DC와 LC에 안전한 SPN 구조 암호 알고리즘

최 은 화[†]·서 창 호[††]·성 수 학[†††]·류 희 수[††††]·전 길 수[†††††]

요 약

본 논문에서는 수학적인 이론에 기반한 안전성이 증명 가능한 128 비트 블록 암호 알고리즘을 제안한다. 제안된 SPN 구조 암호 알고리즘에 사용된 active S-box가 많은 16×16 선형변환을 찾았고, 안전성에 대한 증명 방법으로 차분 해독(Differential Cryptanalysis)와 선형해독(Linear Cryptanalysis)에 대하여 증명하였다. 또한 DC와 LC에 영향을 주는 128 비트 블록 암호 알고리즘의 라운드 별 active S-box의 최소 개수, 최대 차분 확률과 최대 선형확률을 구하였다.

Secure Block Cipher Algorithm for DC and LC

Eun-Hwa Choi[†]·Chang-Ho Seo[††]·Soo-Hak Sung[†††]
Heuisu Ryu[††††]·Kil-Soo Chun[†††††]

ABSTRACT

In this paper, we suggest the design of 128bit block cipher which is provable security based on mathematics theory. We have derived the 16×16 matrix(i.e.,linear transformation) which is numerous active S-box, and we proved for DC and LC which prove method about security of SPN structure cipher algorithm. Also, the minimum number of active S-box, the maximum differential probabilities and the maximum linear probabilities in round function of 128bit block cipher algorithm which has an effect to DC and LC are derived.

키워드 : 암호 알고리즘(Cipher algorithm), SPN 구조(Substitution Permutation Network Structure)

1. 서 론

SPN(Substitution-Permutation Network) 구조는 Shannon [1]의 "혼동(confusion)"과 "확산(diffusion)" 효과를 줄 수 있는 라운드 함수를 구성함으로써 안전하고 실질적인 연결 암호(product cipher)를 설계할 수 있다는 주장을 직접적으로 반영한 블록 암호 알고리즘의 구조이다. 일반적으로 SPN 구조의 한 라운드는 대치(substitution), 선형 변환(linear transformation) 및 키 덧셈(key addition)의 세 가지 단계로 구성된다. 대치 단계는 입력을 몇 개의 작은 블록으로 구분한 후에 각각의 소블록들에 S-box라 부르는 비선형 변환을 적용하여 출력값을 얻는 과정으로 혼동 효과를 주기 위한 단계이다.

선형 변환 단계는 대치 단계의 출력인 각각의 S-box 출력 값을 전체 블록에 골고루 분산시키기 위한 과정으로 확산 효과를 주는 구성 요소이다. 그리고 키 덧셈 단계는 라운드 키를 주입하는 과정을 알고리즘에 따라서 그 위치는 가변적이다.

현재까지 암호 알고리즘의 안전성에 대한 검증 방법으로 입·출력 변화 공격법(Differential Cryptanalysis)와 선형 공격법(Linear Cryptanalysis)등이 있다. DC는 1990년 Biham과 Shamir[2]에 의해서 개발된 암호 알고리즘에 대한 안전성 분석법이며, LC는 1993년 Matsui[6]에 의해 개발된 안전성 분석법이다. 블록 암호를 설계하기 위해서는 적어도 이들 공격법에 대해서는 안전하게 설계되어야 한다. SPN 구조 블록 암호 알고리즘의 DC와 LC에 대한 안전성은 각각 최대 차분 확률과 linear hull의 최대 선형 확률을 기반으로 한다. 그러나, 일반적으로 이들 확률을 계산하는 것은 매우 어렵다. 따라서, 최대 차분 확률 대신에 최대 특성 확률(maximum characteristic probability)의 상한을 이용하며, 또한 linear hull의 최대 선형 확률 대신에 선형 근사 확률(linear approximation probability)의 상한을 이용한다.

본 논문에서는 SPN 구조인 128비트 블록 암호 알고리즘을 제안한다. 또한 DC와 LC에 안전하게 설계하기 위해 제안된 암호 알고리즘에 사용된 선형 변환은 active S-box

※ 본 논문은 2001년도 한국정보보호진흥원에 의해 연구 되었음.
※ 본 논문은 2001년도 한국과학재단(2001-10300-1-1001)에 의해 연구 되었음.
† 준 회 원 : 공주대학교 대학원 수학과
†† 정 회 원 : 공주대학교 응용수학과 교수
††† 정 회 원 : 배재대학교 전산정보수학과 교수
†††† 정 회 원 : 한국전자통신연구원 정보보호기반연구팀 팀장
††††† 정 회 원 : 한국정보보호진흥원 선임연구원
논문접수 : 2002년 2월 18일, 심사완료 : 2002년 4월 29일

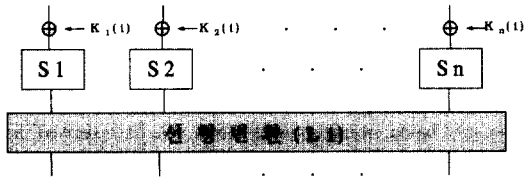
의 개수를 많게 하는 16×16 행렬에 대하여 설명한다. 그리고 최대 차분 확률과 linear hull의 최대 선형 확률을 구하고, DC와 LC에 안전에 대하여 살펴본다.

본 논문의 구성은 다음과 같다. 제2장에서는 일반적인 SPN 구조를 소개하고, S-box와 선형변환에 대한 DC와 LC에 대하여 설명한다. 제3장에서는 SPN 구조 암호 알고리즘을 제안하고, DC와 LC에 대하여 안전성을 분석한다. 마지막으로, 제4장에서는 결론을 맺는다.

2. 준비 단계

2.1 SPN 구조

(그림 1)과 같이 1라운드 SPN 구조는 키 덧셈부분, 대치부분 그리고 선형변환부분으로 나눌 수 있다. 키 덧셈 부분은 라운드의 입력과 서브키 간의 논리합(bitwise-exclusive or)이며, 대치부분은 n개의 S-box로 구성되며, 선형변환부분은 하나의 선형변환으로 구성된다.



(그림 1) 1 라운드 SPN 구조

1 라운드 SPN 구조를 r번 반복하여 만든 것을 r라운드 SPN 구조라고 한다.

S-box와 선형 변환은 복호화를 위하여 역변환을 가져야 하므로 본 논문에서 S-box는 정의역과 공변역이 $\{0,1\}^m$인 전단사 함수라고 가정한다. 또 라운드 키(서브 키)는 서로 독립하며 일양분포를 갖는다고 가정한다. 이 가정 하에서 라운드 키는 DC와 LC 분석에 아무런 영향을 주지 않으므로 앞으로는 라운드 키가 없는 것으로 간주한다.

