情報保護學會論文誌
第14券 第3號, 2004. 6

연관키 차분 특성을 이용한 32-라운드 GOST 공격

이 태 건[a)†] 고 영 대[a)] 홍 석 희[a)] 이 상 진[b)‡]
고려대학교 정보보호기술연구센터[a)]

Related Key Differential Attacks on 32-Round GOST

Tae-keon Lee,[a)†] Young-dai Ko,[a)] Seok-hie Hong,[a)] Sang-jin Lee[b)‡]
Center for Information of Security of Technologies, Korea University[a)]

요 약

이 논문에서는 블록 암호 알고리즘 GOST의 연관키 차분 공격에 대하여 설명한다. GOST는 키 스케줄이 단순하여 연관키 차분 특성식이 발생하는데 이를 이용하여, 우선 랜덤 oracle로부터 GOST 블록 암호 알고리즘을 확률 $1\text{-}2^{-64}$로 써 구별하는 방법에 대하여 언급하고, 그 다음엔 각각 24-라운드와 6-라운드로 이루어진 두 개의 연관키 차분 특성식을 연접하여 30-라운드 차분 특성식을 꾸민 후 31-라운드 GOST의 마지막 라운드 키 32비트를 복구하는 공격방법에 대하여 설명한다. 또한, 전체 32-라운드 GOST의 마지막 32 라운드의 부분키 12 비트를 91.7%의 성공확률로 2^{35}의 선택평문과 2^{36}의 암호화 시간을 이용하여 복구할 수 있는 알고리즘에 대해서 서술한다.

ABSTRACT

In this paper, we present a related key differential attack on Full-round GOST. Firstly, we present a distinguishing attack on full rounds of GOST, which can distinguish it from random oracle with probability $1\text{-}2^{-64}$ using a related key differential characteristic. We will also show that H. Seki et al.'s idea can be applied to attack on 31 rounds of GOST combining our related key differential characteristic. Lastly, we propose a related key differential attack on full rounds of GOST. In this attack, we can recover 12 bits of the master key with 2^{35} chosen plaintexts and 2^{36} encryption times for the 91.7% expectation of success rate.

Keywords : *Related bey differential attack, Distinguishing attack, GOST, Diffential Characteristic*

Ⅰ. 서 론

GOST[3]는 1989년도에 구 소련의 표준 블록 암호로 제안된 전체 32-라운드로 이루어진 알고리즘으로 러시아의 DES라고도 불리우며 'GOST'란 단어의 어원은 "Gosudarstvennyi Standard," 또는 "Government Standard"에서 유래되었다. 알고리즘의 안전성을 결정짓는 S-box의 설계원리가 공개되지 않는 등 여러 가지 형태에서 DES 알고리즘과 종종 비교되기도 하지만, DES[6] 알고리즘과는 달리 아직까지 많은 분석이 이루어지지는 않았다. 그러던 중 Crypto'96에서 J. Kelsey 등이 GOST의 매우 단순한 키 스케줄을 이용한 연관키 차분 공격[4]에 대하여 제안하였다. J. Kelsey 등은 주로 이론적인 면에서 3 라운드 반복 차분 특성식을 꾸민 후 24-라운드 GOST에 대해서 공격이 가능함을 언급하였다. 그러나, GOST의 라운드 함수는 키 부분이 덧셈 연산으로 이루어지기 때문에 차분 확률이 알고리즘의

접수일 : 2004년 3월 3일 ; 채택일 : 2004년 5월 11일
* 본 연구는 고려대학교 특별연구비로 수행하였습니다.
† 주저자, imml97@cist.korea.ac.kr
‡ 교신저자, sangjin@korea.ac.kr

입력 및 출력 차분 값에만 의존하는 것이 아니라 사용되는 부분키 값에도 영향을 받는 성질이 있다. 이러한 성질은 GOST에 대한 차분 특성을 이용한 분석을 어렵게 하는 요인이 되었다.

