韓國 컴퓨터情報學會 論文誌
第10卷 第3號, 2005. 7.

2005-10-3-2-1

대용량 고속화 수행을 위한 변형된 Feistel 구조 설계에 관한 연구

이 선 근*, 정 우 열**

Design of modified Feistel structure for high-capacity and high speed achievement

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

요 약

블록암호알고리즘의 기본 구조인 Feistel 구조는 순차처리 구조이므로 병렬처리가 곤란하다. 그러므로 본 논문은 이러한 순차처리 구조를 변형하여 Feistel 구조가 병렬처리가 가능하도록 하였다. 이를 이용하여 본 논문은 병렬 Feistel 구조를 가지는 DES를 설계하였다. 제안된 병렬 Feistel 구조는 자체의 구조적 문제 때문에 pipeline 방식을 사용할 수 없어 데이터 처리속도와 데이터 보안사이에서 trade-off관계를 가질 수밖에 없었던 DES등과 같은 블록암호알고리즘의 성능을 크게 향상 시킬 수 있었다. 그러므로 Feistel 구조를 적용한 SEED, AES의 Rijndael, Twofish 등에 제안된 방식을 적용할 경우 지금보다 더욱 우월한 보안 기능 및 고속의 처리능력을 발휘하게 될 것이다.

Abstract

Parallel processing in block cryptographic algorithm is difficult, because Feistel structure that is basis structure of block cryptographic algorithm is sequential processing structure. Therefore this paper changes these sequential processing structure and Feistel structure made parallel processing to be possible. This paper that apply this modified structure designed DES that have parallel Feistel structure. Proposed parallel Feistel structure could prove greatly block cryptographic algorithm's performance such as DES and so on that could not but have trade-off relation the data processing speed and data security interval because block cryptographic algorithm can not use pipeline method because of itself structural problem. Therefore, modified Feistel structure is going to display more superior security function and processing ability of high speed than now in case apply way that is proposed to SEED, AES's Rijndael, Twofish etc. that apply Feistel structure.

▸ Keyword : 병렬처리(Parallel Processing), 블록암호 알고리즘(Block Cryptographic Algorithm), Feistel 구조(Feistel Structure)

• 제1저자 : 이선근
• 접수일 : 2005.04.20, 심사완료일 : 2005.05.25
* 원광대학교 전기전자및정보공학부 강의교수, ** 한려대학교 정보통신공학과 교수

Ⅰ. 서 론

현존하는 암호시스템들은 통계적 취약성을 모두 내포하고 있다. 그러므로 암호시스템 자체가 해석되어지는 것은 시간과 노력의 차이만 있을 뿐, 완전하게 막아낼 수는 없다. 즉 비도와 처리시간과는 trade-off 관계가 존재하게 된다. S/W적으로 구현된 암호시스템의 경우, 암호시스템의 구현 및 운용이 용이하다는 장점이 있으나 데이터량이 급증하거나 동시다발적인 다중화 포화상태일 경우 컴퓨터가 다운되거나 처리시간이 오래 걸린다는 단점을 가지게 된다. 이러한 단점을 없애기 위하여 H/W적으로 구현된 암호시스템이 등장하였으나 아직까지는 크게 현실적이며 효율적이지 못하다. 이러한 이유는 암호시스템을 H/W 또는 S/W로 구현할 때 구현의 기본이 되는 암호 알고리즘에 문제점이 내포되어 있기 때문이다. 즉, 효율적인 키 분배와 키 생성, 암호문과 평문사이에서의 변환과정 등이 구조적이거나 수학적인 배경을 가지고 있기 때문에 고속 및 고비도의 암호시스템을 구현하는데 장애로써 존재하기 때문이다[1][2][3].

