Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.28, NO.6, Dec. 2018 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2018.28.6.1379

DLCT를 활용한 향상된 차분선형 분석*

김 현 우,[1†] 김 성 겸,[1] 홍 득 조,[2‡] 성 재 철,[3] 홍 석 희[1]
[1]고려대학교, [2]전북대학교, [3]서울시립대학교

Improved Differential-Linear Cryptanalysis Using DLCT*

Hyunwoo Kim,[1†] Seonggyeom Kim,[1] Deukjo Hong,[2‡]
Jaechul Sung,[3] Seokhie Hong[1]
[1]Korea University, [2]Chonbuk National University, [3]University of Seoul

요 약

차분선형 분석의 복잡도는 라운드 독립성, 선형 근사식 독립성, 차분 경로를 만족하지 못하는 경로에 대한 균일성 가정 아래 계산되는 차분선형 특성의 확률에 큰 영향을 받는다. 따라서 차분선형 특성의 정확한 확률을 계산하는 것은 공격의 유효성과 관련된 매우 중요한 문제이다. 본 논문은 차분선형 분석을 위한 새로운 개념 DLCT(Differential-Linear Connectivity Table)를 제안한다. 그리고 DLCT를 적용하여 선형 근사식 독립성 가정을 완화할 수 있는 차분선형 특성의 향상된 확률 계산 방법을 제안하며, DES와 SERPENT에 적용하여 기존 분석결과를 재분석한다. DES의 7-라운드 차분선형 특성의 확률은 $1/2+2^{-5.81}$, SERPENT의 9-라운드 차분선형 특성의 확률은 $1/2+2^{-57.9}$로 다시 계산되었고 공격에 필요한 데이터 복잡도는 각각 $2^{0.2}$, $2^{2.2}$배 감소한다.

ABSTRACT

The complexity of the differential-linear cryptanalysis is strongly influenced by the probability of the differential-linear characteristic computed under the assumption of round independence, linear approximation independence, and uniformity for the trail that does not satisfy differential trail. Therefore, computing the exact probability of the differential-linear characteristic is a very important issue related to the validity of the attack. In this paper, we propose a new concept called DLCT(Differential-Linear Connectivity Table) for the differential-linear cryptanalysis. Additionally, we propose an improved probability computation technique of differential-linear characteristic by applying DLCT. By doing so, we were able to weaken linear approximation independence assumption. We reanalyzed the previous results by applying DLCT to DES and SERPENT. The probability of 7-round differential-linear characteristic of DES is $1/2+2^{-5.81}$, the probability of 9-round differential-linear characteristic of SERPENT is computed again to $1/2+2^{-57.9}$, and data complexity required for the attack is reduced by $2^{0.2}$ and $2^{2.2}$ times, respectively.

Keywords: Differential-Linear Cryptanalysis, DLCT(Differential-Linear Connectivity Table), DES, SERPENT

Received(10. 05. 2018), Modified(11. 13. 2018), Accepted(11. 14. 2018)

* 본 연구는 고려대 암호기술 특화연구센터(UD170109ED)를 통한 방위사업청과 국방과학연구소의 연구비 지원으로 수행되었습니다.

† 주저자, gusdn0319@gmail.com
‡ 교신저자, deukjo.hong@jbnu.ac.kr(Corresponding author)

I. 서 론

1.1 개요

차분 분석(differential cryptanalysis)은 평문의 차분이 암호화가 진행되면서 암호문의 차분에 어떠한 영향을 미치는지 분석하는 암호 분석 기법으로, Biham과 Shamir에 의해 1990년 처음 소개되었다[1]. 선형 분석(linear cryptanalysis)은 평문과 암호문, 그리고 키 사이의 관계를 선형 식으로 근사시켜서 분석하는 암호 분석 기법으로 Matsui에 의해 1992년 처음 소개되었다[2]. 차분 분석은 한 라운드 또는 그 이상의 라운드의 차분 특성(differential characteristic)을 사용하여 암호 분석이 진행되며, 선형 분석의 경우에는 선형 근사(linear approximation)를 사용하여 암호 분석을 진행한다.

차분 특성 및 선형 근사는 일정 라운드 이상으로 길게 구성할 경우, 해당 특성을 만족할 확률이 매우 낮아지기 때문에 분석에 사용할 수 없게 된다. 따라서 1994년 Langford와 Hellman은 확률 1인 3-라운드 (부정) 차분 특성과 확률이 가장 좋은 3-라운드 선형 근사를 연결하여 6-라운드의 특성을 구성하는 차분선형 분석 기법을 제안했다[3].

차분선형 분석 기법이 제안된 후, Biham 등은 2002년에 확장된 차분선형 분석 기법을 제안했다[4]. Langford와 Hellman은 확률 1의 (부정) 차분 특성을 사용하였지만 Biham 등은 확률이 1보다 작아지는 (부정) 차분 특성에 대해서도 동일하게 차분선형 특성을 구성할 수 있음을 보였다. Biham 등은 확률이 1보다 작은 4-라운드 (부정) 차분 특성을 사용하여 DES[5]에 대한 분석결과를 제시했고, [6]에서는 SERPENT[7]에 대한 차분선형 분석결과를 제시하였다.

2015년에는 Jiqiang Lu가 자동화 탐색 기법을 사용하여 전체 차분 특성을 탐색하고 Biham 등이 사용했던 확장된 차분선형 특성의 확률 계산에 필요한 가정을 완화했다. 그리고 이를 DES에 적용하여 Biham 등의 결과보다 더 긴 라운드의 차분선형 특성을 구성하였다[8]. 차분선형 분석은 소개된 이후에 다양한 암호들에 대한 적용이 이루어지고 있으며, 더 높은 확률을 갖는 차분선형 특성을 구성하기 위한 연구가 이루어지고 있다.

차분선형 분석과 유사한 개념으로 1999년 David Wagner에 의해 제안된 부메랑 공격(boomerang attack)이 있다[9]. 부메랑 공격은 좋은 확률을 갖는 짧은 라운드의 차분 특성을 독립적으로 연결하여 긴 라운드의 쿼텟(quartet)을 구성하여 공격하는 기법이다. 부메랑 공격은 제안된 이후, 지속적으로 연구되어 확장된 부메랑 공격(amplified boomerang attack)[10], 렉탱글 공격(rectangle attack)[11]으로 발전되어 왔다. 또한, 높은 확률을 갖는 쿼텟 구성을 위해 Ladder Switch, S-box Switch, Feistel Switch 등의 기법이 제안되었다[12, 13]. 부메랑 공격은 기본적으로 두 차분 특성이 독립적이라는 가정아래 쿼텟이 구성된다. 그러나 실제로는 두 특성이 독립적이지 않으므로 꾸준히 쿼텟의 존재성에 대한 문제가 지적되어왔다[14]. 2018년 Carlos Cid 등은 부메랑 쿼텟 분석 도구인 BCT(Boomerang Connectivity Table)를 제안하였고 이를 통해 쿼텟의 존재성 문제를 해결하였다. 또한, S-box Switch 기법에서 S-box 입력 값들의 상호 종속성으로 인해 발생하는 일반화된 스위칭 효과(generalized switching effect)를 제안하여 쿼텟의 확률을 개선 시켰다[15].

차분선형 분석도 마찬가지로 차분 특성과 선형 특성이 완전히 독립적으로 구성된다고 볼 수 없으므로 유사한 문제들을 가질 수 있다. 일반적으로 차분선형 분석 기법은 차분선형 특성을 구성하고 특성이 가지는 확률을 계산하여 공격의 복잡도를 결정한다. 따라서 확률을 정확하게 계산하는 것은 실제 공격의 유효성과 관련된 중요한 문제이다. 그러나 기존에 제안된 확률 계산 방법은 실제 분석에서 만족하지 못하는 몇 가지 독립성 가정들을 요구하기 때문에 계산된 확률이 실제 확률과 차이가 생길 수 있다. 따라서 가정을 완화하여 확률 계산을 할 수 있다면 기존의 결과보다 더 정확한 안전성 분석이 가능하다.

