비대칭 피스텔 네트웍을 이용하는 순열 생성기의 유사 랜덤 증명

(Proof of the Pseudorandomness of Permutation Generators that use Unbalanced Feistel Network)

이 광 수 † 신 준 범 †† 이 광 형 †††

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

요 약 Luby-Rackoff의 논문 이후로 유사 랜덤 순열 생성기에 관한 많은 연구가 있었다. 하지만 대부분의 연구는 대칭 피스텔 네트웍 구조를 이용한 유사 랜덤 순열 생성기에 관한 것이었다. 이 논문에서는 비대칭 피스텔 네트웍 구조를 사용하는 순열 생성기가 유사 랜덤 순열 생성기가 되기 위한 조건을 분석한다. 비대칭 피스텔 네트웍 순열 생성기의 입출력의 크기가 (k+1)n 비트인 경우 논문의 결과는 다음과 같다.

비대칭 피스텔 네트웍이 입력 크기가 kn 비트이고 출력 크기가 n 비트인 유사 랜덤 함수 생성기를 사용하는 경우, 전체 라운드 수가 k+2 이상이면 유사 랜덤 순열 생성기이다.

비대칭 피스텔 네트웍이 입력 크기가 n 비트이고 출력 크기가 kn 비트인 유사 랜덤 함수 생성기를 사용하는 경우, 전체 라운드 수가 k+2 이상이면 유사 랜덤 순열 생성기이다.

Abstract Since the publication of Luby-Rackoff's paper, there have been much research on pseudorandom permutation generators. However, many of the previous works have been focused on balanced feistel networks for constructing pseudorandom permutation generators. In this paper, we will analyze the requirements of pseudorandom generators based on unbalanced feistel networks. In the case of unbalanced feistel network which has both (k+1)n bits input and output size, the results of this paper is following:

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

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

1. 서 론

블록 암호기(block cipher)는 블록 단위로 평문(plaintext)을 암호문(ciphertext)으로 변환하는 대칭키 암호 시스템(private-key cryptosystem)이다[1]. 그리고 블록 암호기는 일대일 대응함수(bijective function) 성질을 가진다. 또한 블록 암호기는 대부분의 암호 시스템에서 가장 중요한 요소이다. 따라서 안전한 암호 시스템을 구현하기 위해서는 안전한 블록 암호기를 구현해야 한다. 안전한 블록 암호기(secure block cipher)란 블록 암호기의 출력값이 랜덤한 값이 되는 블록 암호기이다[1]. 좀더 수학적으로 정의하면 블록 암호기가 유사 랜덤 순열 생성기(pseudorandom permutation generator)가 되면 안전한 블록 암호기가 된다. 하지만 실제로 블록 암호기를 구현할 때 가장 큰 문제점은 일대일 대응함수의 성질과 랜덤 성질을 모두 가지도록 구현하는 것이 어렵다는 것이다. 이런 문제점을 해결할 수 있는 방법이 바로 피스텔 네트웍(feistel network) 구조이다.

피스텔 네트웍이란 임의의 함수를 일대일 대응함수로

· 본 연구는 첨단정보기술 연구 센터를 통하여 과학재단의 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

변환해 주는 방법이다. 이 구조는 H. Feistel이 Lucifer 암호기를 구현할 때 고안했다 [2,3]. 블록 암호기를 구현할 때 피스텔 네트웍 구조를 사용한다면 출력값이 랜덤한 임의의 함수를 구현하는 것만으로 안전한 블록 암호기를 구현할 수 있다. 왜냐하면 출력값이 랜덤한 임의의 함수만 존재한다면 피스텔 네트웍 구조가 자동으로 일대일 대응 함수로 변환해 주기 때문이다.

따라서 대부분의 블록 암호기는 피스텔 네트웍 구조를 이용하거나 피스텔 네트웍 구조를 약간 변경한 구조를 사용해서 구현되었다. 특히 Luby와 Rackoff의 논문 이후 안전한 블록 암호기인 유사 랜덤 순열 생성기에 관한 수많은 연구가 있었다 [4,5,6,7,8,9]. 하지만 기존 피스텔 네트웍(대칭 피스텔 네트웍)의 문제점은 입출력의 크기가 큰 블록 암호기를 구현하기 어렵다는 것이다. 왜냐하면 라운드 함수를 구현하는 비용이 라운드 함수의 입력 크기에 비례하기 때문에 입출력이 큰 블록 암호기를 구현할 때는 기존 대칭 피스텔 네트웍 구조가 부적합하다.