본 논문에서는 MAC(Message Authentication Code) 등과 같은 네트워크 서비스 실현을 위해서 사용되어지고 있는 대칭형 암호시스템인 DES의 기본구조인 Feistel 구조를 병렬처리가 가능하도록 변형시켜 키의 길이에 처리속도가 비례관계를 갖지 않도록 하며, 데이터의 고속처리가 가능하도록 하는 병렬 Feistel 구조를 제안한다. 이러한 병렬 Feistel 구조는 DES의 기본구조에서만 적용되는 것이 아니고 키 생성 및 분배에 해당하는 키 생성 알고리즘에도 적용하며 DES의 안전도에 절대적으로 기인하는 키 생성 단계를 보다 효율적으로 구성함으로서 키의 길이는 길고 데이터처리는 고속으로 수행할 수 있도록 하는데 주요한 기능을 수행하게 될 것이다.

Ⅱ. 대칭형 암호시스템

현대 암호학 기술은 암호알고리즘의 공개와 키의 보호라는 특징을 가지고 있다. 암호시스템은 알려진 알고리즘에 입력되는 평문과 키를 섞어 암호문을 만들어 낸다. 즉, 키를 모르면 도저히 원래의 정보를 회복할 수 없으나 키를 알면 손쉽게 원래의 정보를 복원한다는 것이다.

DES는 64비트의 평문을 64비트의 암호문으로 만드는 블록 암호 시스템으로 64비트의 키를 사용한다. 이 64 비트의 키(외부 키) 중 56비트는 실제 키(내부 키)가 되고 나머지 8비트는 검사용 비트가 된다.

DES의 구조는 16 round의 iteration을 수행하며 각 round마다 transposition과 substitution을 거친 평문과 56비트의 내부키에서 나온 48비트의 키가 섞여 암호문을 형성하고 복호화는 암호화 과정과 동일하나 사용되는 키만 역순으로 작용한다.

DES의 기본 요소는 평문을 64비트로 blocking 하는 과정이 있으며 transposition, substitution를 DES에서는 table에 의해 수행하게 된다. 또한 확장(expansion), 압축(compaction)과 배타적 논리합(exclusive-OR)이 기본 연산이 된다. DES의 키는 0과 127사이의 8개의 십진수로 구성되어 있는데 키는 거기서 어떤 규칙성이 나타나지 않도록 난수 발생기가 무작위로 골라낸 숫자들로 만들어진다. DES는 평문 64비트들이 치환(permutation)을 거친 후 32비트씩 두 부분으로 나뉘어진다. 오른쪽 부분은 키와 f라고 표시되는 알고리즘에 의해 섞이고, 그 결과는 왼쪽부분과 Ex-OR 연산으로 결합되어 새로운 round의 시작이 된다. 이런 과정으로 각 블록은 16 round의 처리과정을 반복하게 되며 키 역시 각 round를 지날 때마다 새로운 형태로 바뀌어진다. 각각의 round에 대한 키는 이전 round에서 수행되었던 키에 의해 그 형태가 규정되며 이것이 다시 배타적 논리합으로 암호화되고 있는 평문의 내용과 결합된다. 평문의 암호화 알고리즘의 전체적인 구조는 (그림 1)과 같다. 64비트의 입력은 미리 정해진 표에 의해 initial permutation을 수행한다. 치환된 결과의 64비트 중 왼쪽 32비트를 L_0, 오른쪽 32비트를 R_0라 한다. 16 round 끝난 후 마지막

에서 다시 합쳐져서 정해진 표와 같은 역 초기 치환을 거쳐 64비트의 출력을 산출한다.

i번째 round가 시작되기 전의 왼쪽 32비트를 L_{i-1}이라 표시하고, 오른쪽 32비트를 R_{i-1}일 경우, i번째 round의 왼쪽 32비트 L_i는 바로 이전 round의 오른쪽 32비트 R_{i-1}을 그대로 쓴다. f 에서는 S-box가 있으며 compaction, expansion등의 방법이 쓰인다.