H. Seki[8]등은 이러한 키에 의존하는 차분 확률의 특성을 줄이기 위해서 차분 특성 집합 (a set of differential characteristics)이라는 개념을 사용하였다. 이것은 일종의 부정 차분 특성[5]으로 S-box의 입·출력 차분 값을 특정한 형태의 집합 모양으로 분류한 후 이를 이용하여 S-box의 모든 키에 대한 확률 값을 계산하여 그 평균값을 차분 확률로 사용한 것이다. H. Seki등은 이러한 차분 특성 집합으로 2-라운드 반복 차분 특성식을 꾸민 후 13-라운드 GOST에 대한 차분 공격을 제시하였고, 여기에 J. Kelsey[4]등이 이론적으로 언급한 연관키 차분 공격을 실제 적용하여 21-라운드 GOST에 대한 연관키 차분 공격을 제안하였다.

이 논문에서는 보다 다양한 방법으로 GOST에 대한 연관키 차분 공격을 설명할 것이다. 첫째로, 연관키 차분 특성을 이용하여 블록 암호 알고리즘인 GOST를 랜덤 oracle로부터 $1\text{-}2^{-64}$의 확률로 구별할 수 있는 방법 (distinguishing attack)에 대하여 언급한다. 둘째로, 우리가 꾸민 24-라운드 연관키 차분 특성식과 H. Seki등이 사용한 차분 특성 집합과 그 때 사용한 차분 확률을 이용하여 새로이 꾸민 6-라운드 연관키 차분 특성식을 연접한 후 1-R 연관키 공격[1]을 적용한 31-라운드 GOST에 대한 연관키 차분 공격이 가능함을 보인다. 마지막으로 이 논문에서 처음으로 보이는 전체 32-라운드 GOST에 대한 연관키 차분 공격에 대하여 제시할 것이다. 91.7%의 성공 확률을 위하여 이 공격에는 2^{35}의 선택 평문과 2^{36}의 암호화 과정이 요구된다.

이 논문은 다음과 같은 순서로 이루어져 있다. 2, 3장에서는 GOST 알고리즘에 대한 소개와 현재까지 진행된 GOST에 대한 관련 작업들에 대해서 간략하게 설명하고 4장에서는 GOST에 대한 다양한 연관키 차분 공격에 대하여 설명하고 5장에서 본 논문을 마무리한다.

II. GOST 알고리즘 소개

이 장에서는 블록 암호 알고리즘인 GOST에 대해서 간략하게 설명하도록 하겠다. GOST는 64-비트 블록과 256-비트 비밀키를 사용하는 전체 32-라운드 Feistel 구조로 이루어진 알고리즘이다. GOST는 그림 1.에서 보여지는 것처럼 매우 단순한 연산으로 이루어진 라운드 함수 F를 사용한다. F 함수는 라운드 함수의 입력값 32비트가 법 (modular) 2^{32} 위에서의 키 덧셈 연산 (⊞) 이후에 전체 8개로 이루어진 4×4 S-box들을 통과한 후 왼쪽으로 11-비트 순환 이동 (⟨⟨⟨)하는 과정의 순서로 구성되어 있다.

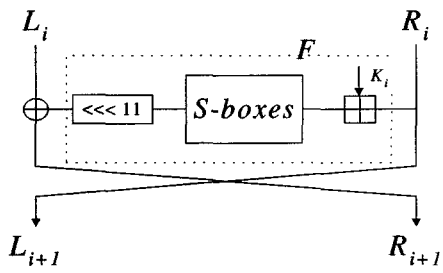


그림 1. i-번째 라운드 함수

그림 1.에서, $1 \le i \le 32$인 i에 대해서 L_i와 R_i ($I=(L_i, R_i)$)는 i-번째 라운드 함수의 각각 왼쪽과 오른쪽 32-비트 입력 값을 각각 나타내고 K_i는 i-번째 라운드의 32-비트 부분키를 가리킨다. 서론에서 언급한 것처럼 매 라운드에서 사용한 8 개의 S-box에 대한 구체적인 내용에 대해서는 나타나 있지 않다. 여기서는 Central Bank of the Russian Federation[7]에서 사용한 S-box에 대하여 다룰 것이고, 다음은 각각의 S-box에 대한 값을 C-코드를 이용하여 표현한 것이다.

