http://dx.doi.org/10.3745/KIPSTC.2011.18C.5.287

DES 키 확장을 이용한 가변 P box 생성에 관한 연구

이 준[+]

요 약

DES 확장키를 이용하여 가변 P box를 생성하고 이용하는 블록암호를 제안하였다. 매 라운드마다 가변 P box를 구현할 때 효율적으로 실행할 수 있는 방안도 제시하였다. 차분 공격과 선형 공격에 관한 반례를 제시함으로 제안된 알고리즘이 기존의 두 공격으로부터 안전함을 보였다. 제안된 알고리즘과 3중 DES(3DES)에서 사용되는 실 키 길이 비교는 전수공격 측면에서 제안된 알고리즘이 3DES보다 더 안전함을 보인다. 3DES와 제안된 알고리즘의 컴퓨터 실험 결과 암호처리 속도측면에서 제안된 알고리즘이 3DES보다 3배정도 빠름을 보였다.

키워드 : DES, 키 확장, 가변 P box

A Study on a Variable P box Generation Using a DES Key Expansion

Jun Lee[+]

ABSTRACT

Using an expanded DES key, we suggest a block cipher algorithm to generate and to use a variable P box. We also present an efficient way for the implementation of variable P box at each round. Using counter examples on Differential Cryptanalysis(DC) and Linear Cryptanalysis(LC), we show that the suggested algorithm is strong enough to overcome those attacks. Compared with the real key bits of triple DES(3DES), the new algorithm is much safer in the points of the exhaustive attack. The results of computer simulations show that the new algorithm is almost 3 times faster than 3DES regarding the cipher process time.

Keywords : DES, Key Expansion, Variable P box

1. 서 론

현대 블록 암호는 112비트 이상의 키 길이에 대해 안전함을 주장한다[1]. DES는 가장 많이 사용되는 블록암호이나, 키 길이가 64비트로 현대 블록 암호가 요구하는 키 길이에 비해 짧은 단점을 갖고 있다. 또한 그 중 8비트를 사용하지 않아 스스로 키 공간을 축소시킴으로 다양한 공격의 가능성을 제공하였다. 환경적으로 계산 능력 향상, 공격 방법의 개발 등으로 전수 공격보다 적은 노력으로 DES 암호문 해독이 가능함을 보였다[2][3].

일반적으로 잘 알려진 DES 공격은 차분 공격 (Differential Cryptanalysis)과 선형 공격(Linear Cryptanalysis)으로 전수공격보다 적은 경우의 수로 키를 추측할 수 있다. 공격 방법은 주로 DES의 비선형 부분에 대한 연구이며, 키 추측 가능성을 합리적으로 제시한다. 또한 이들의 연구에서 선형 부분은 수학적으로 쉬운 부분으로 간주하여 공격범위에 포함하지 않았다. 하지만 선형 부분 P box도 라운드 별로 비공개되는 가변 상황을 DES에 적용한다면 공격이 쉽지 않음을 보일 수 있다.

본문 구성은 2절에서 DES 공격 관련 연구에서 공격이 검토되지 않은 새로운 암호화 가능성을 언급하고, 가변 P box 알고리즘을 3절에서 제안하였다. 4절은 제안된 알고리즘의 안전성을 차분 공격과 선형 공격 가정에 대한 반례를 제시함으로 전수 공격 외에는 다른 공격이 가능하지 않음을 보였고, 5절에서 제안 알고리즘은 DES 실행 시간의 10%정도 추가되며, 전수공격 측면 암호화 강도는 Triple DES(이하 3DES)보다 훨씬 안전하며, 효율성은 3DES에 비해 3배 빠름을 컴퓨터 실험으로 보였다.

논문에 사용될 기호를 다음과 같이 정의한다.