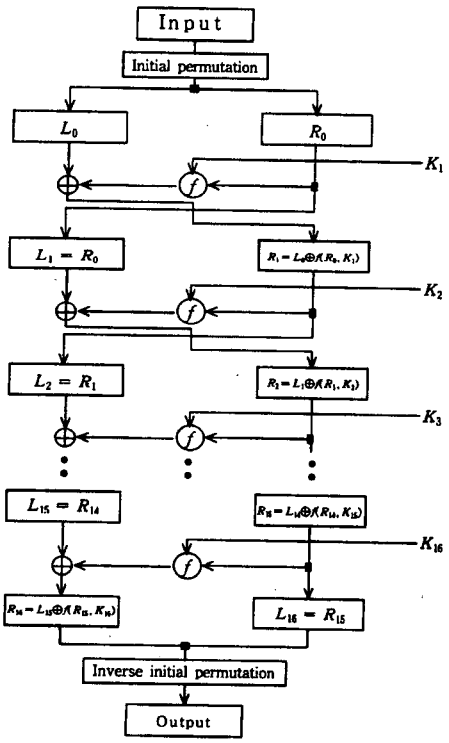


그림 1. DES의 암호화 과정
Fig. 1 DES encryption process

알고리즘 f는 (그림 2)와 같다. 이 알고리즘은 R_{i-1}과 키 K_i를 받아들여서 $f(R_{i-1}, K_i)$를 출력하며 이 출력결과와 L_{i-1}과의 Ex-OR에 의하여 R_i를 생성하게 된다.

(그림 2)의 expansion은 미리 정해진 표에 의해서 수행된다. 이 결과 48비트는 6비트씩 나누어져 처음 6비트는 S-box S_1에, 다음 6비트는 S_2에, …순으로 입력되고 각 S_1, S_2, …, S_8들은 각각 4비트를 출력하게 되며 이 8×4=32비트들은 치환표에 의해 치환된 후 32비트의 출력인 $f(R_{i-1}, K_i)$로 된다. 이러한 S-box의 역할은 미리 정해진 표에 의해 각각 6비트를 받아들여서 4비트로 출력을 하게 되는데 이러한 특징으로 인하여 각 S-box들은 비선형 특성을 가지게 된다. DES의 키 생성 알고리즘은 (그림 3)과 같다. 여기에서 키는 암호화/복호화 키이다[3][5].

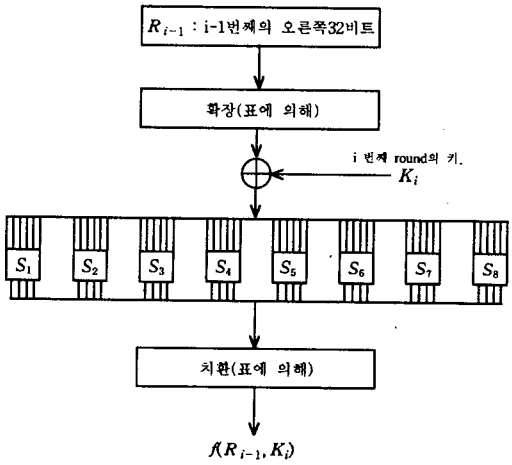


그림 2. f함수의 구조
Fig. 2 f-func. structure

(그림 3)에서 56bits의 키는 표에 의해 치환이 수행되어지고 그 이후 키는 28비트씩으로 나뉜다. 만약 각 round 키가 동일하다면, DES의 수학적 복잡도, 즉 비도가 저하될 수 있다.