이를 위해 본 논문은 DLCT(Differential-Linear Connectivity Table)를 제안하며, DLCT를 적용한 새로운 특성의 확률 계산 방법을 제안한다. 제안하는 방법은 차분의 균일 분포(uniform distribution) 가정을 사용하여 일반화될 수 있으며, 균일 분포 가정을 사용하지 않는다면 실험적인 결과를 바탕으로 차분선형 특성의 확률을 계산할 수 있다. 균일 분포 가정에서 벗어날수록 더 정확한 확률을 계산할 수 있다. 이 방법을 실제로 Biham 등이 제안한 DES의 7-라운드 차분선형 특성과 SERPENT의 9-라운드 차분선형 특성에 적용하여 확률을 다시 계산한다. DES의 경우 확률이 $1/2+2^{-5.81}$으로 계산되었고 기존 분석 결과와 비교하여 데이터 복잡도가 약 $2^{0.2}$배 감소한

다. SERPENT의 경우 확률이 $1/2+2^{-57.9}$로 계산되었고 기존 분석결과와 비교하여 데이터 복잡도는 약 $2^{2.2}$배 감소한다.

본 논문은 다음과 같이 구성된다. 2장에서는 차분 선형 분석에 대해 자세히 설명하고 세 가지 가정을 설명한다. 3장에서는 DLCT를 정의하고 DLCT를 사용한 차분선형 특성 확률 계산 방법을 제안한다. 4장에서는 제안한 방법을 DES와 SERPENT에 적용하여 차분선형 특성의 확률을 다시 계산한 결과를 제시한다. 마지막으로 5장에서는 결론과 향후 연구 방향을 서술한다.

II. 배경지식

본 논문에서 사용하는 표기법은 다음과 같다.

- $\oplus$: 배타적 논리합(exclusive-OR, XOR)
- $\odot$: 내적(boolean inner product)
- $\circ$: 합성 함수(function composition)
- E : 블록 암호(block cipher)
- Ω_P : 입력 차분(input difference)
- Ω_T : 출력 차분(output difference)
- λ_P : 입력 마스크(input mask)
- λ_T : 출력 마스크(output mask)

2.1 차분 분석(Differential Cryptanalysis)

차분 분석은 1990년 Biham과 Shamir에 의해 처음 소개된 블록암호 분석 기법으로, 평문 쌍의 차분이 해당 암호문 쌍의 차분에 어떤 영향을 미치는지를 분석하는 기법이다[1]. 차분은 다양한 방법으로 정의될 수 있지만 일반적으로 배타적 논리합(XOR) 연산으로 정의되며, 입력 값인 평문 쌍의 차분을 입력 차분 그리고 출력 값인 암호문 쌍의 차분을 출력 차분이라고 부른다. 입력 차분과 출력 차분의 조합을 디퍼렌셜(differential)이라고 정의한다. 디퍼렌셜의 확률은 다음과 같이 정의된다.

정의 1. n-비트 차분 Ω_P, Ω_T에 대해 디퍼렌셜 $\Omega_P \rightarrow \Omega_T$의 확률은 다음과 같다.

$$\Pr_E(\Omega_P \rightarrow \Omega_T) = \Pr_{P \in \{0,1\}^n}(E(P) \oplus E(P \oplus \Omega_P) = \Omega_T)$$

$$= \frac{\left|\left\{x | E(x) \oplus E(x \oplus \Omega_P) = \Omega_T, x \in \{0,1\}^n\right\}\right|}{2^n}$$

또한, 차분 특성(differential characteristic) 또는 차분 경로(differential trail)는 다음과 같이 정의한다.

정의 2. r-라운드 블록 암호에 대한 입, 출력 차분을 Ω_P, Ω_T, 각 라운드 함수를 f_{k_i}, $i(2 \le i \le r-1)$-라운드 입, 출력 차분을 α_{i-1}, α_i이라고 할 때, 다음을 차분 특성 또는 차분 경로라고 정의한다.

$$DC = (\Omega_P \xrightarrow{f_{k_1}} \alpha_1 \xrightarrow{f_{k_2}} \alpha_2 \xrightarrow{f_{k_3}} \ldots \alpha_{r-2} \xrightarrow{f_{k_{r-1}}} \Omega_T)$$

임의 함수(random function)의 경우, 임의의 디퍼렌셜 (Ω_P, Ω_T)에 대한 확률은 2^{-n}이다. 따라서 $\Pr_E(\Omega_P \rightarrow \Omega_T)$가 2^{-n}보다 크다면 충분한 수의 선택 평문 쌍을 사용하여 임의 함수와 E를 구별할 수 있다.

2.2 선형 분석(Linear Cryptanalysis)

선형 분석은 1992년 Matsui에 의해 제안된 분석 기법으로, 평문과 암호문, 그리고 키 사이의 관계를 선형 함수로 근사시켜서 분석하는 기법이다[2]. 선형 분석은 블록 암호의 입력의 특정 선형 함수와 출력의 특정 선형 함수 간의 상관관계를 이용한다. 두 선형 함수의 조합을 선형 근사(linear approximation)라고 한다. 가장 널리 사용되는 선형 함수는 특정 이진 벡터(binary vector)를 사용하여 블록의 비트 값들의 내적을 계산하는 것이다. 입력과 결합 된 값을 입력 마스크라고 하며, 출력에 적용된 값을 출력 마스크라고 한다. 선형 근사의 확률은 다음과 같이 정의된다.

정의 3. n-비트 입, 출력 마스크 λ_P, λ_T에 대해 선형 근사의 확률은 다음과 같다.

$$\Pr_E(\lambda_P \rightarrow \lambda_T) = \Pr_{P \in \{0,1\}^n}(P \odot \lambda_P = E(P) \odot \lambda_T)$$

$$= \frac{\left|\left\{x | x \odot \lambda_P = E(x) \odot \lambda_T, x \in \{0,1\}^n\right\}\right|}{2^n}$$

임의 함수의 경우, 임의의 쌍 (λ_P, λ_T)에 대해 선형 근사의 예상 확률은 1/2이다. 선형 근사의 바이어스(bias)는 $|\Pr_E(\lambda_P \rightarrow \lambda_T) - 1/2|$ 로 정의된다.

따라서 바이어스가 충분히 클 경우, 충분한 수의 평문, 암호문 쌍이 주어 지면 임의 함수와 E를 구별할 수 있다.

정의 1과 2는 가능한 모든 평문과 키의 조합을 사용해야만 정확한 값을 구할 수 있다. 그러나 실제 널리 사용되고 있는 64-비트 또는 128-비트의 큰 블록 크기를 갖는 블록 암호에 적용하기는 어렵다.

따라서 실제 확률이 아닌 확률의 근사값을 계산하여 사용하게 된다. 일반적으로 다음 가정 1을 사용하여 반복 블록 암호를 마코브 블록 암호(markov block cipher)로 간주하여 확률을 계산한다[16]. 디퍼렌셜의 확률은 차분 특성을 구성하는 각 한 라운드 차분 특성 확률의 곱으로 전체 확률을 계산하며, 선형 근사의 경우 Piling-up Lemma[2]를 사용하여 계산한다.

가정 1. 각 라운드 함수는 독립이다.

많은 블록 암호 분석 연구는 위의 가정 1을 전제로 이루어져 왔으며 정확하진 않지만, 합리적인 가정으로서 간주 되고 있다. 본 논문은 기본적으로 가정 1을 전제로 한다.

