확장 가능한 비대칭 피스텔 네트웍의 제안 및 유사 랜덤 순열 증명

(Proposal of Extendable Unbalanced Feistel Network and Proof of Pseudorandom Permutation)

이 광 수 † 신 준 범 †† 이 광 형 †††

(Kwang Su Lee) (Jun Bum Shin) (Hyung Lee-Kwang)

요 약 정보 처리량이 증가함에 따라 한번에 많은 양의 평문을 암호화 할 수 있는 입출력이 큰 블록 암호기의 필요성이 대두되고 있다. 하지만 입출력이 큰 블록 암호기를 직접 구현하는 것은 많은 비용이 든다. 따라서 이 논문에서는 기존에 존재하는 블록 암호기를 이용해서 입출력이 큰 블록 암호기를 구현할 수 있는 방법을 제안한다. 그런 뒤 새로 제안된 비대칭 피스텔 네트웍 구조가 안전한 블록 암호기가 되기 위한 조건을 분석한다. 논문의 결과는 다음과 같다.

○확장 가능한 비대칭 피스텔 네트웍이 입력과 출력의 크기가 모두 n 비트인 유사 랜덤 함수 생성기를 사용하는 경우, k가 홀수이고 전체 라운드 수가 2k+1이상이면 유사 랜덤 순열 생성기이다.

Abstract *Because the amount of information processing is increasing, block ciphers with large input-output length are needed. But it is expensive to construct a block cipher with large input-output length. Thus in this paper, we provide a construction of unbalanced feistel network with large input-output length using the original network with small input-output length. And we analyze the condition of pseudorandom permutation generator based on extendable unbalanced feistel networks. The results of this paper is the following:*

○2k+1 rounds extendable unbalanced feistel network using pseudorandom functions from n bits to n bits is a pseudorandom permutation generator.

1. 서 론

블록 암호기(block cipher)는 블록 단위로 평문을 암호화하는 비밀키 암호 시스템(private-key cryptosystem)이다 [1]. 그리고 블록 암호기는 일대일 대응함수(bijective function) 성질을 가진다. 또한 대부분의 암호 시스템에서 블록 암호기는 가장 핵심적인 역할을 수행하기 때문에 안전한 암호 시스템을 구현하기 위해서는 안전한 블록 암호기를 구현해야 한다. 안전한 블록 암호기(secure block cipher)란 출력값이 랜덤한 블록 암호기이다. 즉, 블록 암호기가 유사 랜덤 순열 생성기(pseudorandom permutation generator)가 되면 안전한 블록 암호기가 된다. 하지만 일반적으로 블록 암호기를 구현할 때 일대일 대응함수 성질과 유사 랜덤성을 동시에 만족하도록 구현하는 것이 쉽지 않다. 이런 문제점을 해결할 수 있는 방법이 바로 피스텔 네트웍이다.

피스텔 네트웍(feistel network)은 임의의 함수를 일대일 대응함수로 변환해 주는 방법이다 [2,3]. 따라서 블록 암호기를 구현할 때 피스텔 네트웍 구조를 사용하면 유사 랜덤 함수를 구현하는 것으로 안전한 블록 암호기를 구현할 수 있다. 왜냐하면 피스텔 네트웍이 자동으로 유사 랜덤 함수를 일대일 대응함수로 변환 시켜주기 때문이다. 따라서 이와 같은 피스텔 네트웍의 성질의 장점으로 인해서 대부분의 블록 암호기가 피스텔 네트웍 구조를 이용해서 구현되었다 [1].

한편 컴퓨터의 정보 처리량이 증가함에 따라 많은 양

· 본 연구는 해킹바이러스 연구센터의 지원을 받았음.
† 비 회 원 : 미래산업(주) 소프트포럼 연구원
guspin@monami.kaist.ac.kr
†† 비 회 원 : 한국과학기술원 전자전산학과
jbshin@monami.kaist.ac.kr
††† 종신회원 : 한국과학기술원 전자전산학과 교수
khlee@monami.kaist.ac.kr
논문접수 : 2000년 3월 2일
심사완료 : 2000년 10월 18일

의 평문을 암호화할 수 있는 입출력이 큰 블록 암호기가 필요하게 되었다. 차세대 암호 표준(Advanced Encryption Standard)에서도 기존 64비트 블록 암호기 대신 128비트 블록 암호기를 요구하고 있다 [4]. 하지만 입출력이 큰 새로운 블록 암호기를 구현하기 위해서는 새로운 암호기에 대한 많은 분석이 이루어져야 한다. 또한 일반적으로 입출력이 큰 블록 암호기를 직접 구현하는 것은 많은 비용이 든다. 따라서 기존에 존재하는 블록 암호기를 이용해서 입출력이 큰 블록 암호기를 구현하는 방법이 필요하게 된다. 이를 해결하는 방법중의 하나는 DES 암호기의 사용해서 제안된 ECB, CBC, CFB, 그리고 OFB 방법이다 [5,6]. 하지만 이들 방법은 결코 안전한 블록 암호기인 유사 랜덤 순열이 되지 않는다. Naor와 Reingold는 DES의 ECB방법과 쌍으로 독립인 순열(pairwise independent permutation)을 사용해서 기존에 존재하는 블록 암호기를 이용해서 입출력이 큰 블록 암호기를 구현할 수 있음을 보였다 [7]. 하지만 이 방법은 기존에 존재하는 블록 암호기뿐만 아니라 쌍으로 독립인 순열을 필요로 하는 단점이 있다.

