通信情報保護學會誌
第4卷 第1號, 1994. 3

DES의 선형 해독법에 관한 해설(Ⅲ)

김 광 조*

요 약

본 해설은 1993년 일본의 미쓰비스 전기(주)의 마쯔이가 세계 최초로 발표한 DES의 선형해독법을 1994년 1월 27일부터 1월 30일까지 개최된 SCIS' 94에서 기 발표한 선형 해독법의 고속화 방법을 제안한 논문을 번역하여 소개한다. 이 방법으로 2^{43}개의 랜덤한 평문과 암호문이 주어진다면 DES의 모르는 키 56비트를 80%의 성공확률로 구할 수 있다. 또한, 이론을 실증하기 위해 50일간 12대의 W/S(PA-RISC, 99MHz, 125MIPS)을 이용하여 수행한 결과 DES의 기지 평문 공격에 성공하였다고 한다.

1. 서 론

마쯔이는 참고 문헌 [1]과 [2]에서 블럭암호의 새로운 해독법인 선형해독법(Linear Cryptanalysis)를 제안하고 DES에 적용하였다. 그 결과 DES를 기지 평문공격 또는 암호문 단독공격 방법이 키의 전수검사 방법보다 고속으로 해독 가능함을 제시하였다. 그것에 이어 본 해설은 선형 해독법을 개선한 새로운 이론을 기술하고 이론의 실증을 위하여 DES의 해독 실험한 결과도 소개한다.

DES는 미국 상무성 표준국(NBS)가 공개 가능한 암호 알고리즘 공모에 대하여 IBM이 응모한 방식을 근간으로 미국의 NSA가 안전성을 평가, 그 구성에 수정을 가하여 결정된 것이다[3]. 1977년 연방 정보 표준 규격(FIPS)로 채택[4]된 이래 현재까지 사실상 세계 표준 암호로서 금융망을 중심으로 세계적으로 널리 사용되고 있다. 한편 DES의 안전성에 관하여 논의는 발표 초기부터 활발하였다. 중요한 2가지 점은 암호화 키의 크기에 대한 적정성과 설계기준의 비공개성이다. 암호화 키의 크기에 관하여는 Diffie와 Hellman[5]이 56비트는 너무 작으므로 전수 검사가 가능하리라고 지적하여 NBS와 논쟁이 있었다. 그 후 암호화 키의 전수 검사를 고속화하는 연구가 널리 이루어졌으나 현 상태에서 해독 전용의 고속 LSI 제작은 다수의 병렬 처리가 필요하므로 구현한 예는 아직 없다.

NSA가 DES의 설계에 협력하여 설계 기준을 비공개로 요청한 것을 설계자만이 해독할 수 있는 trapdoor가 있다는 견해도 있었다. 이것은 DES의 공개적인 해독법 연구, 특히 내부 테이블인 S-box의 구조 해석 연구가 추진되어 그 결과가 Hellman[6] 등에 의하여 최초로 제시되었다. 그들은 S-box구성 자체에 외견상 랜덤성을 가진 trapdoor가 실제 삽입되어 있다는 몇가지 지적사항 중 S-box에는 통계적인 특성이 있다는 점을 특히 지적하였다. 그들의 논문에서는 상당히 중요한 2가지 점의 지적이 되어 있는데, 그 하나는 S-box의 입출력 변화간의 상관성이고, 또 하나는 S-box의 입출력간의 선형성이다. S-box의 입출력

* 정회원, 한국전자통신연구소 실장

변화간의 상관성은 그후 Desmedt[7]등이 Hellman의 결과를 확장하여 몇가지 새로운 성질을 규명하였다. 이런 국소적인 성질을 암호해독에 응용한 것이 Biham과 Shamir에 의한 Differential Cryptanalysis (DC)이다. Biham과 Shamir는 S-box의 2개의 입력에 대응하는 출력에 대해 각각 Exclusive-Or 값의 변화를 확률적으로 분석하여 S-box의 국소적 성질을 알고리즘 전체로 확대하는 것에 성공하였다[8]. 이어서 그들은 DES가 선택 평문 공격 조건하에 이론상 해독 가능한 점을 제시하였다[9]. 이것은 암호화키의 전수 검사보다 빠른 DES의 해독법으로 최초의 결과였다.

S-box가 부분적으로 선형함수에 가깝다는 것은 Hellman 등에 의해 지적되었으며 Shamir[10]와 Rueppel[11]은 몇 개의 S-box에 대하여 특정 입출력 비트간에 선형에 가까운 관계가 성립한다는 것을 제시하였다. 이러한 성질을 알고리즘 전체로 확장한 것이 선형해독법이 최초이다. 마쯔이는 S-box의 입출력 비트의 parity의 변화를 확률적으로 고찰하여 평문, 암호문과 암호화 키간의 특정 비트간에 성립하는 확률적 선형 관계식을 유도하였다[1]. 이 관계식을 이용하여 2^{47}개의 랜덤한 기지 평문이 주어지면 암호화 키를 찾을 수 있다는 것을 제시하였다. 이것을 암호화 키의 전수 검사보다 고속으로 DES의 기지평문 공격으로 최초의 결과이다. 이 해독법은 평문이 특별한 확률분포를 가질 때 암호문 단독 공격도 가능함도 발표하였다[2]. 최근에는 DC과 선형해독법간의 관계에 관한 논문[12],[13]이나 선형 해독법을 이용한 DC의 고속화등이 발표되고 있다.

본 해설은 선형해독법을 이용한 기지평문 공격에 필요한 평문의 수를 감소시킬 수 있는 2가지 새로운 방식을 소개한다. 하나는 새로운 해독 방정식을 도입하는 것이고, 또 하나는 해의 후보에 대하여 신뢰도 정보를 부여하는 것이다. 이전의 해독 방정식은 1단부터 $n-1$단까지 선형 근사하고 n단의 확장키를 구하는 방식이 없으나 여기서는 2단부터 $n-1$단까지 선형근사하고 1단과 n단의 확장키를 동시에 구하는 방식으로 해독에 필요한 평문 수를 감소시켰다. 또한 이전의 해독 방식을 해독 방정식으로부터 구한 확장키를 하나로 고정하고 남은 키 정보를 탐색하는데 반해 지금은 확장키의 복수 후보에 신뢰도 정보를 주어 해독 성공 확률을 높였다. 이 결과 DES는 2^{43}개의 랜덤한 기지 평문으로 암호화 키를 높은 확률로 추정하는 것이 가능함을 시사한다. 이 평문의 수는 소프트웨어로 실현하는 것이 가능한 수치로 마쯔이는 이론의 실증을 위해 실제 컴퓨터를 이용하여 해독 실험을 하였다. 실험은 2^{43}개의 평문을 랜덤하게 생성하고 암호화하면서 필요한 정보를 축적하고 최종적으로 암호화 키를 추정하는 것으로 모두 소프트웨어로 실현하였다. 미쓰미시 전기(주)의 제품인 W/S(ME/R, PA-RISC, 99MHz)을 12대 사용하고 50일간 계산한 결과 올바른 암호화 키를 구하였다. 이것은 DES를 소프트웨어로 분석한 최초 해독 실험 결과이다. 2^{43}개라는 평문수는 1 Gbit/s 통신로에 약 1주일간의 정보량에 상당한다. 현재의 통신로로 이러한 양의 평문과 암호문을 동시에 구하는 것을 생각하기 어려우나 통신로를 소프트웨어로 시뮬레이션하는 것은 실현 가능한 시대이다.