이런 문제점을 해결해 줄 수 있는 방법중의 하나가 바로 기존 대칭 피스텔 네트웍(balanced feistel network)을 변형한 비대칭 피스텔 네트웍 구조(unbalanced feistel network)이다 [10]. 비대칭 피스텔 네트웍은 라운드 함수의 입력이 되는 소스-블록(source-block)과 라운드 함수의 출력과 결합되는 타겟-블록(target-block)의 크기가 서로 다른 피스텔 네트웍 구조이다. 따라서 비대칭 피스텔 네트웍 구조에서는 라운드 함수의 입력 크기를 조절하는 것이 가능하므로 대칭 피스텔 네트웍보다 더욱 적은 비용으로 라운드 함수를 구현할 수 있다.

비대칭 피스텔 네트웍을 이용해서 안전한 블록 암호기인 유사 랜덤 순열 생성기를 구현하기 위한 연구가 몇몇 있었다 [11,12]. 하지만 이들 연구는 비대칭 피스텔 네트웍을 이용하는 블록 암호기가 안전하고 효율적인 블록 암호기가 되기 위한 라운드 수의 최소값을 보여주지 못했다. 실제로 블록 암호기를 구현할 때 라운드 수의 최소값이 중요한 이유는 라운드 수가 블록 암호기의 속도와 비용에 큰 영향을 미치기 때문이다. 즉, 블록 암호기를 구현할 때 적은 수의 라운드를 사용해야 빠른 블록 암호기를 적은 비용으로 구현할 수 있다. 그러므로 이 논문에서는 먼저 비대칭 피스텔 네트웍을 이용한 블록 암호기가 안전한 블록 암호기인 유사 랜덤 순열 생성기가 되기 위한 라운드 수의 최소값을 구한다.

먼저 2장에서는 피스텔 네트웍과 유사 랜덤을 정의하고 기존연구를 정리한다. 그런 뒤 3장에서는 비대칭 피스텔 네트웍이 안전한 블록 암호기인 유사 랜덤 순열 생성기가 되기 위한 조건을 분석한다. 마지막으로 4장에서는 결론을 맺고 마치도록 한다.

2. 관련 연구

이 절에서는 피스텔 네트웍과 유사 랜덤의 개념을 정의하고 기존 연구를 정리한다.

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 크기가 동일하면 대칭 피스텔 네트웍, L과 R의 크기가 다르면 비대칭 피스텔 네트웍이다. 이때 f함수의 입력이 되는 블록 R을 소스-블록, 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 균형 피스텔 네트웍과 1라운드 비대칭 피스텔 네트웍을 나타낸다.

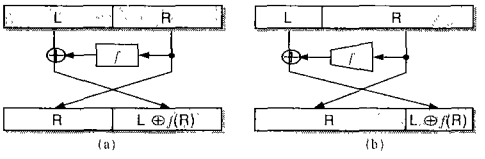

그림 1 피스텔 네트워 : (a) 1라운드 대칭 피스텔 네트웍, (b) 1라운드 비대칭 피스텔 네트웍

비대칭 피스텔 네트웍은 소스-블록이 타겟-블록보다 큰 소스-헤비 비대칭 피스텔 네트웍과 타겟-블록이 소스-블록보다 큰 타겟-헤비 비대칭 피스텔 네트웍으로 구분된다 [10].

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 [13]. 이때 두 사건의 동일성 여부를 판단하는 효과적인 알고리즘은 다음과 같은 오라클 기계(oracle machine)로 정의된다.

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.

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.

정의 2.3.2 (다항시간 구분불능) 두 사건 X와 Y가 다항시간에 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 Luby와 Rackoff에 의해서 소개가 되었다 [5].

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.

Luby-Rackoff의 연구 이후 대부분의 유사 랜덤 순열에 관한 연구는 대칭 피스텔 네트웍 구조를 이용한 순열 생성기에 관한 것이었다 [6,7,8,9]. 비대칭 피스텔 네트웍을 이용한 순열 생성기에 관한 연구는 비교적 최근에 연구가 되기 시작했다 [11,12].