따라서 이 논문에서는 기존에 존재하는 블록 암호기를 사용해서 입출력이 더욱 큰 블록 암호기를 쉽게 구현할 수 있는 비대칭 피스텔 네트웍 구조를 제안한다. 그리고 이 비대칭 피스텔 네트웍 구조가 안전한 블록 암호기인 유사 랜덤 순열 생성기가 되기 위한 조건을 분석한다.

먼저 2장에서는 피스텔 네트웍과 유사 랜덤을 정의하고 기존연구를 정리한다. 그런 뒤 3장에서는 확장이 가능한 새로운 비대칭 피스텔 네트웍 n:kn-UFN2 구조를 제안하고, 이 구조가 안전한 블록 암호기인 유사 랜덤 순열 생성기가 되기 위한 조건을 분석한다. 4장에서는 결론을 맺고 마치도록 한다.

2. 관련 연구

이 절에서는 피스텔 네트웍과 유사 랜덤의 개념을 정의하고 기존 연구를 정리하도록 한다.

2.1 기호

이 논문에서 사용되는 기호는 다음과 같다.

- I_n은 모든 n비트 스트링의 집합을 나타낸다. 즉, $\{0,1\}^n$.
- $F:I_s \mapsto I_t$는 입력이 s비트이고 출력이 t비트인 모든 함수의 집합을 나타낸다.
- F_n은 입력과 출력이 n비트인 함수의 집합을 나타낸다 ($F:I_n \mapsto I_n$).
- P_n은 입력과 출력이 n비트인 순열의 집합을 나타낸다 ($P_n \subset F_n$).
- $|x|$는 비트 스트링 x의 크기를 나타낸다. 즉, x가 n비트인 경우 $|x|$는 n이다.
- $x \oplus y$는 x와 y가 동일한 크기의 비트 스트링일 때 x와 y의 비트 단위당 xor이다.
- $x \| y$는 두 비트 스트링 x와 y의 연결을 나타낸다. 즉, $|x \| y| = |x| + |y|$.
- $f \circ g$는 두 함수 f와 g가 함수의 집합 F_n의 원소일 때 두 함수의 합성을 나타낸다. 즉, $f \circ g(x) = f(g(x))$.

2.2 피스텔 네트웍

피스텔 네트웍은 H. Feistel이 Lucifer를 설계할 때 고안한 방법으로 블록 암호기를 구현하는데 가장 많이 사용된다 [2,3]. 피스텔 네트웍의 가장 큰 장점은 임의의 함수를 일대일대응 함수인 순열이 되도록 변환해 준다는 것이다.

정의 2.2.1 (피스텔 네트웍). 집합 $F:I_s \mapsto I_t$에 속하는 임의의 함수 f에 대해서, 1라운드 피스텔 네트웍은 함수 $D_f(L \| R) := (R \| (L \oplus f(R)))$로 정의가 된다. 이와 마찬가지로 집합 $F:I_s \mapsto I_t$에 속하는 함수 $f_1, f_2, \ldots, f_r$에 대해서, r라운드 피스텔 네트웍은 함수 $D_{fk} \circ \cdots \circ D_{f2} \circ D_{f1}(L \| R) := D_{fk} \circ \cdots \circ D_{f2}(D_{f1}(L \| R))$로 정의가 된다. 이때 $|L|=t$, $|R|=s$, $|L|+|R|=2n$이고 함수 D_f는 집합 P_{2n}의 원소이다.

정의 2.2.1에서 L과 R의 비트 크기가 동일하면 ($|L|=|R|=n$) 대칭 피스텔 네트웍(balanced feistel network)이라 하고, L과 R의 비트 크기가 다르면 비대칭 피스텔 네트웍(unbalanced feistel network)이라고 한다 [8]. 이때 f함수의 입력이 되는 블록 R을 소스-블록이라 하고 f함수의 출력값과 xor되는 블록 L을 타겟-블록이라고 한다. 소스-블록의 크기가 s이고 타겟-블록의 크기가 t인 비대칭 피스텔 네트웍을 s:t-UFN이라고 표기한다. 그림 1은 1라운드 대칭 피스텔 네트웍과 1라운드 비대칭 피스텔 네트웍을 나타낸다.

비대칭 피스텔 네트웍은 소스-블록이 타겟-블록보다 큰 소스-헤비 비대칭 피스텔 네트웍과 타겟-블록이 소

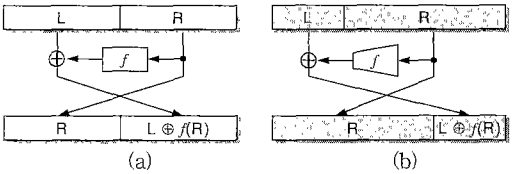

그림 1 피스텔 네트웍 : (a) 1라운드 대칭 피스텔 네트웍, (b) 1라운드 비대칭 피스텔 네트웍

스-블록보다 큰 타겟-해비 비대칭 피스텔 네트웍으로 구분된다 [8].

2.3 유사 랜덤

유사 랜덤을 정의하기 전에 먼저 두 사건이 계산론적으로 동일하다는 것이 무엇인지를 알아보자. 두 사건이 계산론적으로 동일하다는 것(computational equivalence)은 어떤 효과적인 알고리즘도 이 두 사건이 서로 다르다고 판단하지 못하는 것을 의미한다. 따라서 유사 랜덤이란 이상적인 랜덤 사건과 계산론적으로 구분이 불가능한 사건을 의미한다. 이때 두 사건의 동일성 여부를 판단하는 효과적인 알고리즘은 다음과 같은 오라클 기계(oracle machine)로 정의된다 [9].