2. 준 비

(그림1)과 (그림2)는 본 해설에서 취급하는 DES의 암호화 과정 및 F 함수의 구성도이다. 암호화 과정 중 초기 전치 IP및 최종 전치 IP^{-1}는 1대 1 사상이므로 생략한다. 본 해설에는 다음의 기호를 사용하며 특히 단수에 의존하지 않는 경우는 단수를 표시하는 첨자를 생략한다. 또한 각 그림에 있어서 오른쪽은 하위로하고 특히 최하위 비트는 0번째 비트로 약속한다.

- P : 평문 64 비트
- C : 암호문 64 비트
- P_H, P_L : 평문의 상위 32비트, 하위 32비트

- C_H, C_L : 암호문의 상위 32비트, 하위 32비트
- X_i : 제 i 단에서 F 함수의 32비트 입력
- K_i : 제 i 단의 확장키 48 비트
- $F_i(X_i, K_i)$: 제 i 단의 F 함수
- $A[i]$: A의 제 i번째 비트
- $A[i, j, ..., k]$: $A[i] \oplus A[j] \oplus \ldots \oplus A[k]$

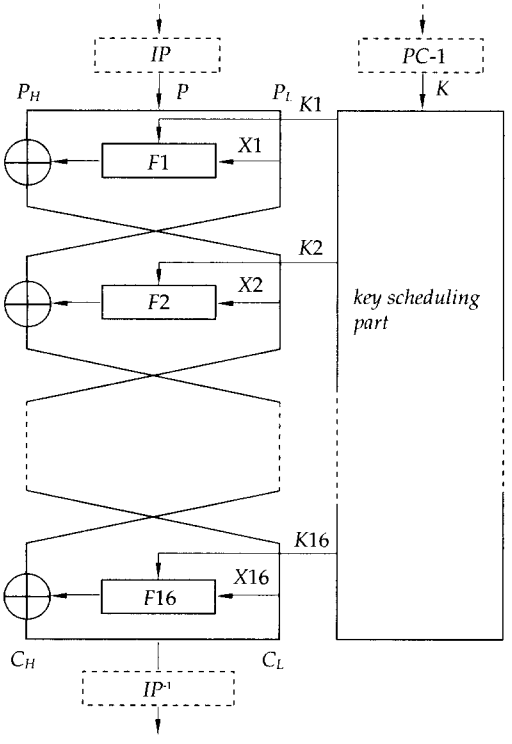


그림 1 : DES의 암호화 과정

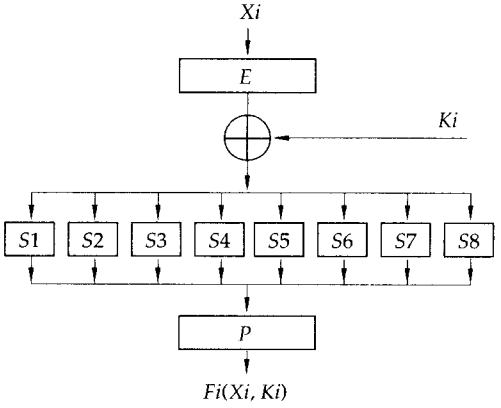


그림 2 : DES의 F 함수

3. 선형 해독법의 개요

본 장에는 참고문헌 [1]을 기반으로 선형 해독법의 개요를 간단히 기술한다. 기술 내용은 참고문헌[1]의 내용을 개선한 것으로 특히 3.1.2 및 3.3.2가 중심적인 개선 내용이다.

3.1 선형해독법의 원리

3.1.1 암호 알고리즘의 선형 근사

선형 해독법의 최초 목표는 랜덤하게 주어진 평문 P와 대응하는 암호문 C 및 고정된 키 K에 대하여 유의 확률 p로 다음과 같은 형태의 선형 근사식을 구성하는데 있다.

$$P[i_1, i_2, \ldots, i_a] \oplus C[j_1, j_2, \ldots, j_b] = K[k_1, k_2, \ldots, k_c] \quad (1)$$

여기서, $i_1, \ldots, i_a, j_1, \ldots, j_b, k_1, \ldots, k_c$는 고정된 비트 위치이다. 식(1)의 양변은 각 1비트의 정보를 나타내면 유의 확률이란 $p \neq 1/2$를 의미한다. 실제 이 식의 성공하면 암호 해독자는 maximum likelihood법을 이용하여 암호화 키의 1비트 $K[k_1, k_2, \ldots, k_c]$를 기지 평문공격으로 추정하는 것이 가능하다.

[알고리즘1]

Step 1 식(1)의 좌변을 0으로하는 기지 평문과 대응하는 암호문 쌍의 갯수를 T라 한다.

Step 2 $T > N/2$(N는 기지 평문 수)라면, $K[k_1, k_2, \ldots, k_c] = 0$($p > 1/2$인 때) 또는 1($p < 1/2$ 인 때)로 추정한다. $T < N/2$(N은 기지 평문수)라면, $K[k_1, k_2, \ldots, k_c] = 1$ ($p > 1/2$ 인 때) 또는 0($p < 1/2$ 인 때)으로 추정한다.

일반적으로 평문과 암호문의 관계를 서로 바꾸어도 암호화 키 1비트를 구할 수 있다. 추정의 성공 확률은 기지평문 수 N과 식(1)의 성립확률 p로

결정되지만, N나 $|p-1/2|$의 값이 크면 분명히 그 확률이 높아질 것이다. 이후 식(1)의 형태를 갖는 선형 근사식 가운데, $|p-1/2|$값이 최대로 되는 경우를 최량 확률이라 한다. 선형해독법의 이론은 다음의 문제를 해결한다.

$Q1$ 암호 알고리즘의 선형 근사식의 구성

$Q2$ n단 DES의 최량 표현과 최량 확률의 도출

$Q3$ 알고리즘1을 이용하여 해독 성공 확률의 계산

$Q1$은 3.2절에서 다루고 $Q2$은 컴퓨터에 의한 검색으로 결정되고 $Q3$은 보조 정리 3으로 확률을 계산한다.

3.1.2 기지 평문 공격법

실제 n단 암호의 효율적인 해독을 위해 n-2단의 최량 표현을 이용한다. 즉, 최초의 단은 확장키 K_1을 이용하여 암호화되고, 최종단은 확장키 K_n으로 복호화되는 것으로 생각하여 근사식중엔 F함수를 집어 넣는 것이다. 그 결과 식(2)와 같이 최량 확률을 갖는 n-2단의 근사식을 구할 수 있다.

$$\begin{aligned} &P[i_1, i_2, \ldots, i_a] \oplus C[j_1, j_2, \ldots, j_b] \\ &\oplus F_1(P_1, K_1)[u_1, u_2, \ldots, u_d] \\ &\oplus F_n(C_1, K_n)[v_1, v_2, \ldots, v_c] \\ &= K[k_1, k_2, \ldots, k_c] \end{aligned} \quad (2)$$

식(2)에 들린 K_1과 K_n을 대입시키면 식의 유의성을 감소한다. 즉, 식(2)의 성립 확률은 보다 1/2에 근접할 것이다. 따라서, 해독자는 다음과 같이 maximum likelihood법을 이용하여 K_1, K_N 및 $K[k_1, k_2, \ldots, k_c]$를 기지 평문공격으로 효율적인 추정이 가능할 것이다.