2.2 S-box에 대한 DC와 LC

SPN 구조에 대한 DC와 LC를 살펴보기 위해 먼저 S-box에 대한 DC와 LC에 관하여 설명한다.

정의 1. 임의의 $a', b', a, b \in \{0,1\}^m$에 대해 S-box의 차분 확률과 선형 확률을 각각

$$DP^S(a' \to b') = \frac{\delta_S(a', b')}{2^m} = \frac{\#\{x \in \{0,1\}^m \mid S(x) \oplus S(x \oplus a') = b'\}}{2^m},$$

$$LP^S(a \to b) = \left(\frac{\lambda_S(a,b)}{2^{m-1}}\right)^2 = \left(\frac{\#\{x \in \{0,1\}^m \mid a \cdot x = b \cdot S(x)\}}{2^{m-1}} - 1\right)^2$$

로 정의한다. 여기서 $x \cdot y$는 $Z_2(=\{0,1\})$ 상에서 계산된 두 벡터의 내적이다.

a'와 b'을 각각 S-box의 입력과 출력 차분이라고 하며, a와 b를 각각 S-box의 입력과 출력mask 값이라고 부른다. DC와 LC에 대한 S-box의 강도는 각각 최대 차분 확률 $DP^S_{\max}$와 최대 선형 확률 $LP^S_{\max}$ 값으로 측정한다.

정의 2. S-box의 최대 차분 확률 $DP^S_{\max}$와 최대 선형 확률 $LP^S_{\max}$ 값은 각각

$$DP^S_{\max} = \max_{a' \neq 0, b'} DP^S(a' \to b'),$$
$$LP^S_{\max} = \max_{a, b \neq 0} LP^S(a \to b)$$

로 정의한다.

정의 3. 입력 차분이 0이 아니거나 출력 mask 값이 0이 아닌 S-box를 active S-box라고 부른다.

S-box는 전단사이므로 입력 차분이 0일 필요 충분 조건은 출력 차분이 0이며, 출력 mask 값이 0일 필요 충분 조건은 입력 mask 값이 0이다.

2.3 선형변환 대한 DC와 LC

이 절에서는 선형변환에 대한 DC와 LC에 관하여 설명한다.

i 번째 라운드의 선형변환은

$$L_i(x) = L_i(x_1, \cdots, x_n) = A_i \cdot \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$$

로 정의할 수 있다.

여기서 $\cdot$는 행렬의 곱이며, A_i의 원소들은 0 또는 1이고, n차 정칙행렬 (즉, $det(A_i) \neq 0$)이다.

행렬 A_i를 다음과 같이 놓자.

$$A_i = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}$$

그리고 $L_i(x) = y$로 놓으면

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

이다.

2.3.1 선형변환에 대한 DC

선형변환 L_i의 DC 성질에 관하여 설명한다. $(x_1, x_2, \cdots, x_n)$과 $(x_1^*, x_2^*, \cdots, x_n^*)$를 L_i의 입력 쌍이라고 두고 $(x_1', x_2',$

$\cdots, x_n')$ 을 두 입력의 차분이라고 놓자.

$$(x_1', x_2', \cdots, x_n') = (x_1 \oplus x_1^*, x_2 \oplus x_2^*, \cdots, x_n \oplus x_n^*)$$

또 $(y_1, y_2, \cdots, y_n)$과 $(y_1^*, y_2^*, \cdots, y_n^*)$를 L_i의 출력 쌍이라고 두고, 두 출력의 차분을 $(y_1', y_2', \cdots, y_n')$이라고 놓자.

그러면

$$y_i = a_{i1}x_1 \oplus a_{i2}x_2 \oplus \cdots \oplus a_{in}x_n,$$
$$y_i^* = a_{i1}x_1^* \oplus a_{i2}x_2^* \oplus \cdots \oplus a_{in}x_n^*$$

이다.

그러므로,

$$\begin{aligned} y_i' &= y_i \oplus y_i^* \\ &= a_{i1}x_1 \oplus a_{i2}x_2 \oplus \cdots \oplus a_{in}x_n \oplus a_{i1}x_1^* \oplus a_{i2}x_2^* \oplus \cdots \oplus a_{in}x_n^* \\ &= a_{i1}(x_1 \oplus x_1^*) \oplus a_{i2}(x_2 \oplus x_2^*) \oplus \cdots \oplus a_{in}(x_n \oplus x_n^*) \\ &= a_{i1}x_1' \oplus a_{i2}x_2' \oplus \cdots \oplus a_{in}x_n' \end{aligned}$$

이다.

따라서 $y' = L_i(x') = A_i x'$이다. 다시 말하면,

$$\begin{pmatrix} y_1' \\ y_2' \\ \vdots \\ y_n' \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1' \\ x_2' \\ \vdots \\ x_n' \end{pmatrix}$$

이다.

결과적으로 L_i가 선형변환일 때 $DP^{L_i}(x' \to L_i(x')) = 1$이므로, $x' \xrightarrow{L_i} L_i(x')$ 차분 확률 1 이다.

2.3.2 선형변환에 대한 LC

선형변환 L_i에 대한 LC에 관하여 설명한다. $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_n)$ 이라고 두자.

단, $\alpha_i \in \{0,1\}^m (1 \le i \le n)$이다. 그리고 A_i의 역행렬 A_i^{-1}을 다음과 같이 놓자.

$$A_i^{-1} = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nn} \end{pmatrix}.$$

$$\begin{aligned} &(\alpha A_i^{-1}) \cdot L_i(x) \\ &= (\alpha A_i^{-1}) \cdot \begin{pmatrix} \oplus_{i=1}^n a_{1i}\, x_i \\ \oplus_{i=1}^n a_{2i}\, x_i \\ \vdots \\ \oplus_{i=1}^n a_{ni}\, x_i \end{pmatrix} \\ &= (\alpha_1, \alpha_2, \cdots, \alpha_n) \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nn} \end{pmatrix} \cdot \begin{pmatrix} \oplus_{i=1}^n a_{1i}\, x_i \\ \oplus_{i=1}^n a_{2i}\, x_i \\ \vdots \\ \oplus_{i=1}^n a_{ni}\, x_i \end{pmatrix} \\ &= (\oplus_{i=1}^n b_{i1}\alpha_i, \oplus_{i=1}^n b_{i2}\alpha_i, \cdots, \oplus_{i=1}^n b_{in}\alpha_i) \begin{pmatrix} \oplus_{i=1}^n a_{1i}\, x_i \\ \oplus_{i=1}^n a_{2i}\, x_i \\ \vdots \\ \oplus_{i=1}^n a_{ni}\, x_i \end{pmatrix} \end{aligned}$$

이다.