정의 2.3.1 (오라클 기계 M) 오라클 기계 M은 오라클-테이프를 가진 튜링 머신(Turing machine)이다. 이 오라클 기계의 입력값은 1^n이고 출력값은 1비트이다. 함수 f에 접근할 수 있는 오라클 기계는 $M^f(1^n)$으로 표기되고 다음과 같이 동작한다. 먼저 오라클 기계는 n비트 길이의 오라클 질의 x_1을 오라클-테이프에 쓴다. 그 다음 상태에서 오라클 회신 $y_1 = f(x_1)$를 오라클-테이프에서 얻는다. 이런 과정을 m번 반복한다. 이렇게 얻은 오라클 질의와 회신 $\langle x_1, y_1\rangle$, $\langle x_2, y_2\rangle$, ..., $\langle x_m, y_m\rangle$을 사용해서 오라클의 1비트 출력값을 계산한다.

이때 두 사건의 동일성을 판단하는 효과적인 알고리즘은 확률론적 다항시간(probabilistic polynomial-time) 안에 출력값을 계산해 주는 오라클 기계를 의미한다. 계산론적 구분불능은 다음과 같은 다항시간 구분불능으로 정의된다 [9].

정의 2.3.2 (다항시간 구분불능) 두 사건 X와 Y가 다항-시간에 구분이 불가능하다는 것은 모든 확률론적 다항시간 알고리즘 D와 모든 다항식 $p(\cdot)$, 그리고 충분히 큰 n값에 대해서 다음 식을 만족하는 경우이다.

$$|Pr(D(X, 1^n)=1) - Pr(D(Y, 1^n)=1)| < 1/p(n).$$

유사 랜덤 순열 생성기는 Luby-Rackoff에 의해서 소개되었다 [10]. 그 뒤 유사 랜덤 순열 생성기에 대한 많은 연구가 있었다 [1,12,13,14,15].

정의 2.3.5 (유사 랜덤 순열 생성기) 유사 랜덤 순열 생성기는 순열의 집합을 생성하는 알고리즘 P로 정의된다. 그리고 모든 확률적 다항시간 오라클 기계 M과, 모든 다항식 $p(\cdot)$ 그리고 충분히 큰 n값에 대해서 다음 식을 만족한다.

$$|Pr(M^P(1^n)=1) - Pr(M^K(1^n)=1)| < 1/p(n).$$

이때 K는 순열을 균일한 확률분포로 생성하는 이상적인 랜덤 순열 생성기이다.

2.4 기존 블록 암호기의 확장

기존 블록 암호기를 이용해서 입출력이 더욱 큰 블록 암호기를 구현하는 것에 관련된 연구는 아직 많지 않다. 가장 대표적인 방법은 DES mode of operation과 Naor와 Reingold가 제안한 방법이다.

DES의 경우 큰 메시지를 암호화하기 위해서 ECB, CBC, CFB, 그리고 OFB 방법이 제안되었다 [6]. 하지만 이들 방법은 절대로 안전한 블록 암호기인 유사 랜덤 순열이 되지 않는다. 왜냐하면 이들 방법의 경우 모든 암호문의 값들이 평문의 첫 번째 몇 블록에 의해서 결정이 되기 때문이다 [7]. 즉, 평문의 첫 번째 몇 블록에 블록 암호기의 안전성을 의존하는 결과를 가져오기 때문에 안전하지 않게 된다.

Naor와 Reingold는 ECB 방법에 아이디어를 얻어서 기존 블록 암호기를 입출력이 더욱 큰 블록 암호기로 확장할 수 있는 방법을 제안했다 [7]. 이 방법은 먼저 큰 평문을 짝으로 독립인 순열을 이용해서 서로 독립인 블록들을 생성한 뒤, 이 블록들을 2 라운드 대칭 피스텔 네트웍에 통과시키는 방법이다. 즉, $D^{\times b}{}_f(x_1, x_2, ..., x_b) := (D_f(x_1), D_f(x_2), ..., D_f(x_b))$이고 $D_f(L \| R) := (R \| (L \oplus f(R)))$라고 정의할 때, 블록 암호기의 출력값은 $D^{\times b}{}_{f2} \circ D^{\times b}{}_{f1} \circ h(x_1, x_2, ..., x_b) = (y_1, y_2, ..., y_b)$이다. 이때 h는 짝으로 독립인 순열이고 L 블록과 R 블록의 크기는 동일하다. 하지만 이 방법의 경우 기존 블록 암호기뿐만 아니라 짝으로 독립인 순열을 필요로 하는 단점이 있다.

3. 확장 가능한 비대칭 피스텔 네트웍 구조

이 절에서는 타겟-해비 비대칭 피스텔 네트웍을 변형해서 확장 가능한 비대칭 피스텔 네트웍 구조 n:kn-UFN2를 제안한다. 그리고 이 구조가 유사 랜덤 순열 생성기가 되기 위한 조건을 분석한다.

3.1 n:kn-UFN2 구조

n:kn-UFN2 구조는 타겟-해비 비대칭 피스텔 네트웍과 유사한 구조이다. 단지 유사 랜덤 함수 생성기로 $F:I_n \mapsto I_{kn}$을 사용하는 대신 $F:I_n \mapsto I_n$을 사용하는 점이 다르다. 입출력 크기가 동일한 유사 랜덤 함수를 사용하므로 기존에 존재하는 블록 암호기를 이용해서 입출력이 더욱 큰 블록 암호기로 확장이 가능한 구조이다. 즉, 기존에 존재하는 블록 암호기를 피스텔 네트웍의 라운드 함수대신 사용하면 입출력이 더욱 큰 블록 암호기를 쉽게 구현할 수 있다. 그림 2는 입출력의 크기가 n비트인 함수를 사용하는 3라운드 비대칭 피스텔 네트웍이다.