- $C(A, B)$ 원소별 5비트 Full Adder 연산, A, B 및 $C(A, B)$는 차원이 같은 행렬
- P는 DES의 P box로 1×32 행렬
- $P[k]$는 키 k로부터 변형된 P box
- One는 모든 원소가 1인 1×32 행렬

† 종신회원 : 공군사관학교 전산학과 교수
논문접수 : 2011년 2월 15일
수 정 일 : 1차 2011년 5월 27일
심사완료 : 2011년 5월 31일

- e_i는 $(i+1)$번 원소가 1이고 나머지 원소가 0인 32×1 단위 열벡터, $0 \le i \le 31$
- P_{key}는 가변 P box 생성에 사용될 확장키 64비트

2. 관련 연구

2.1 차분 공격

Biham과 Shamir에 의해 제안된 선택 평문 공격으로 입력 차분에 대한 출력 차분을 이용하여 키의 특정 비트를 추론할 수 있다[4]. 의도하는 입력 차분에 대한 출력 차분을 충분히 확보할 수 있음을 가정한다. 특정한 비트에 대한 입력 차분에 대해 출력 차분 발생 확률 특성을 이용하여 키를 추론할 수 있다. 차분 공격은 비선형 부분 연구에 집중되었다. 선택 평문 공격에 대한 DES의 차분 공격 기본 가정은 선형 부분인 P box가 고정된 상태로 특정 모델의 라운드를 완료하였을 때 입출력 차분 모델을 확률적으로 추론하여 전수 공격보다 적은 경우의 공격으로 키를 찾는다. 만약 P box를 라운드 별로 가변으로 설정한 후 가변 내용을 공개하지 않을 경우 입출력 차분에 대한 모델의 확률적 특성은 성립하지 않는다.

2.2 선형 공격

Matsui에 의해 제안된 기지 평문 공격으로 입력 비트, 출력 비트 및 키 비트에 대한 관계를 정리한 선형식의 확률적 특성을 이용하여 키를 추측한다[5]. 특정 라운드로 구성된 모델의 특징을 이용하여 키, 평문, 암호문의 선형 관계식을 유도한다. S box에 적용되는 키와 입출력문의 관계를 편차 특성을 이용한 확률로 선형식을 유도한다. 선형 공격 역시 Feistel 구조의 각 라운드에서 선형사상 되는 비트 위치가 고정된 특성을 이용한다. DES에서 P box가 확장된 개인키에 의해서 라운드 별로 비공개 가변으로 정의될 경우 매 라운드 마다 S box의 출력이 비트별로 선형 사상되는 위치는 예측할 수 없게 된다. 이 경우 선형공격에서 유도되는 선형식은 무의미하다.

DES 공격은 P box를 고정시킨 상태에서 선택평문 공격과 기지평문 공격을 고려하였다. 따라서 라운드 별 비공개 가변되는 선형 사상을 P box 위치에 적용할 경우 기존의 공격에 대한 특성은 성립되지 않는다.

DES의 대표적인 두 가지 공격에서 기본가정은 비선형 부분 S box와 선형 부분 P box가 공개되어 공격자가 구조를 파악할 수 있음이다. 이에 대한 대책은 S box를 변경하는 재설계 방안[6][7][8][9]을 다양하게 제안하고 있으나 그에 따른 문제점[10]도 지적되고 있다. 따라서 공개된 S box의 재설계로는 공격에 대한 문제점을 완전히 해결할 수 없으며 암호함수 자체를 수정하는 것이 바람직하다고 판단된다. 다음과 같이 제시되는 가변 P box도 암호함수 자체를 수정하는 방식이 될 수 있다.

만일 P box가 DES의 확장키 P_{key}에 따라 라운드별 비공개 가변으로 선형 사상할 수 있다면, P_{key}를 소유하지 않은 제 3 자는 라운드마다 변하는 P box를 형성할 수 없다. 이는 가변 P box를 이용하여 기존 공격을 방지하는 대책이 될 수 있다. 또한 P_{key}의 이용은 현대 블록 암호가 안전성을 요구하는 이상의 키 길이 확장 문제도 해결할 수 있다.

3. 제안 알고리즘

DES에서 비공개 가변 P box 이용 알고리즘을 다음과 같이 제안한다.

3.1 가변 P box

1) 가변 P box 조건

① 0부터 31로 중복 없이 구성된다.

② P_{key}부터 생성된 n비트 라운드 키 k_i에 의해서 가변 P box가 구성된다.

③ 가변 P box 생성시간은 암호화에 영향이 거의 없어야 한다.