그러므로

$$\begin{aligned} (\alpha A_i^{-1}) \cdot L_i(x) = {} & ((\oplus_{i=1}^n b_{i1}\alpha_i) \cdot (\oplus_{i=1}^n a_{1i}x_i)) \oplus \cdots \\ & \cdots \oplus ((\oplus_{i=1}^n b_{in}\alpha_i) \cdot (\oplus_{i=1}^n a_{ni}x_i)) \end{aligned}$$

이다.

여기서 $\oplus_{i=1}^n b_{ij}\alpha_i$, $\oplus_{i=1}^n a_{ji}x_i$ $(1 \le j \le n)$는 $\{0,1\}^m$ 상의 원소이며, $\cdot$ 는 $Z_2(=\{0,1\})$상에서 계산되는 내적이다. 한편 $\{0,1\}^m$ 상의 원소 u, v, w에 대해 $u \cdot (v \oplus w) = (u \cdot v) \oplus (u \cdot w)$이 성립한다.

따라서

$$\begin{aligned} (\alpha A_i^{-1}) \cdot L_i(x) = {} & [(a_{11}b_{11} \oplus a_{21}b_{12} \oplus \cdots \oplus a_{n1}b_{1n})(x_1 \cdot \alpha_1) \\ & \oplus (a_{11}b_{21} \oplus a_{21}b_{22} \oplus \cdots \oplus a_{n1}b_{2n})(x_1 \cdot \alpha_2) \oplus \\ & \cdots\cdots\cdots \\ & \oplus (a_{11}b_{n1} \oplus a_{21}b_{n2} \oplus \cdots \oplus a_{n1}b_{nn})(x_1 \cdot \alpha_n)] \oplus \\ & \cdots\cdots\cdots \\ & \oplus [(a_{1n}b_{11} \oplus a_{2n}b_{12} \oplus \cdots \oplus a_{nn}b_{1n})(x_n \cdot \alpha_1) \\ & \oplus (a_{1n}b_{21} \oplus a_{2n}b_{22} \oplus \cdots \oplus a_{nn}b_{2n})(x_n \cdot \alpha_2) \oplus \\ & \cdots\cdots\cdots \\ & \oplus (a_{1n}b_{n1} \oplus a_{2n}b_{n2} \oplus \cdots \oplus a_{nn}b_{nn})(x_n \cdot \alpha_n)] \end{aligned}$$

이다. 또 $A_i^{-1}A_i = I_n$ 이므로,

$$\begin{aligned} &(\alpha A_i^{-1}) \cdot L_i(x) \\ &\quad = [(x_1 \cdot \alpha_1) \oplus 0 \oplus \cdots \oplus 0] \oplus \cdots \oplus \\ &\quad\quad [0 \oplus 0 \oplus \cdots \oplus (x_n \cdot \alpha_n)] \\ &\quad = (\alpha_1, \alpha_2, \cdots, \alpha_n) \cdot (x_1, x_2, \cdots, x_n) \\ &\quad = \alpha \cdot x \end{aligned}$$

이다.

위의 결과를 정리하면 다음과 같다.

정리 1. 선형변환을 $L_i(x_1, x_2, \cdots, x_n) = A_i(x_1, x_2, \cdots, x_n)^t$ 라고 하자. 여기서 $x_i \in \{0,1\}^m (1 \le i \le n)$이다. 만일 $\{0,1\}^m$ 상의 덧셈이 xor이고, $n \times n$ 행렬 A_i의 원소가 $\{0,1\}$ 상의 원소이면 선형변환 L_i에 대한 선형 확률은 다음과 같다.

$$\begin{aligned} &LP^{L_i}(\alpha \to \alpha A_i^{-1}) \\ &= \left(\frac{\#\{x = (x_1, x_2, \cdots, x_n) \mid \alpha \cdot x = (\alpha A_i^{-1}) \cdot L_i(x)\}}{2^{nm-1}} - 1 \right)^2 = 1. \end{aligned}$$

2.4 SPN 구조에 대한 DC와 LC

이 절에서는 SPN 구조에 대한 DC과 LC 정의를 설명하고자 한다.

$L_i(x) \oplus L_i(x^*) = L_i(x \oplus x^*)$라는 사실로부터 1 라운드

SPN 구조의 차분 확률을

$$DP(a' \rightarrow L_i(b')) = \prod_{i=1}^{n} DP^{S_i}(a_i' \rightarrow b_i')$$

로 정의할 수 있으며 차분을 $a' \rightarrow b'$으로 쓰기로 한다. 단, $a' = (a_1', \cdots, a_n')$, $b' = (b_1', \cdots, b_n')$이다.

$L_i(B_i') = A'_{i+1}(1 \le i \le r-1)$을 만족하는 r개의 특성 $\{A_i' \rightarrow B_i',\ 1 \le i \le r\}$을 r 라운드 특성(characteristic)이라고 하며, 특성의 확률은 $\prod_{i=1}^{n} DP(A_i' \rightarrow L_i(B_i'))$이다. A_1'과 $L_r(B_r')$을 각각 r 라운드 특성의 입력과 출력이라고 한다. 입력이 A_1', 출력이 $L_r(B_r')$인 r 라운드 특성 전체를 r 라운드 차분(differential)이라고 하며, 특성 확률의 합을 차분 확률이라고 한다. R 라운드 SPN 블록 암호의 DC에 대한 복잡도는 R-1 라운드의 최대 차분 확률의 역수와 같으므로, DC에 안전한지를 평가하기 위해서는 최대 차분 확률을 구해야 한다.

이젠 SPN 구조에 대한 LC 정의를 살펴보자. 위에서 정의 된 선형변환 L_i에 대응되는 새로운 선형변환 L'_i을 $L'_i(x) = (A_i^{-1})^t \cdot \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$로 정의하자. 그러면 $LP^{L_i}(b \rightarrow L'_i(b)) = 1$ 이므로, 1 라운드 SPN 구조에 대한 선형근사 확률을

$$LP(a \rightarrow L'_i(b)) = \prod_{i=1}^{n} LP^{S_i}(a_i \rightarrow b_i)$$

로 정의 할 수 있으며, 선형근사를 $a \rightarrow b$로 쓰기로 한다. $L'_i(B_i) = A_{i+1}(1 \le i \le r-1)$을 만족하는 r개의 선형근사 $\{A_i \rightarrow B_i,\ 1 \le i \le r\}$을 r 라운드 선형근사라고 하며, 선형근사 확률은 $\prod_{i=1}^{n} LP(A_i \rightarrow L'_i(B_i))$이다. A_1과 $L'_r(B_r)$을 각각 r라운드 선형근사(linear approximation)의 입력 mask 값과 출력 mask 값이라고 하며, 입력 mask 값이 A_1, 출력 mask 값이 $L'_r(B_r)$인 r라운드 선형근사 전체를 linear hull이라고 한다. R 라운드 SPN 블록 암호의 LC에 대한 복잡도는 $R-1$ 라운드 linear hull의 최대 확률의 역수와 같으므로, LC에 안전한지를 평가하기 위해서 linear hull의 최대 선형 확률을 구해야 한다.