정의 3.1.1 (n:kn-UFN2) 집합 $F:I_n \mapsto I_n$에 속한 임의의 함수 f에 대해서 1라운드 n:kn-UFN2는 함수 $D_f(L_1$

$\| \ldots \| L_k \| R) \coloneqq (R \| L_1 \oplus f(R) \| \ldots \| L_k \oplus f(R))$로 정의된다. 이와 마찬가지로 집합 $F:I_n \mapsto I_n$에 속한 임의의 함수들 $f_1, f_2, \ldots, f_r$에 대해서 r라운드 n:kn-UFN2는 함수 $D_{fr} \circ \cdots \circ D_{f2} \circ D_{f1}(L_1^0 \| \ldots \| L_k^0 \| R^0) \coloneqq D_{fr} \circ \cdots \circ D_{f2}(D_{f1}(L_1^0 \| \ldots \| L_k^0 \| R^0))$로 정의가 된다. 이때 $|L_i|=|R|=n$ 이다.

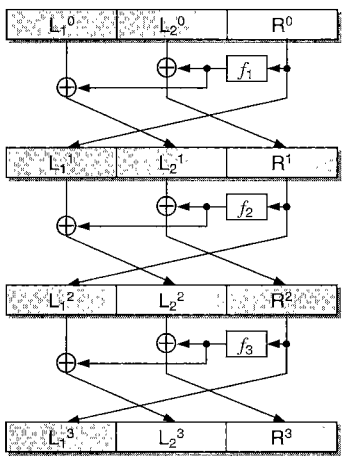

그림 2 3라운드 n:2n-UFN2

3.2 유사 랜덤 순열의 증명

이 절에서는 유사 랜덤 함수들을 사용하는 n:kn-UFN2 구조가 유사 랜덤 순열 생성기가 되기 위한 조건을 분석한다. 유사 랜덤 순열 생성기임을 증명하는데 사용된 방법은 Luby-Rackoff가 대칭 피스텔 네트웍에서 사용한 방법을 비대칭 피스텔 네트웍으로 확장한 것이다 [7,10].

정리 3.2.1 k가 짝수인 경우, n:kn-UFN2는 유사 랜덤이 아니다.

증명 k가 짝수인 경우 xor 연산의 성질에 의해서 r라운드 n:kn-UFN2의 입력값과 출력값 사이에 다음과 같은 관계식이 성립한다.

$$L_1^r \oplus L_2^r \oplus \cdots \oplus L_k^r \oplus R^r = R^r \oplus (L_1^{r-1} \oplus f_r(R^{r-1})) \oplus \cdots \oplus (L_k^{r-1} \oplus f_r(R^{r-1}))$$
$$= L_1^{r-1} \oplus L_2^{r-1} \oplus \cdots \oplus L_k^{r-1} \oplus R^{r-1}$$
$$\vdots$$
$$= L_1^0 \oplus L_2^0 \oplus \cdots \oplus L_k^0 \oplus R^0$$

따라서 이상적인 랜덤 순열 생성기와 k가 짝수인 n:kn-UFN2를 구분하는 오라클 기계가 존재한다. □

정리 3.2.2 k가 홀수인 경우, 2k라운드 n:kn-UFN2는 유사 랜덤이 아니다.

증명 2k라운드 n:kn-UFN2의 입력값과 출력값 사이에 다음과 같은 관계식이 성립한다.

$$L_1^{2k} \oplus \cdots \oplus L_k^{2k} \oplus R^{2k} = (L_1^{2k-1} \oplus \cdots \oplus L_k^{2k-1} \oplus R^{2k-1}) \oplus f_{2k}(R^{2k-1})$$
$$\vdots$$
$$= (L_1^0 \oplus \cdots \oplus L_k^0 \oplus R^0) \oplus \bigoplus_{i=1}^{2k} f_i(R^{i-1})$$

그런데 출력값 중 R^{2k}부분은 다음과 같은 식으로 표현된다.

$$R^{2k} = R^{2k-(k+1)} \oplus (f_{k+1}(R^k) \oplus \cdots \oplus f_{2k}(R^{2k-1}))$$
$$= R^{2k-(k+1)} \oplus (f_k(R^k) \oplus \cdots \oplus f_{2k}(R^{2k-1})) \oplus f_k(R^{k-1})$$
$$\vdots$$
$$= W \oplus \bigoplus_{i=1}^{2k} (f_i(R^{i-1}) \oplus \bigoplus_{i=1}^{\lfloor 2k/(k+1) \rfloor} f_{(2k+1)-i(k+1)}(R^{(2k+1)-i(k+1)-1})$$

이때 W 값은 다음과 같다.

$$W = \begin{cases} L_i^0 & \text{if } 3(2k+1) \bmod (k+1) = i-1 \\ R^0 & \text{if } 3(2k+1) \bmod (k+1) = k \end{cases}$$

따라서 입력값과 출력값 사이에 다음과 같은 관계식을 얻을 수 있다.

$$(L_1^{2k} \oplus \cdots \oplus L_k^{2k}) \oplus (L_1^0 \oplus \cdots \oplus L_k^0 \oplus R^0) \oplus W = f_k(R^{k-1})$$