$$
\begin{aligned}
S_8 &= \{1,15,13,0,5,7,10,4,9,2,3,14,6,11,8,12\}\\
S_7 &= \{13,11,4,1,3,15,5,9,0,10,14,7,6,8,2,12\}\\
S_6 &= \{4,11,10,0,7,2,1,13,3,6,8,5,9,12,15,14\}\\
S_5 &= \{6,12,7,1,5,15,13,8,4,10,9,14,0,3,11,2\}\\
S_4 &= \{7,13,10,1,0,8,9,15,14,4,6,12,11,2,5,3\}\\
S_3 &= \{5,8,1,13,10,3,4,2,14,15,12,7,6,0,9,11\}\\
S_2 &= \{14,11,4,12,6,13,15,10,2,3,8,1,0,7,5,9\}\\
S_1 &= \{4,10,9,2,13,8,0,14,6,11,1,12,7,15,5,3\}
\end{aligned}
$$

GOST의 키 스케줄은 매우 단순한 구조로 이루어져 있다. 256-비트 비밀키 K는 32-비트 8개의 블록

$$K=(K_1, K_2, K_3, K_4, K_5, K_6, K_7, K_8)$$

으로 쪼개진 후 다음과 같은 순서로 매 라운드에서 한 개의 블록씩 사용된다. 1-라운드부터 24-라운드

까지는 $K_1, \cdots, K_8$의 순서로 세 번 반복해서 사용되고, 25-라운드부터 마지막 32-라운드 까지는 $K_8, \cdots, K_1$의 역순서로 사용된다. 다음은 GOST의 키 스케줄을 표로 나타낸 것이다.

표 1. GOST의 키 스케줄

라운드	1 . . . 8	9 . . . 16	17 . . . 24	25 . . . 32
키	$K_1 \cdots K_8$	$K_1 \cdots K_8$	$K_1 \cdots K_8$	$K_8 \cdots K_1$

Ⅲ. GOST에 대한 현재까지의 분석 결과

이 장에서는 현재까지 나타난 GOST 알고리즘에 대한 연관키 차분 공격에 대하여 살펴 보고, 특히 [8]에서 사용한 차분 특성 집합이 우리의 31-라운드 연관키 차분 공격에 어떻게 사용되는지에 대해서 다룰 것이다.

J. Kelsey[4]등은 2장에서 설명한 GOST의 단순한 키 스케줄을 이용한 연관키 차분 공격에 대하여 제시 하였다. J. Kelsey 등은 부분키 K_1이 첫번째 라운드에서 사용된 이후에 9-라운드에서 다시 사용되는 성질을 이용하였다. 비밀키 256-비트에서 첫 번째 블록에서만 0이 아닌 차분 ΔK_1을 갖는 연관키를 고려하면, 이 차분 값은 선택 평문의 차분을 이용하여 1 라운드 함수의 출력 차분 값을 0으로 만들 수 있고, 이것은 공격자로 하여금 처음 8-라운드를 그냥 통과할 수 있게 한다. 즉, 키 차분이 발생하는 9 라운드까지 차분값 (0, 0)을 유지할 수 있게 된다. Kelsey 등은 이러한 생각으로 9 라운드부터 GOST의 S-box를 고려한 3-라운드 반복 차분 특성을 이용하여 20-라운드 특성식을 꾸민 후 GOST에 대한 4R 연관키 공격이 이론적으로 가능함을 보였다. 그러나, J. Kelsey 등은 이것은 단지 S-box의 차분 분포가 매우 나쁜 환경에서만 가능하고 실제로는 12-라운드 특성식을 사용한 24-라운드로 GOST에 대한 연관키 공격을 이론적으로만 보였다.

