블록 암호에서 교환 계층의 MDS 코드 생성 확인 알고리즘

박창수* · 조경연**

MDS code Creation Confirmation Algorithms in Permutation Layer of a Block Cipher

Chang-Soo Park*, Gyeong-Yeon Cho**

요 약

정보통신의 발달과 인터넷의 확산으로 인해 정보보안의 필요성이 증대되면서 다양한 암호알고리즘이 개발되어 활용되고 있다. 이와 더불어 암호 공격 기술도 발전하여서, 공격에 강한 알고리즘에 대한 연구가 활발하게 진행되고 있다. Substitution Permutation Networks(SPN) 등의 블록 암호알고리즘에서 교환계층의 선형변환행렬이 Maximum Distance Separable(MDS) 코드를 생성하면 차분공격과 선형공격에 강한 특성을 보인다.

본 논문에서는 선형변환행렬이 MDS 코드를 생성하는가를 판단하는 새로운 알고리즘을 제안한다. 선행변환행렬의 입력코드는 $GF(2^n)$상의 원소들로 이들을 변수로 해석할 수 있다. 하나의 변수를 다른 변수들의 대수식으로 변환하고 대입하여 변수를 하나씩 소거한다. 변수가 하나이고 모든 계수가 '0'이 아니면 선형변환행렬은 MDS 코드를 생성한다.

본 논문에서 제안한 알고리즘은 기존의 모든 정방부분행렬이 정칙인지를 판단하는 알고리즘과 비교하여 곱셈 및 역수 연산 수를 많이 줄임으로서 수행 시간을 크게 감소 시켰다.

ABSTRACT

According to the necessity about information security as well as the advance of IT system and the spread of the Internet, a variety of cryptography algorithms are being developed and put to practical use. In addition, the technique about cryptography attack also is advanced, and the algorithms which are strong against its attack are being studied. If the linear transformation matrix in the block cipher algorithm such as Substitution Permutation Networks(SPN) produces the Maximum Distance Separable(MDS) code, it has strong characteristics against the differential attack and linear attack.

In this paper, we propose a new algorithm which can estimate that the linear transformation matrix produces the MDS code. The elements of input code of linear transformation matrix over $GF(2_n)$ can be interpreted as variables. One of variables is transformed as an algebraic formula with the other variables, with applying the formula to the matrix the variables are eliminated one by one. If the number of variables is 1 and the all of coefficient of variable is non zero, then the linear transformation matrix produces the MDS code.

The proposed algorithm reduces the calculation time greatly by diminishing the number of multiply and reciprocal operation compared with the conventional algorithm which is designed to know whether the every square submatrix is nonsingular.

키워드

암호, 블록 암호, MDS 코드, 선형공격, 차분공격

*부경대학교 전자컴퓨터정보통신공학부 컴퓨터공학과 박사과정 **부경대학교 전자컴퓨터정보통신공학부 교수

접수일자 : 2003. 7. 30

I. 서 론

정보통신의 발달과 인터넷의 확산으로 정보보안의 필요성이 중요한 문제로 대두되면서 많은 국가들이 독자적으로 암호 알고리즘에 대한 연구를 진행하고 있다. 이에 우리나라도 한국정보보호센터를 주축으로 128 비트 블록 암호 알고리즘인 SEED[1]를 개발하여 공개하였다.

암호 기술이 발달하자 그에 따라서 암호문을 해독하기 위한 공격 기술도 같이 발달하게 되었다. 블록 암호 알고리즘에 대한 공격으로 차분공격[2]과 선형공격[3]이 연구되었고, 이들 공격에 저항성이 강한 블록 암호 알고리즘에 관한 연구도 활발하게 진행되고 있다. Heys와 Tavares는 블록 암호 알고리즘 중의 하나인 Substitution Permutation Networks(SPN)의 교환 계층에 선형변환을 사용하면 확산 특성이 개선되고, 차분공격과 선형공격에 대한 저항성이 증가된다는 연구 결과를 발표하였다[4][5]. 또한 Vaudenary는 교환 계층의 선현변환행렬이 Maximum Distance Separable(MDS) 코드[6]를 생성하면 이들 공격에 더욱 강한 특성을 가진다는 것을 제안하였고[7], 블록 암호알고리즘인 SHARK[8]와 SQUARE[9]에서는 MDS 코드를 생성하는 선형변환행렬을 교환계층에 사용하였다.

본 논문에서는 SPN 구조를 가지는 블록 암호알고리즘이 차분공격과 선형공격에 강한 특성을 가지기 위하여 교환계층에 선형변환행렬을 사용하였을 때, 이 선형변환행렬이 MDS 코드를 생성하는지 판단하는 알고리즘을 제안한다. 본 논문에서 제안하는 알고리즘은 선형변환행렬의 입력이 $GF(2^n)$상의 원소들로 구성되는데, 그 원소들을 변수로 해석하여, 변수를 소거시키면서 연산을 수행하여 선형변환행렬이 MDS 코드를 생성하는지 판단하는 것이다. 종래의 MDS 코드 생성 판단 알고리즘은 선형변환행렬의 모든 정방부분행렬이 정칙인지 확인하는 것이다. 본 논문에서 제안하는 알고리즘과 종래의 알고리즘을 비교하였을 때 본 논문에서 제안하는 알고리즘은 연산 수를 크게 감소시켜, 연산 수행시간을 대폭 줄인 것으로 향후 안전성이 높은 암호 알고리즘을 개발하는데 활용될 것으로 기대된다.

