 4개의 16비트 블록인 ($X_3||X_2||X_1||X_0$)으로



<그림 4> K-SEED 알고리즘의 F 함수 구조

분할하여 2개의 S-box에 ($S_2||S_1||S_2||S_1$) 순서로 적용시켜 ($Y_3||Y_2||Y_1||Y_0$)를 생성하고 4비트만큼 왼쪽으로 회전 이동한 후, 4개의 16비트 블록인 ($Z_3||Z_2||Z_1||Z_0$)을 생성한다. G 함수의 구조는 <그림 5>와 같다.



<그림 5> K-SEED 알고리즘의 G 함수 구조

라. 키 생성 알고리즘

기존의 SEED 암호화에서 키 생성 알고리즘은 128비트의 암호 키를 64비트씩 좌우로 나누어 이들을 교

대로 16비트씩 좌/우 회전 이동한 후에 생성된 64비트 결과 값에 대한 간단한 산술 연산과 G 함수를 적용하여 라운드 키를 생성하고 있다. 본 논문에서 제안하는 K-SEED 알고리즘에서의 키 생성 알고리즘은 기본적으로 하드웨어나 제한된 자원을 갖는, 즉 RFID와 같은 응용에서의 효율성을 위하여 암호화나 복호화 시 암호 키로부터 필요한 라운드 키를 간단히 계산할 수 있도록 설계한다.

라운드 키를 생성하기 위한 키 스케줄(Key Schedule)은 <그림 6>과 같다. RFID 시스템의 적용과 속도향상을 목적으로 암호화 동작을 경량화시키기 위해서 기존의 SEED 알고리즘에서 16비트 사용자 비밀 키를 확장시킨다. 즉, *UserKey*를 32비트로 변경하고 처리된 중간 라운드의 결과는 **AlgInfo*에 저장한다. 사용자 비밀 키를 확장시킴으로써, 암호화 과정에서 처리되는 전체 블록의 개수를 감소시키는 효과를 가져온다.

```
RET_VAL   SEED_KeySchedule  {
          BYTE              *UserKey,
          DWORD             UserKeyLen,
          SEED_ALG_INFO     *AlgInfo);  }
```

<그림 6> K-SEED의 키 스케줄

3.2.2 라운딩 횟수 조정 알고리즘(R-SEED)

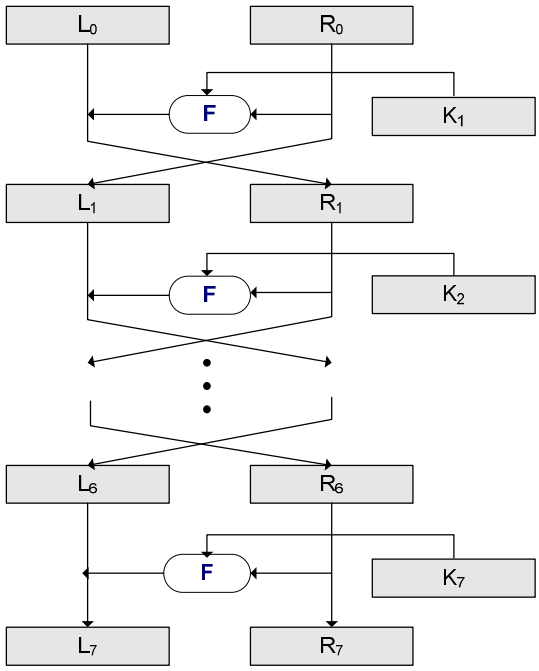
라운딩 횟수 조정에 의해 암호 키를 생성할 경우, 기존 SEED 알고리즘의 라운드 처리 횟수를 줄임으로써, 라운드 당 처리시간을 감소시킬 수 있다. 본 논문에서는 이러한 방법을 R-SEED 알고리즘으로 정의하고, 다음과 같이 설계한다.

R-SEED 알고리즘은 Feistel 구조로 이루어지며, t 비트인 L_0, R_0 블록으로 이루어진 $2t$비트 크기의 평문 블록을 8라운드를 거쳐 암호문 L_r, R_r을 생성해 내는 반복 구조이다. 반복 구조란 평문 블록이 몇 번의 라운드를 거쳐 암호화를 수행하는 것을 말하고, 라운드 $i(1\le i\le r)$란 암호키 K로부터 유도된 각 서브키인 K_i를 주요 입력으로 하는 $L_i=R_{i-1}$, $R_i=L_{i-1}\bigcirc f(R_{i-1},K_i)$를 통해 $(L_{i-1},R_{i-1})->(L_i,R_i)$로 변환해주는 함수를 의미한다.