GOST에 대한 차분 특성을 이용한 공격은 SAC 2000에서 H. Seki[8]등에 의하여 구체화 되었다. H. Seki 등은 GOST S-box의 차분 확률이 입·출력 차분 뿐만 아니라 키 값에 많이 의존한다는 성질을 파악하고, 차분 확률에 키가 관여하는 부분을 줄이기 위하여 부정 차분 특성 (truncated differential characteristics)의 일종인 차분 특성 집합(a set of differential characteristics)의 개념을 이용하였다. H. Seki등은 두 개의 0이 아닌 차분 집합 Δ와 ∇를 사용하였는데, Δ는 S-box의 입력 차분으로 4-비트 중에서 최상위 비트가 0인 차분 값을 나타내는 집합이고, ∇는 그에 대응하는 S-box의 출력 차분으로 4-비트 중에서 최하위 비트가 0인 차분 값을 갖는 집합을 나타낸다. 또한 그들은 각각의 S-box에 대하여 다음의 평균 확률을 계산하였다.

표 2. S-box에 대한 차분 확률 분포

p_{S_1}	p_{S_2}	p_{S_3}	p_{S_4}	p_{S_5}	p_{S_6}	p_{S_7}	p_{S_8}
0.43	0.38	0.37	0.37	0.37	0.35	0.47	0.45

$$p_{S_i} : \mathrm{Prob}\left\{\Delta \xrightarrow{S_i} \nabla\right\}$$

각각의 확률값들은 0.30에서부터 0.75까지 S-box의 번호와 키 값들에 따라 다양하게 나타난다 (여기서 p_{S_i}는 모든 키 값에 대한 S_i-box의 평균값을 나타낸다). 여기서 사용한 확률 분포들은 4장에서 설명할 우리의 31-라운드 연관키 차분 공격에도 사용되므로 주목할 필요가 있겠다.

이러한 차분 특성 집합을 이용하여 H. Seki등은 13-라운드 GOST에 대한 차분 공격을 언급하였다. 또한, 이 차분 특성 집합을 J. Kelsey[4]등이 제시한 GOST에 대한 연관키 차분 공격에 적용하여 키 차분 ΔK_1을 e_{31}로 구체화 시킨 후 21-라운드 GOST에 대한 연관키 공격을 제안하였다. 앞으로 e_i는 i-번째 비트에서만 값이 1이고 나머지는 0인 32-비트 이진 수열을 가리킨다. 여기서 i는 오른쪽부터 0, 1, ···, 31의 인덱스를 갖는다.

Ⅳ. GOST에 대한 연관키 차분 공격

지금부터, 앞에서 언급한 GOST의 취약한 키 스케줄을 이용하여 다양한 방법의 연관키 차분 공격들에 대하여 소개할 것이다. 우선 연관키 특성을 이용하여 확률 $1\text{-}2^{-64}$로써 1-라운드 반복 차분 특성을 사용한 32-라운드 특성식을 꾸민 후, 전체 32-라운드 GOST 블록 암호 알고리즘을 랜덤 oracle로부터 구별하는 방법에 대하여 설명한다. 그리고, 2라운드

반복 차분 특성을 사용한 24라운드 특성식과 3절에서 언급한 차분 특성 집합을 이용한 6-라운드 특성식을 연접하여 30-라운드 특성식을 꾸민 후, 31-라운드 GOST의 마지막 라운드의 부분키 32-bit를 복구하는 공격에 대해서 설명한다. 그리고 마지막으로 본 논문에서 처음으로 언급하고 있는 전체 32-라운드 GOST에 대한 연관키 공격에 대해서 소개한다.

4.1. 랜덤 oracle과의 구별 방법 (Distinguishing attack)

우리는 단지 2쌍의 선택평문 P, P'를 연관키 K, K'으로 암호화하는 과정을 통해서 확률 $1\text{-}2^{-64}$로써 GOST 블록 암호를 랜덤 oracle과 구별할 수 있다. 공격을 위하여 우선, 다음과 같은 랜덤 oracle O와 O′을 고려한다. O는 평문 $P=(P_L,P_R)$에 대하여 키 $K=(K_1,\cdots,K_8)$를 이용하여 암호문 $C=(C_L,C_R)$를 출력하고, O′은 평문 $P'=(P_L\oplus e_{31},P_R\oplus e_{31})$에 대하여, 키 $K'=(K_1\oplus e_{31},K_2\oplus e_{31},\cdots,K_8\oplus e_{31},)$을 이용하여 암호문 $C'=(C'_L,C'_R)$를 출력하는 랜덤한 oracle이다. 공격자는 키 $K=(K_1,\cdots,K_8)$를 알 수는 없지만, 두 개의 키에 대한 다음과 같은 연관성 $K\oplus K'=(e_{31},e_{31},\cdots,e_{31})$을 알고 있다고 가정한다.