2.3 차분선형 분석(Differential-Linear Cryptanalysis)

차분 분석과 선형 분석이 제안된 이후 1994년 Langford와 Hellman은 차분선형 분석을 제안하면서 두 가지 분석 기법이 조합되어 사용될 수 있음을 보였다[3]. Langford와 Hellman은 확률 1을 가지는 3-라운드 (부정) 차분 특성을 3-라운드 선형 근사와 연결시켜 6-라운드 차분선형 특성을 구성했다. 차분선형 분석도 기본적으로 가정 1을 전제로 하며, 특성의 확률 계산을 위해 추가적으로 다음 가정을 사용한다. 전체 블록 암호 E는 차분 특성이 적용되는 E_0와 선형 근사가 적용되는 E_1의 합성으로 정의한다.

가정 2. 전체암호 E를 $E=E_1 \circ E_0$라고 하면, E_1에 적용되는 선형 근사의 두 입력 값 $E_0(P)$와 $E_0(P\oplus\Omega_P)$는 각각 독립적이다.

가정 2 역시 선형 근사에 두 입력 값이 완전히 독립적이라고 볼 수 없으므로 정확하진 않지만, 합리적인 가정으로서 간주 되고 있다. Langford와 Hellman은 차분 특성의 확률이 1, 선형 근사의 확률이 $1/2+q$인 차분선형 특성의 확률을 다음과 같이 계산한다.

두 평문 P_1, $P_2=P_1\oplus\Omega_P$에 대해, $\Omega_T\odot\lambda_P=0$인 경우 $\lambda_P\odot E_0(P_1)=\lambda_P\odot E_0(P_2)$를 만족하고 $\Omega_T\odot\lambda_P=1$인 경우 $\lambda_P\odot E_0(P_1)\neq\lambda_P\odot E_0(P_2)$이다. 선형 근사의 바이어스를 q라고 하면, 선형 근사는 $1/2+q$의 확률로 $\lambda_P\odot P=\lambda_T\odot E_1(P)$를 만족시킨다. 따라서 $1/2+q$의 확률로 다음 두 식이 성립한다.

$$\lambda_P\odot E_0(P_1)=\lambda_T\odot E_1(E_0(P_1)) \quad (1)$$

$$\lambda_P\odot E_0(P_2)=\lambda_T\odot E_1(E_0(P_2)) \quad (2)$$

Langford와 Hellman은 $\lambda_P\odot E_0(P_1)=\lambda_P\odot E_0(P_2)$를 만족하면서 확률이 1인 차분 특성을 사용하기 때문에 등식 $\lambda_P\odot E_0(P_1)=\lambda_P\odot E_0(P_2)$은 1의 확률로 만족하게 되며, 식 (1)과 (2)에 의해 $1/2+2q^2$의 확률로 $\lambda_T\odot C_1=\lambda_T\odot C_2$ 등식을 만족하게 된다. C_1과 C_2는 각각 P_1, P_2에 대한 암호문을 의미한다.(즉, $C_i=E_1(E_0(P_i))$).

차분선형 분석 기법이 소개된 이후, 2002년에 Biham 등은 차분선형 기법을 확률이 1보다 작은 (부정) 차분 특성에도 적용할 수 있도록 확장하였다[4]. Langford와 Hellman의 경우, 확률 1인 차분 특성을 사용하므로 등식 $\lambda_P\odot E_0(P_1)=\lambda_P\odot E_0(P_2)$가 항상 성립한다. 그러나 확률이 $p'(<1)$인 차분 특성을 사용하게 되면, 차분 특성을 따르지 않는 출력 차분이 존재하므로 $\lambda_P\odot E_0(P_1)=\lambda_P\odot E_0(P_2)$가 성립하지 않는 경우가 발생한다. 이 경우의 확률을 계산하기 위해 Biham 등은 차분 특성을 따르지 않는 경우, 등식 $\lambda_T\odot E(P_1)=\lambda_T\odot E(P_2)$의 성립 확률을 다음 가정을 사용하여 계산한다.

가정 3. 두 내적 값 $\lambda_T\odot E(P)$와 $\lambda_T\odot E(P\oplus\Omega_P)$는 $E_0(P)\oplus E_0(P\oplus\Omega_P)\neq\Omega_T$인 경우(설정한 차분 경로를 따르지 않을 때)에는 균일하고 독립적인 분포를 따른다. 즉,

$$\Pr(\lambda_T\odot E(P)=\lambda_T\odot E(P\oplus\Omega_P)\mid E_0(P)\oplus E_0(P\oplus\Omega_P)\neq\Omega_T)=\frac{1}{2}$$

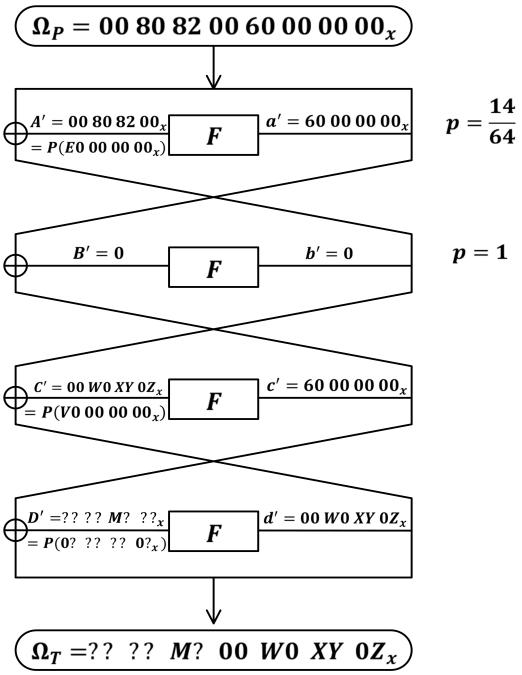

Fig. 1. The 4-Round Differential Characteristic Used in [4] (where $V \in \{1,...,F_x\}$, $W \in \{0,8\}$, $X \in \{0,8\}$, $Y \in \{0,2\}$, $Z \in \{0,2\}$, $M \in \{0,...,7\}$ and '?' means random value in $\{0,1\}$)

Biham 등은 $\lambda_P \odot E_0(P_1) = \lambda_P \odot E_0(P_2)$가 성립할 확률을 $1/2+p$로 선형 근사시켜 표현한다. 이때, p를 차분 연결 바이어스(differential connection bias)라고 정의한다. 차분 연결 바이어스는 가정 3에 의해 다음과 같이 계산된다.

$$\frac{1}{2}+p \approx p'+(1-p')\times\frac{1}{2}=\frac{1}{2}+\frac{p'}{2} \quad (3)$$

Langford와 Hellman의 경우, $p=1/2$, $p'=1$이다. 따라서 세 개의 가정 아래에서 전체 차분선형 특성의 확률은 $1/2+4pq^2$으로 계산할 수 있다. $4pq^2$은 차분선형 특성의 바이어스라고 지칭한다. 차분선형 특성의 확률은 $1/2+2p'q^2$으로도 나타낼 수 있으므로 $2p'q^2$도 차분선형 특성의 바이어스를 나타낸다.

Biham 등은 4-라운드 부정 차분 특성과 3-라운드 선형 근사를 사용해서 7-라운드 차분선형 특성을 구성하였고 차분 연결 바이어스는 다음과 같이 계산하였다.