기존의 SEED 알고리즘에서 처리되는 라운드의 조건은 키 전수조사 공격에 필요한 계산복잡도 및 평문과 암호문 쌍의 크기가 2^{128}비트 이하가 되지 않아야 하며, 효율성 요구조건을 만족하여야 한다. 그리고 키 생성 알고리즘은 2라운드마다 일정한 규칙으로 서브키를 생성하며, 서브키를 생성하기 위한 입력 값은 (8+i)라운드의 값과 i라운드에서의 값이 동일하다. 즉, 1라운드와 9라운드에서 동일한 입력 값을 이용하여 키를 생성하게 된다. 또한, 서브키 간의 관계를 이용하여 암호를 공격하는 Related Key 공격의 경우 5라운드 이상이 되면 공격이 거의 불가능하다[10]. 따라서 기존의 SEED 알고리즘에서 처리되는 16라운드를 8라운드로 감소시켜 RFID 시스템에 적용하게 되더라도 암호 공격으로부터 비교적 안전한 성능을 보장하면서 암호화 속도를 향상시킬 수 있다.

가. R-SEED 알고리즘의 구조

R-SEED 알고리즘은 <그림 7>과 같이 128비트 단위의 평문 블록 당 128비트 크기의 키로부터 생성된 8개의 64비트 라운드 키를 입력받아 총 8라운드를 거쳐 128비트 크기의 암호문 블록을 출력한다.



<그림 7> R-SEED 알고리즘의 구조

나. F 함수

R-SEED 알고리즘의 F 함수는 128비트 크기를 단위로 하는 Feistel 암호 알고리즘으로 구성된다. F 함

수는 두 개의 64비트 크기의 블록을 입력받아 두 개의 64비트 크기의 블록을 출력한다. 즉, 암호화 과정에서 64비트의 블록(C, D)와 64비트의 키 K_i($K_i = K_{i,0}, K_{i,1}$)를 F 함수의 입력으로 하고, 처리한 결과 (C', D')를 출력한다.

다. G 함수

G 함수는 기존의 SEED 알고리즘과 같이 두 개의 16비트 S-box를 이용하여 입력의 각 비트를 비선형 변환한 후, 그 결과인 32비트를 4비트 왼쪽으로 회전 이동한 후 출력한다. 즉, G 함수의 입력 값(32비트)을 4개의 16비트 블록인 ($X_3||X_2||X_1||X_0$)으로 분할하여 2개의 S-box에 ($S_2||S_1||S_2||S_1$) 순서로 적용시켜 ($Y_3||Y_2||Y_1||Y_0$)를 생성하고 8비트만큼 왼쪽으로 회전 이동한 후, 4개의 16비트 블록인 ($Z_3||Z_2||Z_1||Z_0$)을 생성한다.

라. 키 생성 알고리즘

R-SEED 알고리즘에서의 키 생성 알고리즘은 64비트의 암호화 키를 32비트씩 좌우로 나누어 이들을 교대로 16비트씩 좌/우로 회전 이동한 후, 그 결과인 32비트에 대해 간단한 산술 연산과 G 함수를 적용하여 <그림 8>과 같이 기존의 16라운드 키를 8라운드로 조정하여 생성한다.

```
for   (i = 1; i = 8; i++) {
      K_{i,0} ← G(A + C − KC_{i-1})
      K_{i,1} ← G(B − D + KC_{i-1})
      if (i % 2 == 1)
          A||I ← (A||B)^{>>8}
      else C||I ← (C||D)^{<<8} }
```

<그림 8> R-SEED의 키 생성 알고리즘

4. 실험 및 분석

본 장에서는 3장에서 제안한 경량화된 SEED 암·복호화 알고리즘에 대한 실험 방법 및 결과에 대해 설명한다. 그리고 실험 결과에 기반을 두어 본 논문에서 제안하고 있는 암호화 알고리즘의 특징과 성능에 대해 분석한다.

SEED 알고리즘의 키 크기를 변형시킨 K-SEED 알고리즘과 라운드 횟수를 변경한 R-SEED 알고리즘의 성능을 평가하기 위해, 두 알고리즘에 대해 각각 암·복호화 처리 속도를 측정하였다. 또한, 기존의 SEED 알고리즘에 대해서도 암·복호화 처리 속도를 함께 측정하여 그 결과를 비교, 분석한다. 각 암호화 방식에 따라 실험한 결과는 <표 1>과 같으며, 실험에 사용된 컴퓨터는 Pentium IV 2.80, 메모리 크기는 2GB이다. 암호화와 복호화 처리 속도의 평균값은 최곳값과 최젓값을 뺀 나머지 값들을 대상으로 한 결과이다.

