

Twofish 암호알고리즘의 성능향상을 위한개선 된 MDS 블록 설계

정우열*, 이선근**

Design of Modified MDS Block for Performance Improvement of Twofish Cryptographic Algorithm

Woo-Yeol Jeong *, Seon-Keun Lee **

요약

Twofish 암호알고리즘은 AES인 Rijndael 암호알고리즘에 비하여 알고리즘 자체가 간결하며, 구현의 용이성이 좋지만 처리속도가 느린 단점을 가진다. 그러므로 본 논문은 Twofish 암호알고리즘의 속도를 향상시키기 위하여 개선된 MDS 블록을 설계하였다. 기존 MDS 블록은 Twofish 암호시스템의 critical path를 점유하게 되는 블록으로서 처리속도의 병목현상으로 인한 속도저하의 문제점이 존재하였다. MDS 블록에서 연산자로 사용되는 곱셈연산을 감소시키기 위하여 본 논문은 LUT 연산과 modulo-2연산을 사용하여 MDS 자체에 대한 속도저하 및 병목현상을 제거하였다. 이러한 결과로 새롭게 설계된 MDS 블록을 포함하는 Twofish 암호시스템은 기존 Twofish 암호시스템에 비하여 10%정도 처리속도의 향상을 가져옴을 확인하였다.

Abstract

Twofish cryptographic algorithm is concise algorithm itself than Rijndael cryptographic algorithm as AES, and easy of implementation is good, but the processing speed has slow shortcoming. Therefore this paper designed improved MDS block to improve Twofish cryptographic algorithm's speed. Problem of speed decline by a bottle-neck phenomenon of the processing speed existed as block that existing MDS block occupies Twofish cryptosystem's critical path. To reduce multiplication that is used by operator in MDS block this paper removed a bottle-neck phenomenon and low-speed about MDS itself using LUT operation and modulo-2 operation. Twofish cryptosystem including modified MDS block designed by these result confirmed that bring elevation of the processing speed about 10% than existing Twofish cryptosystem.

▶ Keyword : Twofish cryptographic algorithm, MDS(Maximum Distance Separable), Multiplication and LUT operation

• 제1저자 : 정우열

• 접수일 : 2005.08.25, 심사완료일 : 2005.09.13

* 한려대학교 멀티미디어정보통신공학과 교수, **원광대학교 전기전자 및 정보공학부 강의교수

Distance Separable) 행렬 곱셈기를 곱셈연산이 아닌 LUT를 적용한 MDS로 변환함으로써 Rijndael 암호알고리즘이 갖는 속도에 가깝도록 개선시켰다.[4]

I. 서론

1977년 암호 표준안으로 제정되어 사용되던 DES(Data Encryption Standard) 암호알고리즘이 플랫폼의 발달 및 네트워크 환경의 다변화로 인하여 더 이상 암호 표준안으로의 기능이 상실되었다. 따라서 NIST는 AES(Advanced Encryption Standard)를 공모하여 2000년도에 Rijndael 암호알고리즘을 최종 표준안으로 제정하였다. 그러나 Rijndael이 차세대 암호알고리즘으로 제정된 이유는 다른 후보 알고리즘들에 비하여 처리속도가 우수하다는 장점 때문이었다. 2000년도 이후 채택된 Rijndael 암호알고리즘은 암호화 및 복호화를 동시에 수행할 수 없으며 구현상의 어려움이 많기 때문에 아직까지 Rijndael 암호알고리즘은 활발하게 사용되지 못하고 있다.[1][3]

다섯 개의 후보 알고리즘(Twofish, Rijndael, MARS, RC6, Serpent) 중 암호의 안전성과 성능에서 인정을 받은 것이 Twofish 암호 알고리즘이다. Twofish 암호알고리즘은 Rijndael 암호알고리즘에 비하여 처리속도는 낮지만 구현상의 용이성 및 암호/복호화가 동시에 수행될 수 있다는 장점 때문에 Rijndael 암호알고리즘에 비하여 폭넓게 사용되고 있다. 그러므로 본 논문에서는 이러한 Twofish 암호알고리즘의 처리속도를 향상시켜 Rijndael 암호알고리즘의 처리속도에 비슷한 속도를 가질 수 있도록 하였다.[2][4]