그런데 두 오라클 질의에서 x_p, x_q에서 L_1^0 블록 값만 서로 다르도록 선택하면 $f_k(R^{k-1})$값이 동일하게 된다. 따라서 다음과 같은 선형 관계식을 얻을 수 있다.

$$(L_{p,1}^{2k} \oplus \cdots \oplus L_{p,k}^{2k}) \oplus (L_{q,1}^{2k} \oplus \cdots \oplus L_{q,k}^{2k}) \oplus (L_{p,1}^0 \oplus L_{q,1}^0) \oplus (W_p \oplus W_q) = 0$$

이 선형 관계식을 이용하면 이상적인 랜덤 순열 생성기와 2k라운드 n:kn-UFN2 순열 생성기를 구분하는 오라클 기계를 만들 수 있다. □

정의 3.2.1 (2k+1라운드 n:kn-UFN2의 BAD 사건 ξ) 확률변수 ξ^i는 $1 \le p < q \le m$인 두 오라클 질의 인덱스 p와 q에 대해서 R'_p 블록 값과 R'_q 블록 값이 동일한 사건으로 정의된다. 이때 BAD 사건 ξ는 확률변수로서 $\vee_{i=k}^{2k} \xi^i$로 정의된다.

정의 3.2.1에 정의된 BAD 사건에 대해서, 아래의 도움정리 3.2.1과 도움정리 3.2.2는 이상적인 랜덤 함수 생성기를 이용하는 2k+1라운드 피스텔 네트웍이 유사 랜덤 순열 생성기가 됨을 보이는데 이용된다.

도움정리 3.2.1 이상적인 랜덤 함수 생성기를 이용하는 2k+1라운드 n:kn-UFN2 순열 생성기에서 BAD 사건이 일어나지 않는 경우 n:kn-UFN2는 이상적인 순열 생성기와 동일하다. 즉, 모든 가능한 $\sigma_1, \ldots, \sigma_m \in \{0,1\}^{(k+1)n}$에 대해서

$$\Pr(\wedge_{i=1}^{m} (y_i = \sigma_i) \mid \neg \xi) = 1/2^{(k+1)nm}.$$

이때 y_i는 2k+1라운드 n:kn-UFN2 순열 생성기의 출력값이다.

증명 주어진 오라클 질의 x_p의 회신 y_p가 n:kn-UFN2

순열 생성기에 의해서 생성되는 경우, 오라클 회신은 다음과 같은 알고리즘으로 계산된다. 이때 x_p는 $(L_{p,1}^{\ 0} \| \cdots \| L_{p,k}^{\ 0} \| R_p^{\ 0})$이고, y_p는 $(L_{p,1}^{\ 2k+1} \| \cdots \| L_{p,k}^{\ 2k+1} \| R_p^{\ 2k+1})$이다.

알고리즘 :

Input $(L_{p,1}^{\ 0} \| \cdots \| L_{p,k}^{\ 0} \| R_p^{\ 0})$

Select random variable $h_p^{\ 1}$, $h_p^{\ 2}$, ..., $h_p^{\ 2k+1}$ ∈uniform distribution

For i=1 to 2k+1 do

$l \Leftarrow \min\{q : 1 \le q \le p \text{ and } R_q^{\ i-1}=R_p^{\ i-1}\}$

$L_{p,1}^{\ i} \Leftarrow R_p^{\ i-1}$,

$L_{p,2}^{\ i} \Leftarrow L_{p,1}^{\ i-1} \oplus h_l^{\ i}$,

⋮

$L_{p,k}^{\ i} \Leftarrow L_{p,k-1}^{\ i-1} \oplus h_l^{\ i}$,

$R_p^{\ i} \Leftarrow L_{p,k}^{\ i-1} \oplus h_l^{\ i}$,

End for

Output $(L_{p,1}^{\ 2k+1} \| \cdots \| L_{p,k}^{\ 2k+1} \| R_p^{\ 2k+1})$

이 알고리즘을 이용하면 오라클 회신 y_p는 $h_p^{\ k}$, $h_p^{\ k+1}$, ..., $h_p^{\ 2k+1}$와 알고리즘에서 사용되는 중간 값인 $(L_{p,1}^{\ k-1} \| \cdots \| L_{p,k}^{\ k-1} \| R_p^{\ k-1})$로 표현할 수 있다.

$$\begin{pmatrix} L_{p,1}^{k-1} \\ L_{p,2}^{k-1} \\ L_{p,3}^{k-1} \\ \vdots \\ L_{p,k-1}^{k-1} \\ L_{p,k}^{k-1} \\ R_p^{k-1} \end{pmatrix} \oplus \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 & 1 & 0 \\ 1 & 1 & 1 & \cdots & 1 & 0 & 1 \\ 1 & 1 & 1 & \cdots & 0 & 1 & 1 \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots \\ 1 & 1 & 0 & \cdots & 1 & 1 & 1 \\ 1 & 0 & 1 & \cdots & 1 & 1 & 1 \\ 0 & 1 & 1 & \cdots & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} h_p^{k} \\ h_p^{k+1} \\ h_p^{k+2} \\ \vdots \\ h_p^{2k-1} \\ h_p^{2k} \\ h_p^{2k+1} \end{pmatrix} = \begin{pmatrix} L_{p,1}^{2k+1} \\ L_{p,2}^{2k+1} \\ L_{p,3}^{2k+1} \\ \vdots \\ L_{p,k-1}^{2k+1} \\ L_{p,k}^{2k+1} \\ R_p^{2k+1} \end{pmatrix}$$