〔알고리즘2〕

Step 1 각 K_1과 K_n의 후보값을 $K_1^{(i)}(i=1, 2, \ldots)$과 $K_n^{(j)}(j=1, 2, \ldots)$로 하고, 각 후보의 쌍$(K_1^{(i)}, K_n^{(j)})$에 대해서 식(2)의 우변을 0으로 하는 기지평문과 암호문의 쌍의 수를 $T_{i,j}$로 한다.

Step 2 $T_{i,j}$의 최대, 최소치를 각각 T_{max} 및 T_{min}으로 하고 N을 기지 평문수라 하면,

- $|T_{max}-N/2| > |T_{min}-N/2|$이면 T_{max}에 대응하는 K_1과 K_n을 채택하고 식(2)의 우변은 $0(p > 1/2$인 경우) 또는 $1(p < 1/2$인 경우)로 추정한다.
- $|T_{max}-N/2| > |T_{min}-N/2|$이면 T_{min}에 대응하는 K_1과 K_n을 채택하고 식(2)의 우변은 $1(p > 1/2$인 경우) 또는 $0(p < 1/2$인 경우)로 추정한다.

일반적으로 평문과 암호문의 관계를 서로 바꾸어도 더 많은 확장키를 구할 수 있는 데 이를 위해서는 다음 문제는 해결하여야 한다.

$Q4$ 알고리즘 2를 이용한 해독 성공 확률의 계산

실제 이 값은 직접 계산 가능한 형태의 수식으로 나타내는 것은 곤란하다. 그러나, 짧은 단수의 암호에 대해 해독 실험결과를 근거로 보다 긴 단수의 암호에의 해독 성공확률을 예측할 수 있다(보조정리 4참조). 이것은 실용적으로는 충분한 결과라고 생각한다.

3.2 선형 해독법의 국소 이론

3.2.1 S-box의 선형근사

S-box의 선형 근사 방법으로 특정 입/출력 비트가 S-box의 64개 입력 중 몇 회가 일치하는가를 조사한다. 예를들면, 3번째 S-box의 입력의 3번째 비트는 출력의 1번째 비트와 64회 중 38회가 동일한 값을 갖는다. 즉, $S_3(x)[1]=x[3]$이 확률 38/64=0.59로 성립함을 의미한다.

모든 S-box에 대하여 몇 개의 입력 비트의 Excl-

usive Or값과 몇 개의 출력 비트의 Exclusive Or값이 64개 중 몇 개가 일치하는가를 우선 조사한다.

▨ 정 의 1

S-box $S_a(a=1, 2, \ldots, 8)$에 있어서, S_a의 입력 64개 중에 $1\leq\alpha\leq63$으로 마스크된 입력 비트 위치와 $1\leq\beta\leq15$로 마스크된 출력 비트의 Exclusive Or값이 일치되는 경우의 수를 $NS_a(\alpha, \beta)$로 표현한다. 즉,

$$NS_a(a, b)=\#\{x|0\leq x<64, \quad Parity(x\cdot\alpha)=Parity(S_a(x)\cdot\beta)\}$$

여기서 "•"는 비트 product 연산을 의미한다.

◐ 예 1 S-box의 최량 선형 근사식

$$NS_5(16, 15)=12 \tag{3}$$

이 값이 32가 되지 않는 경우 S-box의 입출력 비트간에 상관이 있다고 생각된다. 예를들면, 식(3)은 S_5 입력의 4번째 비트는 확률 12/64=0.19로 모든 출력비트의 Exclusive Or와 같아진다는 것을 의미한다. 식(3)은 모든 S-box중에 편차가 가장 큰 것, 즉 $|NS_a(i, j)-32|$를 최대로 하는 것으로 Shamir[10]에 의해 최초로 지적된 바가 있다. 또한 S-box의 구성법에 의해 다음과 같은 사실을 쉽게 알 수 있다.

▨ 보 조 정 의 1

(1) $NS_a(i, j)$는 짝수 값을 갖는다.

(2) i값이 1,32 또는 33의 경우, 임의의 S-box와 j에 대하여 $NS_a(i, j)=32$이다.

3.2.2 F함수의 선형근사

S-box의 선형근사를 F함수의 선형근사로 확장한다. 예를 들면, 식(3)은 키 K를 임의로 고정하고 F함수에 랜덤한 입력 X가 주어졌을 때 식(3)이 확률 0.19로 성립하는 것을 의미한다. 따라서 식(4)는 F함수의 최량 선형 근사이다.

$$X[15]\oplus F(X, K)[7, 18, 24, 29]=K[22] \tag{4}$$

달리 말하면 다음식은 확률 1−0.19=0.81로 성립한다.

$$X[15]\oplus F(X, K)[7, 18, 24, 29]=K[22]\oplus1 \tag{5}$$

3.2.3 DES의 선형근사

F함수의 선형근사를 1단씩 중첩시키므로 암호문 및 키간의 관계를 (중간값을 포함하지 않음)나타내는 선형근사식을 구할 수 있다. 물론 중첩 방법은 선형경로가 중단되거나 모순이 되지 않아야 한다. 달리 말하면, 제i단의 F함수에 입력 비트 마스크 값과 출력 비트 마스크값을 각각 ΓY_i, ΓY_i로 하면 다음식이 성립하여야 한다.

$$\Gamma Y_{i+2}=\Gamma Y_i\oplus\Gamma Y_{i+1}(1\leq i\leq n-2) \tag{6}$$

또한 이때의 제i단 F함수의 근사 확률 p_i는 위의 기호를 이용하면 다음과 같다.

$$p_i=Prob\{Parity(X_i\cdot\Gamma X_i) =Parity(F_i(X_i, K_i)\cdot\Gamma Y_i)\} \tag{7}$$

여기서 X_i와 K_i는 랜덤하게 선택한 것이다. 이와같이 F함수의 선형 근사를 n회 중첩하여 얻은 암호 알고리즘 전체의 선형 근사식의 성립 확률은 식(8)과 같으며 증명은 n에 관한 귀납법으로 쉽게 가능하다.

▨ 보 조 정 의 2 Piling-up Lemma

독립적인 확률변수 $X_i(1\leq i\leq n)$가 확률 p_i로 0, 확률 $1-p_i$로 1을 취할때, $X_1\oplus X_2\oplus\ldots\oplus X_n=0$이 되는 확률은 다음식과 같다.

$$2^{n-1}\prod_{i=1}^{n}(p_i-1/2)+1/2 \tag{8}$$

따라서, n단 암호의 최량 확률은 식(6)이 성립하는 조건하에 식(8)의 최대치이다. 이것을 구하기 위하여 컴퓨터에 의해 효율적인 검색법이 필요하며 검색 알고리즘에 대하여는 참고 문

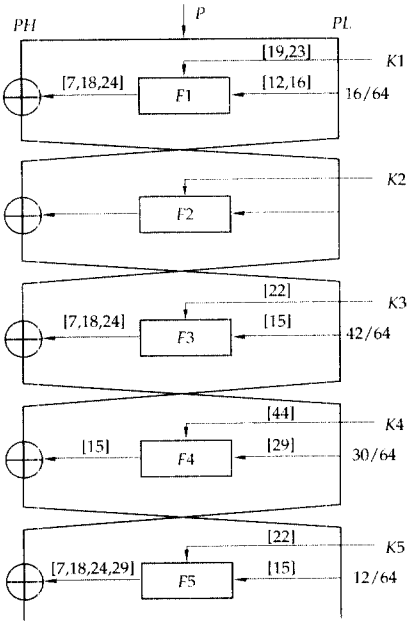
헌[13]에 그 개요를 서술하였다.