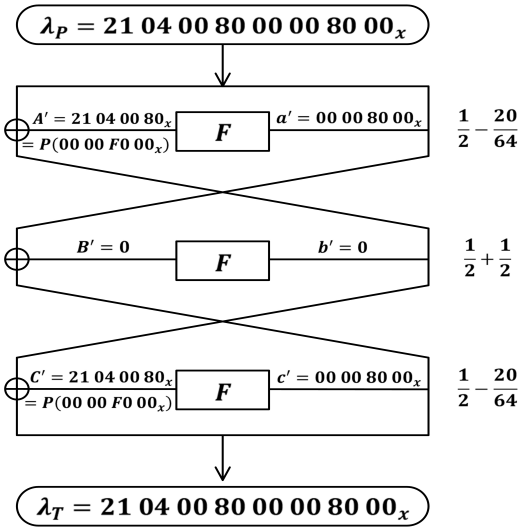

Fig. 2. The 3-Round Linear Approximation Used in [4]

$$\frac{1}{2}+p \approx \frac{14}{64}+(1-\frac{14}{64})\times\frac{1}{2}=\frac{1}{2}+0.109$$

또, 선형 근사 바이어스 $q=0.195$로 최종 차분선형 특성의 확률 $1/2+4pq^2$을 다음과 같이 계산하였다.

$$\frac{1}{2}+4\times 0.109\times(0.195)^2=\frac{1}{2}+0.0167$$

Fig.1은 [4]에서 사용한 4-라운드 부정 차분 특성을 나타낸 그림이며, Fig.2는 [3, 4]에서 사용한 3-라운드 선형 근사를 나타낸 그림이다.

Jiqiang Lu는 [8]에서, 자동화 탐색 기법을 사용하여 모든 특성을 탐색함으로써 가정 3을 완화하고 확률 계산을 진행하였다. 이를 통해 전체 차분 특성을 탐색할 수 있는 실험 범위 내에서 Biham 등의 결과보다 더 긴 라운드의 차분선형 특성을 구성하였다.

III. DLCT

본 장에서는 DLCT(Differential-Linear Conn-ectivity Table)에 대해 정의하고 DLCT를 적용한 차분선형 특성의 확률 계산에 대해 설명한다. 앞에서 설명한 것처럼 [3, 4]에서는 차분선형 특성의 확률을 계산하기 위해 세 가지 가정을 사용한다. 그

러나 P_1과 P_2는 $P_2 = P_1 \oplus \Omega_P$ 관계를 만족하기 때문에 완전히 독립적인 값은 아니다. 따라서 차분 특성이 적용되는 E_0후의 출력 값 $E_0(P_1)$과 $E_0(P_2)$는 서로 완전히 독립적인 값이라고 할 수 없다. 즉, 가정 2를 사용하여 선형 근사의 두 입력 값에 대한 선형 근사 확률을 서로 독립적으로 계산하는 것은 차분선형 특성의 정확한 확률을 계산하지 못할 수도 있다. 본 논문에서는 가정 2를 완화 시키기 위해 다음과 같은 DLCT를 정의한다.

정의 4. n-비트 S-box에 대해, 입력 차분을 $\triangle$, 출력 값에 대한 마스크 값을 α라고 할 때 $DLCT_S$ (Differential-Linear Connectivity Table of S-box)는 다음과 같이 정의한다.

$$DLCT_S(\triangle, \alpha) = \#\ \{x | S(x) \odot \alpha = S(x \oplus \triangle) \odot \alpha\}$$

확률 $DLCP_S$(Differential-Linear Connectivity Probability of S-box)는 다음과 같이 정의한다.

$$DLCP_S(\triangle, \alpha) = \frac{DLCT_S(\triangle, \alpha)}{2^n}$$

정의된 DLCT는 S-box에 대해서 특정 입력 차분 $\triangle$을 가지는 두 입력 값 x, $x \oplus \triangle$에 대해 각 출력 값에 α를 내적 한 값이 같을 경우의 수를 의미한다.

실제 S-box에 대한 DLCT를 만들어보면 흥미로운 결과를 관찰할 수 있다. Table 1은 PRESENT[9]

Table 1. The Part of DLCT of PRESENT S-box(row means input difference and column means output mask of S-box).

	$0x1$	$0x2$	$0x3$	$0x4$
$0x0$	16	16	16	16
$0x1$	**0**	8	8	8
$0x2$	8	8	4	4
$0x3$	8	4	8	8
$0x4$	8	8	4	4
$0x5$	8	12	8	8
$0x6$	8	4	12	8
$0x7$	8	8	8	8
$0x8$	**0**	4	12	8
$0x9$	**16**	8	8	4

에 사용된 4-비트 S-box에 대한 DLCT의 일부분이다. 행은 S-box의 입력 차분, 열은 출력 값에 대한 마스크 값을 의미한다. Table 1의 (1,1), (8,1) 그리고 (9,1) 값을 살펴보면, 각각 0과 16의 값을 가진다. 이는 마스크 값 0x1에 대해 입력 차분이 각각 0x8과 0x9일 경우 $S(x) \odot \alpha$과 $S(x \oplus \triangle) \odot \alpha$가 같을 확률이 완전히 한쪽으로 치우치고 있다는 것을 의미한다. 이러한 현상이 생기는 이유는 가정 2의 선형 근사의 두 입력 값이 독립적이라는 가정이 실제론 맞지 않기 때문이다. 따라서 DLCT를 사용하면 선형 근사의 적어도 한 라운드는 가정 2를 완화하여 계산할 수 있게 된다. 다시 말해서, 선형 근사의 전체 확률을 가정 2를 사용하지 않고 계산할 수는 없지만, 적어도 선형 근사의 처음 한 라운드는 DLCT를 사용함으로써 좀 더 정확한 값을 계산할 수 있다.

기존의 차분선형 특성의 확률 계산은 차분 특성의 출력 차분 Ω_T와 선형 근사의 입력 마스크 λ_P에 대해, $\Omega_T \odot \lambda_P = 0$을 만족하기 때문에 차분 연결 바이어스의 확률이 식 (3)과 같이 계산되지만 DLCT를 적용하면 선형 근사의 한 라운드 이후의 마스크 값이 같을 확률을 구할 수 있으므로 기존보다 한 라운드 이후의 차분 연결 바이어스를 다음과 같이 계산할 수 있다.

$$\begin{aligned} &\frac{1}{2} + p \approx p' \times DLCP_S(\triangle, \alpha) + \frac{1-p'}{2} \\ &= \frac{1}{2} + (DLCP_S(\triangle, \alpha) - \frac{1}{2})p' \end{aligned} \tag{4}$$

위의 식으로 선형 근사의 한 라운드 이후의 마스크 값에 대한 차분 연결 바이어스를 계산하게 되고, 첫 라운드 입력 마스크에 대해서는 $\Omega_T \odot \lambda_P = 0$이 성립하므로 전체 차분선형 특성의 확률을 계산할 때에는 선형 근사의 처음 한 라운드를 제외하고 계산한다. 첫 라운드를 제외한 선형 근사의 확률을 $1/2 + q'$이라고 하면 차분선형 특성의 확률은 다음과 같다.

$$\frac{1}{2} + 4(DLCP_S(\triangle, \alpha) - \frac{1}{2})p'(q')^2 \tag{5}$$

따라서 위의 식으로 선형 근사 한 라운드의 확률을 차분 연결 바이어스를 계산할 때 계산 함으로써 좀 더 정확하게 계산할 수 있다.