2 라운드 SPN 구조에 대한 active S-box의 최소 개수의 정의를 살펴보자.

정의 4. 2 라운드 특성에 작용하는 active S-box의 최소 개수를

$$n_d = \min_{a \neq 0} \{Hw(a) + Hw(M \cdot a)\}$$

로 정의하며, 2 라운드 선형근사에 작용하는 active S-box의 최소 개수를

$$n_l = \min_{b \neq 0} \{Hw(b) + Hw(M^t \cdot b)\}$$

로 정의할 수 있다[3]. 여기서 $a = (a_1, \cdots, a_n) \in GF(2^m)^n$, $Hw(a) = \#\{i \mid a_i \neq 0\}$, M은 선형변환에 대응되는 행렬, M^t는 행렬 M의 전치 행렬이며, $\cdot$는 행렬의 곱이다.

아래 보조정리는 본 논문의 정리를 증명하는데 사용된다[9].

보조정리 1. $F : \{0,1\}^n \rightarrow \{0,1\}^m$가 부울함수일 때 다음이 성립한다.

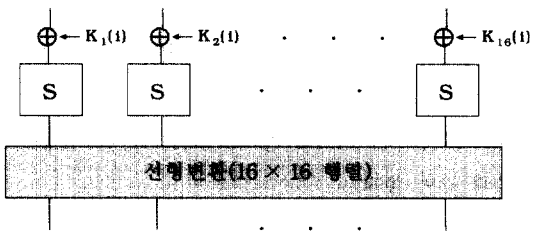
$$\sum_b DP^F(a \rightarrow b) = 1$$

3. 제안한 SPN 구조 암호 알고리즘 구조 및 안전성 분석

3.1 암호 알고리즘 구조

본 장에서는 안전성이 증명 가능한 새로운 128비트 SPN 블록 암호 알고리즘을 설명한다. SPN 블록 암호는 서브키와의 덧셈부분, S-box 부분 그리고 선형변환으로 구성되며, 안전한 블록 암호가 되기 위해서는 S-box와 선형변환을 설계하여야 한다. 또 서브키에 취약점이 없기 위해서는 서브키 생성 알고리즘도 잘 설계하여야 한다.

(그림 2)와 같이 128비트 블록 암호의 1라운드 구조는 128비트 서브 키(16 바이트), 8개의 8×8 S-box 그리고 선형변환(16×16 행렬)으로 구성된다.



(그림 2) 1 라운드 구조

암호화 및 복호화 하는데 필요한 메모리를 최소화하기 위해 S-box는 모두 같으며, 각 라운드의 선형변환도 모두 같다. 특히 복호화 과정에서 필요한 메모리를 최소화하기 위해 선형변환과 역 선형변환이 같게 설계하였다. 또 128비트 블록 암호의 라운드의 수는 12이며, 마지막 라운드는 그 이전 라운드와 달리 한번 더 서브키와의 논리합(bitwise-exclusive or)을 하였다.

3.2 핵심 논리

3.2.1 S-box

제안된 암호 알고리즘의 1 라운드에 16개의 S-box가 사용

되나, 암호화 및 복호화시 필요한 메모리의 양을 최소화하기 위해 동일한 S-box를 사용한다. 그리고 S-box 설계 단계는 다음과 같다.

[단계 1] $m(x)=x^8+x^6+x^5+x^4+1(171_x$에 대응)을 이용하여 유한체 $GF(2^8)$을 만든다.

[단계 2] 갈로아체 $GF(2^8)$ 상에서 변환 $x^{-1}=x^{254}$ 선택한다.

[단계 3] 다시 $\{0,1\}^8$ 상에서 affine 변환이 작용한다.

$$\begin{pmatrix} y_7\\ y_6\\ y_5\\ y_4\\ y_3\\ y_2\\ y_1\\ y_0 \end{pmatrix} = \begin{pmatrix} 0&0&0&1&1&1&1&0\\ 0&0&1&0&1&1&0&1\\ 0&1&0&0&1&0&1&1\\ 1&0&0&0&0&1&1&1\\ 1&1&1&0&0&0&1&1\\ 1&1&0&1&0&1&1&0\\ 1&0&1&1&1&1&0&0\\ 0&1&1&1&1&0&0&1 \end{pmatrix} \begin{pmatrix} x_7\\ x_6\\ x_5\\ x_4\\ x_3\\ x_2\\ x_1\\ x_0 \end{pmatrix} \oplus \begin{pmatrix} 1\\0\\1\\0\\0\\0\\1\\1 \end{pmatrix}$$

위의 affine 변환시 사용된 행렬은 대칭행렬(즉 $A=A^t$)이며 non-singular($det(A)\neq 0$)이다. 그 역행렬은 다음과 같다.

$$\begin{pmatrix} 0&0&1&1&1&1&0&1\\ 0&1&1&0&1&0&1&1\\ 1&1&0&0&0&1&1&1\\ 1&0&0&1&1&1&1&0\\ 1&1&0&1&0&0&1&0\\ 1&0&1&1&0&1&0&0\\ 0&1&1&1&1&0&0&0\\ 1&1&1&0&0&0&0&1 \end{pmatrix}$$

역행렬 역시 대칭행렬이다. 또 $x_0(y_0)$은 LSB이며 $x_7(y_7)$은 MSB이다. 따라서 affine 변환을 다음과 같이 간단히 쓸 수 있다.

$$\begin{pmatrix} y_7\\ y_6\\ y_5\\ y_4\\ y_3\\ y_2\\ y_1\\ y_0 \end{pmatrix} = \begin{pmatrix} 1e_x\cdot x\oplus 1\\ 2d_x\cdot x\oplus 0\\ 4b_x\cdot x\oplus 1\\ 87_x\cdot x\oplus 0\\ e3_x\cdot x\oplus 0\\ d6_x\cdot x\oplus 0\\ bc_x\cdot x\oplus 1\\ 79_x\cdot x\oplus 1 \end{pmatrix}$$

여기서 $\cdot$는 $Z_2(=\{0,1\})$ 상에서 계산된 두 벡터의 내적이다. 따라서 S-box는 다음과 같다.

$$S(x)=S(x_7,x_6,x_5,x_4,x_3,x_2,x_1,x_0)= \begin{pmatrix} 1e_x\cdot x^{-1}\oplus 1\\ 2d_x\cdot x^{-1}\oplus 0\\ 4b_x\cdot x^{-1}\oplus 1\\ 87_x\cdot x^{-1}\oplus 0\\ e3_x\cdot x^{-1}\oplus 0\\ d6_x\cdot x^{-1}\oplus 0\\ bc_x\cdot x^{-1}\oplus 1\\ 79_x\cdot x^{-1}\oplus 1 \end{pmatrix}$$