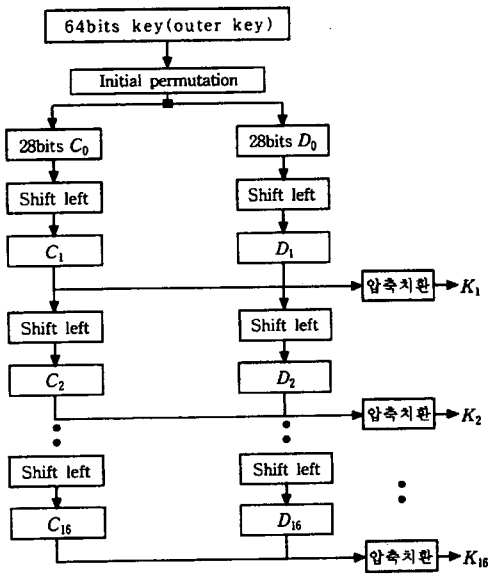


그림 3. 키 생성 과정
Fig. 3 Key generation

III. 병렬 Feistel 구조를 갖는 대칭형 암호시스템

(그림 4)와 같은 Feistel 암호화 시스템은 r-round의 과정을 거치는 동안에 평문 2m 비트(L_0, R_0)에 대하여 암호문 m 비트(L_r, R_r)를 생성하게 되며 이때 생성방정식은 식 (1)과 같다[3][4][5]. 여기에서 $r \geq 1$. For $1 \leq i \leq r$.

$$L_i = R_{i-1} \qquad \cdots\cdots (1)$$
$$R_i = L_{i-1} \oplus F(R_{i-1}, K_i)$$

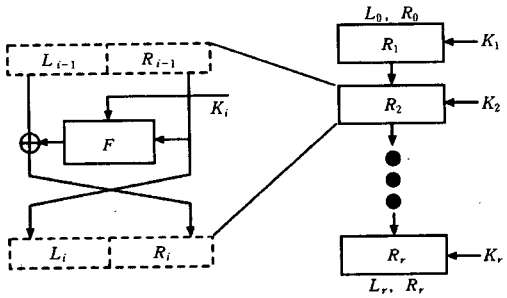


그림. 4 기본 Feistel 구조
Fig. 4 Based Feistel structure

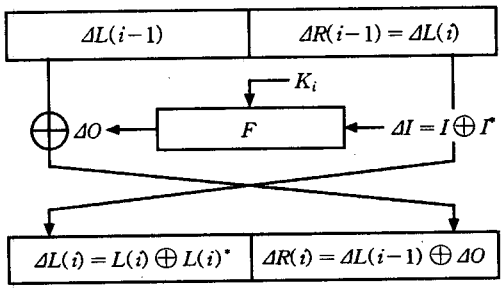


그림. 5 Feistel 구조에서 DC 특성
Fig. 5 DC characteristic in Feistel

DC(Differential Cryptanalysis) 해석법은 입력과 출력의 관계로부터 Chosen-plaintext를 이용하여 공격하는 방법으로써 (그림 5)에서와 같은 특징을 Feistel 구조가 가지고 있기 때문에 Feistel 구조에 대한 변형은 DC 해석법에 대한 방어의 기능을 향상시킬 수 있다[6][7].

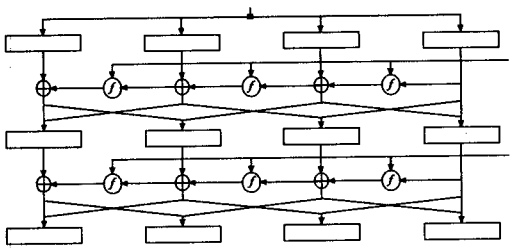
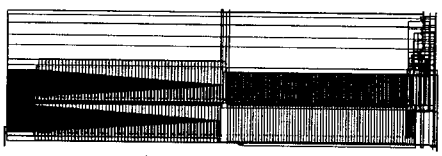