최근의 여러 통신기술의 발전과 함께 인터넷은 단순한 웹 서핑이나 전자우편의 용도로만 사용되는 것에서 벗어나, 전자상거래를 비롯한 다양한 영역에서 활용되고 있다. 이에 따라 인터넷 응용 서비스는 점차 대규모화되고 복잡해지고 있으며, 이에 따라 인터넷 보안기술 역시 외부의 불법적인 침입을 방어하거나, 통신 내용을 보호하는 기능뿐만 아니라, 사용자 인증, 데이터 무결성 보장, 송수신 부인 봉쇄 등 보다 다양하고 복잡한 기능의 제공을 요구받고 있음에 따라 다른 무엇보다도 빠른 암호/복호화 처리가 중요시 되고 있다. 이에 따라 구현하기 용이하며 암호/복호화 처리가 동시에 처리 가능한 Twofish 암호 알고리즘을 인터넷 보안 기술에 적용하고자 한다. 그러나 처리속도면에서 Rijndael 암호알고리즘에 비하여 저속의 처리율을 가지므로 Twofish 암호알고리즘에서 가장 시간이 많이 소요되는 MDS(Maximum

II. Twofish 암호알고리즘

대칭키 블록 암호 알고리즘인 Twofish 암호 알고리즘은 블록 크기 128 비트에 대하여 키 길이를 128, 192, 256 비트로 다양한 길이를 가질 수 있는 가변 블록 대칭형 암호 알고리즘이다. (그림 1)은 Twofish 암호 알고리즘의 기본 구조를 나타낸다. Twofish 암호 알고리즘은 입/출력의 whitening 과정을 포함하며, F 함수의 출력이 한 비트씩 순환된다는 점을 제외하면 16라운드 Feistel 네트워크 구조와 유사하다.[1][2]

128 비트 평문은 little-endian convention을 사용하여 4개의 32비트 워드로 나누어지고, 각각의 워드는 32비트 부분키(subkey) 4개와 XOR 되어 입력 whitening 과정을 거친다. 각 라운드에서는 좌측 2개의 워드가 F-함수내의 두 g-함수의 입력으로 사용된다. 이때 1개의 입력 워드는 8비트 좌측 순환을 거쳐 입력된다. g-함수는 4개의 8-by-8비트 키값에 종속된 S-box들과 MDS 행렬 곱셈기로 구성된다. 두 g-함수의 출력은 Pseudo-Hadamard Transform(PHT)을 이용하여 결합되고, 2개의 부분키가 modulo 2 덧셈에 의해 더해진다. F-함수의 두 출력은 다음 라운드를 위하여 자리바꿈 한다. 16라운드 후에는 마지막 라운드의 결과가 다시 자리바꿈 되고, 출력 whitening을 거쳐 128비트 평문에 적용한 little-endian convention과 같은 방법으로 128 비트의 암호문을 생성한다. 복호화 과정은 암호화 과정에서 적용한 40개의 부분키의 순서 및 F-함수의 출력이 우측 2개의 워드와 XOR 및 한 비트 되는 과정을 역으로 적용하여 이루어진다.[5]