이젠 S-box의 값을 구체적으로 계산해 보자. $0^{-1}=0$이므로 $S(0)=(1,0,1,0,0,0,1,1)=a3_x=163$이다. $1^{-1}=1$이므로

$$S(1)=\begin{pmatrix}0\\1\\1\\1\\1\\0\\0\\1\end{pmatrix}\oplus\begin{pmatrix}1\\0\\1\\0\\0\\0\\1\\1\end{pmatrix}=\begin{pmatrix}1\\1\\0\\1\\1\\0\\1\\0\end{pmatrix}$$

이다. 따라서 $S(1)=da_x=218$이다. $2^{-1}=b8_x$이므로

$$S(2)=\begin{pmatrix} 0&0&0&1&1&1&1&0\\ 0&0&1&0&1&1&0&1\\ 0&1&0&0&1&0&1&1\\ 1&0&0&0&0&1&1&1\\ 1&1&1&0&0&0&1&1\\ 1&1&0&1&0&1&1&0\\ 1&0&1&1&1&1&0&0\\ 0&1&1&1&1&0&0&1 \end{pmatrix}\begin{pmatrix}1\\0\\1\\1\\1\\0\\0\\0\end{pmatrix}\oplus\begin{pmatrix}1\\0\\1\\0\\0\\0\\1\\1\end{pmatrix}=\begin{pmatrix}0\\0\\1\\1\\0\\0\\0\\1\end{pmatrix}\oplus\begin{pmatrix}1\\0\\1\\0\\0\\0\\1\\1\end{pmatrix}=\begin{pmatrix}1\\0\\0\\1\\0\\0\\1\\0\end{pmatrix}$$

이다. 따라서 $S(2)=92_x=146$이다. 제안된 암호 알고리즘에서 사용된 S-box는 <표 1>과 같다.

〈표 1〉 S-box

	00x	01x	02x	03x	04x	05x	06x	07x	08x	09x	0ax	0bx	0cx	0dx	0ex	0fx
00x	163	218	146	23	60	202	38	215	97	186	79	155	185	111	75	237
10x	55	105	82	204	85	165	16	36	123	222	199	167	160	216	1	20
20x	180	56	203	241	81	34	67	191	141	6	175	68	250	240	64	195
30x	239	192	103	223	150	22	86	236	138	231	107	182	247	219	15	88
40x	162	90	148	42	159	118	213	77	120	212	76	101	92	80	119	46
50x	193	70	43	225	170	30	128	233	0	51	5	149	117	179	99	144
60x	248	253	74	234	59	100	183	61	201	174	246	108	164	131	209	28
70x	29	19	4	151	50	83	91	62	217	104	66	106	7	87	84	136
80x	115	208	173	109	48	207	176	95	229	113	54	133	178	45	172	157
90x	198	130	98	153	124	47	194	37	161	17	168	53	230	33	69	12
a0x	154	21	121	147	96	200	8	94	143	227	10	196	24	57	244	114
b0x	39	156	65	232	2	190	224	134	31	110	126	14	206	214	197	132
c0x	249	26	220	242	112	184	221	137	189	169	18	73	139	44	177	32
d0x	102	235	122	13	9	63	238	254	127	11	49	72	71	40	243	245
e0x	35	142	211	166	210	3	25	89	145	205	129	188	125	135	152	181
f0x	187	41	27	255	140	158	226	252	251	58	116	171	93	78	228	52

갈로아체 $GF(2^8)$ 상의 함수 x^k중에서 $x^{127}, x^{191}, x^{223}, x^{239}, x^{247}, x^{251}, x^{253}, x^{254}=x^{-1}$은 DC, LC 그리고 고계차분공격법에 안전한 S-box이다(Moriai[7]). 그런데 위의 변환은 모두 0 과 1을 각각 0 과 1로 보내며(즉 0 과 1은 고정점임), 이 문제를 해결하기 위해 다시 선형변환을 작용하여 S-box를 설계한다. 선형변환 Z_{256} 상의 선형변환이 아닌 $GF(2^8)$ 상의 선형변환을 사용해야 좋은 성질이 유지된다. E2[8]의 S-box는 갈로아체 $GF(2^8)$ 상의 변환 x^{127}을 선형변환 $97x+225\ (\mathrm{mod}256)$으로 작용하여 설계되었다. 즉 $S(x)=97\times(x^{127}\in GF(2^8))+225\ (\mathrm{mod}256)$이다. 갈로아체 $GF(2^8)$ 상의 변환 x^{127}에대한 최대 차분 확률과 최대 선형 확률은 각각

$$1/2^6(\Delta=\max_{a'\neq 0,b'}\delta(a',b')=4,\ \Lambda=\max_{a,b\neq 0}\lambda(a,b)=16)$$

으로 DC와 LC에 안전하나, E2 S-box에 대한 Δ 값은 10, Λ 값은 28로 DC와 LC에 아주 안전한 것은 아니다.

<표 1>에서와 같이 갈로아체 $GF(2^8)$ 상의 변환 x^{223} 와 x^{239}를 사용하였으며 E2와는 달리 $GF(2^8)$상의 선형변환을 사용하였다. 즉 적당한 8×8 비트 행렬 $A^{(i)}$ 와 8×1 행렬 ($GF(2^8)$ 상의 원소) $b_{(i)}(i=1,2)$ 을 이용한 선형변환 $A^{(1)}x^{223} \oplus b_{(1)}$ 와 $A^{(2)}x^{239} \oplus b_{(2)}$ 로, 두 변환 모두 Δ 값은 4, Λ 값은 16으로 DC와 LC에 안전한 S-box이다.

3.2.2 선형 변환

S-box를 안전하게 설계할 수 있는 방법은 앞에서 언급하였듯이 아주 간단하다. SPN 블록 암호는 S-box와 선형변환으로 구성되어 있으므로 안전한 SPN 암호 설계는 선형변환의 설계에 달려 있다. S-box의 개수가 n일 때 선형변환은 $n \times n$ 행렬로 나타낼 수 있다. 선형변환은 DC와 LC에 안전하게 설계하여야 하며, 이를 위해 active S-box의 개수가 많도록 설계하면 된다. S-box의 개수가 n일 때 2라운드 SPN의 active S-box 개수의 하한은 $n+1$ 이하이다. 하한이 최대가 되는 행렬은 존재하나 $GF(2^m)$ (단, m은 S-box의 입출력 크기) 상의 원소로 구성된 행렬로 선형변환을 계산하는데 많은 시간이 소요된다. 본 논문에서는 선형변환을 쉽게 계산할 수 있는 $GF(2)(=\{0,1\})$ 상의 행렬로 active S-box의 개수를 많게 하는 것을 찾았다.