(그림3)은 16단 DES의 최량표현을 나타낸 것으로 성립화를 즉, 최량 확률은 Piling-up Lemma에 의해 $1/2+2^{11}(-20/64)^4(-10/64)(-2/64)^3(10/64)^3(-16/64)=1/2-1.49\times2^{-24}$이다.

◐ 예 2 16단 DES의 최량 표현

$$P_H[7,18,24]\oplus P_L[12,16]\oplus C_H[15]\oplus C_L[7,18,24,27,28,29,30,31]=K_1[19,23]\oplus K_3[22]\oplus K_4[44]\oplus K_5[22]\oplus K_7[22]\oplus K_8[44]\oplus K_9[22]\oplus K_{11}[22]\oplus K_{12}[44]\oplus K_{13}[22]\oplus K_{15}[22]\oplus K_{16}[42,43,45,46] \quad (9)$$

식(9)는 P와 C 및 K_i와 K_{17-i}를 서로 교환하여 또 하나의 최량 표현을 구할 수 있다는 점을 주의하자. 표(1)은 16단까지 DES의 최량 확률을 나타내며 각 단에 있어서 F함수의 근사식을 포함한 표는 참고문헌[15]에 기재되어 있다.



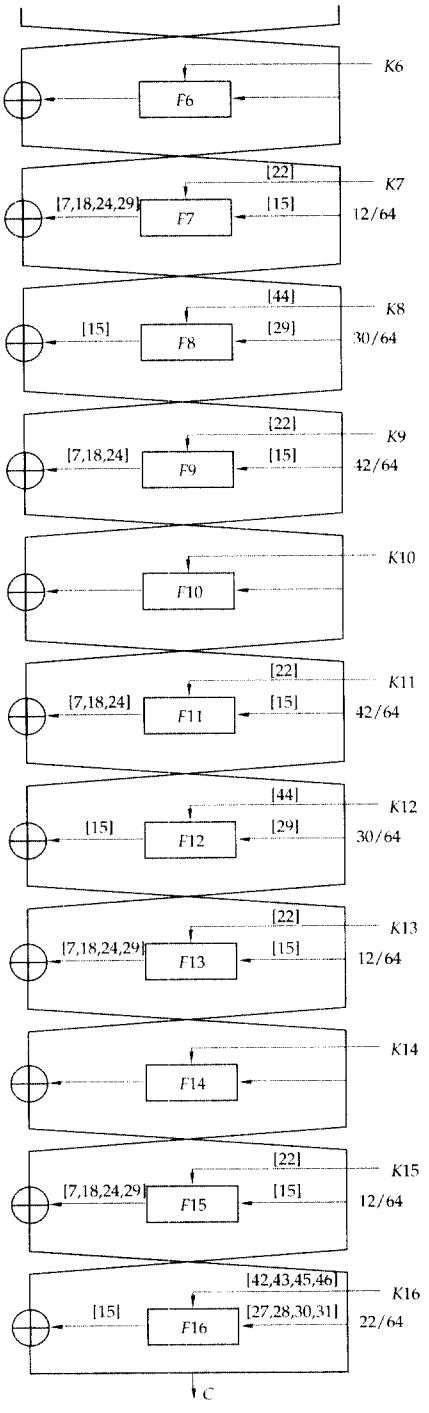


그림3 : 16단 DES의 최량 표현

표 1 : DES의 최량 확률

단 수	최 량 확 률
4	$1.2-1.95\times2^{-5}$
6	$1.2-1.95\times2^{-9}$
8	$1.2-1.22\times2^{-11}$
10	$1.2-1.53\times2^{-15}$
12	$1.2-1.19\times2^{-17}$
14	$1.2-1.19\times2^{-21}$
16	$1.2-1.49\times2^{-24}$

3.3 선형 해독법의 대역 이론

3.3.1 알고리즘1

여기서는 알고리즘1을 이용하여 식(1)을 푸는 경우, 올바른 우변을 추정할 확률을 계산한다. 지금 N개의 기지 평문을 이용할 때, 알고리즘 1에 있어서 카운터 T의 값이 $N/2$보다 커질 확률은

$$\sum_{t=N/2+1}^{N} {}_{N}C_t p^t (1-p)^{N-t} \qquad (10)$$

이고 이항 분포를 정규분포로 근사하면 다음의 보조정리를 구할 수 있다.

▨ 보 조 정 의 3

N개의 랜덤 기지 평문을 이용하여 알고리즘 1을 실행할 때, 그 성공확률은 다음과 같다.

$$\int_{2\sqrt{N}|p-1/2|}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx \qquad (11)$$

▨ 따 름 정 의 1

N개의 랜덤 기지 평문을 이용하여 알고리즘 1을 실행할 때, 그 해독 성공확률은 $\sqrt{N}|p-1/2|$에만 의존한다.

표 2 : 알고리즘1의 성공 확률(%)

기지 평문수	추정 성공 확률
$1/4|p-1/2|^{-2}$	84.1%
$1/2|p-1/2|^{-2}$	92.1%
$|p-1/2|^{-2}$	97.7%

식(11)의 계산값은 표(2)와 같다. 따라서 식(9)를 알고리즘1로 풀 경우 $1/2|1.49\times2^{-24}|^{-2}=1.80\times2^{45}$개의 기지 평문이 주어지면 92.1%의 확률로 우변의 1비트로 구할 수 있다.

3.3.2 알고리즘2

다음은 알고리즘2를 이용하여 식(2)를 푸는 경우 올바른 우변 및 K_1, K_n이 추정되는 확률을 계산한다. 이 경우 다항 분포를 고려해야 하므로 보조정리3에 상응하는 식의 형태는 복잡하고 수치 계산이 곤란하다. 그러나 따름정리1을 확장하면 다음 사실이 증명 가능하다(증명은 생략).

▨ 보 조 정 의 4

N개의 랜덤한 깊 평문을 이용하여 알고리즘2를 실행할 때, 해독 성공 확률은 $N|p-1/2|$ 및 $u_1, u_2, \ldots, u_a$와 $v_1, v_2, \ldots, v_b$에만 의존한다.

다음 예를 들어 위의 보조정리가 무엇을 의미하는지를 살펴본다. 아래의 식(12)와 식(13)은 각각 6단, 14단의 최량 표현에서 얻은 8단 DES 및 16단 DES의 근사식이다. 이것과 함께 제1단과 최종단의 F함수를 식에 삽입할 것으로 근사식의 구성은 (그림 3)에 제1단과 제8단(또는 제16단)의 근사를 생략한 것과 동일하다. 성립확률은 각각 Piling-up Lemma에 의해 $1/2-1.95\times2^{-11}$ 및 $1/2-1.9\times2^{-21}$임을 알 수 있다.