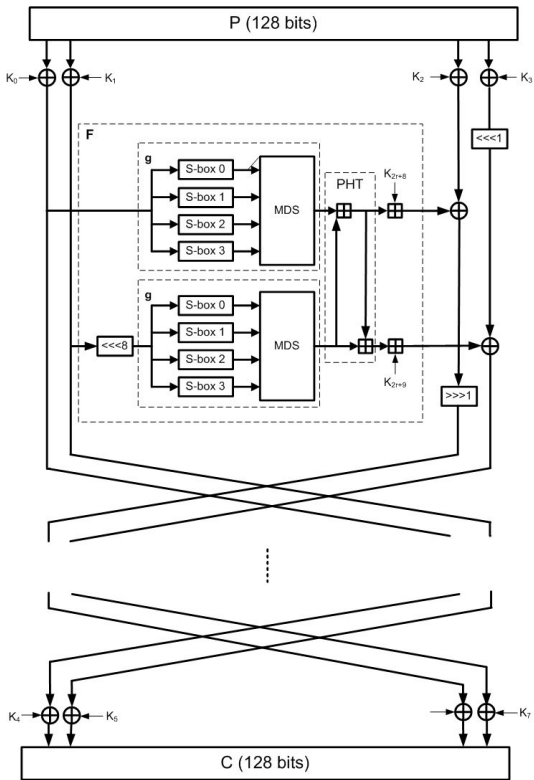


그림 1. Twofish 암호 알고리즘의 블록도
Fig. 1 Block diagram of Twofish cryptographic algorithm

g -함수내의 4-by-4 바이트 MDS 행렬 곱셈기는 Twofish 내의 주요한 확산 메카니즘으로 사용되며 원시다항식은 식 (1)과 같다.

$$\begin{aligned} & \text{Primitive Polynomial} \dots\dots\dots (1) \\ & = x^8 + x^6 + x^5 + x^3 + x + 1 \end{aligned}$$

Galois Field(28)상에서의 MDS 행렬 곱셈은 다음 식 (2)와 같이 표현된다.

$$\begin{bmatrix} Y_0 \\ Y_1 \\ Y_2 \\ Y_3 \end{bmatrix} = \begin{bmatrix} 01 & EF & 5B & 5B \\ 5B & EF & EF & 01 \\ EF & 5B & 01 & EF \\ EF & 01 & EF & 5B \end{bmatrix} \cdot \begin{bmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \end{bmatrix} \dots\dots\dots (2)$$

또한, 외부 128비트 키를 입력으로 받아 40개의 32비트 부분키 및 S-box에서 사용되는 S_0, S_1 부분키를 생성하는 키 스케줄링 부분은 암호/복호 부분의 두 g -함수와 PHT 및 추가적인 고정된 순환, RS-곱셈기 등으로 구성된다.[3][4]

III. 처리속도 향상을 위한 LUT2-MDS 블록

Twofish 암호알고리즘을 구현할 경우 처리시간이 가장 많이 소요되는 블록이 MDS 블록이다. MDS 블록은 F 함수 내에 존재하며 식 (2)와 같이 곱셈연산을 수행해야 한다. 곱셈연산 수행으로 인하여 Twofish 암호시스템의 가장 중요한 블록인 MDS 블록에서 병목현상이 발생하고 이로 인하여 처리시간의 감소를 가져온다.[5][6]

본 논문에서는 이러한 곱셈연산에 의한 처리시간의 감소를 억제하고자 LUT2(Look-Up Table and modulo-2)를 사용하여 데이터 처리시간을 높이고자 하였다.

식 (2)와 같은 MDS 연산을 식 (3)과 같이 변형한다.

$$\begin{aligned} Y_0 &= 01 \cdot X_0 + EFX_1 + 5BX_2 + 5BX_3 \dots\dots\dots (3) \\ Y_1 &= 5BX_0 + EFX_1 + EFX_2 + 01X_3 \\ Y_2 &= EFX_0 + 5BX_1 + 01X_2 + EFX_3 \\ Y_3 &= EFX_0 + 01 \cdot X_1 + EFX_2 + 5BX_3 \end{aligned}$$

이때 $01 \cdot X$ 연산의 결과값은 자기 자신이 되므로 식 (3)에서 $01 \cdot X$ 의 부분을 생략하여 정리하면 식 (4)와 같다.