Naor와 Reingold는 첫 라운드에 짝으로 독립인 순열(pairwise independent permutation)을 사용하고 나머지 라운드에 소스-블록의 크기가 타겟-블록보다 k배 큰 소스-헤비 비대칭 피스텔 네트웍 구조를 사용하는 경우, 피스텔 네트웍의 f함수로 유사 랜덤 함수를 사용하고 전체 라운드 수가 k+2이면 유사 랜덤 순열 생성기가 된다는 것을 보였다 [12]. Jutla는 타겟-블록의 크기가 소스-블록보다 k배 큰 타겟-헤비 비대칭 피스텔 네트웍 구조를 사용하는 경우, 피스텔 네트웍의 f함수로 유사 랜덤 함수를 사용하고 전체 라운드 수가 2k+2이면 유사 랜덤 순열 생성기가 된다는 것을 보였다 [11]. 하지만 이들 연구는 비대칭 피스텔 네트웍을 사용하는 유사 랜덤 순열 생성기가 되기 위한 라운드 수의 최소값을 밝히지 못했다. 블록 암호기를 구현할 때 라운드 수의 최소값이 중요한 이유는 라운드 수가 블록 암호기의 속도와 비용에 큰 영향을 미치기 때문이다. 즉, 블록 암호기를 구현할 때 적은 수의 라운드를 사용해야 빠른 블록 암호기를 적은 비용으로 구현할 수 있다. 따라서 이 논문에서는 비대칭 피스텔 네트웍을 이용하는 순열 생성기가 유사 랜덤이 되기 위한 라운드 수의 최소값을 밝히도록 한다.

3. 유사 랜덤 순열의 증명

이 절에서는 Luby-Rackoff의 증명 방법을 비대칭 피스텔 네트웍으로 확장하여 비대칭 피스텔 네트웍을 이

용하는 순열 생성기가 유사 랜덤 순열 생성기가 되기 위한 라운드 수의 조건을 밝힌다.

3.1 kn:n-UFN 순열 생성기

kn:n-UFN 순열 생성기는 소스-블록의 크기가 kn비트이고 타겟-블록의 크기가 n비트인 소스-헤비 비대칭 피스텔 네트웍이다. 그림 2는 소스-블록의 크기가 2n비트이고 타겟-블록의 크기가 n 비트인 3라운드 비대칭 피스텔 네트웍이다.

정의 3.1.1 (kn:n-UFN) 집합 $F:I_{kn}\mapsto I_n$에 속한 임의의 함수 f에 대해서 1라운드 kn:n-UFN은 함수 $D_f(L \| R_1 \| \cdots \| R_k) := (R_1 \| \cdots \| R_k \| (L\oplus f(R_1 \| \cdots \| R_k)))$로 정의된다. 이와 마찬가지로 집합 $F:I_{kn}\mapsto I_n$에 속한 임의의 함수들 $f_1, f_2, \ldots, f_r$에 대해서 r라운드 kn:n-UFN은 함수 $D_{fr}\circ\cdots\circ D_{f2}\circ D_{f1}(L^0 \| R_1^0 \| \cdots \| R_k^0) := D_{fr}\circ\cdots\circ D_{f2}(D_{f1}(L^0 \| R_1^0 \| \cdots \| R_k^0))$로 정의가 된다. 이때 $|L|=|R_i|=n$이다.

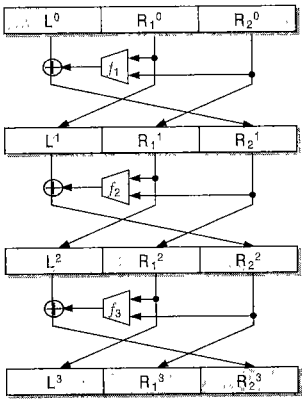

그림 2 3라운드 2n:n-UFN

정리 3.1.1 k+1라운드 kn:n-UFN은 유사 랜덤이 아니다.

증명 먼저 kn:n-UFN의 정의에 의해서 다음과 같은 식을 얻을 수 있다.

$L^{k+1} = R_1^k = R_2^{k-1} = \cdots = L^0\oplus f_1(R_1^0 \| \cdots \| R_k^0)$.