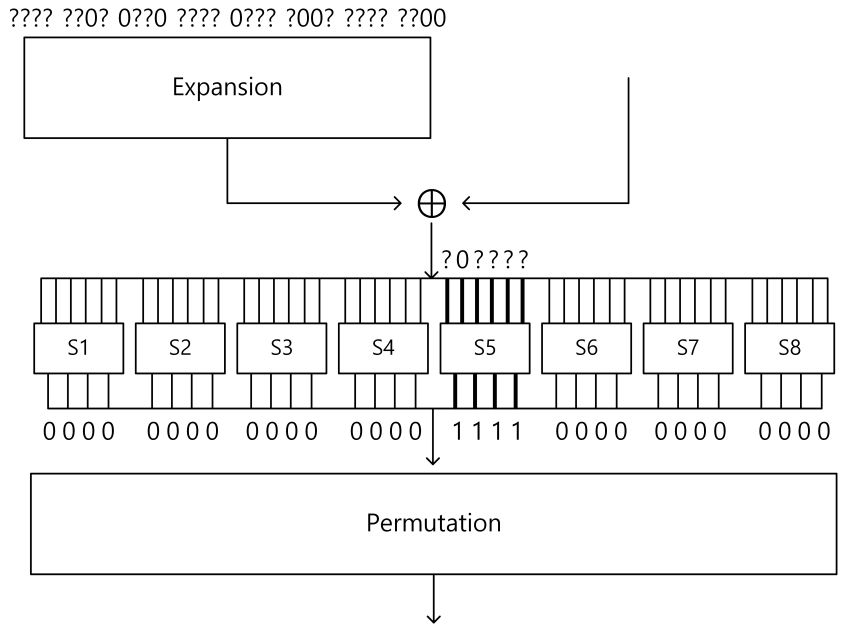

Fig. 3. The input difference and output mask for the 5-Round S-box 5 where the DLCT is applied after the 4-Round Differential Characteristic (the value '?0????' before S-box indicates the difference. 0 means zero difference, '?' means random value in $\{0,1\}$ and the value '1111' after the S-box means the bits to be masked).

선형 근사 입력 값의 종속성에 의해, 전체 차분선형 특성의 확률은 기존 결과보다 증가할 수도 있고 감소할 수도 있다. 그러나 대부분 DLCT의 입력 차분은 부정 차분 형태이므로 가능한 입력 차분의 분포를 측정하면 더 정확한 확률을 계산 수 있다. 부정 차분이 가지는 가능한 모든 입력 차분을 $\triangle_i$, 그 가짓수를 m, 각각의 확률을 $w_i(0 \le i \le m-1)$라고 할 때, 차분 연결 바이어스는 다음과 같이 계산한다.

$$\frac{1}{2}+p \approx p' \times \left(\sum_{i=0}^{m-1} w_i \times DLCP_S(\triangle_i, \alpha) \right) + \frac{1-p'}{2}$$
$$= \frac{1}{2} + \left(\sum_{i=0}^{m-1} w_i \times DLCP_S(\triangle_i, \alpha) - \frac{1}{2} \right) p'$$

$$(w_i = \Pr(x \oplus x' = \triangle_i | P_1 \oplus P_2 = \Omega_P)) \quad (6)$$

따라서 입력 차분의 분포를 적용한 차분선형 특성의 확률은 다음과 같이 계산할 수 있다.

$$\frac{1}{2} + 4\left(\sum_{i=0}^{m-1} w_i \times DLCP_S(\triangle_i, \alpha) - \frac{1}{2} \right) p'(q')^2 \quad (7)$$

일반적으로 각 차분의 확률 w_i의 값이 $1/m$로 균일하다는 가정을 사용하여 차분선형 특성의 확률을 계산할 수 있으며, 값이 정확할수록 차분선형 특성의 확률을 더 정확하게 계산할 수 있다.

IV. 적 용

본 장에서는 앞서 소개한 식 (6)과 (7)을 사용하여 실제 DES와 SERPENT에 대한 차분선형 특성의 확률을 다시 계산하고 더 정확한 확률을 얻은 결과를 제시한다. 실험은 2^{40}의 데이터를 사용하였으며 평문과 키의 비율을 조정하고, 임의의 난수 값으로 설정하여 진행되었다.

4.1 DES에 대한 적용

DES(Data Encryption Standard)는 64-비트 블록 크기와 56-비트 키 크기를 가지는 16-라운드 블록 암호이다[5]. 1994년 Langford는 DES의 8-라운드 변형에 대해 차분선형 분석을 제안하였고, Biham 등은 Langford의 방법을 확장시켜서 DES의 8-라운드 변형에 대해 개선된 차분선형 분석 결

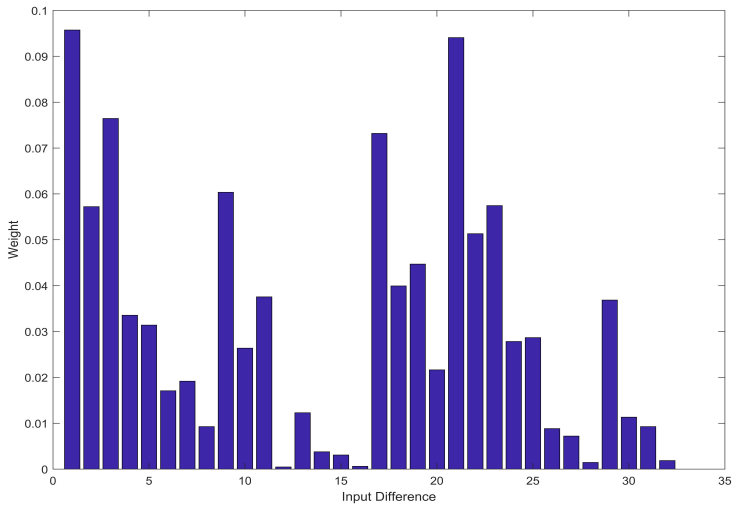

Fig. 4. Differential Distribution Graph of 32 input difference values in DES.

과를 제시하였다. 앞서 설명한 것처럼 Biham 등은 전체 차분선형 특성의 확률을 $1/2+2^{-5.91}$으로 계산했다.

DLCT를 적용하여 차분선형 특성의 확률을 다시 계산하기 위해선 부정 차분 특성의 출력 차분의 형태와 선형 근사의 입력 마스크 값이 S-box를 지난 이후의 마스크 값의 형태를 알아야 한다. 기존의 7-라운드 특성은 4-라운드 부정 차분 특성과 3-라운드 선형 근사로 구성되어 있는데 부정 차분 특성의 출력 차분과 선형 근사의 마스크 값이 만나는 부분의 모양은 Fig.4와 같다. 따라서 가능한 차분의 종류는 0x0~0xF 16가지와 0x20~0x2F 16가지, 총 32가지가 있으며 출력 마스크의 형태는 0xF이다. 가능한 32가지의 입력 차분과 출력 마스크 값 0xF에 대한 DLCT는 Table 2와 같다. Table 2는 전체 DLCT 중 필요한 부분만을 표시한 것이다. 이를 이용한 확률 계산은 다음과 같다.

Table 2. The Part of DLCT of the S-box 5 in DES. Input difference($\triangle_i$) have 32 values, output mask(α) is 0xF and c_i means $DLCT_S(\triangle_i,\alpha)$

$\triangle_i$	c_i	$\triangle_i$	c_i	$\triangle_i$	c_i	$\triangle_i$	c_i
$0x0$	64	$0x8$	48	$0x20$	40	$0x28$	40
$0x1$	44	$0x9$	44	$0x21$	40	$0x29$	40
$0x2$	40	$0xA$	40	$0x22$	52	$0x2A$	44
$0x3$	40	$0xB$	40	$0x23$	48	$0x2B$	48
$0x4$	48	$0xC$	48	$0x24$	40	$0x2C$	40
$0x5$	52	$0xD$	52	$0x25$	40	$0x2D$	40
$0x6$	40	$0xE$	40	$0x26$	52	$0x2E$	44
$0x7$	40	$0xF$	40	$0x27$	48	$0x2F$	48

$$\frac{1}{2}+p \approx \frac{1}{2}+\left(\sum_{0}^{31} w_i \times DLCP_S(\triangle_i, 0xF)-\frac{1}{2}\right)p'$$

각각의 차분 $\triangle_i$가 가지는 확률을 균일하다고 가정하면 $w_i=1/32$이고, S-box 5의 DLCT를 이용하여 평균적인 확률을 계산할 수 있다. 이때의 차분 연결 바이어스는 다음과 같다.