이때 $y^{k-1}=(L_{p,1}^{\ k-1} \| \cdots \| L_{p,k}^{\ k-1} \| R_p^{\ k-1})$, $y^{2k+1}=(L_{p,1}^{\ 2k+1} \| \cdots \| L_{p,k}^{\ 2k+1} \| R_p^{\ 2k+1})$, $x=(h_p^{\ k} \| h_p^{\ k+1} \| \cdots \| h_p^{\ 2k+1})$ 그리고 (k+1)×(k+1) 행렬 A는 $A_{i,k+2-i}$원소만 0이고 다른 원소는 1인 행렬이다. 그러면 위의 식은 선형 관계식 $y^{k-1} \oplus Ax=y^{2k+1}$이다. 이때 $y=y^{k-1} \oplus y^{2k+1}$라고 두면 위의 식은 선형 관계식 Ax=y로 표현된다. 이 것은 $T(x)=Ax$인 함수 $T:I_{(k+1)n} \mapsto I_{(k+1)n}$와 동일하다.

함수 T(x)=Ax의 출력값 y가 균등한 확률 분포임을 증명하기 위해서는 함수 T가 일대일 함수라는 것만 보이면 된다. 왜냐하면 T의 입력값이 균등한 확률분포를 가지므로 T가 일대일 대응함수이면 출력값 y역시 균등한 확률분포를 가지기 때문이다. 함수 T(x)=y가 일대일 대응이라면 T의 역함수가 존재해야 한다. 즉, T(x)=Ax이므로 행렬 A의 역행렬 A^{-1}이 존재해야 한다. 그런데 만일 A의 행렬식(determinant) 값이 0이 아니면 역행렬이 존재한다. 따라서 $\det(A) \neq 0$임을 증명하면 된다.

사각 행렬 A의 행렬식 값이 0이 아니라는 것을 보이기 위해서는 먼저 A를 사다리꼴(echelon form) A′로 변환한다. 이때 행 덧셈(row addition)과 행 교환(row interchange)만을 사용한다. 만일 사다리꼴 A′이 0으로 이루어진 행을 포함하고 있다면 $\det(A)=0$이고, 그렇지 않다면 $\det(A) \neq 0$이다. 따라서 A′이 0으로 이루어진 행을 포함하고 있지 않다는 것을 보이면 된다.

먼저 (1,1)을 첫번째 피보트(pivot) 원소로 선택하고 행 덧셈을 하면 다음과 같은 행렬을 얻는다. 그리고 i번째 피보트는 (i,k+2−i)위치의 원소로 선택한다 ($2 \le i \le k$).

$$A \Rightarrow \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 & 1 & 0 \\ & & & & & 1 & 1 \\ & O & & & 1 & & 1 \\ & & & \cdots & & & \vdots \\ & & 1 & & O & & 1 \\ & 1 & & & & & 1 \\ 0 & 1 & 1 & \cdots & 1 & 1 & 1 \end{pmatrix}$$

위의 행렬은 k+1번째 열(column)을 제외한 모든 열이 피보트 원소를 포함하고 있다. 따라서 (k+1,k+1)위치의 원소를 k+1번째 피보트로 선택해야 한다. 그러기 위해서 k+1번째 행벡터의 k+1번째 원소를 제외한 모든 값을 0로 만들어야 한다. 위 행렬의 i번째 행벡터를 r_i라고 표기하자. k+1번째 피보트 원소는 행 덧셈 $r_2 \oplus r_3 \oplus \ldots \oplus r_k \oplus r_{k+1}$을 수행하면 얻어진다. XOR 연산의 성질에 의해서 행 덧셈의 결과는 (0, 0, ..., 0, 0, X)이다. 이때 k가 짝수이면 X=0, k가 홀수이면 X=1이다. 이렇게 얻은 행렬에서 각 피보트가 대각선에 위치하도록 행 교환을 수행하면 사다리꼴 A′이 얻어진다.

따라서 k가 홀수인 경우, A′은 0으로 이루어진 행을 포함하지 않는다. 그러므로 k가 홀수인 경우 k+2 라운드 n:kn-UFN2의 출력값은 균등한 확률 분포를 가진다. □

도움정리 3.2.2 2k+1라운드 n:kn-UFN2 순열 생성기에서 BAD 사건이 일어날 확률은 다음과 같다.

$$\Pr(\xi) \le (k+1)m^2/2^{n+1}.$$

증명 먼저 $\Pr(\xi^i) \le m^2/2^{n+1}$임을 보이자. 임의의 두 오라클 질의 p와 q에 대해서 $\Pr(\xi^i) = \Pr(R_p^{\ i}=R_q^{\ i})$이므로 다음과 같은 식을 얻는다.

$$\Pr(R_p^{\ i}=R_q^{\ i}) = \begin{cases} \Pr(L_{p,k}^{\ i-1} \| R_p^{\ i-1} = L_{q,k}^{\ i-1} \| R_q^{\ i-1}) & \text{if } (L_{p,k}^{\ i-1} \| R_p^{\ i-1} \neq L_{q,k}^{\ i-1} \| R_q^{\ i-1}) \\ 1/2^n & \text{otherwise} \end{cases}$$

$\Pr(L_{p,k}^{\ i-1} \| R_p^{\ i-1}=L_{q,k}^{\ i-1} \| R_q^{\ i-1})$에 대해서도 위와 같은 관계식을 계속 만들어 나갈 수 있다. 이들 관계식과 모든 오라클 질의 값이 서로 다른 값이라는 것을 이용하면 $\Pr(\xi^i)$는 다음과 같이 구해진다.

$$\Pr(\xi^i) = {}_mC_2 \Pr(R_p^{\ i}=R_q^{\ i}) = {}_mC_2 (1/2^n + \cdots + 1/2^{kn}) \le m^2/2^{n+1}.$$

따라서 $\Pr(\xi) \le (k+1)m^2/2^n$. □

정리 3.2.3 k가 홀수인 경우, 2k+1라운드 n:kn-UFN2는 유사 랜덤이다.