◐ 예 3

$$P_H[7,18,24]\oplus C_H[15]\oplus C_L[7,18,24,29]\oplus F_1(P_L,K_1)[7,18,24]\oplus F_8(C_L,K_8)[15]=K_3[22]\oplus K_4[44]\oplus K_5[22]\oplus K_7[22] \qquad (12)$$

◐ 예 4

$$P_H[7,18,24]\oplus C_H[15]\oplus C_L[7,18,24,29]\oplus F_1(P_L,K_1)[7,18,24]\oplus F_{16}(C_L,K_{16})[15]=K_3[22]\oplus K_4[44]\oplus K_5[22]\oplus K_7[22]\oplus K_8[44]\oplus K_9[22]\oplus K_{11}[22]$$

$$\oplus K_{12}[44]\oplus K_{13}[22]\oplus K_{15}[22] \quad (13)$$

위의 2개 예에서 $u_1, u_2, \ldots, u_d$ 및 $v_1, v_2, \ldots v_e$는 항상 일치한다. 따라서 N_8개의 기지 평문을 이용하여 식(12)를 알고리즘2를 이용하여 푼 때의 해독 성공 확률과 N_{16}개의 기지평문을 이용하여 식(13)을 알고리즘2를 이용하여 푼 때의 해독 성공 확률이 동일하므로 다음 관계가 성립한다.

$$\sqrt{N_8}|1.95\times 2^{-n}| = \sqrt{N_{16}}|1.19\times 2^{-21}| \quad (14)$$

식(14)를 정리하면 다음 관계가 구해진다.

$$1.49\times 2^{-26}\times N_{16}=N_8 \quad (15)$$

이 사실은 8단 DES의 해독 실험 결과로부터 16단 DES의 해독 성공확률을 예측 가능함을 의미하고 다음 장에는 이것을 이용하여 16단 DES의 해독 실험을 한다.

4. 16단 DES의 해독 방침

기본적으로는 식(13)을 알고리즘2를 이용하여 푸는 것이 목표이다.

$$P_H[7,18,24]\oplus C_H[15]\oplus C_L[7,18,24,29]\oplus F_1(P_L,K_1)[7,18,24]\oplus F_{16}(C_L,K_{16})[15]=K_3[22]\oplus K_4[44]\oplus K_5[22]\oplus K_7[22]\oplus K_8[44]\oplus K_{11}[22]\oplus K_{11}[22]\oplus K_{12}[44]\oplus K_{13}[22]\oplus K_{15}[22] \quad (16)$$

여기서 식(16)의 좌변에 영향을 주는 평문과 암호문 및 확장키의 비트들을 구체적으로 조사한다. 이것은 각각 다음과 같이 되는 것을 쉽게 알 수 있다.

- 평문과 암호문 비트(13비트)
  $P_L[11]\sim P_L[16]$, $C_L[0]$, $C_L[27]\sim C_L[31]$,
  $P_H[7,18,24]\oplus C_H[15]\oplus C_L[7,18,24,29]$
- 확장키(12비트)
  $K_1[18]\sim K_1[23]$, $K_{16}[42]\sim K_{16}[47]$

$P_H[7,18,24]\oplus C_H[15]\oplus C_L[7,18,24,29]$는 1비트 정보를 나타냄을 주의한다. 따라서, 기지평문과 암호문으로부터 얻은 13비트의 정보로부터 확장키 12비트와 식(16)의 우변 1비트를 더하여 모두 13비트를 알고리즘2로 구해질 수 있다.

이 식에 P와 C를 교환하고 K_i와 K_{17-i}를 교환하여 얻어진 식도 동일한 확률 $1/2-1.19\times 2^{-21}$로 성립한다.

$$C_H[7,18,24]\oplus P_H[15]\oplus P_L[7,18,24,29]\oplus F_{16}(C_L,K_{16})[7,18,24]\oplus F_1(P_L,K_1)[15]=K_{14}[22]\oplus K_{13}[44]\oplus K_{12}[22]\oplus K_{10}[22]\oplus K_9[44]\oplus K_8[22]\oplus K_6[22]\oplus K_5[44]\oplus K_4[44]\oplus K_2[22] \quad (17)$$

식(17)에 영향을 주는 평문과 암호문 및 확장키의 비트들은

- 평문과 암호문 비트(13비트)
  $C_L[11]\sim C_L[16]$, $P_L[0]$, $P_L[27]\sim P_L[31]$,
  $C_H[7,18,24]\oplus P_H[15]\oplus P_L[7,18,24,29]$
- 확장키(12비트)
  $K_{16}[18]\sim K_{16}[23]$, $K_1[42]\sim K_1[47]$

이 식으로부터 마찬가지로 확장키 12비트와 우변의 1비트를 더해서 모두 13비트를 구할 수 있다. 이 2개식으로부터 확장키 26비트가 알고리즘2에 의해 구해지는 것이다. 26비트의 확장키는 암호화 키의 다음 정보에 상당한다.

0,1,3,4,8,9,14,15,18,19,24,25,31,32,38,39,41,42,44,45,50,51,54,55,
5⊕13⊕17⊕20⊕46,2⊕7⊕11⊕22⊕26⊕37⊕52

구체적인 해독 방법은 다음과 같다.

16단 DES의 기본 해독 알고리즘

Data Counting Phase

Step 1 2^{13}개 배열을 갖는 2개의 카운터 UA_{i_A}, UB_{i_B}($0\le i_A,\ i_B\le 2^{13}$)를 초기화한다. 여기서, i_A, i_B는 각각 식(16)과 식(17)의 좌

변에 영향을 주는 평문과 암호문의 13비트에 1대 1 대응하여 생각한다.

Step 2 기지 평문 P와 대응하는 암호문 C의 쌍에 대하여 i_A, i_B를 계산하고 대응하는 UA_{i_A} 및 UB_{i_B}의 값을 1씩 증가시킨다.

Key Counting Phase

Step 3 2^{12}개 배열을 갖는 2개의 카운터 TA_{j_A}, $TB_{j_B}(0 \le j_A, j_B < 2^{12})$를 초기화한다. 여기서, j_A, j_B는 각각 식(16)과 식(17)의 좌변에 영향을 주는 확장키의 12비트에 1대 1 대응하여 생각한다.

Step 4 i_A, i_B, j_A, j_B가 결정되면 식(16)과 식(17)의 우변이 결정된다. 각 j_A, j_B에 대하여 식(16) 또는 식(17)을 0으로 하는 i_A 및 i_B에 대응하는 가운데 UA_{i_A} 및 UB_{i_B}의 합을 TA_{j_A} 및 TB_{j_B}에 저장한다.

Step 5 모든 TA_{j_A}에 대한 최대치와 최소치를 TA_{max}와 TA_{min}로, 모든 TB_{j_B}에 대한 최대치와 최소치를 TB_{max}와 TB_{min}로 한다.