2) 가변 P box 제안

라운드마다 가변 P box를 생성하는 방법 중 하나는 P_{key}로부터 유도된 라운드 키 k_i를 seed로 하여 0부터 31까지 중복 없이 생산되는 난수로 구성하는 것이다. 이 방법은 DES 에 난수 발생 알고리즘을 추가하는 부담이 있으며, 암호화 시간보다는 라운드마다 중복이 없는 가변 P box 생성 시간이 더 많이 소요되므로 비현실적이다.

가능한 다른 방법은 P box의 구성 원소에 대해 라운드마다 동일한 키 값 k_i를 5비트 Full Adder 연산 후 carry를 무시한 5비트 sum을 원소로 가변 P box를 구성한다. 이 방법은 가변 P box의 생성과정에서 추가조치 없이 원소 중복을 방지할 수 있으며, 라운드 키에 의해 생성되는 조건 등을 만족한다. 라운드 키 k_i로부터 재구성 된 가변 P box인 $P[k_i]$는 다음과 같다.

$$P[k_i] = C(P, k_i One)$$

이 경우 P의 j번 원소를 p_j라 할 때 $P[k_i]$의 j번 원소는 수학적으로 $(p_j + k_i) Mod 32$가 된다. 예를 들면 $k_i = 11$은 라운드 키에 의해서 P가 <표 1>과 같이 변한다.

〈표 1〉 $P[11]$

26	17	30	31	7	22	6	27
11	25	1	4	15	28	9	20
12	18	2	24	10	5	13	19
29	23	8	16	0	21	14	4

<표 1>의 $P[11]$은 라운드 키 k_i에 의해서 0부터 31까지 중복 없이 비트의 위치를 지정한다.

이 방법은 난수 발생기로 가변 P box를 만드는 것보다 효율적이나 매 라운드마다 5 비트 Full Adder 덧셈 연산 32회가 요구되며, 하드웨어 구성은 32개의 5 비트 Full Adder가 라운드마다 필요하므로 효율적이라 할 수 없다. S box 처리된 32비트 R에 대해 다음과 같은 k_i비트 circular shift left 연산 1회로 동일한 결과를 얻을 수 있다.

3) Circular shift left

S box에 의해서 처리된 32비트 R과 P는 다음과 같이 표시된다.

$$R=[r_0,r_1,\cdots,r_{30},r_{31}]$$
$$P=[e_{p_0},e_{p_1},\cdots,e_{p_{30}},e_{p_{31}}]$$

R에 대한 P box 선형 사상은 비트 위치를 다음과 같은 RP로 바꾼다.

$$RP=[r_{p_0},r_{p_1},\cdots,r_{p_{30}},r_{p_{31}}]$$

라운드 i에서 라운드 키 k_i를 적용한 가변 $P[k_i]$는 다음과 같이 표시할 수 있다.

$$P[k_i]=[e_{C(p_0+k_i)},e_{C(p_1+k_i)},\cdots,e_{C(p_{31}+k_i)}]$$

그리고 S box에 의해 처리된 R을 $P[k_i]$로 선형 사상하여 비트 위치를 변환시킨 $RP[k_i]$는 다음과 같다.

$$RP[k_i]=[r_{C(p_0+k_i)},r_{C(p_1+k_i)},\cdots,r_{C(p_{31}+k_i)}]$$

S box에 의해 처리된 R을 왼쪽으로 n 비트 왼쪽 순환 시프트 한 결과를 $R(n)$이라 하자.

$$R(n)=[r_n,r_{n+1},\cdots,r_{31},r_0,r_1,\cdots,r_{n-1}]$$
$$=[r_{C(0,n)},r_{C(1,n)},\cdots,r_{C(30,n)},r_{C(31,n)}]$$

$R(n)$에 P를 적용한 결과는 다음과 같다.

$$R(n)P=[r_{C(p_0+n)},r_{C(p_1+n)},\cdots,r_{C(p_{31}+n)}]$$

위의 결과에서 $n=k_i$를 적용하면 $RP[k_i]$와 $R(k_i)P$는 동일한 결과를 얻는다. 라운드 키 k_i로 $P[k_i]$를 얻기 위해 매 라운드마다 32개의 5비트 Full Adder 연산을 실시하는 대신 S box 처리된 결과 R에 대해 라운드 키 k_i만큼 왼쪽 순환 시프트 한 후 $R(k_i)$에 P를 적용하면 $RP[k_i]$와 동일한 결과를 얻을 수 있다.