입출력의 크기가 8인 S-box를 사용할 때, 128비트 SPN 블록 암호를 설계하기 위해서는 16×16 행렬인 선형변환이 필요하다. 128비트 SPN 블록 암호용 선형변환은 다음과 같다.

$$\begin{pmatrix}
1&1&0&0&0&1&1&1&1&0&1&0&1&0&1&0\\
1&1&1&1&1&0&0&0&0&1&0&1&0&1&0&1\\
0&1&0&0&0&1&0&0&1&1&1&1&1&1&1&0\\
0&1&0&0&1&0&1&1&0&0&1&1&1&1&1&0\\
0&1&0&1&0&0&1&1&1&1&0&0&1&1&1&0\\
1&0&1&0&0&1&1&1&0&1&0&1&0&1&0&1\\
1&0&0&1&1&1&1&0&0&1&1&0&0&1&0&1\\
1&0&0&1&1&1&0&1&1&0&0&0&1&1&0&1\\
1&0&1&0&1&0&0&1&1&1&0&1&1&0&0&1\\
0&1&1&0&1&1&1&0&1&0&1&1&0&0&1&0\\
1&0&1&1&0&0&1&0&0&1&1&1&1&0&0&1\\
0&1&1&1&0&1&0&1&1&1&1&0&0&0&1&0\\
1&0&1&1&1&0&0&0&1&0&1&0&1&1&0&1\\
0&1&1&1&1&1&1&1&0&0&0&0&1&0&1&0\\
1&0&1&1&1&0&0&0&0&1&0&1&0&1&1&1\\
0&1&0&0&0&1&1&1&1&0&1&0&1&0&1&1
\end{pmatrix}$$

제안한 암호 알고리즘에 사용되는 16×16행렬(선형변환)의 특징은 다음과 같다.

(1) 선형변환과 역 선형변환은 같다. 즉 $A^{-1}=A$이다.

(2) $n_d(A)=n_l(A)=9$이다

(3) 대칭 행렬이다. 즉 $A^t=A$이다.

(4) 각 행 벡터 또는 열 벡터의 hamming weight는 9이다.

선형변환의 입력을 $(x_1,\cdots,x_{16})$, 출력을 $(y_1,\cdots,y_{16})$ 라고 할 때 입출력간의 관계는 다음과 같다.

$$\begin{aligned}
y_1 &= x_1\oplus x_2\oplus x_6\oplus x_7\oplus x_8\oplus x_9\oplus x_{11}\oplus x_{13}\oplus x_{15}\\
y_2 &= x_1\oplus x_2\oplus x_3\oplus x_4\oplus x_5\oplus x_{10}\oplus x_{12}\oplus x_{14}\oplus x_{16}\\
y_3 &= x_2\oplus x_6\oplus x_9\oplus x_{10}\oplus x_{11}\oplus x_{12}\oplus x_{13}\oplus x_{14}\oplus x_{15}\\
y_4 &= x_2\oplus x_5\oplus x_7\oplus x_8\oplus x_{11}\oplus x_{12}\oplus x_{13}\oplus x_{14}\oplus x_{15}\\
y_5 &= x_2\oplus x_4\oplus x_7\oplus x_8\oplus x_9\oplus x_{10}\oplus x_{13}\oplus x_{14}\oplus x_{15}\\
y_6 &= x_1\oplus x_3\oplus x_6\oplus x_7\oplus x_8\oplus x_{10}\oplus x_{12}\oplus x_{14}\oplus x_{16}\\
y_7 &= x_1\oplus x_4\oplus x_5\oplus x_6\oplus x_7\oplus x_{10}\oplus x_{11}\oplus x_{14}\oplus x_{16}\\
y_8 &= x_1\oplus x_4\oplus x_5\oplus x_6\oplus x_8\oplus x_9\oplus x_{12}\oplus x_{14}\oplus x_{16}\\
y_9 &= x_1\oplus x_3\oplus x_5\oplus x_8\oplus x_9\oplus x_{10}\oplus x_{12}\oplus x_{13}\oplus x_{16}\\
y_{10} &= x_2\oplus x_3\oplus x_5\oplus x_6\oplus x_7\oplus x_9\oplus x_{11}\oplus x_{12}\oplus x_{15}\\
y_{11} &= x_1\oplus x_3\oplus x_4\oplus x_7\oplus x_{10}\oplus x_{11}\oplus x_{12}\oplus x_{13}\oplus x_{16}\\
y_{12} &= x_2\oplus x_3\oplus x_4\oplus x_6\oplus x_8\oplus x_9\oplus x_{10}\oplus x_{11}\oplus x_{15}\\
y_{13} &= x_1\oplus x_3\oplus x_4\oplus x_5\oplus x_9\oplus x_{11}\oplus x_{13}\oplus x_{14}\oplus x_{16}\\
y_{14} &= x_2\oplus x_3\oplus x_4\oplus x_5\oplus x_6\oplus x_7\oplus x_8\oplus x_{13}\oplus x_{15}\\
y_{15} &= x_1\oplus x_3\oplus x_4\oplus x_5\oplus x_{10}\oplus x_{12}\oplus x_{14}\oplus x_{15}\oplus x_{16}\\
y_{16} &= x_2\oplus x_6\oplus x_7\oplus x_8\oplus x_9\oplus x_{11}\oplus x_{13}\oplus x_{15}\oplus x_{16}
\end{aligned}$$

$$\begin{pmatrix} y_1\\y_2\\y_3\\y_4\\y_5\\y_6\\y_7\\y_{18}\\y_{19}\\y_{10}\\y_{11}\\y_{12}\\y_{13}\\y_{14}\\y_{15}\\y_{16}\end{pmatrix} = \begin{pmatrix}
1&1&0&0&0&1&1&1&1&0&1&0&1&0&1&0\\
1&1&1&1&1&0&0&0&0&1&0&1&0&1&0&1\\
0&1&0&0&0&1&0&0&1&1&1&1&1&1&1&0\\
0&1&0&0&1&0&1&1&0&0&1&1&1&1&1&0\\
0&1&0&1&0&0&1&1&1&1&0&0&1&1&1&0\\
1&0&1&0&0&1&1&1&0&1&0&1&0&1&0&1\\
1&0&0&1&1&1&1&0&0&1&1&0&0&1&0&1\\
1&0&0&1&1&1&0&1&1&0&0&0&1&1&0&1\\
1&0&1&0&1&0&0&1&1&1&0&1&1&0&0&1\\
0&1&1&0&1&1&1&0&1&0&1&1&0&0&1&0\\
1&0&1&1&0&0&1&0&0&1&1&1&1&0&0&1\\
0&1&1&1&0&1&0&1&1&1&1&0&0&0&1&0\\
1&0&1&1&1&0&0&0&1&0&1&0&1&1&0&1\\
0&1&1&1&1&1&1&1&0&0&0&0&1&0&1&0\\
1&0&1&1&1&0&0&0&0&1&0&1&0&1&1&1\\
0&1&0&0&0&1&1&1&1&0&1&0&1&0&1&1
\end{pmatrix} \begin{pmatrix} x_1\\x_2\\x_3\\x_4\\x_5\\x_6\\x_7\\x_{18}\\x_{19}\\x_{10}\\x_{11}\\x_{12}\\x_{13}\\x_{14}\\x_{15}\\x_{16}\end{pmatrix}$$