본 논문의 구성으로 2장에서는 관련연구로 SPN 구조를 가지는 블록 암호알고리즘의 치환계층과 교환계층의 특성 및 선형변환행렬의 구성방법을 소개하고, 3장에서는 본 논문에서 제안하는 알고리즘을 설명하고, 4장에서는 기존 알고리즘과 제안하는 알고리즘을 비교하고, 5장에서는 결론을 맺는다.

II. 관련 연구

SPN 구조를 가지는 블록 암호알고리즘의 한 라운드는 치환계층, 교환계층 및 키 덧셈계층의 3개 계층으로 구성된다. 치환계층은 비선형 치환을 수행하며 주로 비선형함수인 S 박스로 구성된다. 교환계층은 $GF(2^n)$상에서의 선형변환 계층으로 선형변환행렬로 나타낼 수 있다. 키 덧셈계층은 라운드 키를 더하는 계층이다.

키 덧셈 연산은 주로 배타적 논리합(XOR)으로 구현되고, 차분공격 및 선형공격 특성에 영향을 주지 않으므로 안전성 분석과정에서 생략할 수 있다. 따라서 SPN의 한 라운드에서 키 덧셈계층을 생략하면 치환-교환 계층으로 다룰 수 있다. 두 라운드의 치환-교환 계층을 하나로 묶으면 치환-교환-치환-교환 계층으로 된다. 여기서 마지막 교환계층은 선형공격과 차분공격 특성을 분석하는데 있어서 복잡성을 낮추기 위해서 생략하기도 한다. 즉, 두 라운드의 SPN은 하나의 치환-교환-치환 계층으로 간략화하여 다룰 수 있다. 본 논문에서는 이것을 SPS(substitution permutation substitution) 함수로 정의한다. 그림 1에 블록 크기 N=20, S 박스의 수 m=4, S 박스의 길이 n=5인 SPS 함수를 나타내었다.

SPN 구조에서 교환계층이 전단수함수일 때의 차분공격 특성에 관해서는 O'Connor가 연구하였으며[10], SPS 함수의 차분공격 및 선형공격에 대한 특성은 강 주성등[11] 및 Tavares등[12]의 연구 결과가 있다. 또한 이들 연구결과를 SEED에 적용한 연구[13]도 수행되었다.

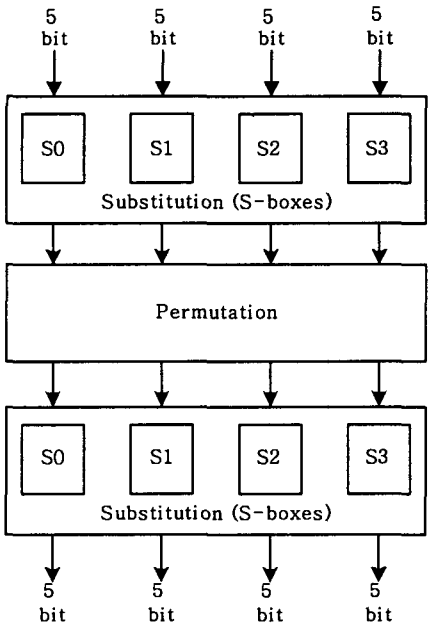


그림 4. SPS 함수
FIG. 1 SPS function

치환계층은 비선형 특성을 가지는 다수의 S 박스로 구성한다. 예를 들어 128 비트 블록 암호에서 S 박스를 'S: Z_2^8 -> Z_2^8'로 정의하면 치환계층은 16개의 S 박스로 구성된다. S 박스들은 교환계층의 특성에 따라서 차분적 및 선형적 활동성을 가지게 된다. 차분적 활동성을 가지는 S 박스는 입력 값이 '0'이 아닌 S 박스이며, 선형적 활동성을 가지는 S 박스는 입력과 출력 마스크 값이 '0'이 아닌 S 박스이다.

SPS 함수에서 입력과 출력을 각각 x, y라 하면 차분적 및 선형적 활동성을 가지는 S 박스의 최소수는 다음과 같다.

$$\beta_d(P) = \min_{\Delta x \neq 0} \{ W_c(\Delta x) + W_c(\Delta y) \}$$
$$\beta_l(P) = \min_{b \neq 0} \{ W_c(a) + W_c(b) \}$$
$$while,\ \Delta x, \Delta y, a, b \in Z_2^n$$

여기서 W는 해밍가중치이다.