이는 64비트 암호문 처리에 512번의 5비트 Full Adder 연산으로 얻는 결과 대신 R을 16번 왼쪽 순환 시프트로 처리하여 동일한 결과를 효율적으로 얻을 수 있음을 의미한다.

3.2 P_{key}구성

5비트 Full Adder 이용 알고리즘에서 각 라운드마다 가변시키는 키 k_i의 범위는 5비트 이하이다. 이 경우 DES는 16라운드로 구성되었으므로 기본키 64비트 외에 확장키 80비트가 필요하다. 소프트웨어 측면에서 변수는 8, 16, 32, 64비트로 구성되므로 k_i는 라운드 당 5비트보다는 4비트로 구성하는 것이 암호화 및 복호화 알고리즘 키 구성에 효율적이다. 5비트를 사용할 경우 전수 공격에 대한 암호화 강도는 4비트보다 크지만, 4비트도 공격으로부터 안전하다 가정할 경우 암호화 및 복호화 절차에 대한 프로그래밍의 효율성을 고려하여 라운드 키를 4비트로 제안한다.

3.3 알고리즘

제안 알고리즘은 64비트 메시지 입력에 대한 DES 암호 알고리즘과 유사하다. 차이점은 입력키 부분에서 기본키 key 외에 확장키 P_{key}가 추가되었다. P box 원소에 대해 5비트 Full Adder로 처리하는 가변 과정을 S box 처리된 32비트 R에 대해 왼쪽 순환 시프트를 적용하는 절차로 대체하여 높은 효율성을 구현한다. <표 2>의 DES 알고리즘에 다음과 같이 R의 왼쪽 순환 시프트를 추가함으로 가변 P box 대체 효과를 얻는다.

Circular_shift(&R, round, P_{key}) ;

P_{key}는 64비트이며 round마다 4비트씩 키 k_i를 생산한다. S box에서 처리된 R을 k_i비트 왼쪽 순환 시프트 시킨 후 P를 적용하면, R에 가변 $P[k_i]$를 적용한 것과 동일한 결과를 얻는 알고리즘이다.

〈표 2〉 제안 알고리즘

```
unsigned long  *Enc_un32_Pshft_DES ( unsigned long
*x, unsigned long *key, unsigned long *Pkey )
// x 평문 입력 64비트
// key DES key 64비트, Pkey 확장키 64비트
{
unsigned long L, R, tmp_L, P[2] = { 0L } ;
unsigned long keyL, keyR ;
short round, i ;

// round 키 생성 초기값
PC_1( KEY, &keyR, &keyL ) ;

Intial_P ( X );
R = X[0] ;
L = X[1] ;
```

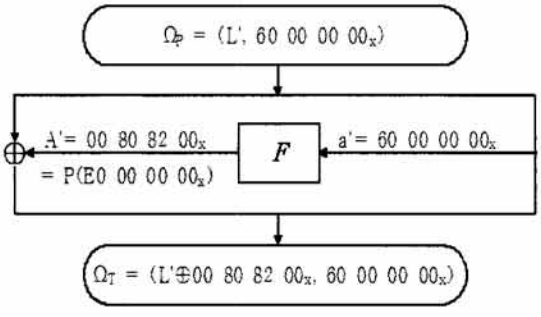
```
for ( round = 0 ; round < 16 ; round ++ )
{
// 부분 key 생산 부분
      L_shft( &keyR, LS[round] ) ;
      L_shft( &keyL, LS[round] ) ;
      PC_2( KEY, keyR, keyL ) ;

// round 암호화 부분
      tmp_L = L ;
      L = R ;
      E_table ( R, P ) ;
      for ( i = 0 ; i < 2 ; i ++ )
            P[i] ^= KEY[i] ;
      S_box( P, &R ) ;
      Circular_shift( &R, round, Pkey ) ;
      P_table( &R ) ;
      R ^= tmp_L ;
}

X[0] = L ;
X[1] = R ;
return R_Intial_P ( X );
}
```

4. 안전성 검토

제안된 알고리즘에 대해 차분 공격과 선형 공격 기본 모델을 적용하여 안전성을 검토한다.