$GF(2)(=\{0,1\})$ 상의 원소를 가진 16×16행렬 중 n_d 와 n_l 의 최대값은 각각 10이다. 위의 선형변환에 대한 n_d와 n_l도 각각 10으로 DC와 LC에 안전한 선형변환이다. $x_1,\cdots,x_{16}$의 순서를 바꾸어도 active S-box의 최소 개수에는 변화가 없으므로 위의 선형변환으로 부터 active S-box의 최소 개수가 10인 16!개의 새로운 선형변환을 만들 수 있다. 위의 선형변환을 SPN 블록 암호에 사용할 경우 라운드 별 active S-box의 최소 개수는 <표 2>와 같다.

〈표 2〉 128비트 블록 암호의 라운드 별 active S-box의 최소 개수

라운드 수	active S-box의 최소 개수	예(각 라운드별 active S-box의 수)
2	10	(1,9)
3	11	(1,9,1)
4	20	(1,9,1,9)
5	21	(1,9,1,9,1)
6	30	(1,9,1,9,1,9)
7	31	(1,9,1,9,1,9,1)
8	40	(1,9,1,9,1,9,1,9)
9	41	(1,9,1,9,1,9,1,9,1)
10	50	(1,9,1,9,1,9,1,9,1,9)
11	51	(1,9,1,9,1,9,1,9,1,9,1)
12	60	(1,9,1,9,1,9,1,9,1,9,1,9)

3.3 안전성 분석

3.3.1 제안한 알고리즘의 DC와 LC

이 장에서는 제안한 SPN 암호 알고리즘에 대한 DC와 LC 안전도를 살펴본다.

선형변환에 대응되는 16×16 행렬 M의 역행렬은 M^t이므로, $(M^{-1})^t = M$이다. 따라서 LC에 작용하는 선형변환 L'은 L과 같기 때문에 LC 분석은 DC 분석에서 S-box의 차분 확률을 선형 확률로 바꾸기만 하면 된다.

Active S-box의 개수는 특성 확률과 선형근사 확률의 상한을 예측하는데 사용할 수 있으나, DC와 LC에 대한 정확한 안전도를 측정하기 위해서는 차분 확률과 linear hull의 선형 확률을 계산하여야 한다. 그러나 라운드의 수가 클때 차분 확률과 linear hull의 선형 확률을 계산하는 것은 쉽지 않기 때문에, 다음과 같은 방법으로 차분 확률과 linear hull의 선형 확률을 예측한다.

[차분 확률 측정 알고리즘]

- [단계 1] active S-box의 개수가 최소인 특성을 찾는다.
- [단계 2] [단계 1]에서 구한 특성과 입력과 출력은 같고, 각 라운드 별로 동일한 active S-box가 되는 모든 특성을 찾는다.
- [단계 3] [단계 2]에서 구한 모든 특성 확률의 합을 차분 확률의 예측 값으로 사용한다.

Linear hull의 선형 확률도 차분 확률을 구하는 방법과 같이(특성 대신 선형근사 사용) 구할 수 있다. 그러므로 라운드 별로 차분 확률과 선형 확률을 구하면,

- 2 라운드 SPN일때 active S-box의 최소 개수는 10이며, 특성은

$$(0,0,0,0,\ 0,0,0,0,\ 0,0,0,0,\ 0,0,0,0,\ a_{16}) \rightarrow (0,0,0,0,\ 0,0,0,0,\ 0,0,0,0,\ 0,0,0,0,\ x)$$

$$(0,x,0,0,\ 0,x,x,x,\ x,0,x,0,\ x,0,x,x) \rightarrow (0,b_2,0,0,\ 0,b_6,b_7,b_8,\ b_9,0,b_{11},0,\ b_{13},0,b_{15},b_{16})$$

이므로, 차분 확률은 다음과 같이 계산할 수 있다.

보조정리 1에 의해서 $\sum_x DP^{S_{16}}(a_{16} \rightarrow x) = 1$ 이므로,

$$\begin{aligned}&\sum_x DP^{S_{16}}(a_{16} \rightarrow x)\ DP^{S_2}(x \rightarrow b_2)\\ &DP^{S_6}(x \rightarrow b_6)\ DP^{S_7}(x \rightarrow b_7)\ DP^{S_8}(x \rightarrow b_8)\\ &DP^{S_9}(x \rightarrow b_9)\ DP^{S_{11}}(x \rightarrow b_{11})\ DP^{S_{13}}(x \rightarrow b_{13})\\ &DP^{S_{15}}(x \rightarrow b_{15})\ DP^{S_{16}}(x \rightarrow b_{16})\\ &\le p^9 \sum_x DP^{S_{16}}(a_{16} \rightarrow x)\\ &= p^9.\end{aligned}$$

- 3 라운드 SPN일 때 active S-box의 최소 개수는 11이며, 특성은

$$(0,0,0,0,\ 0,0,0,0,\ 0,0,0,0,\ 0,0,0,0,\ a_{16}) \rightarrow (0,0,0,0,\ 0,0,0,0,\ 0,0,0,0,\ 0,0,0,0,\ x)$$

$$(0,x,0,0,\ 0,x,x,x,\ x,0,x,0,\ x,0,x,x) \rightarrow (0,y_2,0,0,\ 0,y_6,y_7,y_8,\ y_9,0,y_{11},0,\ y_{13},0,y_{15},y_{16})$$

$$(0,0,0,0,\ 0,0,0,0,\ 0,0,0,0,\ 0,0,0,0,\ y_{16}) \rightarrow (0,0,0,0,\ 0,0,0,0,\ 0,0,0,0,\ 0,0,0,0,\ b)$$

이므로, 차분 확률은 다음과 같이 계산할 수 있다.