$$\begin{aligned} Y_0 &= EFX_1 + 5BX_2 + 5BX_3 \dots\dots\dots (4) \\ Y_1 &= 5BX_0 + EFX_1 + EFX_2 \\ Y_2 &= EFX_0 + 5BX_1 + EFX_3 \\ Y_3 &= EFX_0 + EFX_2 + 5BX_3 \end{aligned}$$

식 (4)에서 MDS는 0x5B와 0xEF에 해당하는 연산을 수행하면 되므로 MDS의 0x5B와 0xEF에 대하여 modulo-2를 이용하여 전개하면 식 (5)와 식(6)과 같이 표현할 수 있다.

$$\begin{aligned}
 y_7 &= x_7 \oplus x_1 \dots\dots\dots (5) \\
 y_6 &= x_6 \oplus x_0 \\
 y_5 &= x_7 \oplus x_5 \oplus x_1 \\
 y_4 &= x_6 \oplus x_4 \oplus x_1 \oplus x_0 \\
 y_3 &= x_5 \oplus x_3 \oplus x_0 \\
 y_2 &= x_4 \oplus x_2 \oplus x_1 \\
 y_1 &= x_3 \oplus x_1 \oplus x_0 \\
 y_0 &= x_2 \oplus x_0
 \end{aligned}$$

$$\begin{aligned}
 y_7 &= x_7 \oplus x_1 \dots\dots\dots (6) \\
 y_6 &= x_7 \oplus x_6 \\
 y_5 &= x_7 \oplus x_6 \oplus x_5 \oplus x_1 \\
 y_4 &= x_6 \oplus x_5 \oplus x_4 \oplus x_1 \\
 y_3 &= x_5 \oplus x_4 \oplus x_3 \oplus x_0 \\
 y_2 &= x_4 \oplus x_3 \oplus x_2 \oplus x_1 \oplus x_0 \\
 y_1 &= x_3 \oplus x_2 \oplus x_1 \oplus x_0 \\
 y_0 &= x_2 \oplus x_1 \oplus x_0
 \end{aligned}$$

식 (5)와 식 (6)과 같이 곱셈연산은 단순 modulo-2 연산을 이용하면 쉽게 해결된다. 이때 식 (5)와 식 (6)을 하나의 수식으로 합하면 식 (7)과 같다. 이때 a 는 0x5B를, b 는 0xEF항을 의미한다.

$$\begin{aligned}
 y_{6a} &= x_6 \oplus x_0, & y_{7a} &= y_{7b} = x_7 \oplus x_1 \\
 y_{5a} &= y_{7a} \oplus x_5, & y_{6b} &= x_7 \oplus x_6 \\
 y_{4a} &= y_{6a} \oplus x_4 \oplus x_1, & y_{5b} &= y_{6b} \oplus x_5 \oplus x_1 \\
 y_{3a} &= x_5 \oplus x_3 \oplus x_0, & y_{4b} &= y_{4a} \oplus x_5 \\
 y_{2a} &= x_4 \oplus x_2 \oplus x_1, & y_{3b} &= y_{3a} \oplus x_4 \\
 y_{1a} &= x_3 \oplus x_1 \oplus x_0, & y_{2b} &= x_4 \oplus y_{1b} \\
 y_{0a} &= x_2 \oplus x_0, & y_{1b} &= x_3 \oplus y_{0b} \\
 & & y_{0b} &= y_{0a} \oplus x_1
 \end{aligned}
 \dots\dots\dots (7)$$

식 (7)과 같이 단순화 시킨 식에서 최소항에 해당하는 식으로 변환하면 식 (8)과 같다.

$$MDS(x_i, y_{ia}, y_{ib}) \dots\dots\dots (8)$$

식 (8)에서 $i = 0..7$ 의 범위를 갖는다. 이때 식 (8)은 MDS에 대한 곱셈 수행을 위한 최종 함수이며 이를 LUT에 고정할 경우, 기존 MDS보다 stage 연산이 감소되어 전체 처리시간의 감소를 가져오게 된다.