활동성을 가지는 S 박스의 최소수 β를 branch number라고 부르며, 교환계층의 선형변환행렬 M의 특성에 의하여 결정된다. 행렬 M의 출력 값과 입력 마스크 값 사이의 관계는 전치된 행렬인 M^t로 나타낼 수 있다. 따라서,

$$\Delta y = M\Delta x \ , \qquad a = M^t b$$

이다. 이 식을 위 식에 적용하면 차분적 및 선형적 branch number는 다음 식이 된다.

$$\beta_d(P) = \min_{\Delta x \neq 0} \{ W_c(\Delta x) + W_c(M\Delta x) \}$$
$$\beta_l(P) = \min_{b \neq 0} \{ W_c(M^t b) + W_c(b) \}$$

위 식으로부터 행렬 M이 대칭행렬 또는 직교행렬이면 차분적 branch number와 선형적 branch number가 동일하게 됨을 알 수 있다.

한편 SPS 함수의 치환계층의 S 박스의 차분적 및 선형적 공격 최고 확률을 각각 D 및 L이라고 각각 정의하면, SPS 함수의 차분적 및 선형적 공격 최고 확률 D_f 및 L_f는 다음 식이 된다.

$$D_f \leq D^\beta, \qquad L_f \leq L^\beta$$

차분공격 및 선형공격에 강한 특성을 가지기 위해서는 D_f 및 L_f가 작아야하며, D와 L은 1보다 항상 작으므로, branch number β가 최대로 되어야 한다.

한편 임의의 필드상의 선형 (n, k, β) 코드에서 $\beta \leq n - k + 1$이 되며, $\beta = n - k + 1$로 최대값을 가지는 코드를 Maximum Distance Separable 코드 또는 줄여서 MDS 코드라고 부른다[6]. SPN 구조의 블록 암호 알고리즘에서 교환계층의 선형변환행렬 M은 $GF(2^n)$상의 $(2m, m, \beta)$ 코드가 된다. 이때 생성행렬 $G = [\ I \mid M\ ]$ 이라 하면, M은 $m \times m$ 정칙행렬이고, I는 $m \times m$ 항등행렬이다. 행렬 M이 MDS 코드를 생성하면, branch number가 $\beta = m + 1$ 로 최대값을 가진다[14].

따라서 SPS 함수가 차분공격 및 선형공격에 강하기 위해서는 교환계층의 선형변환행렬은 MDS 코드를 생성하는 대칭행렬 또는 직교행렬이 되어야

한다.

MDS 코드를 생성하는 선형변환행렬을 구성하는 방식은 행렬의 원소를 무작위로 선정하는 방식과 Cauchy 행렬을 사용하는 방식이 연구되었다[14]. 일정한 특성을 가지는 원소들로 선형변환행렬을 구성하는 경우[13]에는 생성된 행렬이 MDS 코드를 생성하는지를 판단해야 하는데, 행렬의 차수가 커지면 연산시간이 많이 걸린다.

III. MDS 코드 생성 확인 알고리즘

SPN 블록 암호 알고리즘에서 교환계층의 선형변환행렬 M은 $GF(2^n)$상의 m개의 n 비트 코드 X를 m개의 n 비트 코드 Y로 선형 변환한다. 이 때 교환계층은 식(1)로 표현된다.

$$\begin{bmatrix} Y_0 \\ Y_1 \\ Y_2 \\ \cdots \\ Y_{m-1} \end{bmatrix} = \begin{bmatrix} M_{0,0} & M_{0,1} & M_{0,2} & \cdots & M_{0,m-1} \\ M_{1,0} & M_{1,1} & M_{1,2} & \cdots & M_{1,m-1} \\ M_{2,0} & M_{2,1} & A_{2,2} & \cdots & A_{2,m-1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ M_{m-1,0} & M_{m-1,1} & M_{m-1,2} & \cdots & M_{m-1,m-1} \end{bmatrix} \times \begin{bmatrix} X_0 \\ X_1 \\ X_2 \\ \cdots \\ X_{m-1} \end{bmatrix} \quad (1)$$

식(1)에서 선형변환의 입력코드 X의 해밍 가중치와 선형변환 후의 출력코드 Y의 해밍 가중치를 각각 $W_h(X)$, $W_h(Y)$라 하면, 선형변환행렬 M의 branch number는 식(2)와 같이 주어진다.

$$\beta = \min(W_h(X) + W_h(Y)) \quad while \quad X \neq 0 \quad (2)$$

Branch number $\beta = m + 1$이 되면 선형변환행렬 M이 MDS 코드를 생성한다고 한다. 따라서 선형변환행렬 M이 MDS 코드를 생성하는지를 판단하기 위해서는 입력코드 X의 모든 원소에 대하여 branch number $\beta = m + 1$을 만족하는지 확인해야 한다. 이를 위해서 본 논문에서는 입력코드 X의 원소 $X_i(0 \leq i \leq m-1)$를 변수로 취급하여 이들을 소거하면서 연산하여, MDS 코드를 생성하는지 판단하는 알고리즘을 제안한다.

Lemma 3.1

m개의 원소를 가진 입력코드 X의 해밍 가중치 '$W_h(X) = 1$'이면, 선형변환행렬 M의 모든 원소가 '0'이 아니면 MDS 코드를 생성한다.