보조정리 1에 의해서 $\sum_x DP^{S_{16}}(a_{16} \rightarrow x) = 1$ 이므로,

$$\begin{aligned}&\sum_x DP^{S_{16}}(a_{16} \rightarrow x)\ DP^{S_2}(x \rightarrow y_2)\ DP^{S_6}(x \rightarrow y_6)\ DP^{S_7}(x \rightarrow y_7)\\ &DP^{S_8}(x \rightarrow y_8)\ DP^{S_9}(x \rightarrow y_9)\ DP^{S_{11}}(x \rightarrow y_{11})\ DP^{S_{13}}(x \rightarrow y_{13})\\ &DP^{S_{15}}(x \rightarrow y_{15})\ DP^{S_{16}}(x \rightarrow y_{16})\ DP^{S_{16}}(y_{16} \rightarrow b)\\ &\le p^{10} \sum_x DP^{S_{16}}(a_{16} \rightarrow x)\\ &= p^{10}.\end{aligned}$$

같은 방법으로 linear hull의 선형 확률의 상한은 q^9이다. 여기서 p와 q는 S-box의 최대 차분 확률과 최대 선형 확률이다. 즉, $p = \max_j DP^{S_j}_{\max}$, $q = \max_j LP^{S_j}_{\max}$이다. 각 라운드 별 차분 확률과 linear hull의 상한은 <표 3>와 같다. p와 q가 각각 $1/2^6$ 되게 좋은 S-box를 선택하면, 8 라운드 일 때 차분 확률과 linear hull의 선형 확률의 상한은 각각 $1/2^{234}$ 이며 12 라운드 일 때 $1/2^{354}$ 이다.

〈표 3〉 128 비트 블록 암호의 라운드 별 차분 확률과 선형확률

라운드 수	차분 확률의 상한	linear hull의 선형 확률이 상한
2	p^9	p^9
3	p^{10}	p^{10}
4	p^{19}	p^{19}
5	p^{20}	p^{20}
6	p^{29}	p^{29}
7	p^{30}	p^{30}
8	p^{39}	p^{39}
9	p^{40}	p^{40}
10	p^{49}	p^{49}
11	p^{50}	p^{50}
12	p^{59}	p^{59}

4. 결 론

본 논문에서는 DC와 LC에 안전한 SPN 구조를 갖는 128 비트 블록 암호 알고리즘을 제안하였다. 제안된 암호 알고리즘에 사용된 active S-box가 많은 16×16 행렬인 선형변환을 찾았으며, 최대 차분 확률과 linear hull의 최대 선형 확률을 구하였다. 그러므로 제안된 SPN 구조의 암호 알고

리즘은 암호학적인 측면에서 안전하다. 또한 E2[8] 암호 알고리즘과 같이 제안된 SPN 구조 암호 알고리즘을 Feistel 구조의 블록 암호 알고리즘의 라운드 함수로 사용할 수 있다. 제안된 128비트 SPN 구조 암호 알고리즘을 라운드 함수로 사용하면 256비트 Feistel 구조의 블록 암호 알고리즘을 설계할 수 있다.

참 고 문 헌

[1] C. E. Shannon, Communication theory of secrecy systems, Bell System Technical Journal 28, pp.656-715, 1949.

[2] E. Biham and A. Shamir, "Differential cryptanalysis of DES-like cryptosystem," Advance in Cryptology- Crypto'90, LNCS Vol.537, Springer-Verlag, pp.2-21, 1990.

[3] J. Daemem, R. Govaerts, and J. Vandewlle, "Correlation matrices," Proceedings of the first international workshop of the Fast Software Encryption, LNCS Vol.1008, Springer-Verlag, pp.275-285, 1994.

[4] S. Hong, S Lee, J. Lim, J. Sung, and D. Cheon, "Provable security against differential and linear cryptanalysis for the SPN structure," FSE'2000, 2000.

[5] M. Kanda, Y. Takashima, T. Matsumoto, T. Matsumoto, K. Aoki, and K. Ohta, "A strategy for constructing fast round functions with practical security against differential and linear cyrptanalysis," Proceedings of SAC'98, LNCS Vol.1556, Springer-Verlag, pp.264-279, 1998.

[6] M. Matsui, "Linear cryptanalysis method of DES cipher," Advance in Cryptology-Eurocrypt'90, LNCS Vol.1039, Springer-Verlag, pp.386-397, 1993.

[7] S. Moriai, "How to design secure S-boxes against differential, linear, higher order differential, and interpolation attacks," SCES'98, 1998.

[8] NTT-Nippon Telegraph and Telephone Corporation, "Specification of E2-a 128 bit block cipher," AES proposal (available at http://info.isl.ntt.co.jp/e2/), 1998.

[9] M Matsui, New structure of block ciphers with provable security against differentail and linear cryptanalysis, in Fast Software Encryption(Springer, Berlin) pp.205-218, 1996.

최 은 화

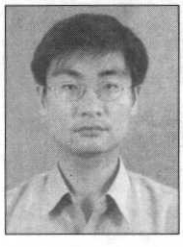
e-mail : euchoi@kongju.ac.kr
2002년 공주대학교 응용수학과 졸업
2002년 공주대학교 일반대학원 수학과
관심분야 : 블록 암호 알고리즘

서 창 호

e-mail : chseo@kongju.ac.kr
1990년 고려대학교 수학과 졸업(학사)
1992년 고려대학교 일반대학원 수학과 (이학석사)
1996년 고려대학교 일반대학원 수학과 (이학박사)
1996년~1997년 국방과학연구소 선임연구원
1997년~2000년 한국전자통신연구원 선임연구원, 팀장
2000년~현재 공주대학교 응용수학과 조교수
관심분야 : 암호 알고리즘, PKI, 시스템 보안 등

성 수 학

e-mail : sungsh@mail.pcu.ac.kr
1982년 경북대학교 수학과(학사)
1985년 KAIST 응용수학과(석사)
1996년 KAIST 응용수학과(박사)
1988년~1991년 한국전자통신연구원 선임연구원
1991년~현재 배재대학교 전산정보수학과 교수
관심분야 : 암호 이론, 암호프로토콜 등



류 희 수

e-mail : hsryu@etri.re.kr
1990년 고려대학교 수학과(학사)
1992년 고려대학교 수학과(석사)
1999년 Johns Hopkins Univ. 수학과 (박사)
1999년~2000년 홍익대학교 전자과 post-doc
2000년~현재 한국전자통신연구원 선임연구원(팀장)
관심분야 : 암호 이론, 통신망 정보보호, 정수론 등

전 길 수

e-mail : kschun@kisa.or.kr
1991년 서강대학교 수학과 졸업
1993년 서강대학교 수학과 석사
1998년 서강대학교 수학과 박사
1998년~1999년 서강대학교 기초과학연구소 박사후 연구원
2001년~2001년 서강대학교 컴퓨터학과 연구교수
2001년~현재 한국정보보호진흥원 선임연구원
관심분야 : 암호이론, 군론