$$\frac{1}{2}+p \approx \frac{1}{2}+\left(0.695313-\frac{1}{2}\right)p'$$

따라서 전체 차분선형 특성의 확률은 식 (7)을 이용하여 계산할 수 있다. 3-라운드 선형 근사의 첫 라운드를 제외한 바이어스 q'은 $-20/64$이며 부정 차분 특성의 확률 p'은 $14/64$이므로 전체 확률은 다음과 같이 계산된다.

$$\frac{1}{2}+4\times(0.695313-\frac{1}{2})\times\frac{14}{64}\times\left(-\frac{20}{64}\right)^2$$
$$\approx \frac{1}{2}+2^{-5.91}$$

그러나 실제로 각 차분이 가지는 확률은 균일하지 않다. 따라서 더 정확한 확률을 계산하기 위해선 실제와 근사한 w_i값들을 알아야 한다. 본 논문에서는 각 w_i의 값의 평균을 실험적으로 측정하였으며 실험값의 분포는 Fig. 5와 같다. 측정한 w_i값들을 이용하여 다시 전체 차분선형 특성의 확률을 계산할 수 있고, 전체 차분선형 특성의 확률은 식 (7)을 이용하여 계산하면 다음과 같다.

$$\frac{1}{2}+4\times(0.708362448-\frac{1}{2})\times\frac{14}{64}\times\left(-\frac{20}{64}\right)^2 \approx \frac{1}{2}+2^{-5.81}$$

위의 결과처럼, 각 w_i의 값을 실제와 근사한 값으로 적용하면 전체 차분선형 특성의 확률을 더 정확하게 계산할 수 있다. 다시 계산된 차분선형 특성의 확률은 기존에 계산한 DES의 7-라운드 차분선형 특성의 확률보다 증가했음을 알 수 있다.

4.2 SERPENT에 대한 적용

SERPENT는 128-비트 블록 크기와 0~256-비트 키 크기를 가지는 32-라운드 SPN 구조 블록 암호이다[7]. 각 라운드는 키 혼합(key mixing), 대체(S-box layer), 선형 변환(linear transformation) 세 개의 단계로 구성되며 대체 단계는 4-비트 입력 및 출력을 가지는 8개의 S-box를 사용하며, 각 라운드에서 하나의 S-box가 32번 병렬적으로 사용된다.

본 논문에서는 차분 특성 및 선형 근사를 표현하기 위하여 [6]에서 Biham 등이 사용한 표기법을 동일하게 사용한다. 1/4의 확률로 이동하는 차분은 굵은 화살표로 표시를 하고 1/8의 확률로 이동하는 차분은 얇은 화살표로 표시한다. 표기법에 대한 예시는 Fig. 5와 같다. Biham 등이 SERPENT에 대해 구성한

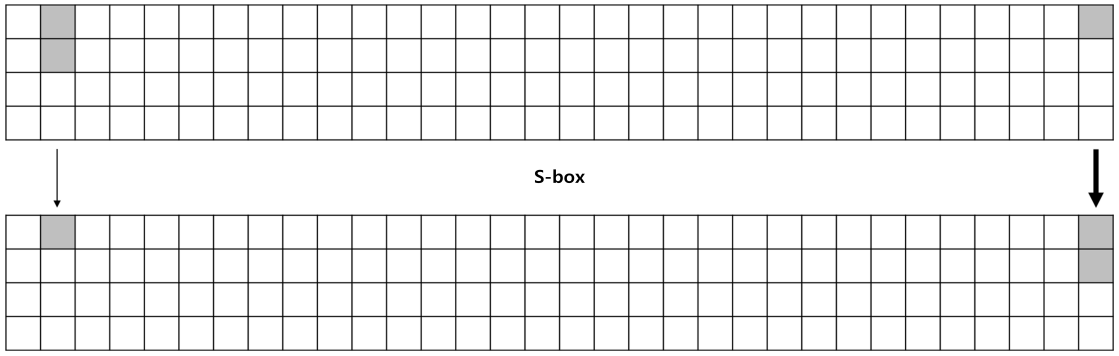

Fig. 5. Differential and Linear Representation Example

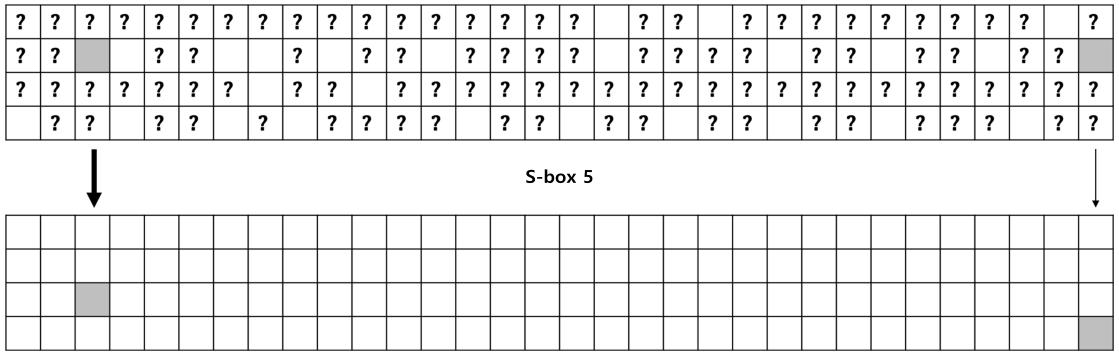

Fig. 6. The input difference and output mask (the state before S-box 5 is the difference and the state after the S-box 5 is the mask bits) for 4-Round S-box 5 to which DLCT is applied after 3-Round Differential Characteristic. '?' means random value in $\{0,1\}$ and a shaded portion denotes a mask bit).

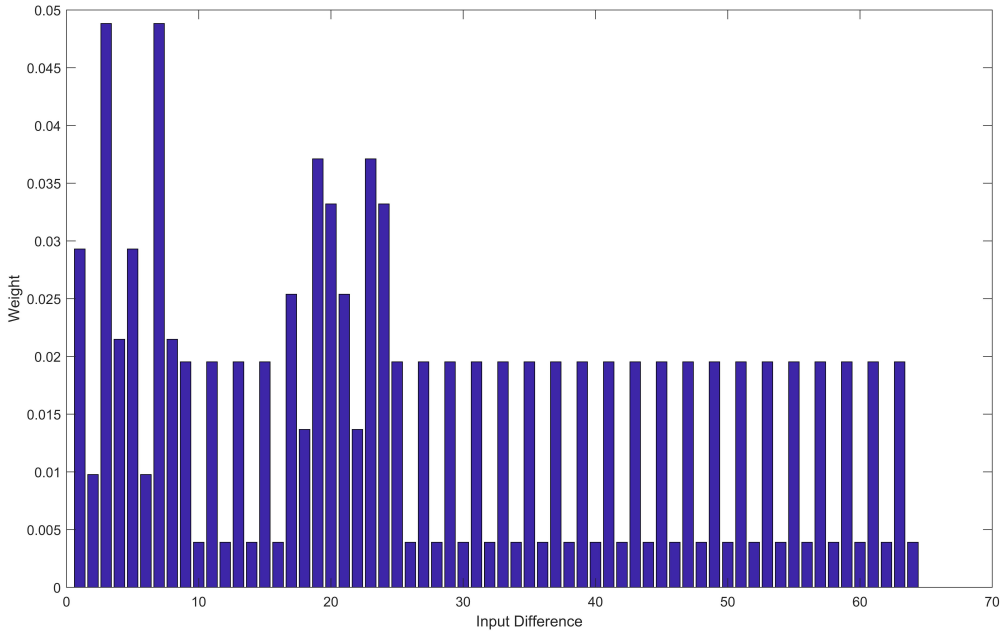

Fig. 7. Differential Distribution Graph of 64 input difference values in SERPENT.