증명

입력코드 X의 해밍 가중치 '$W_h(X) = 1$'이므로 X의 원소중 하나만이 '0'이 아니다. 그 원소를 $X_i(0 \leq i \leq m-1)$라 하면 식(1)은 식(3)과 같이 된다.

$$\begin{aligned} Y_0 &= M_{0,i} \times X_i \\ Y_1 &= M_{1,i} \times X_i \\ Y_2 &= M_{2,i} \times X_i \\ &\cdots\cdots \\ Y_{m-1} &= M_{m-1,i} \times X_i \end{aligned} \quad (3)$$

식(3)에서 $X_i \neq 0$이므로 선형변환행렬 M의 모든 원소가 '0'이 아니면 출력코드 Y의 모든 원소는 '0'이 아니고, 해밍 가중치 '$W_h(Y) = m$'이 된다. 따라서 선형변환행렬 M의 branch number $\beta = m + 1$이 되어 MDS 코드를 생성한다. □

Lemma 3.2

입력코드 X의 해밍 가중치 '$W_h(X) = 2$'이면, 출력코드 Y의 해밍 가중치 최소값이 '$W_h(Y) = m-1$'인 경우에 MDS 코드를 생성한다. 입력코드의 원소 가운데 두 개의 원소만이 '0'이 아니고 그 원소를 X_i, X_j라 하면, 출력코드 원소 가운데 $Y_p(0 \leq p \leq m-2)$ 원소를 '0'으로 만드는 X_j가 반드시 존재한다. X_j를 X_i의 함수로 표현하고, 이것을 $Y_q(p < q \leq m-1)$에 대입하여 '$Y_q \neq 0$'가 되면 선형변환행렬 M은 MDS 코드를 생성한다.

증명

입력코드 X의 해밍 가중치 '$W_h(X) = 2$'이므로 X의 원소 가운데 두개만이 '0'이 아니다. 그 원소를 X_i, X_j라 두면 식(1)은 식(4)와 같이 표현된다.

$$\begin{aligned} Y_0 &= (M_{0,i} \times X_i) + (M_{0,j} \times X_j) \\ Y_1 &= (M_{1,i} \times X_i) + (M_{1,j} \times X_j) \\ Y_2 &= (M_{2,i} \times X_i) + (M_{2,j} \times X_j) \\ &\cdots\cdots\cdots \\ Y_{m-1} &= (M_{m-1,i} \times X_i) + (M_{m-1,j} \times X_j) \end{aligned} \quad (4)$$

식(4)에서 X_j를 소거시키기 위하여 'Y_0 = 0'으로 가정하고, X_j에 대해서 정리하면 식(5)가 된다.

$$X_j = -\frac{M_{0,i} \times X_i}{M_{0,j}} \quad (5)$$

식(5)를 식(4)에 대입하여 정리하면 식(6)과 같이 된다.

$$\begin{aligned} Y_1 &= \left(M_{1,i} - \frac{M_{1,j} \times M_{0,i}}{M_{0,j}}\right) \times X_i \\ Y_2 &= \left(M_{2,i} - \frac{M_{2,j} \times M_{0,i}}{M_{0,j}}\right) \times X_i \\ Y_3 &= \left(M_{3,i} - \frac{M_{3,j} \times M_{0,i}}{M_{0,j}}\right) \times X_i \\ &\cdots\cdots\cdots \\ Y_{m-1} &= \left(M_{m-1,i} - \frac{M_{m-1,j} \times M_{0,i}}{M_{0,j}}\right) \times X_i \end{aligned} \quad (6)$$

식(6)의 출력코드 Y_1에서 Y_{m-1}의 식에서 X_i의 모든 계수 값이 '0'이 아니면 $X_i \neq 0$이므로 Y_1에서 Y_{m-1}는 모두 '0'이 아니다. 따라서 출력 Y의 해밍 가중치 최소값은 '$W_h(Y) = m-1$'이다.

이어서 식(4)에서 $Y_1 = 0$이라고 가정하면, 앞에서 $Y_0 = 0$이면서 $Y_1 = 0$인 경우가 없음을 확인하였으므로 $Y_0 \neq 0$이다. 이 경우 X_j는 식(7)이 된다.

$$X_j = -\frac{M_{1,i} \times X_i}{M_{1,j}} \quad (7)$$

X_j를 소거하기 위하여 식(7)을 식(4)에 대입하여 정리를 하면 식(8)과 같이 된다.

$$\begin{aligned} Y_2 &= \left(M_{2,i} - \frac{M_{2,j} \times M_{1,i}}{M_{1,j}}\right) \times X_i \\ Y_3 &= \left(M_{3,i} - \frac{M_{3,j} \times M_{1,i}}{M_{1,j}}\right) \times X_i \\ Y_4 &= \left(M_{4,i} - \frac{M_{4,j} \times M_{1,i}}{M_{1,j}}\right) \times X_i \\ &\cdots\cdots\cdots \\ Y_{m-1} &= \left(M_{m-1,i} - \frac{M_{m-1,j} \times M_{1,i}}{M_{1,j}}\right) \times X_i \end{aligned} \quad (8)$$

식(8)의 출력코드 Y_2에서 Y_{m-1}의 식에서 X_i의 모든 계수값이 '0'이 아니면, $Y_0 \neq 0$이므로 Y의 해밍 가중치 최소값은 '$W_h(Y) = m-1$'이 된다.