4.1 차분 공격

차분 공격은 선택평문 공격을 가정한다. 따라서 원하는 입력에 대한 출력 차분도 얻을 수 있다. key의 특정 비트를 찾기 수월한 입력 차분을 가정하고 그에 대한 출력 차분을 얻는다. (그림 1)의 1라운드 특성은 14/64 확률로 입력 차분에 대한 출력 차분의 관계가 성립된다[4].



(그림 1) one round characteristic

제안된 알고리즘과 같이 P box를 라운드 별로 키에 따라 가변인 경우 (그림 1)과 같은 고정된 $\acute{A}$을 얻을 수 없다. 예를 들면 라운드 1에서 라운드 키 k_1가 1인 경우 P box는 $P[1]$이 되며 F 처리된 후 값은 다음과 같다.

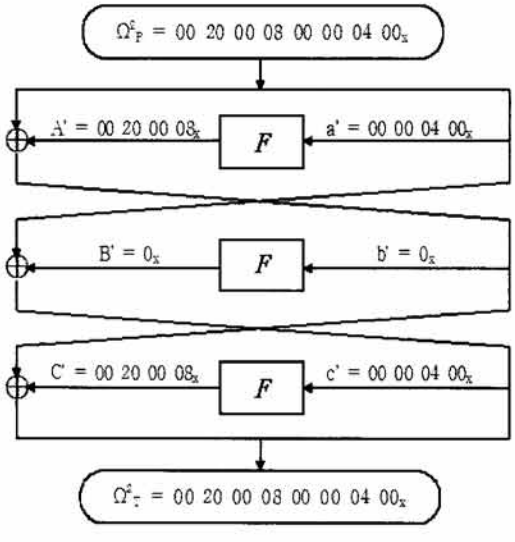
$$\acute{A} = P[1](E0\ 00\ 00\ 00_x) = 00\ 80\ 88\ 00_x$$

만일 k_1이 2 인 경우 F 처리된 후 값은 다음과 같다.

$$\acute{A} = P[2](E0\ 00\ 00\ 00_x) = 00\ 82\ 08\ 00_x$$

라운드 키 k_1 값에 따라 출력 차분 $\acute{A}$이 변화한다.

(그림 2)에서 제시하는 예제 역시 P box가 고정된 경우에 1/16 확률로 입력 차분에 대한 출력 차분 특성을 얻을 수 있다[4]. (그림 1)과 같은 라운드 키($k_i \neq 0, i = 0,1,2$)에 의한 가변 P box를 적용할 경우 $\acute{A} \neq 00\ 20\ 00\ 08_x$ 이다. 연쇄적인 결과로 $\acute{b} \neq 0_x$, $\acute{B} \neq 0_x$이 되며, $\acute{c} \neq 00\ 00\ 04\ 00_x$와 $\acute{C} \neq 00\ 20\ 00\ 08_x$된다. 따라서 Ω_T^2는 의도된 출력 차분을 얻을 수 없다. 제안된 알고리즘과 같이 P box가 라운드 키에 따라 가변되는 경우 어떠한 선택 평문 공격에 대해서 적당한 확률로 입력 차분에 대한 출력 차분을 유도할 수 없다. 라운드 별로 가변 P box를 적용시킨 입출력 차분에 대해서 DC table은 의미를 가질 수 없으므로 차분 공격 가능한 모델을 만들 수 없다. 따라서 모든 라운드에 비공개 가변 P box를 적용한 경우 차분 공격은 불가능하다.



(그림 2) 3 round characteristic

4.2 선형 공격

선형공격은 무작위로 제시된 평문과 대응하는 암호문 및 키에 대해 유의 확률 p로 다음과 같은 선형 근사식을 구성한다. 기호와 예문은 Matsui 논문[5]을 인용한다.

$$P[i_1, i_2, \cdots, i_a] \oplus C[j_1, j_2, \cdots, j_b] = K[k_1, k_2, \cdots, k_c]$$

(그림 3)에서 제 1 단의 F 함수에 식을 적용하면 12/64 확률로 다음 식들이 성립한다.