IV. LUT2-MDS를 사용한 Twofish 암호시스템 설계

식 (7)과 식 (8)을 이용하여 MDS를 설계하면 (그림 2)와 같다.

(그림 2)는 LUT 블록과 modulo-2 연산만으로 구성된 블록으로 곱셈연산만을 수행하게 된다.

식 (4)를 식 (7)과 같이 단순화 할때 0x5B와 0xEF에 대한 항목은 식 (7)의 a b 항으로 구별된다.

입력 8 비트씩 4블록은 MDS의 상수항과 곱셈연산을 수행하게 되고 이에 대한 출력은 8 비트씩 4블록의 출력으로 산출된다.

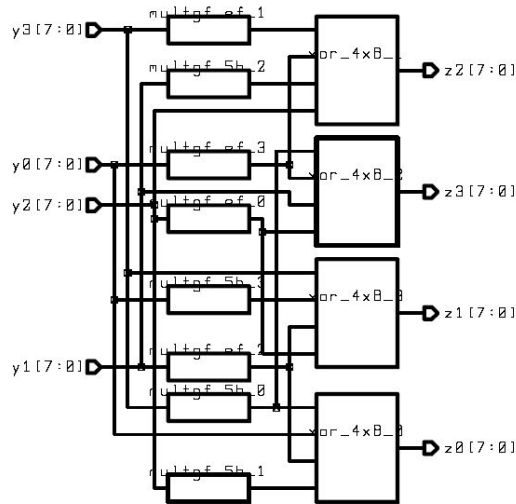


그림 2. LUT2-MDS 블록
Fig. 2 LUT2-MDS block

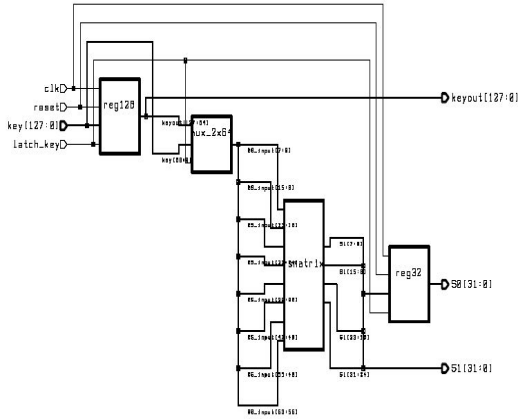


그림 3. 키 스케줄러 블록
Fig. 3 Key scheduler block

(그림 3)은 Twofish 암호시스템에서 키 데이터를 생성하는 키 스케줄러 블록이다. S-box들에 의한 S 함수와 g 함수 그리고 q 치환 기능을 수행하는 블록으로서 RS(Reed-Solomon) 곱셈에 의한 연산을 수행한다.

(그림 4)는 LUT2-MDS를 적용하여 설계한 Twofish 암호시스템의 전체 합성 회로이다. 입력데이터 128 비트, 입력 키는 128 비트이며 출력되는 암호문은 128 비트이다.

<표 1>은 LUT2-MDS를 이용한 Twofish와 기존 Twofish 암호시스템들과의 비교표이다.

<표 1>에서 보논바와 같이 제안된 LUT2-MDS 구조를 갖는 Twofish 암호시스템이 기존 암호시스템에 비하여 10% 정도 속도가 향상됨을 확인하였다.