이와 같은 과정을 $Y_i(2 \leq i \leq m-2)=0$에 대하여 반복 적용하여 모든 출력코드 Y_{i+1}에서 Y_{m-1}의 식에서 X_i의 모든 계수값이 '0'이 아니면, Y의 해밍 가중치 최소값은 '$W_h(Y) = m-1$'이 된다. □

Lemma 3.3

입력코드 X의 해밍 가중치 '$W_h(X) = 3$'이면, 출력코드 Y의 해밍 가중치 '$W_h(Y) = m-2$'인 경우에 MDS 코드를 생성한다. 입력코드 X의 원소 가운데 세 개만이 0이 아니고 그 원소를 X_i, X_j 그리고 X_k라 하면, 출력코드의 원소 $Y_p(0 \leq p \leq m-3)$를 '0'으로 가정하여 X_k를 X_i와 X_j의 함수로 표현하고, 이것을 $Y_q(p < q \leq m-2)$에 대입하여 X_k를 소거한다. Y_q는 '$W_h(X) = 2$'인 $m-q$ 차원의 행렬이 되며, Y_q가 *Lemma* 3.2를 만족하면 선형변환행렬 M은 MDS 코드를 생성한다.

증명

입력코드 X의 해밍 가중치 '$W_h(X) = 3$'이므로 X의 원소 가운데 세 개만이 '0'이 아니다. 그 원소를 X_i, X_j 그리고 X_k 라 두면 식(1)은 식(9)와 같이 된다.

$$\begin{aligned}
Y_0 &= (M_{0,i}\times X_i)+(M_{0,j}\times X_j)+(M_{0,k}\times X_k)\\
Y_1 &= (M_{1,i}\times X_i)+(M_{1,j}\times X_j)+(M_{1,k}\times X_k)\\
Y_2 &= (M_{2,i}\times X_i)+(M_{2,j}\times X_j)+(M_{2,k}\times X_k)\\
&\cdots\cdots\cdots\\
Y_{m-1} &= (M_{m-1,i}\times X_i)+(M_{m-1,j}\times X_j)+(M_{m-1,k}\times X_k)
\end{aligned} \tag{9}$$

식(9)에서 X_k를 소거시키기 위하여 $Y_0 = 0$이라고 가정하고 X_k에 대해서 정리하면 식(10)이 된다.

$$\begin{aligned}
Y_1 &= \left\{\left(M_{1,i}-\frac{M_{0,i}\times M_{1,k}}{M_{0,k}}\right)\times X_i\right\}+\left\{\left(M_{1,j}-\frac{M_{0,j}\times M_{1,k}}{M_{0,k}}\right)\times X_j\right\}\\
Y_2 &= \left\{\left(M_{2,i}-\frac{M_{0,i}\times M_{2,k}}{M_{0,k}}\right)\times X_i\right\}+\left\{\left(M_{2,j}-\frac{M_{0,j}\times M_{2,k}}{M_{0,k}}\right)\times X_j\right\}\\
Y_3 &= \left\{\left(M_{3,i}-\frac{M_{0,i}\times M_{3,k}}{M_{0,k}}\right)\times X_i\right\}+\left\{\left(M_{3,j}-\frac{M_{0,j}\times M_{3,k}}{M_{0,k}}\right)\times X_j\right\}\\
&\cdots\cdots\cdots\\
Y_{m-1} &= \left\{\left(M_{m-1,i}-\frac{M_{0,i}\times M_{m-1,k}}{M_{0,k}}\right)\times X_i\right\}+\left\{\left(M_{m-1,j}-\frac{M_{0,j}\times M_{m-1,k}}{M_{0,k}}\right)\times X_j\right\}
\end{aligned} \tag{10}$$

여기서 식을 간소화하기 위하여

$$\left(M_{1,i}-\frac{M_{0,i}\times M_{1,k}}{M_{0,k}}\right)=M'_{1,i},$$

$$\left(M_{1,j}-\frac{M_{0,j}\times M_{1,k}}{M_{0,k}}\right)=M'_{1,j},\ \cdots,$$

$$\left(M_{m-1,i}-\frac{M_{0,i}\times M_{m-1,k}}{M_{0,k}}\right)=M'_{m-1,i},$$

$$\left(M_{m-1,j}-\frac{M_{0,i}\times M_{m-1,k}}{M_{0,k}}\right)=M'_{m-1,j}$$

로 치환하면 식(10)은 식(11)로 나타낼 수 있다.

$$\begin{aligned}
Y_1 &= (M'_{1,i}\times X_i)+(M'_{1,j}\times X_j)\\
Y_2 &= (M'_{2,i}\times X_i)+(M'_{2,j}\times X_j)\\
Y_3 &= (M'_{3,i}\times X_i)+(M'_{3,j}\times X_j)\\
&\cdots\cdots\cdots\\
Y_{m-1} &= (M'_{m-1,i}\times X_i)+(M'_{m-1,j}\times X_j)
\end{aligned} \tag{11}$$

식(11)은 식(4)와 동일하다. 따라서 식(11)이 ***Lemma 3.2***를 만족하면 Y_1-Y_{m-1}의 해밍 가중치의 최소값은 '$m-2$'이다. 한편 $Y_0 = 0$이므로 Y의 해밍 가중치의 최소값은 '$W_h(Y) = m-2$'이다.