그림 6. 변형된 Feistel 구조
Fig. 6 Modified Feistel structure

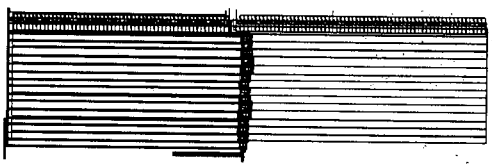
(그림 6)은 일반적인 DES에 대한 보안기능 강화 및 구현을 용이하게 하기 위한 방법으로써 Feistel 구조를 병렬처리가 가능하도록 한 구조이다. 입력에 대하여 초기 치환을 수행하고 128비트들에 대하여 32 비트 4개의 데이터를 일반 DES와 같이 처리하도록 하였다. 그러므로 본 논문에서 제안한 방식은 기본 32 비트 처리를 이용하기 때문에 별도의 데이터 포맷이 필요 없고 초기 및 역치환 수행시에도 비도의 차이가 없어지게 된다. 또한 round 처리시에도 16 round를 수행할 필요가 없이 단지 1회의 round로도 보안기능을 수행할 수가 있다. 이때 1회 round 수행에 따르는 비도의 저하가 문제될 수 있다. 그러나 단순한 Feistel 구조에서 1회 round 수행은 안전성에 문제가 야기될 수 있지만 제안된 Feistel 구조는 병렬적인 상호 신호의 흐름으로 인하여 16회 round 수행의 결과를 가져온다. 그러므로 제안된 병렬 Feistel 구조는 기존 16 round에 준하는 안전성을 제공함으로서 안전성에 커다란 문제점을 나타내지 않는다[8].

(그림 7)은 일반 DES의 H/W 구현을 수행한 것으로써 (그림 7(a))는 단일 round만을 구현하여 제어신호에 의하여 16번의 round 처리를 수행한것이며

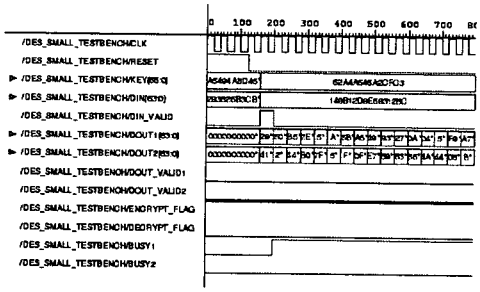
(그림 7(b))는 전체 16 round에 대하여 모든 부분을 H/W로 구현한 것이다. (그림 7)의 모든 부분들의 시스템은 round처리에 대한 부분을 수행하는 동안 출력을 얻을 수 없기 때문에 encryption을 수행하기 위해서는 시스템 처리시간이 절대적으로 늘어나게 되며 H/W의 구현시에도 구현의 어려움이 존재하게 된다. (그림 7(c))는 이러한 기존의 DES에 대한 모의실험 결과이다.



(a) 단일 round 사용한 DES



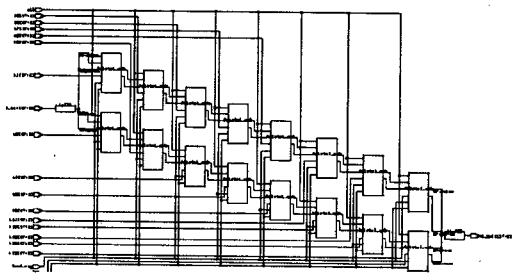
(b) 16 round를 사용한 DES



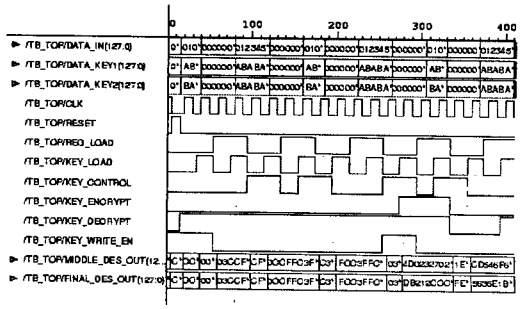
(c) 기존 시스템의 모의실험결과

그림 7. 기존의 DES 시스템
Fig. 7 Existed DES system

(그림 8(a))는 본 논문에서 제안한 병렬처리 DES의 구조이다. 입력값에 대한 초기 치환을 수행한 후 바로 32비트씩 분할되어 동시에 f 함수와의 계산을 수행하게 되고 1 round만을 수행하고 나서 바로 출력이 나타나게 된다. 그럼으로써 시스템 처리시간 및 H/W의 구현상의 문제를 용이하게 해결할 수 있었다. (그림 8(b))는 제안된 시스템의 모의실험결과이다.