따라서 두 오라클 질의 x_p와 x_q에서 L 블록 값만 다르도록 질의를 선택하면 오라클 질의와 회신 $<x_p, y_p>$, $<x_q, y_q>$ 사이에 다음과 같은 선형 관계식이 존재하게 된다.

$L_p^{k+1}\oplus L_q^{k+1} = L_p^0\oplus L_q^0$.

이때 $<x_p, y_p> = <(L_p^0 \| R_{p,1}^0 \| \cdots \| R_{p,k}^0), (L_p^{k+1} \| R_{p,1}^{k+1} \| \cdots \| R_{p,k}^{k+1})>$, $<x_q, y_q> = <(L_q^0 \| R_{q,1}^0 \| \cdots \| R_{q,k}^0), (L_q^{k+1} \| R_{q,1}^{k+1} \| \cdots \| R_{q,k}^{k+1})>$로 표기한다.

오라클 기계 M은 L 블록 값만 다른 두 질의 x_p와 x_q를 만들어 오라클 회신을 받아 $<x_p, y_p>$, $<x_q, y_q>$값을 얻게 된다. 이때 $<x_p, y_p>$, $<x_q, y_q>$값에서 $L_p^{k+1}\oplus L_q^{k+1} = L_p^0\oplus L_q^0$ 관계식이 성립하면 1을 출력한다. 그렇지 않으면 0을 출력한다. 따라서 만일 오라클 회신 y_p, y_q 값이 kn:n-UFN에 의해서 생성된 것이라면 오라클 기계의 출력값은 항상 1이다. 하지만 y_p, y_q 값이 이상적인 랜덤 순열 생성기에 의해서 생성된 것이라면 위의 선형 관계식이 성립할 확률은 1/2n이다. 그러므로 kn:n-UFN을 P라고 하면 다음과 같은 식을 얻는다.

$|Pr(M^P(1^{(k+1)n})=1) - Pr(M^K(1^{(k+1)n})=1)| = 1 - 1/2^n > 1/p(n)$.

따라서 k+1라운드 kn:n-UFN은 유사 랜덤 순열 생성기가 아니다. □

정의 3.1.2 (k+2라운드 kn:n-UFN의 BAD 사건 ξ) 확률변수 ξ^i는 $1\le p<q\le m$인 두 오라클 질의 인덱스 p와 q에 대해서 $R_{p,1}^i \| \cdots \| R_{p,k}^i$ 블록 값과 $R_{q,1}^i \| \cdots \| R_{q,k}^i$ 블록 값이 동일한 사건으로 정의된다. 이때 BAD 사건 ξ는 확률변수로서 $\vee_{i=1}^{k+1}\xi^i$로 정의된다.

정의 3.1.2에 정의된 BAD 사건에 대해서, 아래의 도움정리 3.1.1과 도움정리 3.1.2는 이상적인 랜덤 함수 생성기를 이용하는 k+2라운드 피스텔 네트웍이 유사 랜덤 순열 생성기가 됨을 보이는데 이용된다.

도움정리 3.1.1 이상적인 랜덤 함수 생성기를 이용하는 k+2라운드 kn:n-UFN 순열 생성기에서 BAD 사건이 일어나지 않는 경우 kn:n-UFN은 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 k+2라운드 kn:n-UFN 순열 생성기의 출력값이다.

증명 이상적인 랜덤 함수 생성기를 사용하는 k+2라운드 kn:n-UFN의 정의에 의해서 오라클 기계의 p번째 질의 x_p의 회신 y_p는 다음과 같다.

$y_p = (L_p^{k-2} \| R_{p,1}^{k+2} \| \cdots \| R_{p,k}^{k+2})$

$= (L_p^1\oplus h_2(\cdot) \| L_p^2\oplus h_3(\cdot) \| \cdots \| L_p^{k+1}\oplus h_{k+2}(\cdot))$

이때 $h_1, h_2, \ldots, h_{k+2}$는 입력이 kn비트이고 출력이 n비트인 이상적인 랜덤 함수 생성기에서 생성된 함수들이고 $h_i(\cdot)$의 입력값은 $(R_{p,1}^{i-1} \| \cdots \| R_{p,k}^{i-1})$이다. 그런데 BAD 사건 ξ의 정의에 의해서 $1\le q<p\le m$인 오라클 인덱스 q와 p에 대해서 다음과 같은 식을 얻게 된다.