- $|TA_{max}-N/2| > |TA_{min}-N/2|$이면 TA_{max}에 대응하는 확장키 12비트를 선택하고 식(16)의 우변은 1로 추정한다.
- $|TA_{max}-N/2| > |TA_{min}-N/2|$이면 TA_{min}에 대응하는 확장키 12비트를 선택하고 식(16)의 우변은 0로 추정한다.
- $|TB_{max}-N/2| > |TB_{min}-N/2|$이면 TB_{max}에 대응하는 확장키 12비트를 선택하고 식(17)의 우변은 1로 추정한다.
- $|TB_{max}-N/2| > |TB_{min}-N/2|$이면 TB_{min}에 대응하는 확장키 12비트를 선택하고 식(17)의 우변은 0로 추정한다.

Exchustive Search Phase

Step 6 나머지의 암호화 키 비트 30비트는 전수검색에 의해 결정한다.

본 해독법에 필요한 총카운터의 수는 $2^{13}\times 2+2^{12}\times 2$정보이고 계산량은 Step 2에만 의존함을 주의한다.

5. 해독 성공 확률

5.1 기본 알고리즘의 해독 성공 확률

4장에서 제시한 기본 해독 알고리즘의 성공 확률을 평가한다. 식(15)에 의하면 16단 DES를 예를들어 2^{45}개의 기지 평문으로 해독할 때, 확장키 26비트를 올바르게 구할 확률은 동일한 알고리즘으로 독립적인 확장키를 갖는 8단 DES에서 구한 식(18)과 식(19)(이것은 각각 확률 $1/2-1.95\times 2^{-9}$로 성립한다. [예3]참조)에 적용하면 $1.49\times 2^{45-26}=1.49\times 2^{19}$개의 기지 평문을 이용하여 확장키 26비트를 올바르게 구할 확률과 동일하게 된다.

$$P_H[7,18,24]\oplus C_H[15]\oplus C_L[7,18,24,29]\oplus F_1(P_L,K_1)[7,18,24]\oplus F_8(C_L,K_8)[15]=K_3[22]\oplus K_4[44]\oplus K_5[22]\oplus K_7[22] \quad (18)$$

$$C_H[7,18,24]\oplus P_H[15]\oplus P_L[7,18,24,29]\oplus F_8(C_L,K_8)[7,18,24]\oplus F_1(P_L,K_1)[15]=K_6[22]\oplus K_7[44]\oplus K_1[22]\oplus K_2[22] \quad (19)$$

여기서 기본 알고리즘을 이용하여 8단 DES를 각각 1.49×2^{17}, 1.49×2^{18}, 1.49×2^{19}개의 기지 평문(이것은 16단 DES에 2^{43}, 2^{44}, 2^{45}개의 기지 평문으로 해독하는 경우에 상응함)으로 실험을 1,000회 실시하여 성공확률은 다음과 같다. 이 결과로 2^{45}개의 기지 평문과 암호문쌍이 주어진다면 16단 DES의 암호화 키는 거의 틀림없이 결정된다.

표 3 : 기본 알고리즘의 해독성공 확률

8단 DES	16단 DES	해독 성공 확률
1.49×2^{17}	2^{43}	10.6%
1.49×2^{18}	2^{44}	60.4%
1.49×2^{19}	2^{45}	98.8%

5.2 개선 알고리즘

다음은 이 해독 성공 확률을 높이는 방법을 생각하자. 기본 알고리즘은 식(16)과 식(17)로부터 얻은 총 26비트의 후부를 1개만 선택하고 나머지 30비트는 전수 검사에 의해 결정하는 방법이다. 여기서 26비트의 후보 중 신뢰성을 높은 것을 복수 선택하여 각각에 대하여 30비트의 검사를 반복하는 방법을 도입한다. 즉, 기본 알고리즘의 Step 5 및 Step 6를 다음과 같이 변경한다.

16단 DES의 해독 개선 알고리즘

Step 5 Step 4에서 구한 카운터 TA_{j_A} 와 TA_{j_B}($0 \le j_A, j_B < 2^{13}$)의 값을 기준으로 26비트의 확장키의 후보를 각 신뢰도의 순서를 $W_1, W_2, W_3, \ldots$로 놓는다.

Step 6 각 W_i에 대하여 나머지 30비트의 암호화 키를 검색하여 올바른 값을 발견하면 종료한다.

그러면 $W_1, W_2, W_3, \ldots$을 구체적으로 결정하는 방법을 고찰한다. 이를 위해 우선 TA_{j_A}를 $|TA_{j_A} - N/2|$의 크기순으로 나열한 것은 $\overline{TA_{j_A}}$로 하고 또한 TB_{j_B}를 $|TB_{j_B} - N/2|$의 크기순으로 나열한 것을 $\overline{TB_{j_B}}$로 정의한다. 이 때 가장 신뢰도가 높은 26비트의 확장키 후보는 $(\overline{TA_1}, \overline{TB_1})$에 대응하는 것으로 기본 알고리즘에서는 이것을 하나만으로 선택하였다. 다음으로 신뢰성이 높은 26비트의 확장키 후보는 $(\overline{TA_1}, \overline{TB_2})$또는 $(\overline{TA_2}, \overline{TB_1})$일 것이다. 그 다음의 신뢰도가 높은 것은 $(\overline{TA_1}, \overline{TB_3})$ 또는 $(\overline{TA_3}, \overline{TB_1})$일까? 또는 $(\overline{TA_2}, \overline{TB_2})$를 검사해야 되는 것은 아닐까? 26비트의 후보를 몇 개를 골라 놓아야 좋은 것일까? 이것을 결정하기 위하여 다음과 같이 8단 DES에 대하여 실험을 시행하였다.

5.3 개선 알고리즘의 해독성공확률

다음의 표는 기본 알고리즘을 이용하여 8단 DES를 식(18)을 이용하여 1.49×2^{17}개의 기지 평문으로 해독하였을 때(16단 DES의 경우, 식(16)을 2^{43}개의 기지평문으로 해독하는 경우에 상당)올바른 키가 $\overline{TA_{j_A}}$의 몇 번째의 있는가를 실험적으로 구한 것이다. 이 결과 상위 40번째 이내에 올바른

표 4 : 개선 알고리즘의 해독성공 확률(Ⅰ)

순 위	j_A	확 률	총 확 률
1	1-8	60.7%	60.7%
2	9-16	8.6%	69.2%
3	17-24	4.8%	74.0%
4	25-32	3.6%	77.6%
5	33-40	2.1%	79.6%
6	41-48	1.8%	81.4%
7	49-56	1.4%	82.8%
8	57-64	1.4%	84.2%
9	65-72	0.9%	85.0%
10	73-80	0.9%	86.0%
11	81-88	0.7%	86.6%
12	89-96	0.5%	87.2%

13비트의 키가 있을 확률은 거의 80%임을 알 수 있다. 여기서 표(4)를 근거로 $(\overline{TA_i}, \overline{TB_j})$의 신뢰성의 순위를 계산하여 표(5)를 구하였다. 이 표에서 나타난 순서로 26비트의 후보를 선택하고 나머지 30비트의 검색을 하면 가장 효율적이라고 생각된다.