<표 1> 암호화와 복호화 처리 속도

(E: 암호화, D: 복호화, 단위: Mbps)

No	SEED (E / D)	K-SEED (E / D)	R-SEED (E / D)
1	17.220 / 17.517	22.002 / 23.532	30.103 / 31.894
2	17.484 / 16.279	21.261 / 21.669	30.163 / 31.059
3	17.249 / 16.987	22.947 / 23.418	30.280 / 31.738
4	16.961 / 17.364	22.489 / 21.486	31.102 / 32.954
5	17.278 / 17.249	22.539 / 23.098	31.281 / 32.738
6	17.305 / 17.133	22.590 / 23.209	31.280 / 33.151
7	16.848 / 15.971	21.955 / 23.263	30.720 / 31.280
8	16.357 / 17.334	22.534 / 23.209	32.079 / 33.312
9	17.308 / 17.305	22.685 / 23.263	31.662 / 32.687
10	17.018 / 17.397	21.767 / 22.890	28.403 / 30.281
11	17.264 / 16.357	22.539 / 22.947	31.862 / 32.687
12	17.397 / 17.364	21.440 / 23.046	31.079 / 31.894
평균	**17.185 / 17.077**	**22.254 / 23.001**	**30.953 / 32.208**

<표 1>의 실험 결과를 바탕으로 기존의 SEED 알고리즘에 대해 제안한 K-SEED 알고리즘의 암호화와 복호화의 속도를 비교하여 성능을 계산하면 다음과 같다.

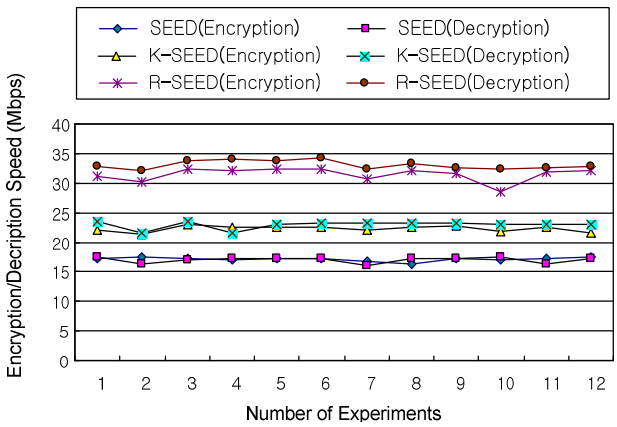
$$EncryptionPerformance = (K\text{-}SEED - SEED)/SEED*100$$
$$= (22.254 - 17.185)/17.185*100$$
$$= 29.496(\%)$$
$$DecryptionPerformance = (K\text{-}SEED - SEED)/SEED*100$$
$$= (23.001 - 17.077)/17.077*100$$
$$= 34.689(\%)$$

즉, K-SEED 알고리즘을 적용하여 암호화를 수행하는 경우, 기존의 SEED 알고리즘보다 29% 정도의 성능향상을 보여주고 있으며, 복호화의 경우에도 약 34%의 성능향상을 확인할 수 있다.

같은 방법으로 기존의 SEED에 대해 제안한 R-SEED 알고리즘의 암호화와 복호화의 성능을 비교 계산하면 다음과 같다.

$$EncryptionPerformance = (R\text{-}SEED - SEED)/SEED*100$$
$$= (30.953 - 17.185)/17.185*100$$
$$= 80.116(\%)$$
$$DecryptionPerformance = (R\text{-}SEED - SEED)/SEED*100$$
$$= (32.208 - 17.077)/17.077*100$$
$$= 88.604(\%)$$

계산 결과, R-SEED 알고리즘의 경우 암호화와 복호화의 속도는 기존의 SEED 알고리즘에 비해 80% 이상의 실험 결과들을 볼 때, K-SEED 및 R-SEED 알고리즘의 암·복호화 처리속도가 기존의 SEED 알고리즘에 비해 일정 수준의 향상을 가져온 것은 RFID 시스템에서의 태그정보를 보호하는데 SEED 알고리즘이 응용될 수 있음을 보여주는 것이다. 기존의 SEED 알고리즘은 네트워크 보안 분야의 표준으로 자리 잡을 만큼 견고함과 안정성이 입증된 알고리즘이지만 연산량이 많고 복잡하여 RFID 시스템에 적용하기에는 무리한 측면이 있다. 그러나 K-SEED 및 R-SEED 알고리즘과 같이 SEED 알고리즘을 경량화 시킬 경우 의미 있는 수준의 보안성은 확보되면서 연산 속도가 개선됨으로써 RFID 시스템과 같이 저사양의 하드웨어 환경 하에서 저수준의 보안 요구가 있는 분야에 충분히 적용될 수 있다.