$\neg\xi = \wedge_{i=1}^{k+1}\neg\xi^i = \wedge_{i=1}^{k+1}(R_{p,1}{}^i \| \cdots \| R_{p,k}{}^i \neq R_{q,1}{}^i \| \cdots \| R_{q,k}{}^i)$

따라서 모든 오라클 질의에 대해서 h_2, ..., h_{k+2}의 입력값들이 서로 다르고 h는 이상적인 랜덤 함수 생성기에 의해서 생성된 함수이므로 h의 출력값은 균등한 확률분포 값이 되고 $L'^{-1}\oplus h_i(\cdot)$값 역시 균등한 확률분포 값이 된다. □

도움정리 3.1.2 k+2라운드 kn:n-UFN 순열 생성기에서 BAD 사건이 일어날 확률은 다음과 같다.

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

증명 BAD 사건의 정의에 의해서 $\xi = \vee_{i=1}^{k+1}\xi^i$ 이다. 먼저 ξ^i 사건이 일어날 확률 값을 계산해보자. 확률변수 ξ^i는 $1\le p<q\le m$인 오라클 질의 인덱스 p와 q에 대해서 $R_{p,1}{}^i \| \cdots \| R_{p,k}{}^i$ 값과 $R_{q,1}{}^i \| \cdots \| R_{q,k}{}^i$ 값이 같게 되는 사건을 나타낸다. 그리고 kn:n-UFN 구조의 정의에 의해서 다음과 같은 식을 얻게 된다. 여기서 $h(\cdot)$는 이상적인 랜덤 함수 생성기에서 생성된 함수들이다.

$R_1{}^i \| \cdots \| R_{k-i}{}^i \| R_{k-i+1}{}^i \| \cdots \| R_k{}^i = R_{i+1}{}^0 \| \cdots \| R_k{}^0 \| L^0\oplus h_1(\cdot) \| \cdots \| L^{-i}\oplus h_i(\cdot)$

따라서 ξ^i 사건이 일어나도록 하기 위한 최선의 선택은 $i+1\le j\le k$에 대해서 $R_{p,j}{}^0=R_{q,j}{}^0$인 오라클 질의를 선택하는 것이다. 이 경우 ξ^i 사건이 일어날 확률은 $Pr(\xi^i) = {}_mC_2\ 1/2^n$ 이다. 그러므로 $Pr(\xi)\le(k+1)m^2/2^{n+1}$. □

정리 3.1.2 k+2라운드 kn:n-UFN은 유사 랜덤이다.

증명 도움정리 3.1.1과 도움정리 3.1.2에 의해서 이상적인 랜덤 함수 생성기를 이용하는 kn:n-UFN 순열 생성기 P는 유사 랜덤 순열 생성기임을 다음과 같이 보일 수 있다.

$|Pr(M^P(1^{(k+1)n})=1) - Pr(M^K(1^{(k+1)n})=1)|$

$= |Pr(M^P(1^{(k+1)n})=1|\xi) - Pr(M^K(1^{(k+1)n})=1)| \cdot Pr(\xi)$

$+ |Pr(M^P(1^{(k+1)n})=1|\neg\xi) - Pr(M^K(1^{(k+1)n})=1)| \cdot Pr(\neg\xi)$

$\le Pr(\xi)$

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

이제 유사 랜덤 함수 생성기를 이용하는 k+2라운드 kn:n-UFN 순열 생성기 P가 유사 랜덤 순열 생성기임을 보이자. 먼저 유사 랜덤 함수 생성기를 사용하는 kn:n-UFN 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. $0\le i\le k+2$인 i에 대해서 p_i^D를 kn:n-UFN 순열 생성기에서 첫 번째 라운드부터 i번째 라운드까지는 이상적인 랜덤 함수 생성기를 사용하고 i+1라운드에서 k+2라운드까지는 유사 랜덤 함수 생성기를 이용하는 순열 생성기 $D_{fk+2}\circ\cdots\circ D_{fi+1}\circ D_{hi}\circ\cdots\circ D_{h1}(\cdot)$에 접근 가능한 오라클 기계가 출력값 1을 생성할 확률이라고 하자. 즉,

$p_i^D = Pr(M^{D_{f_1}\circ\cdots\circ D_{f_i}\circ D_{h_i}\circ\cdots\circ D_{h_1}}(1^{(k+1)n}) = 1)$

여기서 f_{i+1}, ..., f_{k+2}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 k+2라운드 kn:n-UFN이 유사 랜덤이 아니라고 가정했으므로 다음과 같은 식을 얻게 된다.