이어서 식(9)에서 $Y_1 = 0$이라고 가정하고 식(9)가 ***Lemma 3.2***를 만족하면 $Y_2 - Y_{m-1}$의 해밍 가중치의 최소값은 '$m-3$'이 된다. 앞에서 $Y_0 = Y_1 = 0$이면서 또 다른 $Y_i = 0$인 경우가 없음을 확인하였으므로, $Y_0 \neq 0$이다. 따라서 Y의 해밍 가중치의 최소값은 '$W_h(Y) = m-2$'이다.

이와 같은 과정을 $Y_i(3 \le i \le m-3)$에 대하여 반복하여 모두 ***Lemma 3.2***를 만족하면 Y_i-Y_{m-i}의 해밍 가중치의 최소값은 '$m-i-1$'이 된다. 그리고 동시에 3개의 $Y_i = 0$이 되는 경우가 없음을 확인하였으므로, Y의 해밍 가중치 최소값은 '$W_h(Y) = m-2$'이 된다. □

Lemma 3.4

입력코드 X의 해밍 가중치 '$W_h(X) = k$ ($4 \le k \le m$)'이면, 출력코드 Y의 해밍 가중치 최소값이 '$W_h(Y) = m - k + 1$'이면 MDS 코드를 생성한다. 입력코드 X의 원소 가운데 k 개의 원소가 '0'이 아니라면, 출력코드 Y의 원소 $Y_p(0 \le p \le m-k)$를 '0'으로 가정하여 X의 원소 가운데 하나를 다른 X의 원소로 표현하고 이것을 $Y_q(p < q \le m-k+1)$에 대입하여 X의 원소 하나를 소거하면, Y_q는 '$W_h(X) = k-1$'인 $m-1$ 차원의 행렬이 된다. 이러한 과정을 재귀적으로 반복하여 '$W_h(X) = 2$'인 $m - k + 2$ 차원의 행렬이 되고, 이 행렬이 ***Lemma 3.2***를 만족하면 선형변환행렬 M은 MDS 코드를 생성한다.

증명

Lemma 3.3의 증명을 재귀적으로 적용하여 증명할 수 있다. □

IV. 구현 및 비교 검토

MDS 코드 생성을 확인하는 알고리즘에서 수행 시간이 많이 소요되는 연산은 $GF(2^n)$상에서의 곱셈 연산과 역수 연산이다. 이 장에서는 본 논문에서 제안한 알고리즘과 기존 알고리즘의 곱셈 연산과 역수 연산의 횟수를 비교한다.

제안한 알고리즘에서 m개의 원소를 가진 입력코드 X의 해밍 가중치가 '$W_h(X) = 1$'이면 *lemma* 3.1에 의하여 곱셈 연산을 수행하지 않는다. '$W_h(X) = k$, $(2 \le k \le m)$' 이면 m개의 원소 중에서 '0'이 아닌 k개의 원소를 선정하므로 그 경우의 수는 $\binom{m}{k}$가 된다. 각 경우에 있어서 k의 값에 따라서 *lemma* 3.2부터 *lemma* 3.4를 적용한다. 이를 수식으로 나타내면 식-12가 된다.

$$F(m) = \sum_{k=2}^{m} \left\{ \binom{m}{k} Y_P(m, k) \right\} \tag{12}$$

식(12)의 Y_p에서 $k = 2$이면 *lemma* 3.2에 의하여 곱셈 연산의 횟수가 결정된다. $k > 2$ 이면 *lemma* 3.3 및 *lemma* 3.4에서 k 값을 '1'씩 감소시켜가면서 재귀적으로 연산해서 $k = 2$가 되면 *lemma* 3.2를 적용한다. 이를 수식화하면 식(12)의 Y_p는 식(13)이 된다.

$$Y_P(m, k) = \begin{cases} \sum_{i=2}^{m} i \\ \quad \text{if } k = 2 \\ \sum_{i=0}^{m-k} [\{(m-i)\times(k-1)\} \\ \quad + Y_P(m-i-1,\ k-1)] \\ \quad \text{if } k > 2 \end{cases} \tag{13}$$

동일한 방법으로 역수 연산 횟수를 수식화하면 식(14)가 된다.

$$G(m) = \sum_{k=2}^{m} \left\{ \binom{m}{k} Y_R(m, k) \right\}$$

where (14)

$$Y_R(m, k) = \begin{cases} m-1 \\ \quad \text{if } k = 2 \\ \sum_{i=0}^{m-k} \{1 + \\ \quad Y_R(m-i-1,\ k-1)\} \\ \quad \text{if } k > 2 \end{cases}$$