이 표에 의하면 순위 9까지 60%의 해가 있으며 순위 24까지 70%의 해가 존재한다. 계산을 수행함에 따라 순위 70까지 해가 존재하는 확률은 80%가 됨을 알 수 있다. 표의 한 entry에는 64개의 (j_A, j_B)의 후보를 갖고 있으므로, 달리말하면 $64 \times 9 = 576$ 종류의 26비트 후보를 조사하면 60%, $64 \times 24 = 1,536$ 종류의 26비트 후보를 조사하면 70%, $64 \times 70 = 4,480$ 종류의 26비트 후보를 조사하면 80%의 확률로 올바른 키를 구하는 것이 가능하다. 이 결과를 16단 DES에 적용하면, 1개의 26비트 후보에 대하여 나머지 30비트의 검색을 하지 않으므로 결국 Step 6에 있어서 총 검색 횟수는 각각 1.13×2^{39}, 1.5×2^{40}, 4.09×2^{42}에 해당한다.

표 5 : 개선 알고리즘의 해독성공확률(Ⅱ)

순 위	j_A, j_B	확 률	총 확 률
1	1-8, 1-8	36.8%	36.8%
2	1-8, 9-16	5.2%	42.0%
3	9-16, 1-8	5.2%	47.2%
4	1-8, 17-24	2.9%	50.1%
5	17-24, 1-8	2.9%	53.1%
6	1-8, 25-32	2.2%	53.1%
7	25-32, 1-8	2.2%	57.3%
8	1-8, 33-40	1.2%	58.5%
9	33-40, 1-8	1.2%	60.0%
10	1-8, 41-48	1.1%	60.9%
11	41-48, 1-8	1.1%	62.0%
12	1-8, 49-56	0.8%	62.8%
13	49-56, 1-8	0.8%	63.7%
14	1-8, 57-61	0.8%	64.5%
15	57-64, 1-8	0.8%	65.3%
16	9-16, 9-16	0.7%	66.0%
17	1-8, 65-72	0.6%	66.6%
18	65-72, 1-8	0.6%	67.2%
19	1-8, 73-80	0.5%	67.7%
20	73-80, 1-8	0.5%	68.2%
21	1-8, 81-88	0.5%	68.7%
22	81-88, 1-8	0.5%	69.2%
23	1-8, 89-96	0.4%	69.7%
24	89-96, 1-8	0.4%	70.1%

6. 소프트웨어에 의한 실현

이상의 고찰에서 16단 DES는 2^{43}개의 기지 평문과 대응하는 암호문의 쌍이 주어진다면 2^{30}에서 2^{42}개 정도의 암호화 키 검색 시간에 모든 암호화 키를 높은 확률로 찾을 수 있다. 본 해독 알고리즘은 평문이나 암호문을 축적할 메모리가 필요없으므로 위의 평문수나 키 검색 시간은 소프트웨어에 의해 시뮬레이션이 가능한 범위이다.

실제 해독시험을 수행하는 데는 프로그램의 효율화가 상당히 중요한 역활을 한다. 본장에는 마쯔이가 해독 실험에 이용한 스프트웨어의 실현 방법을 기술한다.

6.1 난수 생성 프로그램

가정이 기지평문공격이므로 평문과 암호문의 생성 과정은 해독 과정과는 독립적으로 볼 수 있다. 실험을 위하여는 평문을 자체적으로 생성하고 암호화하여야 한다. 해독프로그램에는 다음과 같이 난수 생성 프로그램을 이용하여 2^{43}개의 평문을 준비하였다.

이 난수는 64비트을 한 단위로 보았을 때, 적어도 2^{43}이라 하는 긴 주기를 가질 필요가 있어 병렬처리를 고려하면 각 컴퓨터상에 발생하는 난수가 중복되지 않아야 한다. 예를들면, UNIX 등에 제공하는 라이브러리 함수인 mrand48() 또는 random()은 비교적 양질의 난수를 발생해 주지만 지금의 목적은 난수로서 좋은 특성이 반드시 필요하지는 않고 가능하면 고속 발생이 요구되므로 본 실험에는 유한체의 2진 표현을 이용하였다.

구체적으로는 유한체 $GF(2^{64})$상의 원시근 g를 하나로 고정하고 수열 $1, g, g^2, g^3, \ldots$를 $GF(2)$의 적당한 기저에 관하여 2진 표현을 한 난수 발생 프로그램을 대응하였다. 이것은 난수로서는 난수성이 약하지만 본 목적에는 충분하고 몇 번째의 값을 즉시 계산 가능하여 병렬처리에 효과적이다.

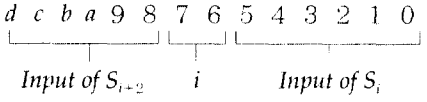
6.2 DES의 암호화 프로그램

해독 실험에 가장 많은 연산 시간이 필요하는 것이 기지 평문에서 암호문을 생성하는 프로그램이다. 이 과정도 기지 평문 공격과는 독립적이나 반드시 필요하다. 이 프로그램의 고속화를 위해 실험에서는 W/S의 CPU인 PA-RISC의 어셈블리 언어로 구현하였다.

PA-RISC는 32개의 32비트 레지스터를 가지고 있고 기능도 거의 유사하다. 스택 포인터 등의 변경이 안되는 소수의 레지스터를 제외하고 거의 모든 레지스터를 (필요에 따라 레지스터의 내용을 스

택 포인터에 대피시키는 것) 자유로 사용하는 것이 가능하다[16,17]. 본 프로그램에는 CPU의 유효한 파이프라인 처리를 위해 8블럭의 평문을 최소 단위로 암호화하고 16개의 레지스터를 8블럭 데이터의 보존 전용으로 할당하였다.

S-box의 계산은 S_i와 S_{i+2}를 하나로 하여 F 함수 내부의 메모리 참조는 4회로 하였다. 인접 S-box를 페어로 하지 않은 이유는 확대 전치 E에 의한 비트의 중복을 피하기 위함이다. 4회의 메모리 참조는 다음과 같은 형식의 14비트 입력 어드레스를 갖는 1개의 테이블로 구현하였다.



여기서 0번째에서 5번째 어드레스는 S_i-box에는 입력치를 나타내고, 8번째에서 13번째 어드레스는 S_{i+2}-box에의 입력치를 나타내고 6번째에서 7번째의 값은 0, 1, 2, 3에 따라 i는 0, 1, 4, 5를 나타내는 것이다. S-box의 출력은 전치 P를 포함하는 것이 가능하므로 테이블의 크기는 $2^{14}\times$ 4byte=66Kbyte이다.

암호화키에 대하여는 고정되었다고 생각해서 미리 확장키를 위해 형식에 맞도록 최초에 생성하여 둔다. 1단의 확장키를 저장하는 데는 4개의 레지스터를 할당하고 파이프라인 처리로 최대한의 효율화를 꾀하였다.

이 결과 암호화키를 고정한 경우, 암호화 속도는 약 19Mbit/s이고 필요한 메모리 크기는 코드 및 데이터 영역을 합쳐서 약 80Kbyte이다.

6.3 DES의 암호키 탐색 프로그램

기본적으로는 DES의 암호화 과정과 동일하나 본 실험에는 해독에 필요한 30비트의 키 검색을 위한 프로그램이다. 이것은 평문을 하나로 고정하고 암호화키를 매번 바꾸는 조건으로 작성하였다. 따라서 Key Scheduling부분을 어떻게 실현하는가에 따라 암호화 속도에 주요한 영향을 준다. 이 부분도 고속을 위하여 어셈블리 언어로 실현하였다.