$$X_2[7,18,24,29] \oplus P_H[7,18,24,29] \oplus P_L[15] = K_1[22]$$

$$X_2[7,18,24,29] \oplus C_H[7,18,24,29] \oplus C_L[15] = K_3[22]$$

위 두 식에서 X2를 소거하면

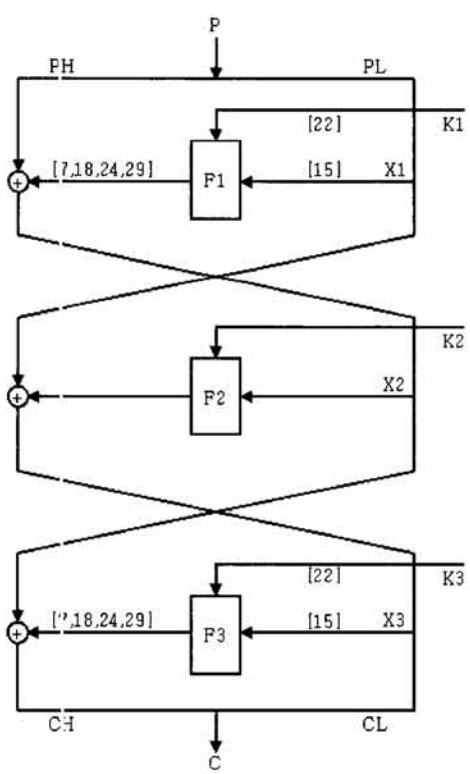
$$P_H[7,18,24,29] \oplus C_H[7,18,24,29] \oplus P_L[15] \oplus C_L[15] = K_1[22] \oplus K_3[22]$$

이러한 관계에 대해 제안된 알고리즘을 적용하여 라운드 1과 3에 키 k_1과 k_3를 각각 적용할 경우 식은 다음과 같다.

$$X_2[C(7,k_1), C(18,k_1), C(24,k_1), C(29,k_1)] \oplus P_H[C(7,k_1), C(18,k_1), C(24,k_1), C(29,k_1)] \oplus P_L[15] = K_1[22]$$

$$X_2[C(7,k_3), C(18,k_3), C(24,k_3), C(29,k_3)] \oplus C_H[C(7,k_3), C(18,k_3), C(24,k_3), C(29,k_3)] \oplus C_L[15] = K_3[22]$$

만일 $k_1 \neq k_3$ 경우 X_2는 소거될 수 없다. 또한 $k_1 = k_3$ 이라도 평문과 암호문의 정확한 비트 위치를 파악은 곤란하다. 이와 같이 간단한 3단 DES에 대해서도 가변 P box를 적용할 때 입력 평문과 출력 암호문 및 키 관계를 설정할 수 있는 최량 표현은 불가능하다. 따라서 모든 라운드에 비공개 가변 P box를 적용할 경우 선형 공격은 불가능하다.



(그림 3) 3-round DES cipher

5. 실 험

64비트 메시지 M에 대해 암호화 및 복호화 알고리즘 정확성과 효율성을 다음과 같이 측정하였다.

5.1 실험 일반

실험에 사용된 알고리즘들은 C++로 작성하였다. 암호화 및 복호화 과정은 업무용 PC에서 실행하였다. PC 성능은 Intel Atom CPU 230, 1.6GHz, 0.99 GB RAM이다.

DES 암호화 최소 블록단위 64비트 M에 대한 암호화 및 복호화 시간을 측정하였다. 임의로 선택한 64비트 메시지 M은 다음과 같다.

M ={ 65, 66, 67, 68, 69, 'F', 'G', 'H' }

실험은 알고리즘 DES, 3DES 및 제안 알고리즘 3 개를 구현한 후 각 알고리즘의 정확성 확인과 메시지 M의 암호화 및 복호화 과정 시간측정으로 효율을 비교하였다.

5.2 실험 결과

세 암호화 알고리즘은 M에 대해 암호화/복호화 과정에서 동일한 M을 생산하였다.

DES의 암호화 및 복호화 시간은 대단히 짧아 1회 측정으로 정확한 값을 얻을 수 없다. 64비트 M에 대한 암호화/복호화 과정을 1000만 번 반복하여 각 알고리즘에 대한 암호화/복호화 실행 평균시간을 <표 3>과 같이 얻었다.