한편, 기존 알고리즘에서는 선형변환행렬의 모든 정방부분행렬이 정칙이면 MDS 코드를 생성한다는 것이 연구되었다[14]. 행렬이 정칙이기 위한 필요조건은 행렬의 determinant(det)가 '0'이 아니어야 한다. 본 논문에서는 행렬을 상삼각행렬로 변환하고, 그 대각선 원소들이 모두 '0'이 아니면 det 또한 '0'이 아니라는 점을 이용하여 정칙행렬을 판단하였다. 기존 알고리즘에서 곱셈 연산(Fs)과 역수 연산(Gs)의 횟수를 행렬의 차수(m)에 대한 함수로 나타내면 다음과 같이 된다.

$$F_S(m) = \sum_{k=2}^{m} \left\{ \binom{m}{k} \binom{m}{k} Y_S(k) \right\}$$

$$\textit{where} \quad Y_S(k) = \sum_{i=0}^{k-2} \sum_{j=i+1}^{k-1} (k-1)$$

$$G_S(m) = (m-1) + \sum_{k=1}^{m-2} \left\{ k \binom{m}{k+1} \binom{m}{k+1} \right\}$$

제안한 알고리즘과 기존 알고리즘은 IBM-PC상에서 GNU-C로 프로그램을 작성해서 동작을 검증하였다. $GF(2^8)$상에서 선형변환행렬 M의 차수가 m = 4, ⋯, 16인 경우에 곱셈 연산과 역수 연산의 횟수를 측정해서 수식과 일치함을 확인하였다. m = 24, 32인 경우는 수행 시간이 너무 많이 걸려서 프로그램으로는 확인하지 못했다.

표 1에 $GF(2^8)$상에서 선형변환행렬 M의 차수 m에 따른 MDS 코드 생성을 판단하기 위한 곱셈 및 역수 연산 횟수를 제안한 알고리즘과 기존 알고리즘을 비교하였고, 이를 그림 2에 그래프로 보인다.

표 3. 행렬의 차수에 따른 연산 횟수
Table 1. The number of operation by dimension of Matrix

연산 \ m		12	16	24	32
기존 알고리즘	곱셈	2.15 $\times 10^8$	1.11 $\times 10^{11}$	1.96 $\times 10^{16}$	2.61 $\times 10^{21}$
	역수	1.35 $\times 10^7$	4.21 $\times 10^9$	3.55 $\times 10^{14}$	2.75 $\times 10^{19}$
제안 알고리즘	곱셈	2.03 $\times 10^7$	4.87 $\times 10^9$	2.79 $\times 10^{14}$	1.64 $\times 10^{19}$
	역수	2.49 $\times 10^6$	5.66 $\times 10^8$	3.10 $\times 10^{13}$	1.78 $\times 10^{18}$

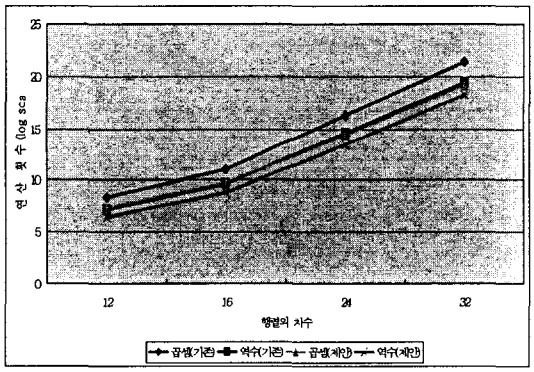


그림 5. 연산 횟수 그래프
FIG. 2 Graph about the number of operation

두 알고리즘의 연산 수를 비교하여 보면 행렬의 차수가 커질수록 본 논문에서 제안한 알고리즘의 연산 수가 큰 폭으로 감소하는 것을 알 수 있다. 그 결과 연산 수행 속도가 빨라져서 연산 수행 시간을 많이 단축할 수 있다.

V. 결 론

최근에 인터넷의 급속한 발전과 정보통신의 발달로 네트워크를 이용한 정보처리의 양이 급속히 늘어나게 되었다. 그 결과 정보보안의 필요성이 대두되어, 정보를 보호하고 불법적인 정보의 유출을 방지할 수 있는 암호 관련 연구들이 활발하게 진행되고 있다.

현재 많이 사용되고 있는 블록 암호 알고리즘에 대한 공격법으로 차분공격과 선형공격이 대표적이다. SPN 구조를 가지는 블록 암호 알고리즘은 치환 계층, 교환 계층 및 키 덧셈 계층으로 구성되는데, 이들 공격에 강한 특성을 지니기 위해서 교환 계층에 선형변환행렬을 이용한다. 특히 선형변환행렬이 MDS 코드를 생성하면 차분공격과 선형공격에 강한 특성을 가진다.

본 논문에서는 SPN의 두 라운드를 하나로 묶어 치환-교환-치환 계층으로 다루고, 이것을 SPS 함수로 정의하였다. SPS 함수의 교환계층으로 선형변환행렬을 이용하였을 때 선형변환행렬이 MDS 코드를 생성하는지 확인하는 알고리즘을 제안하였다.