$1/n^c \le |p^K - p_0{}^D|$

$\le |p^K - p_{k+2}{}^D| + |p_{k+2}{}^D - p_{k+1}{}^D| + \cdots + |p_{i+1}{}^D - p_i{}^D| + \cdots + |p_1{}^D - p_0{}^D|$

그런데 이상적인 랜덤 함수 생성기를 사용하는 k+2라운드 kn:n-UFN이 유사 랜덤 순열 생성기라는 것을 보였으므로 $|p^K - p_{k+2}{}^D| \le (k+1)m^2/2^{n+1}$. 따라서 $|p_{i+1}{}^D - p_i{}^D| \ge 1/(k+3)^{nc}$ 인 i값이 존재한다. 이 것을 이용해서 이상적인 랜덤 함수 생성기와 유사 랜덤 함수 생성기를 $1/(k+3)^{nc}$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 k+2라운드 kn:n-UFN은 유사 랜덤 순열 생성기이다. □

3.2 n:kn-UFN 순열 생성기

n:kn-UFN 순열 생성기는 소스-블록의 크기가 n비트이고 타겟-블록의 크기가 kn비트인 타겟-해비 비대칭 피스텔 네트웍이다. 그림 3은 소스-블록이 n비트이고 타겟-블록이 2n비트인 3라운드 비대칭 피스텔 네트웍이다.

정의 3.2.1 (n:kn-UFN) 집합 $F:I_n\mapsto I_{kn}$에 속한 임의의 함수 f에 대해서 1라운드 n:kn-UFN은 함수 $D_f(L_1 \| \cdots \| L_k \| R) = (R \| L_1\oplus C_1(f(R)) \| \cdots \| L_k\oplus C_k(f(R)))$로 정의된다. 이와 마찬가지로 집합 $F:I_n\mapsto I_{kn}$에 속한 임의의 함수들 f_1, f_2, ..., f_r에 대해서 r라운드 n:kn-UFN은 함수 $D_{fr}\circ\cdots\circ D_{f2}\circ D_{f1}(L_1{}^0 \| \cdots \| L_k{}^0 \| R^0) = D_{fr}\circ\cdots\circ D_{f2}(D_{f1}(L_1{}^0 \| \cdots \| L_k{}^0 \| R^0))$로 정의가 된다. 이때 $C_1(f(R)) \| \cdots \| C_k(f(R)) = f(R)$이고 $|L_i|=|R|=|C_i(\cdot)|=n$이다.

정리 3.2.1 k+1라운드 n:kn-UFN은 유사 랜덤이 아니다.

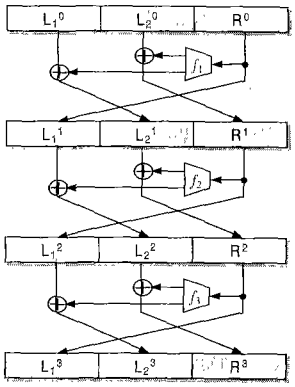

그림 3 3라운드 n:2n-UFN

증명 n:kn-UFN의 정의에 의해서 입력값과 출력값 사이에 $L_1^{k+1}=L_1^0 \oplus \oplus_{i=1}^{k} C_i(f_i(R^{i-1}))$ 관계식이 존재한다. 그리고 두 오라클 질의 x_p와 x_q에서 가장 왼쪽 블록 L_1^0 값만 다르게 질의를 선택하면 $1 \le i \le k$인 i에 대해서 R_p^{i-1}값과 R_q^{i-1}값이 같아진다. 따라서 오라클 질의와 회신 $<x_p,y_p>$, $<x_q,y_q>$에 대해서 다음과 같은 선형 관계식을 얻을 수 있다.