〈표 3〉 효율성 비교(시간 μ-sec)

구 분	시간	키 길이	실 키 길이
DES	52	64	56
3DES	168	128	112
제안 DES	58	128	120

DES와 3DES의 암호화 및 복호화 시간비교에서 효율성은 라운드 수에 의존함을 알 수 있다. 제안 알고리즘은 라운드마다 가변 P box 생성과 동일한 효과를 주는 왼쪽 순환 시프트에 약 10% 시간이 추가되었다. 이는 3DES보다 훨씬 적은 시간이 소요되며, DES 시간과 유사하다. 또한 실험결과 제안된 알고리즘은 다음과 같이 요약 할 수 있다.

① DES는 차분 공격과 선형 공격에 취약점이 있으나 제안

된 알고리즘은 차분 공격과 선형 공격모델이 적용될 수 없다.

② 3DES의 실 키 길이는 112비트이나 제안된 알고리즘의 실 키 길이는 120비트로 전수 공격에 더 강하다.

③ 제안 알고리즘의 암호화 속도측면에서 효율성은 DES와 유사하며 3DES보다 약 3배 빠르다.

6. 맺음말

본 연구는 확장된 DES 키로부터 각 라운드에 사용되는 P box를 비공개 가변 형태로 효율적으로 변형할 수 있는 알고리즘을 제안하였다. 5비트 Full Adder 연산 대신 S box 처리된 32비트를 왼쪽 순환 시프트 시킴으로 동일한 효과가 있음을 보인 후 이 부분을 DES알고리즘에 적용 하였다. 제시된 가변 P box 생성은 키에 의존하므로 확장키가 다를 경우 입력과 출력에 대한 고정된 분석이 불가능하다. 따라서 차분 공격과 선형 공격 가능한 모델을 제시할 수 없음을 보이면서 기존의 공격에 대해 안전함을 보였다.

또한 컴퓨터 실험을 통해 제안된 알고리즘이 3DES보다 약 3배 정도 빠르며, DES 성능과 유사한 시간이 요구되지만 암호화 강도는 전수 공격을 가정할 경우 3DES보다 훨씬 좋음도 보였다. 제안된 알고리즘은 키를 제외한 모든 과정이 공개되는 Kerckhoffs 조건[11]을 모두 만족하는 블록 알고리즘이다.

참 고 문 헌

[1] 홍성룡, 조정호, "SDR System 적용을 위한 한국형 알고리즘(SEED) 구현 및 성능분석", 한국정보과학회 2003년 봄 학술발표 논문집 제 30 권 제 1 호, pp.319-321, 2003. 4.

[2] E. Biham and A. Shamir "Differentila Cryptanalysis of the full 16 round DES", Crypto'92 Extended Abstracts, 1992.

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

[4] E. Biham and A. Shamir "Differentila Cryptanalysis of DES like Crypto systems", Jour. of CRYPTOLOY, Vol.4, No.1, 1991.

[5] M Matsui, "Linear Cryptanalysis Method for DES Cipher", Abstracts of EUROCRYPT'93, pp.W112-W123, May, 1993.

[6] E. Biham and A. Shamir "Differentila Cryptanalysis of FEAL and N-hash", technical report, Weizmann Institute of Science, Israel, 1991.

[7] H. Fiestel, "Cryptography and Computer Privacy", Science America, Vol.228, No.5. 1973.

[8] J. B. Kam and G. I. Davida, "Structured Design of Substitution Permutation Encryption Network", IEEE Tran. on Computer, Vol.C-28, No.10, Oct., 1979.

[9] M. H. Deason and S. E. Tavares, "An Expanded Set of S box Design Criteria based on information theory and its Relation to differential-Like attacks" Proc. of EUROCRYPTO'91, Springer-Verlag, 1991.

[10] L. Brown, M. Kwan, J. Pieprzyk and J. Seberry, "Improving Resistance to Differential Cryptoanalysis and Redesign of LOKI", Abstract of ASIACRYPT'91, 1991.

[11] Mark Stamp, "Information Security-Principles and Practice", Wiley, 2005.

이 준

e-mail : jlee@afa.ac.kr

1981년 공군사관학교

1985년 서울대학교 수학과

1988년 Naval Postgraduate School 전산공학(석사)

1994년 University of Florida 전산공학(박사)

1989년~현 재 공군사관학교 전산학과 교수

관심분야 : 암호알고리즘