선형변환행렬의 입력 코드는 $GF(2^n)$상의 원소들로 구성되는데, 이 원소들을 변수로 해석하여 소거시키면서 연산을 수행하여 선형변환행렬이 MDS 코드를 생성하는지 확인하였다. MDS 코드 생성을 확인하는 알고리즘에서 수행시간이 많이 걸리는 연산은 $GF(2^n)$상에서 곱셈과 역수 연산이다. 본 논문에서 제안하는 알고리즘은 종전의 선형변환행렬의 모든 정방부분행렬이 정칙인지를 확인하는 알고리즘과 비교하여 곱셈 및 역수 연산 수를 크게 감소시켰다. 그 결과 연산의 수행 시간이 많이 단축되었다.

향후 안전성이 강화된 블록 암호 알고리즘을 개발하는데 본 논문에서 제안한 알고리즘이 활용될 수 있을 것으로 기대된다.

참고 문헌

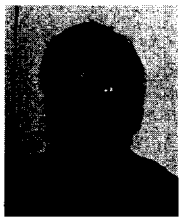
[1] 한국정보보호센터, "128 비트 블록 암호 알고리즘(SEED) 개발 및 분석보고서", Dec. 1998.
[2] E. Biham and A. Shamir, "Differential cryptanalysis of DES-like crytosystems, Journal of Cryptology", vol. 4, no. 1, pp. 3-72, 1991.
[3] M. Matsui, "The first experimental cryptanalysis of the Data Encryption Standard, Advances in Cryptology", Proc. Of EUROCRYPT '91, Springer-Verlag, Berlin, pp. 1-11, 1994.

[4] H.M. Heys and S.E. Tavares, "The design of substitution-permutation networks resistant to differential and linear cryptanalysis", Proceedings of 2nd ACM Conference on Computer and Communications Security, Fairfax, Virginia, pp. 148-155, 1994.
[5] H.M. Heys and S.E. Tavares, "The design of product ciphers resistant to differential and linear cryptanalysis", Journal of Cryptology, Vol. 9, no. 1, pp. 1-19, 1996.
[6] F.J. MacWilliams and N.J.A. Sloane, "The theory of error correcting codes", North-Holland Publishing Company, 1977.
[7] S. Vaudenay, "On the need for multipermutations: Crytanalysis of MD4 and SAFER", Proc. of Fast Software Encryption (2), LNCS 1008, Springer-Verlag, pp. 286-297, 1995.
[8] V. Rijmen, J. Daemen, B. Preneel, A. Bosselaers, and E. De Win, "The cipher SHARK", Fast Software Encryption, LNCS 1039, D. Gollmann, Ed., Springer-Verlag, pp. 99-112, 1996.
[9] J. Daemen, L. Knudsen, and V. Rijmen, "The block cipher SQUARE", Proc. of Fast Software Encryption (4), LNCS, Springer-Verlag, 1997.
[10] Luke O'Connor, "On the Distribution of Characteristics in Bijective Mapping," Advances in Cryptology, Proc. of EuroCrypt'93, Springer-Verlag, pp. 99-112, 1996.
[11] Ju-Sung Kang, Choonsik Park, Sangjin Lee, and Jong-In Lim, "On the Optimal Diffusion Layers with Practical Security against Differential and Linear Cryptanalysis", Proceedings of ICISC'99, LNCS 1787, Spring-Verlag pp. 33-52, 1999.
[12] A.M. Youssef and S.E. Tavares, "Resistance of Balanced S-boxes to Linear and Differential Crypanalysis," Information Processing Letters, Vol. 56, pp. 249-252, 1995
[13] 박창수, 조경연, 송홍복, "SEED 형식 암호에서 공격에 강한 S 박스와 G 함수의 실험적 설계", 한국해양정보통신학회, TBD
[14] A.M. Youssef, S. Mister, S.E. Tavares, "On the Design of Linear Transformation for Substitution Permutation Encryption Networks", in the Workshop Record of the Workshop on Selected Areas in Cryptography (SAC '97), pp. 40-48, Aug. 11-12, 1997.

저자 소개

박창수(Chang-Soo Park)

1995년 인제대학교 전자공학과 졸업(공학사)
2001년 부경대학교 산업대학원 컴퓨터공학과 졸업(공학석사)
2002년-현재 부경대학교 대학원 컴퓨터공학과 박사과정
※ 관심분야 : 반도체회로설계, 암호 알고리즘, 컴퓨터 구조



조경연(Gyeong-Yeon Cho)

1990 인하대학교 공과대학 전자공학과 정보공학전공 (공학박사)
1983-1991 삼보컴퓨터 기술연구소 책임연구원
1991-현재 부경대학교 공과대학 전자컴퓨터정보통신공학부 교수
1991-2001 삼보컴퓨터 기술연구소 비상임기술고문
1998-현재 에이디칩스 사외이사 겸 비상임기술고문

※ 관심분야 : 전산기구조, 반도체회로설계, 암호 알고리즘