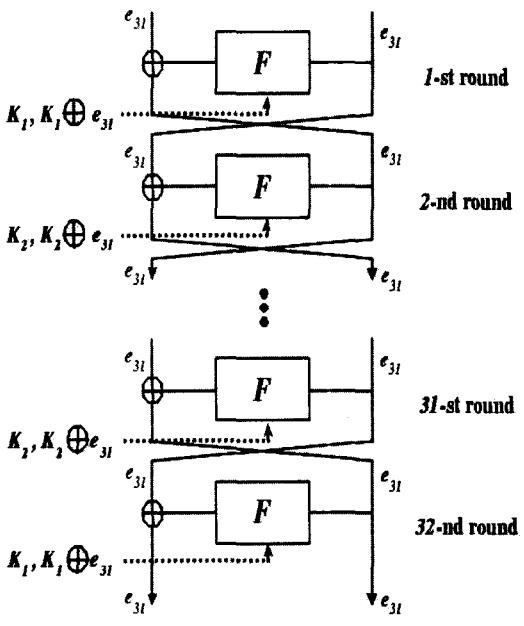


그림 2. 32-라운드 차분 특성식

GOST에 대한 연관키 차분 공격을 고려하면, 키 스케줄에 의하여 모든 라운드에서 사용되는 부분키의 차분값은 e_{31}과 같고 선택한 평문의 입력 차분 또한 $(e_{31},\ e_{31})$이므로, 이러한 1-라운드 반복 차분 특성식을 꾸밀수 있고 (키 덧셈 과정 이후 S_8를 포함한 모든 S-box의 입력 차분이 0이 된다. 즉, 라운드 함수의 입력 차분이 0이므로 출력 차분 또한 0이 되기 때문이다). 그림 2. 와 같은 32 라운드 차분 특성식을 확률 1로써 꾸밀 수 있다. 그러면 암호문의 차분 값이 $(e_{31},\ e_{31})$를 만족하는 지의 여부에 따라서 확률 $1\text{-}2^{-64}$로써 블록 암호 GOST를 랜덤 oracle과 구별할 수 있다. 앞에서 언급했듯이, 우리가 설명한 연관키 차분 특성을 이용한 구별 공격에는 단지 2쌍의 선택평문과 연관키들에 대한 암호화 과정만이 필요할 뿐이다.

4.2. 31-라운드 GOST에 대한 연관키 차분 공격

이 절에서 설명하는 공격은 2-라운드 반복 차분 특성을 이용한 24-라운드 차분 특성식과 3장에서 언급한 H. Seki 등의 차분 특성 집합을 이용한 6-라운드 차분 특성식을 연접하여 만든 전체 30-라운드 연관키 차분 특성식을 사용할 것이다. 우선 공격을 위하여 다음과 같은 연관키를 고려한다.

$$K=(K_1,K_2,K_3,K_4,K_5,K_6,K_7,K_8)$$

$$K'=(K_1\oplus e_{31},K_2,K_3\oplus e_{31},K_4,K_5\oplus e_{31},K_6,K_7\oplus e_{31},K_8)$$

선택평문 쌍 $P=(P_L,P_R)$와 $P'=(P_L,P_R\oplus e_{31})$를 각각 K와 K'에 대응하여 사용하면 확률 1로써 그림 3.과 같은 24-라운드 차분 특성식을 꾸밀 수 있다.