차분선형 특성은 9-라운드 특성으로, 확률 $p' = 2^{-6}$인 3-라운드 부정 차분 특성과 바이어스 $q = 2^{-27}$인 6-라운드 선형 근사를 사용하여 구성하였다. 따라서 차분 연결 바이어스 p는 2^{-7}이고, 전체 차분선형 특성의 확률은 $1/2 + 4pq^2 = 1/2 + 2^{-59}$으로 계산된다.

앞에서 설명한 DES와 마찬가지로 SERPENT도 DLCT를 적용하여 차분선형 특성의 확률을 다시 계산할 수 있다. DLCT를 적용하기 위해선 먼저 SERPENT의 기존 차분선형 특성에서 부정 차분 특성의 출력 차분과 선형 근사의 첫 라운드 S-box 이후의 마스크 값을 알아야 한다.

SERPENT의 경우 4-라운드에 적용되는 2개의 S-box 5에 DLCT를 적용해야 하며, 각 S-box의 입력 차분과 출력 마스크의 모양은 Fig.6과 같다. 각 S-box는 8가지의 입력 차분을 가지기 때문에 두 S-box는 총 64가지의 입력 차분을 가지는 하나의 8-비트 S-box로 간주할 수 있다. 즉, 두 개의 S-box에 적용될 DLCT를 입력 차분 $\triangle_i = ??0???0?$로 하고 출력 마스크 $\alpha = 01001000$으로 하는 8-비트 S-box에 대한 하나의 DLCT로 정의할 수 있다. 설명한 8-비트 S-box의 DLCT는 Table 3에 제시되어 있으며, 우리가 필요한 일부분만 표시되어 있다.

DES와 마찬가지로 가능한 64가지의 입력 차분의 각 확률 w_i를 1/64로 균일하다고 가정하면 차분 연결 바이어스는 다음과 같다.

$$\frac{1}{2} + p \approx \frac{1}{2} + \left(0.501953125 - \frac{1}{2}\right)p'$$

$0.501953125 \approx 1/2 + 2^{-9}$이고, 차분 특성의 확률은 2^{-6}, 첫 라운드를 제외한 선형 근사의 바이어스는 2^{-23}이므로, 식 (7)을 이용하여 전체 차분선형 특성의 확률을 계산하면 다음과 같다.

$$\frac{1}{2} + 4 \times \left(0.501953125 - \frac{1}{2}\right) \times 2^{-6} \times \left(2^{-23}\right)^2$$
$$\approx \frac{1}{2} + 2^2 \times 2^{-15} \times 2^{-46} = \frac{1}{2} + 2^{-59}$$

DES와 마찬가지로 SERPENT도 64가지 차분의 확률을 실험적으로 측정하여 전체 차분선형 특성의 확률을 계산할 수 있다. 64가지 입력 차분에 대한 w_i 값의 분포는 Fig.7과 같다. 측정한 w_i값을 적용하여 다시 계산한 차분 연결 바이어스는 다음과 같다.

$$\frac{1}{2}+p \approx \frac{1}{2}+\left(0.504150293-\frac{1}{2}\right)p'$$

$0.504150293 \approx 1/2+2^{-7.9}$이므로, 다시 계산한 차분 연결 바이어스의 값을 이용하여 전체 차분선형 특성의 확률을 계산하면 다음과 같다.

$$\frac{1}{2}+4\times\left(0.504150293-\frac{1}{2}\right)\times 2^{-6}\times(2^{-23})^2$$
$$\approx \frac{1}{2}+2^2\times 2^{-13.9}\times 2^{-46}=\frac{1}{2}+2^{-57.9}$$

실제 값과 근사한 w_i를 사용하여 차분선형 특성의 확률을 다시 계산하면 기존에 계산된 확률보다 약 $2^{1.1} \approx 2.14$배 증가한 확률을 얻을 수 있다.

4.3 공격 복잡도 분석

차분선형 특성을 구성하여 실제 암호에 대한 공격을 진행할 때에는 $O(p^{-2}q^{-4})$의 데이터 복잡도가 필요하다. 따라서 기존에 계산된 차분선형 특성의 확률과 비교하여 본 논문에서 다시 계산한 확률이 더 좋을 경우, 공격 복잡도가 줄어든다. 본 논문은 기존에 사용했던 차분선형 특성을 바꾸지 않고 새로운 확률 계산 방법으로 다시 확률을 계산했기 때문에 기존에 사용했던 공격 알고리즘은 똑같이 적용되며, 향상된 확률 정도에 따라 공격 복잡도가 개선된다.

Table 3. The Part of DLCT of two S-boxes in SERPENT (entire size of DLCT is 256 by 256). Input difference($\triangle_i$) have 64 values, output mask is 0x48 and c_i means $DLCT_S(\triangle_i,\alpha)$.

$\triangle_i$	c_i	$\triangle_i$	c_i	$\triangle_i$	c_i	$\triangle_i$	c_i
$0x0$	256	$0x40$	64	$0x80$	64	$0xC0$	192
$0x1$	128	$0x41$	128	$0x81$	128	$0xC1$	128
$0x4$	128	$0x44$	128	$0x84$	128	$0cC4$	128
$0x5$	64	$0x45$	160	$0x85$	160	$0xC5$	96
$0x8$	64	$0x48$	160	$0x88$	160	$0xC8$	96
$0x9$	128	$0x49$	128	$0x89$	128	$0xC9$	128
$0xC$	128	$0x4C$	128	$0x8C$	128	$0xCC$	128
$0xD$	192	$0x4D$	96	$0x8D$	96	$0xCD$	160
$0x10$	128	$0x50$	128	$0x90$	128	$0xD0$	128
$0x11$	128	$0x51$	128	$0x91$	128	$0xD1$	128
$0x14$	128	$0x54$	128	$0x94$	128	$0xD4$	128
$0x15$	128	$0x55$	128	$0x95$	128	$0xD5$	128
$0x18$	128	$0x58$	128	$0x98$	128	$0xD8$	128
$0x19$	128	$0x59$	128	$0x99$	128	$0xD9$	128
$0x1C$	128	$0x5C$	128	$0x9C$	128	$0xDC$	128
$0x1D$	128	$0x5D$	128	$0x9D$	128	$0xDD$	128

DES의 경우, Biham 등이 7-라운드 차분선형 특성의 바이어스를 $0.0167 \approx 2^{-5.91}$으로 계산하였고, DES의 7-라운드 변형에 대해서 약 $2^{11.81}$의 데이터 복잡도를 사용하여 약 84.13%의 성공률로 구별 공격(distinguishing attack)을 진행하였다. 본 논문에서는 7-라운드 차분선형 특성의 확률을 DLCT를 적용하여 $0.0178 \simeq 2^{-5.81}$으로 다시 계산하였으며, 실제 구별 공격 시 데이터 복잡도가 약 $2^{0.2} \approx 1.15$배 줄어든다. 이러한 결과는 동일한 차분선형 특성을 사용한 키 복구 공격(key recovery attack)에도 똑같이 적용된다. 7-라운드 차분선형 특성을 사용한 공격에서 분석 결과는 기존과 큰 차이가 없지만, 확률이 작아지는 더 긴 라운드의 차분선형 특성에 적용할 경우 차이는 더 커질 수 있다.

SERPENT는 [6]에서 Biham 등이 9-라운드 차분선형 특성을 구성하였고 바이어스는 약 2^{-59}로 계산하였다. 구성한 특성으로 SERPENT의 11-라운드 변형에 대한 키 복구 공격을 진행하였으며, 약 $2^{125.3}$의 평문을 사용하여 72.1%의 성공률로 공격했다. 본 논문에서 SERPENT에 대한 9-라운드 차분선형 특성의 바이어스는 약 $2^{-57.9}$로 계산되었으며 이 결과는 기존의 분석 결과와 비교하여 데이터 복잡도가 약 $2^{2.2} \approx 4.6$배 줄어든다.