S-box 참조 방법도 DES의 암호화 프로그램과 동일하게 하였고 암호화키 값과 각 S-box쌍에 공급되는 확대키의 값 간에 대응 테이블을 미리 만들어 놓아 56비트의 암호화키의 참조 2회로 1쌍의 S-box를 계산하는데 필요한 확장키를 구하도록 하였다. 이 대응 테이블은 약 51Kbyte이다.

1회의 F 함수 연산에 8회의 암호화키 메모리를 액세스하여야 하므로 8개의 레지스터를 전용으로 할당하였다. 이 결과 평문을 고정한 경우, 암호화키의 검색 속도는 약 9Mbit/s이고 필요한 메모리는 코드와 데이터영역을 포함하여 약 130Kbyte이였다.

6.4 해독 실험의 개요

이상의 프로그램은 C언어와 PA-RISC의 어셈블리 언어로 기술하였다. 각각 1,000 line 정도로 실현되고 실행시의 사용 메모리는 1MB였다. 해독 실험의 기간은 1993년 8월부터 10월까지로 마쓰비시 전기(주)제품의 W/S인 ME/R(PA-RISC, 99MHz, 125MIPS)12대를 이용하였다.

각각의 W/S는 중복이 되지 않고 개별적으로 난수를 생성하고 암호화하면서 카운터 TA_{iA} 및 TB_{iB}를 생성하고 파일에 출력한다(기본 알고리즘의 Step 1-Step 4에 상당). 이것을 종료하는데 약 40일이 소요되었다.

그 후, 모든 W/S에서 생성한 카운터 파일을 1개의 W/S으로 모아 합산하여 최종적인 TA_{iA} 및 TA_{iB}를 구하고 이것을 $|TA_{iA}-N/2|$ 및 $|TB_{iB}-N/2|$의 크기 순으로 나열하여 TA_{iA} 및 TB_{iB}를 구하였다. 이 과정은 2종류의 2^{30}개의 카운터 조작이므로 간단히 계산된다(개선 알고리즘의 Step 5에 상당).

계속하여 26비트의 후보키를 표(5)를 이용하여 신뢰성이 높은 순서로 뽑아서 나머지 30비트를 검색하였다. 이 과정도 12대의 W/S을 이용하여 약 10일이 소요된 후 올바른 암호화 키 56비트를 찾

아내었다(개선 알고리즘의 Step 6에 상당).

올바른 키는 표(5)의 22번째 값이었고 해독에 이용한 식(16)의 해는 84번째, 식(17)의 해는 1번째에 각각 올바른 키가 존재하였다.

7. 결 론

본 해설은 기존의 선형 해독법을 개선하는 새로운 이론을 기술하고 이것을 실증하기 위하여 16단 DES의 해독 실험 결과도 소개하였다. 본 실험에 필요한 2^{33}개라는 기지 평문수는 1 Gbit/s의 통신로에 약 1주일간의 정보량에 상당한다. 이 방법은 암호화키의 검색 횟수를 증가하는 대신에 소요되는 기지 평문의 수를 감소시키는 방법도 가능하다. 달리말하면, 본회의 실험에 사용한 것보다 적은 수의 기지 평문으로 Step 6에서 암호화키의 검색 횟수를 증가시키면 높은 확률로 해독이 성공할 것이다.

해독 실험에 이용한 확장키의 신뢰도 정보는 완전히 실험적으로 구한 것이므로 샘플 수의 한계로 유의한 결과를 얻기에는 표(4)와 표(5)는 8순위까지만 정리하였다. 그러나 표(4)는 이론적인 곡선을 구해야 할 것이다. 이를 이론적으로 정립하면 곡선의 방정식에 기지 평문 수를 변수로 추가하면 표(5)는 엄밀히 정리되어 기지 평문의 수, Step 6의 키 검색 횟수, 그리고 해독 성공 확률간의 관계가 구해진다. 새로운 이런 관계를 이용하면 16단 DES의 해독 성공 확률은 더욱 높아질 것이 예상된다.

참 고 문 헌

[1] M.Matsui, "Linear Cryptanalysis of DES Cipher (Ⅰ)", SCIS93-3C, Jan., 1993.

[2] M.Matsui, "Linear Cryptanalysis of DES Cipher (Ⅱ)", ISEC92-64, Mar., 1993.

[3] D.W. Davies and W.L. Price, "Security for Computer Networks", John Wiley & Sons Ltd, 1984.

[4] National Bureau of Standards, "Data Encryption Standard", FIPS.NO.16, U.S. Dep't of Commerce, 1977.

[5] W. Diffie and M.E. Hellman, "Exhaustive cryptanalysis of the NBS Data Encryption Standard", Computer, 10, 6,74, 1977.

[6] M. Hellman, R. Merkle, R. Schroeppel, L. Washington, W. Diffie, S. Pohlig, and P. Schweizer, "Results an initial attempt to cryptanalyze the NBS Data Encryption Standard", Information Systems Laboratory, SEL 76-041, Standford Univ., 1976.

[7] Y. Desmedt, J.J. Quisquater and M. Davio, "Dependence of output on input in DES : Small avalanche characteristics", Proc. of Crypto'84-Advances in Cryptology, Lecture notes in Computer Science, Vol. 196, Springer-Verlag, pp.356~376, 1984.

[8] E.Biham and A.Shamir, "Differential Cryptanalysis of DES-like Cryptosystems", J. of Cryptology, Vol.4,pp.3~72, 1991.

[9] E. Biham and A. Shamir, "Differential Cryptanaylsis of the full 16-round DES", Crypto'92 Extended Abstracts, pp.12-1~12-5, 1992.

[10] A. Shamir, "On the security of DES", Proc. of Crypto'85-Advances in Cryptology, Lecture notes in Computer Science, Vol.218, Springer-Verlag, pp. 280~281, 1985.

[11] R.A. Rueppel, Analysis and design of stream ciphers, Springer-Verlag, 1986.

[12] E.Biham, "On Matsui's Linear Crytanalysis", presented at Dagstuhl Seminar, Sep., 1993.

[13] M.Matsui, "On correlation between the order of S-boxes and the strength of DES", preprint

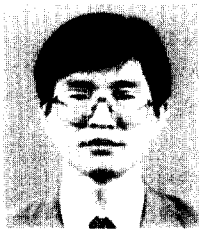
[14] S.Langford and M.E. Hellman, "New, improved 8-round DES Attack", presented at RSA conference, Jan., 1994.

[15] M.Matsui, "Linear Cryptanalysis Method of DES Cipher", Eurocrypt'93 Extended Abstracts, pp.W-112~W-123, 1993.

[16] "Precision Architecture and Instruction Set Reference Manual", HP Part No. 09740-90014, Hewlett-Packard Company.

[17] "Procedure Calling Conventions Reference Manual", HP Part No. 09740-90015, Hewlett-Packard Company.

□ 著者紹介



김 광 조(정회원)

1980년 연세대학교 전자공학과(학사)
1983년 연세대학교 대학원 전자공학과(석사)
1990년 요꼬하마 국립대학 대학원 전자 정보공학과(박사)
현 한국전자통신연구소 실장
※ 관심 분야 : 암호학 및 응용 분야, M/W 통신