<그림 9> 암·복호화 성능 비교

5. 결 론

본 논문에서는 다중 객체를 지원할 수 있는 RFID 태그 구조를 제안하였다. 이는 사용자가 RFID 응용별로 별도의 태그들을 소유하지 않고 하나의 태그 내에 인증 가능한 정보를 통합 관리할 수 있도록 한 것이다. 또한 이러한 다중 객체를 지원하는 태그정보를 보호하기 위한 암호화 알고리즘을 제안하였다. 제안한 알고리즘은 기존의 대표적 암호화 방식인 SEED 알고리즘을 수정하여 RFID 환경에 맞게 경량화한 것이다. SEED 알고리즘은 암호화에 소요되는 연산량과 속도 문제로 인해 RFID 시스템에 그대로 적용할 수 없으므로 키 크기와 라운드 횟수를 변형시켜 태그 식별 정보를 암호화하도록 하였다. 결과적으로 태그 정보의 보안 수준에 맞게 암호화하면서 암호화 단계에 필요한 시간을 감소시키는 효과를 얻을 수 있었다.

본 논문에서 제안한 K-SEED 및 R-SEED 알고리즘을 포함한 암호화 방식은 암호화 속도의 성능향상에 그 중요성을 두었으므로 암호화된 결과 값에 대한 안전성 문제가 세밀히 고려되지 않았다. 따라서 향후 연구에서는 제안한 알고리즘들에 대해 안전성 분석 및 연구가 추가되어야 한다.

참 고 문 헌

[1] G. Avoine, and P. Oechslin, “RFID Traceability: A Multilayer Problem,” EPFL, 2004.

[2] K. Finkenzeller, “RFID Handbook,” John Wiley & Sons, 1999.

[3] S. E. Sarma, S. A. Weis, and D. W. Engels,

"RFID Systems and Security and Privacy Implications", Workshop on Cryptographic Hardware and Embedded Systems, 2002.
[4] S. S. Yeo and S. K. Kim, "Scalable and Flexible Privacy Protection Scheme for RFID System," Proc. the 2nd European Workshop on Security and Privacy in Adhoc and Sensor Networks (ESAS2005), LNCS 3813, pp.153-163, 2005.
[5] S. E. Sarma, "Towards the Fivecent tag," MIT AutoID Center, 2002.
[6] A. Juels, and R. Pappu, "Squealing Euros: Privacy Protetction in RFID-Enabled Banknotes", FC'03, LNCS 2742, pp.103-121, 2003.
[7] A. J. Elbirt, W. Yip, B. Chetwynd, and C. Parr, "An FPGA-based Performance Evaluation of the AES Block Cipher Candidate Algorithm Finalists," IEEE Transactions on Very Large Scale Integration System, vol. 9, no. 4, 2001.
[8] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer, "Strong Authentication for RFID Systems Using the AES Algorithm, "Workshop on Cryptographic Hardware and Embedded Systems, LNCS 3156, pp.357-370, 2004.
[9] 최병윤, "AES Rijndael 암호 프로세서의 설계," 한국통신학회논문지, vol. 26, no. 10B, pp.1491-1500, 2001.
[10] 한국정보보호진흥원, "128비트 블록 암호 알고리즘(SEED) 개발 및 분석 보고서," 2003.
[11] 김지연, 정종진, 조근식, 이균하, 다중 객체 지원을 위한 RFID 시스템에서 보안 레벨 기반의 인증 기법에 관한 연구, 한국전자거래학회논문지, 13권 1호, pp.22-32, 2008. 2.
[12] H. Y. Park, B. Y. Sohn, and Y. T. Shin, "Safe Authentication Method for Security Communication in Ubiquitous Environment," IICSA 2005 LNCS, pp.442-448, 2005.

김 지 연 (Ji-Yeon Kim)

- 1992년 2월 : 인하대학교 전자계산공학과 (공학학사)
- 1997년 2월 : 인하대학교 전자계산공학과 (공학석사)
- 2008년 2월 : 인하대학교 전자계산공학과 (공학박사)
- 1997년 1월 ~ 2001년 2월 : LG정보통신 이동교환실 주임연구원
- 2001년 3월 ~ 2005년 8월 : 청강문화산업대학 인터넷비즈니스과 조교수
- 2006년 3월 ~ 2009년 8월 : 대진대학교 컴퓨터공학과 초빙교수
- 2009년 9월 ~ 현재 : University of North Texas 연구원
- 관심분야 : 이동통신, RFID/센서 네트워크, 정보보안

정 종 진 (Jong-Jin Jung)

- 1992년 2월 : 인하대학교 전자계산공학과 (공학학사)
- 1995년 2월 : 인하대학교 전자계산공학과 (공학석사)
- 2000년 2월 : 인하대학교 전자계산공학과 (공학박사)
- 1998년 3월 ~ 2002년 8월 : 경문대학 인터넷미디어정보과 조교수
- 2002년 9월 ~ 현재 : 대진대학교 컴퓨터공학과 부교수
- 관심분야 : 전문가시스템, 지능형 에이전트, RFID 시스템, 정보보안

논 문 접 수 일 : 2010년 08월 12일
1차수정완료일 : 2010년 09월 01일
게 재 확 정 일 : 2010년 09월 03일