V. 결 론

본 논문에서는 논문에서 제안하는 DLCT(Differential-Linear Connectivity Table)를 사용하여 두 선형 근사의 입력 값이 독립적이라는 가정 2를 완화하여 차분선형 특성의 확률을 계산하는 방법을 제안하고 실제로 DES와 SERPENT에 적용하여 다시 계산한 결과, 기존에 제시된 결과보다 개선된 확률을 얻을 수 있었다.

적용하는 암호에 따라서 확률의 개선 정도는 다를 수 있는데, SERPENT의 경우 바이어스가 2배 이상 개선되었으며 이는 데이터 복잡도를 4배 이상 감소시키게 된다. 이러한 결과를 통해 기존에 차분선형 특성의 확률 계산이 독립성 가정으로 인해 다소 정확하지 못하게 계산되어 온 것을 확인할 수 있다. 본 논문에서 제안하는 DLCT를 적용한 계산 방법을 사용하면

이러한 문제를 조금이나마 해결할 수 있다.

나아가 차분선형 특성을 구성하는 단계에서 DLCT를 적용할 수도 있다. 본 논문에서 설명한 것처럼 DLCT를 적용한 확률 계산에서 선형 근사의 첫 라운드 확률은 DLCT를 통해 계산된 차분-연결 바이어스에 포함되므로 전체 확률 계산 시에 이를 제외하고 확률을 계산하게 된다. 따라서 차분선형 특성을 MILP와 SAT-solver와 같은 자동화 탐색 도구로 탐색할 경우, 탐색한 차분 경로와 선형 경로를 DLCT로 연결하여 한 라운드 더 긴 차분선형 경로를 구성할 수 있게 된다. 선형 경로의 두 입력값의 종속성이 크다면, 기존보다 더 좋은 확률을 가지는 경로가 구성될 가능성이 있다. 따라서 자동화 탐색 도구를 이용한 차분선형 경로 탐색은 좀 더 정확한 안전성 분석을 위한 연구 주제가 될 수 있다.

References

[1] Biham, Eli, and Adi Shamir. "Differential cryptanalysis of DES-like cryptosystems." Journal of CRYPTOLOGY 4(1), pp. 3-72, 1991.

[2] Matsui, Mitsuru. "Linear cryptanalysis method for DES cipher." Workshop on the Theory and Application of of Cryptographic Techniques. Springer, Berlin, Heidelberg, pp. 386-397, 1993.

[3] Langford, Susan K., and Martin E. Hellman. "Differential-linear cryptanalysis." Annual International Cryptology Conference. Springer, Berlin, Heidelberg, pp. 17-25, 1994.

[4] Biham, Eli, Orr Dunkelman, and Nathan Keller. "Enhancing differential-linear cryptanalysis." International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin, Heidelberg, pp. 254-266, 2002.

[5] FIPS, PUB. "46-3: Data encryption standard (des)." National Institute of Standards and Technology 25(10) 1-22, 1999

[6] Biham, Eli, Orr Dunkelman, and Nathan Keller. "Differential-linear cryptanalysis of Serpent." International Workshop on Fast Software Encryption. Springer, Berlin, Heidelberg, pp. 9-21, 2003.

[7] Biham, Eli, Ross Anderson, and Lars Knudsen. "Serpent: A new block cipher proposal." International Workshop on Fast Software Encryption. Springer, Berlin, Heidelberg, pp. 222-238, 1998.

[8] Lu, Jiqiang. "A methodology for differential-linear cryptanalysis and its applications." Designs, Codes and Cryptography 77(1), pp. 11-48, 2015

[9] Wagner, David. "The boomerang attack." International Workshop on Fast Software Encryption. Springer, Berlin, Heidelberg, pp. 156-170, 1999.

[10] Kelsey, John, Tadayoshi Kohno, and Bruce Schneier. "Amplified boomerang attacks against reduced-round MARS and Serpent." International Workshop on Fast Software Encryption. Springer, Berlin, Heidelberg, pp. 75-93, 2000.

[11] Biham, Eli, Orr Dunkelman, and Nathan Keller. "The rectangle attack −rectangling the Serpent." International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Berlin, Heidelberg, pp. 340-357, 2001.

[12] Biryukov, Alex, and Dmitry Khovratovich. "Related-key cryptanalysis of the full AES-192 and AES-256." International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin, Heidelberg, pp. 1-18, 2009.

[13] Dunkelman, Orr, Nathan Keller, and Adi Shamir. "A practical-time re-

lated-key attack on the KASUMI cryptosystem used in GSM and 3G telephony." Annual Cryptology Conference. Springer, Berlin, Heidelberg, pp. 393-410, 2010.

〔14〕 Murphy, Sean. "The return of the cryptographic boomerang." IEEE Transactions on Information Theory 57(4), pp. 2517-2521, 2011.

〔15〕 Cid, Carlos, et al. "Boomerang Connectivity Table: A New Cryptanalysis Tool." Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Cham, pp. 683-714, 2018.

〔16〕 Lai, Xuejia, James L. Massey, and Sean Murphy. "Markov ciphers and differential cryptanalysis." Workshop on the Theory and Application of of Cryptographic Techniques. Springer, Berlin, Heidelberg, pp.17-38, 1991.

〈저 자 소 개〉

김 현 우 (Hyunwoo Kim) 학생회원
2017년 2월: 숭실대학교 수학과 졸업
2017년 3월~현재: 고려대학교 정보보호대학원 석사과정
〈관심분야〉 암호 알고리즘 설계 및 분석, 대칭키 암호

김 성 겸 (Seonggyeom Kim) 학생회원
2016년 8월: 한양대학교 수학과 졸업
2018년 8월: 고려대학교 정보보호대학원 석사
〈관심분야〉 암호 알고리즘 설계 및 분석, 대칭키 암호, 난수발생기

홍 득 조(Deukjo Hong) 종신회원
1999년 8월: 고려대학교 수학과 학사
2001년 8월: 고려대학교 수학과 석사
2006년 2월: 고려대학교 정보보호대학원 박사
2006년 3월~2007년 12월: 고려대학교 정보보호기술연구센터 연구교수
2007년 12월~2015년 8월: 국가보안기술연구소 선임연구원
2015년 9월~현재: 전북대학교 IT정보공학과 조교수
〈관심분야〉 암호 알고리즘 설계 및 분석

성 재 철 (Jaechul Sung) 종신회원
1997년 8월 : 고려대학교 수학과 학사
1999년 8월 : 고려대학교 수학과 석사
2002년 8월 : 고려대학교 수학과 박사
2002년 8월~2004년 1월 : 한국정보보호진흥원 선임연구원
2004년 2월~현재 : 서울시립대학교 수학과 전임강사, 조교수, 부교수, 교수
〈관심분야〉 암호 알고리즘 설계 및 분석

홍 석 희 (Seokhie Hong) 종신회원
1995년: 고려대학교 수학과 학사
1997년: 고려대학교 수학과 석사
2001년: 고려대학교 수학과 박사
1999년 8월~2004년 2월: ㈜시큐리티 테크놀로지 선임연구원
2003년 3월~2004년 2월: 고려대학교 정보보호기술연구센터 선임연구원
2004년 4월~2005년 2월: K.U. Leuven ESAT/SCD-COSIC 박사후 연구원
2005년 3월~2013년 8월: 고려대학교 정보보호대학원 부교수
2013년 9월~현재: 고려대학교 정보보호대학원 정교수
〈관심분야〉 대칭키 및 공개키 암호 알고리즘, 부채널 공격 및 대응기법, 디지털 포렌식