표 1. LUT2-MDS와 기존 MDS에 대한Twofish 암호시스템 비교

	LUT2-MDS	Multiplication MDS
Throughout	68Mbps	61Mbps
Cell counter (MDS)	112	101
Total counter	3,523	3,215

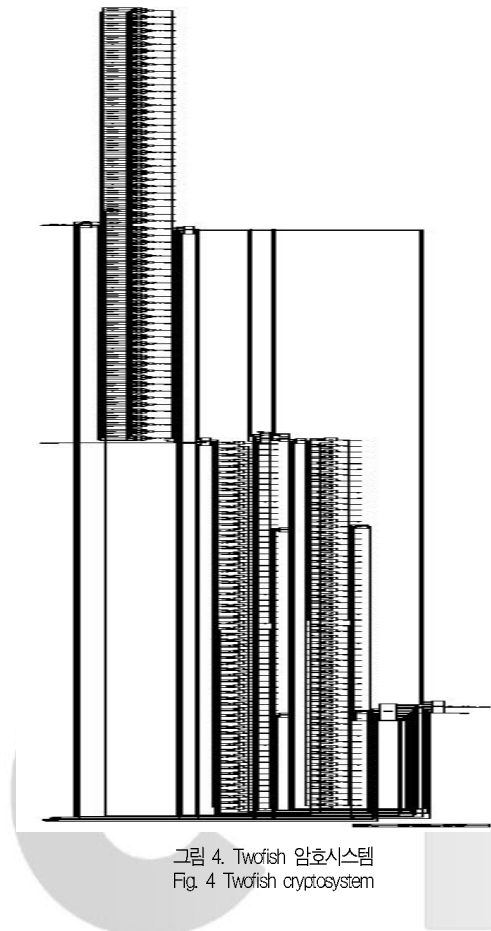


그림 4. Twofish 암호시스템
Fig. 4 Twofish cryptosystem

VI. 결 론

인터넷과 같은 네트워크 환경에 적합한 암호시스템의 기준은 속도이다. 그러므로 AES 표준 암호알고리즘인 Rijndael 암호알고리즘이 가장 적합한 암호알고리즘이지만 구현했을 경우 Rijndael 암호알고리즘은 암호/복호화의 동시 수행이 불가능하다는 이유 등으로 인하여 대중화하기가 어렵다.

그러므로 본 논문에서는 Rijndael 암호알고리즘에 비하여 다소 속도가 떨어지지만 암호/복호화기 동시에 가능하며 구현상의 용이성이 높은 Twofish 암호알고리즘의 속도를 향상시키기 위하여 LUT & modulo-2를 적용한 MDS를 설계하고 Twofish 암호알고리즘에 적용하여 Twofish 암호시스템을 Synopsys 1999.10과 MaxplusII 10.1을 이용하여 설계하였다. 설계된 암호시스템의 모의실험 결과 기존 방식의 Twofish 암호시스템보다 처리속도면에서 10% 정도 속도의 향상을 보임을 확인하였다.

그러므로 본 논문에서 제안한 LUT2-MDS를 적용한 Twofish 암호시스템은 인터넷 보안기술을 요구하는 네트워크 환경에 매우 적합한 암호시스템이라 사료된다.

참고문헌

- [1] National Institute Standards & Technology, <http://csrc.nist.gov/encryption/aes/>
- [2] Pawel Chodowice, Kris Gaj, "Implementation of the Twofish Cipher Using FPGA Devices", <http://csrc.nist.gov/encryption/aes>
- [3] J. Daeman, V. Rijmen, "AES Proposal : Rijndael" <http://csrc.nist.gov/encryption/aes>
- [4] Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall, Niels, "Twofish : A 128 bit Block Cipher", <http://counterpane.com/twofish.html>
- [5] 정우열, 이선근, "다중 비선형 S-box 함수를 이용한 블록 암호시스템 설계", 한국OA학회, 제 6권 제 2, pp.90-96, 2001
- [6] 이선근, 정우열, "대용량 고속화 수행을 위한 변형된 Feistel 구조 설계에 관한 연구", 한국컴퓨터정보학회, 제 10권 3호, pp.183-188, 2005

저자 소개



정우열

현재 한려대학교 멀티미디어정보통신
공학과 교수
<관심분야> 암호시스템, SoC 설계



이선근

2003년 2월 원광대학교 전자공학과
공학박사
2004년~현재 원광대학교 전기전자
및정보공학부 강의교수
<관심분야> 암호시스템, SoC 설계