$L_{p,1}^{k+1} \oplus L_{q,1}^{k+1} = L_{p,1}^0 \oplus L_{q,1}^0$

이 관계식을 이용하면 이상적인 랜덤 순열 생성기와 k+1라운드 n:kn-UFN 순열 생성기를 구분하는 오라클 기계를 만들 수 있다. □

정의 3.2.2 (k+2라운드 n:kn-UFN의 BAD 사건 ξ) 확률변수 ξ_i는 $1 \le p < q \le m$인 두 오라클 질의 인덱스 p와 q에 대해서 R_p^i 블록 값과 R_q^i 블록 값이 동일한 사건으로 정의된다. 이때 BAD 사건 ξ는 확률변수로서 $\vee_{i=k}^{k+1} \xi^i$로 정의된다.

도움정리 3.2.1 이상적인 랜덤 함수 생성기를 이용하는 k+2라운드 n:kn-UFN 순열 생성기에서 BAD 사건이 일어나지 않는 경우 n:kn-UFN은 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 k+2라운드 n:kn-UFN 순열 생성기의 출력값이다.

증명 p번째 오라클 질의x_p의 회신 y_p는 다음과 같다.

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

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

$= (L_{p,k}^{k} \oplus C_k(h_{k+1}(R_p^k)) \| (L_{p,1}^{k+1} \| \cdots \| L_{p,k}^{k+1}) \oplus h_{k+2}(R_p^{k+1}))$

그런데 BAD 사건 ξ이 일어나지 않으므로 모든 오라클 질의에 대해서 h_{k+1}, h_{k+2}의 입력 값은 서로 다르다. 따라서 h_{k+1}과 h_{k+2} 함수는 이상적인 랜덤 함수 생성기에 의해서 생성된 것이므로 출력값은 균등한 확률분포를 가진다. □

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

$\Pr(\xi) \le m^2/2^n$.

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

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

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

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

$\Pr(\xi^{k+1})$역시 위와 같은 방법으로 구하고, 두 값을 합하면 $\Pr(\xi) \le m^2/2^n$. □

정리 3.2.2 k+2라운드 n:kn-UFN은 유사 랜덤이다.

증명 정리 3.1.2와 유사한 방법으로 증명이 가능하다. □

4. 결 론

이 논문에서는 비대칭 피스텔 네트웍이 유사 랜덤 순열 생성기가 되기 위한 라운드 수의 최소값을 분석했다. 먼저 비대칭 피스텔 네트웍 순열 생성기의 입력과 출력의 크기가 (k+1)n 비트라고 하자. 논문의 결과는 다음과 같다.

- 비대칭 피스텔 네트웍이 입력값의 크기가 kn 비트이고 출력값의 크기가 n 비트인 유사 랜덤 함수 생성기를 사용하는 경우: 전체 라운드 수가 k+2이상이면 유사 랜덤 순열 생성기가 된다.
- 비대칭 피스텔 네트웍이 입력값의 크기가 n 비트이고 출력값의 크기가 kn 비트인 유사 랜덤 함수 생성기를 사용하는 경우: 전체 라운드 수가 k+2이상이면 유사 랜덤 순열 생성기가 된다.

이 두 가지 경우 임의의 알고리즘이 m개의 평문과 암호문의 쌍을 살펴보고 이상적인 랜덤 순열 생성기와 유사 랜덤 순열 생성기를 구분할 수 있는 확률은 모두 $O(m^2/2^n)$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] M. Luby, C. Rackoff, "How to construct pseudorandom permutations from pseudorandom functions," SIAM J. Comput., vol. 17, pp. 373--386, 1988.

[5] U.M. Maurer, "A Simplified and Generalized Treatment of Luby-Rackoff Pseudorandom Permutation Generators," Advances in Cryptology - EUROCRYPT '92, Lecture Notes in Computer Science, vol. 658, Springer-Verlag, Berlin, pp. 239-255, 1992.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[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] M. Naor, O. Reingold, "On the Construction of Pseudorandom Permutations: Luby-Rackoff Revisited," J. Cryptology, vol. 12, pp. 29-66, 1999.

[13] O. Goldreich, Foundations of Cryptography (Fragments of book), available on line : http://theory.lcs.mit.edu/~oded/, 1998.

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 Stanford Research Intitute. 관심분야는 퍼지이론 및 응용, 인공지능, 전문가 시스템 등.