그러면 24-라운드 이후의 출력 차분 $(0,e_{31})$을 확률 1로써 얻을 수 있다. 그리고 24-라운드 이후의 출력 차분, 다시 말하면 25-라운드의 입력 차분 값인 $(0,e_{31})$과 H. Seki의 차분 특성 집합을 이용하여 그림 4와 같은 또 다른 6-라운드 차분 특성식을

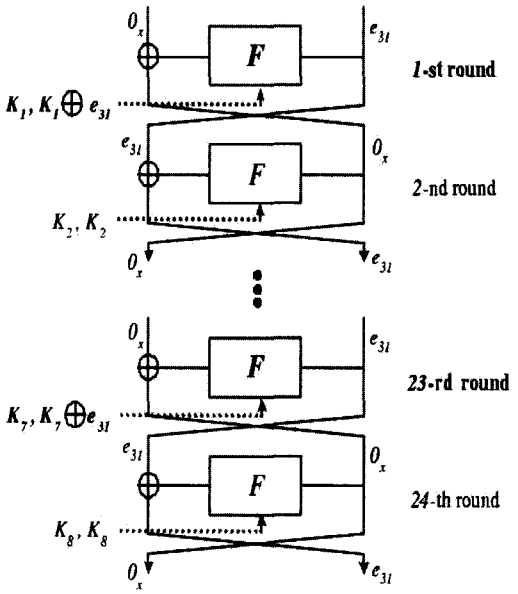


그림 3. 24-라운드 차분 특성식 (1~24 라운드)

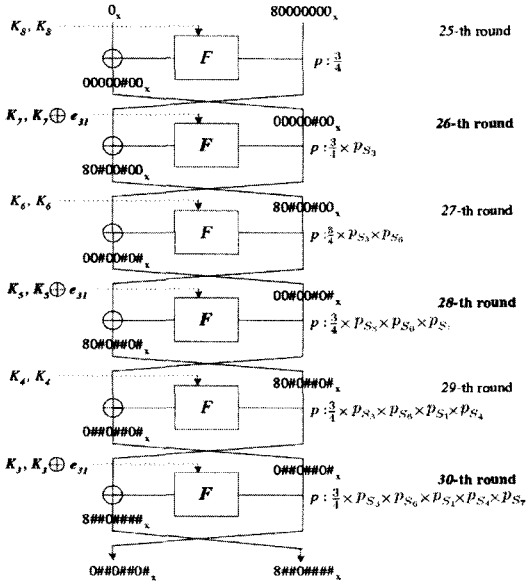


그림 4. 6-라운드 차분 특성식 (25~30 라운드)

꾸밀 수 있다. 그림 4에서 키 차분이 사용되는 라운드 함수는 키 스케줄에 의하여 26, 28, 30번째 라운드의 함수가 되고 #은 차분 집합 $\Delta=\{0abc\}$ (a, b, c$\in\{0, 1\}$)의 원소를 나타낸다. 25라운드의 F함수 입력값이 80000000x이면 출력값이 00000#00x가 될 확률은 $\frac{3}{4}$이다. 이 확률은 F함수의 특성상 $Prob\left\{1000 \xrightarrow{S_8} \nabla=\{abc0\}\right\}$만을 계산하면 되는데 이는 시뮬레이션과 [8]에 의해 쉽게 알 수 있다.

표 2.를 이용하여 6-라운드 차분 특성식의 확률을 계산하면 $2^{-23.33}$이고, 따라서 위의 두 개의 차분 특성식을 연접한 전체 30-라운드 차분 특성식의 확률 또한 $2^{-23.33}$이 된다. 그러므로 31 라운드의 부분키 32-비트를 찾기 위하여 전형적인 1-R공격[1]을 고려하면 올바르지 않은 평문 쌍들 (wrong pairs)은 2^{-14}의 확률로 걸러진다. 따라서, 이 차분 특성식의 S/N[1]는 대략 $2^{39.67}$ $(=\frac{2^{-23.33}}{2^{-17}\cdot 2^{-14}/2^{32}})$의 높은 값을 갖는다. 그래서 약 2^{26}의 선택 평문들을 고려하면, 이 중에는 약 5개의 올바른 쌍이 존재한다고 할 수 있고, 2^{26}의 선택 평문이 여과 과정을 거친 후에는 2^{26}의 올바른 쌍 후보들이 남게 되고, 이 후보들 중 올바르지 않은 쌍 (wrong pairs)들이 올바른 키 (right key)의 카운트 값을 증가시킬 확률은 차분 특성식에 의하여 2^{-17}이므로 대략 5번 이상 카운트 되는 키 값을 올바른 키 (right key)로 판단할 수 있다.