(a) 제안한 병렬처리 DES의 구조



(b) 제안된 시스템의 모의실험결과

그림 8. 제안된 병렬처리 DES 시스템
Fig. 8 Proposed parallel processing DES system

Ⅳ. 결론

본 논문에서는 정보통신의 눈부신 발달과 인터넷의 급격한 확산으로 인한 여러 가지 첨단기능을 수행함에 있어 보다 안전하게, 보다 투명성이 있는 네트워크의 보장을 요구하며, 보다 빠른 네트워크의 성능을 기대하기 위하여 기존의 DES를 H/W의 구현에 대하여 병렬로써 처리하도록 하였다. 기존의 경우, 입력을 64비트로 하였으나 본 시스템은 128 비트로써 처리하도록 하였다.

본 논문에서 사용된 병렬기법의 Feistel 구조는 처리시간 및 H/W 구현의 용이성 등을 매우 효율적으로 처리하는 것으로 사료됨으로써 전자상거래, 전자화폐, 전자서명 등의 여러 가지 첨단기능을 수행하며 미래에는 더욱 진보된 서비스를 제공하게 될 것이다.

본 논문에서는 DES의 기본 구조인 Feistel 구조를 병렬로 변화시킨 병렬 Feistel 구조를 가지는 DES를 제안하였다. 제안된 병렬 Feistel 구조는 자체의 구조적 문제 때문에 pipeline 방식을 사용할 수 없어 데이터 처리속도와 데이터 보안사이에서의 trade-off관계를 가질 수밖에 없었던 DES의 성능을 크게 향상시킬 수 있으며 더불어 Feistel 구조를 적용한 SEED, AES인 Rijndael, Twofish 등에 제안된 방식을 적용할 경우 지금보다 더욱 우월한 보안 기능 및 고속의 처리능력을 발휘하게 될 것이다.

참고문헌

[1] C. E. Shannon, "Communication theory of secret systems", Bell Syst. Tech. J., vol. 28, pp656-715, Oct. 1949

[2] H. Feistel, "Cryptography and Computer Privacy", Sci. Amer., vol. 228, no. 5, pp.15-23, May 1973

[3] D. E. Denning, "The data encryption standard : Fifteen years of public security", Dist. Lecture, 6th Annual Comp. Security Appl. Conf., IEEE Comp, Soc. Press, pp x-x v, 1990

[4] A. Shimizu, S. Miyaguchi, "Fast Data Encipherment Algorithm, FEAL", Proc. of Eurocrypt 91, 1991

[5] X. Lai, "On the Design and Security of Block Ciphers", ETH Series in Information Processing, vol. 1, 1992

[6] E. Biham, A. Shamir, "Differential Cryptanalysis of Data Encryption Standard", Springer-Verlag, 1993

[7] Lee Seon Keun, "효율적인 정보검출을 위한 NIDS 시스템 설계에 관한 연구", 한국컴퓨터정보학회 논문지, 제 8권 제 3호, pp.156-162, 2003

[8] Jung Woo Yeol, "다중 비선형 S-box 함수를 이용한 블록 암호시스템 설계", 한국OA학회 논문지, 제 6권 제 2호, pp.90-96, 2001

저자소개

이 선 근

2003년 2월 : 원광대학교 전자공학과 공학박사

2004년~현재 : 원광대학교 전기전자및정보공학부 강의교수

〈관심분야〉 암호시스템, SoC 설계

정 우 열

현재 : 한려대학교 정보통신공학과 교수

〈관심분야〉 암호시스템, SoC 설계