증명 도움정리 3.2.1과 도움정리 3.2.2에 의해서 이상적인 랜덤 함수 생성기를 이용하는 n:kn-UFN2 순열 생성기 P는 유사 랜덤 순열 생성기임을 다음과 같이 보일 수 있다.

$$
\begin{aligned}
&|\Pr(M^P(1^{(k+1)n})=1) - \Pr(M^K(1^{(k+1)n})=1)| \\
&= |\Pr(M^P(1^{(k+1)n})=1|\xi) - \Pr(M^K(1^{(k+1)n})=1)| \cdot \Pr(\xi) \\
&+ |\Pr(M^P(1^{(k+1)n})=1|\neg\xi) - \Pr(M^K(1^{(k+1)n})=1)| \cdot \Pr(\neg\xi) \\
&\le \Pr(\xi) \\
&\le (k+1)m^2/2^{n+1}.
\end{aligned}
$$

이제 유사 랜덤 함수 생성기를 이용하는 2k+1라운드 n:kn-UFN2 순열 생성기 P가 유사 랜덤 순열 생성기임을 보이자. 먼저 유사 랜덤 함수 생성기를 사용하는 n:kn-UFN2 순열 생성기 P는 유사 랜덤이 아니라고 가정하자. 그러면 어떤 상수 c에 대해서 이상적인 랜덤 순열 생성기 K와 P를 $1/n^c$보다 높은 확률로 구분하는 오라클 기계 M이 존재한다. $0 \le i \le 2k+1$인 i에 대해서 p_i^D를 n:kn-UFN2 순열 생성기에서 첫 번째 라운드부터 i번째 라운드까지는 이상적인 랜덤 함수 생성기를 사용하고 i+1라운드에서 2k+1라운드까지는 유사 랜덤 함수 생성기를 이용하는 순열 생성기 $D_{f_{2k+1}} \circ \cdots \circ D_{f_{i+1}} \circ D_{h_i} \circ \cdots \circ D_{h_1}(\cdot)$에 접근 가능한 오라클 기계가 출력값 1을 생성할 확률이라고 하자. 즉,

$$p_i^D = \Pr(M^{D_{f_{2k+1}} \circ \cdots \circ D_{f_{i+1}} \circ D_{h_i} \circ \cdots \circ D_{h_1}}(1^{(k+1)n}) = 1)$$

여기서 f_{i+1}, ..., f_{2k+1}는 유사 랜덤 함수 생성기가 생성한 함수이고, h_1, ..., h_i는 이상적인 랜덤 함수 생성기가 생성한 함수이다. 그리고 이상적인 랜덤 순열 생성기에 접근 가능한 오라클 기계가 1을 출력할 확률을 p^K라고 정의하자. 즉, $p^K = \Pr(M^K(1^{(k+1)n})=1)$이다.

유사 랜덤 함수 생성기를 사용하는 2k+1라운드 n:kn-UFN2가 유사 랜덤이 아니라고 가정했으므로 다음과 같은 식을 얻게 된다.

$$
\begin{aligned}
&1/n^c \le |p^K - p_0^D| \\
&\le |p^K - p_{2k+1}^D| + |p_{2k+1}^D - p_{2k}^D| + \cdots + |p_{i+1}^D - p_i^D| + \cdots + |p_1^D - p_0^D|
\end{aligned}
$$

그런데 이상적인 랜덤 함수 생성기를 사용하는 2k+1라운드 n:kn-UFN2가 유사 랜덤 순열 생성기라는 것을 보였으므로 $|p^K - p_{2k+1}^D| \le (k+1)m^2/2^{n-1}$. 따라서 $|p_{i+1}^D - p_i^D| \ge 1/(2k+2)n^c$ 인 i값이 존재한다. 이 것을 이용해서 이상적인 랜덤 함수 생성기와 유사 랜덤 함수 생성기를 $1/(2k+2)n^c$보다 높은 확률로 구분하는 오라클 기계를 구현할 수 있다. 하지만 이것은 유사 랜덤 함수 생성기가 유사 랜덤이라는 것에 모순이 된다. 따라서 유사 랜덤 함수 생성기를 이용하는 2k+1라운드 n:kn-UFN2는 유사 랜덤 순열 생성기이다. □

정리 3.2.3에 의해서 유사 랜덤 함수들을 사용하는 2k+1라운드 n:kn-UFN2 순열 생성기는 유사 랜덤이다. 일예로 64 비트 블록 암호기를 이용한 확장을 고려해 보자. 이 경우, 2k+1라운드 n:kn-UFN2의 라운드 함수로 64비트 암호기를 사용하면 (k+1)64 비트 블록 암호기를 구현할 수 있다. 즉, k가 3인 경우 256비트 블록 암호기가 된다. 실제 적용 측면에서 볼 때, 우리는 유사 랜덤성이 증명된 임의의 64 비트 블록 암호기로 이용할 수 있다. 또 다른 방법으로는 유사랜덤성은 증명되지 않았으나 매우 좋은 유사 랜덤성을 가지고 있다고 알려진 DES 등의 블록 암호기를 이용할 수 있다[16].

4. 결 론

이 논문에서는 기존에 존재하는 블록 암호기를 입출력이 더욱 큰 블록 암호기로 확장이 가능한 비대칭 피스텔 네트웍 구조 n:kn-UFN2를 제안했다. 그리고 이때 n:kn-UFN2가 유사 랜덤 순열 생성기가 되기 위한 조건을 살펴보았다. 먼저 확장 가능한 비대칭 피스텔 네트웍의 입력과 출력의 크기가 (k+1)n 비트라고 하자. 논문의 결과는 다음과 같다.

○ 확장 가능한 비대칭 피스텔 네트웍이 입력값의 크기가 n 비트이고 출력값의 크기가 n 비트인 유사 랜덤 함수 생성기를 사용하는 경우: k가 홀수이고 전체 라운드 수가 2k+1이상이면 유사 랜덤 순열 생성기가 된다.

이때 임의의 알고리즘이 m개의 평문과 암호문의 쌍을 살펴보고 이상적인 랜덤 순열 생성기와 유사 랜덤 순열 생성기를 구분할 수 있는 확률은 $O(m^2/2^n)$보다 작거나 같다.

참 고 문 헌

[1] A.J. Menezes, P.C. van Oorschot and S.A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.

[2] H. Feistel, "Cryptography and Computer Privacy," Scientific American, vol. 228, pp. 15-23, 1973.

[3] H. Feistel, W.A. Nots, "Some Cryptographic Techniques for Machine-to-Machine Data Communications," Proc. of the IEEE, vol. 63, pp 1545-1554, 1975.

[4] National Institute of Standard and Technology, "Anouncing Request for Candidate Algorithm Nominations for The Advanced Encryption Standard (AES)," Federal Register, vol. 62, pp. 48051-48058. 1997.

[5] National Bureau of Standard, NBS FIPS PUB 46, "Data Encryption Standard," National Bureau of Standards, U.S. Depardment of Commerce, 1977.

[6] National Bureau of Standard, NBS FIPS PUB 81, "DES Mode of Operation," National Bureau of Standards, U.S. Depardment of Commerce, 1980.
[7] M. Naor, O. Reingold, "On the Construction of Pseudorandom Permutations: Luby-Rackoff Revisited," J. Cryptology, vol. 12, pp. 29-66, 1999.
[8] B. Schneier, J. Kelsey, "Unbalanced Feistel Networks and Block-Cipher Design," Proc. Fast Software Encryption, Lecture Notes in Computer Science, vol. 1039, Springer-Verlag, Berlin, pp. 121-144, 1996.
[9] O. Goldreich, *Foundations of Cryptography (Fragments of book)*, available on line : http://theory.lcs.mit.edu/~oded/, 1998.
[10] M. Luby, C. Rackoff, "How to construct pseudorandom permutations from pseudorandom functions," SIAM J. Comput., vol. 17, pp. 373--386, 1988.
[11] C.S. Jutla, "Generalized Birthday Attacks on Unbalanced Feistel Networks," Advances in Cryptology - CRYPTO '98, Lecture Notes in Computer Science, vol. 1462, Springer-Verlag, Berlin, pp. 186-199, 1998.
[12] J. Patarin, "How to construct pseudorandom and super pseudorandom permutation from one single pseudorandom function," Advances in Cryptology - EUROCRYPT '92, Lecture Notes in Computer Science, vol. 658, Springer-Verlag, Berlin, pp. 256-266, 1992.
[13] J. Pieprzyk, "How to construct pseudorandom permutations from single pseudorandom functions," Advances in Cryptology - EUROCRYPT '90, Lecture Notes in Computer Science, vol. 473, Springer-Verlag, Berlin, pp. 140-150, 1991.
[14] B. Sadeghiyan, J. Pieprzyk, "A construction for super pseudorandom permutations from a single pseudorandom function," Advances in Cryptology - EUROCRYPT '92, Lecture Notes in Computer Science, vol. 658, Springer-Verlag, Berlin, pp. 267-284, 1992.
[15] Y. Zheng, T. Matsumoto, and H. Imai, "Impossibility and optimality results on constructing pseudorandom permutations," Advances in Cryptology - EUROCRYPT '89, Lecture Notes in Computer Science, vol. 434, Springer-Verlag, Berlin, pp. 412-422, 1990.
[16] A. J. Menezes, P. C. Oorschot, and S. A. Vanstone, Handbook of Applied Cryptography, CRC Press, 1997

이 광 수
1998년 연세대학교 컴퓨터과학과 학사. 2000년 한국과학기술원 전산학과 석사. 2000년 ~ 현재 미래산업(주) 소프트포럼. 관심분야는 암호학 등.

신 준 범
1995년 한국과학기술원 수학과 졸업(학사). 1998년 한국과학기술원 수학과 졸업(석사). 1998년 ~ 현재 한국과학기술원 전자전산학과 전산학전공 박사과정. 관심분야는 암호 프로토콜 및 알고리즘, 전자상거래, 인터넷 보안, 퍼지이론

이 광 형
1978년 서울공대 산업공학 학사. 1980년 한국과학원 산업공학 석사. 1982년 프랑스 INSA 전산학과 석사(DEA). 1985년 프랑스 INSA 전산학과 공학박사. 1988년 1월 프랑스 국가박사(전산학INSA LYONI대). 1985년 ~ 1995년 한국과학기술원 전산학과 조교수 및 부교수. 1995년 ~ 현재 한국과학기술원 전자전산학과 교수. 1985년 프랑스 INSA. 1995년 미국 Standford Research Intitute. 관심분야는 퍼지이론 및 응용, 인공지능, 전문가 시스템 등.