따라서, [9]에 의하여 우리는 성공확률 97.9%로 $(2^{32}\times 2^{26}\times 2^{-14}\times\frac{1}{31})\approx 2^{39}$의 암호화 시간과 2^{26}의 선택 평문을 이용하여 31-라운드 GOST의 마지막 라운드 키 32비트를 복구할 수 있다.

4.3. 전체 32-라운드 GOST에 대한 연관키 차분 공격

지금부터 본 논문의 핵심인 전체 32-라운드 GOST의 마지막 라운드의 부분키 12-비트를 높은 성공확률을 가지고 복구하는 알고리즘을 제안한다. 앞으로 설명할, 이 알고리즘에서 사용한 연관키 차분 특성식은 단지 한 가지 예일 뿐이다. 미리 살펴보면, 평문과 연관키에 관련된 차분을 여기서 사용한 e_{30}이 아닌 e_i로 $(1\le i\le 29)$로 변화시킴으로써, 우리가 사용한 방법과 유사하게 다양한 특성식들을 꾸며 공격에 이용할 수 있다.

A를 32-비트 이진수열이라고 할 때, 일단 $A[i]$를 A의 i-번째 비트라 하고 $A[i\sim j]$를 A의 i-번째 비트에서부터 j-번째 비트라 하자. 즉, $A[i\sim j]=A[j]\|A[j-1]\|\cdots\|A[i]$ (||는 비트간 연접

(concatenation)을 의미한다)이다. 선택 평문 $P=(P_L, P_R)$와 $P'=(P_L \oplus e_{30}, P_R \oplus e_{30})$을 키 $K=(K_1, \cdots, K_8)$와 $K'=(K_1 \oplus e_{30}, K_2 \oplus e_{30}, \cdots, K_8 \oplus e_{30})$에 대해서 각각의 암호화 과정을 고려하면 키 덧셈 과정 이후 라운드 함수의 입력 차분은 확률 2^{-1}로써 0이 된다. 이 특성을 이용하면 30-라운드 연관키 차분 특성식을 확률 2^{-30}으로 꾸밀 수 있다. 게다가, $C=(C_L, C_R)$와 $C'=(C_L', C_R')$을 각각 P와 P'에 대응하는 암호문이라 하고, P와 P'을 올바른 평문 쌍(right pair)이라고 가정하면 암호문의 차분값에 대하여 다음과 같은 4개의 특정한 형태의 모양을 고려하여 해당되는 부분키 비트들을 복구할 수 있다.

Case. 1.

$$C_R \oplus C_R' = e_{30} \ \& \ C_L \oplus C_L' = e_{30}.$$

이 경우는 31 라운드와 32 라운드의 모든 S-box의 입력 차분이 0임을 뜻한다. 따라서, 우리는 $K_1[30]$의 값을 확률 $1-2^{-64}$로 얻을 수 있다.

Case. 2.

$$C_R \oplus C_R' = e_{30} \& (C_L \oplus C_L')[0\sim6]$$
$$= (C_L \oplus C_L')[11\sim29] = (C_L \oplus C_L')[31] = 0$$

이 경우는 31-라운드의 모든 S-box의 입력 차분은 0이고 32-라운드 S-box의 입력 차분은 특히, S_8-box에서만 0이 아님을 뜻한다. 따라서, 우리는 $K_1[30]$의 값을 확률 $1-2^{-60}$로 얻을 수 있다. 또한, 주어진 올바른 평문 쌍과 S_8-box의 특성을 이용하면 대략 확률 0.62로 $K_1[28]$, $K_1[29]$, $K_1[31]$을 복구할 수 있다.

Case. 3.

$$C_R \oplus C_R' \neq e_{30} \& (C_R \oplus C_R')[7] = 0.$$

이 경우는 당연히 31-라운드의 S-box의 입력 차분이 특히, S_8-box에서만 0이 아니고 그로 인해 발생하는 암호문의 오른쪽 출력 차분 중 7번째 비트가 0임을 뜻한다. 그러면,

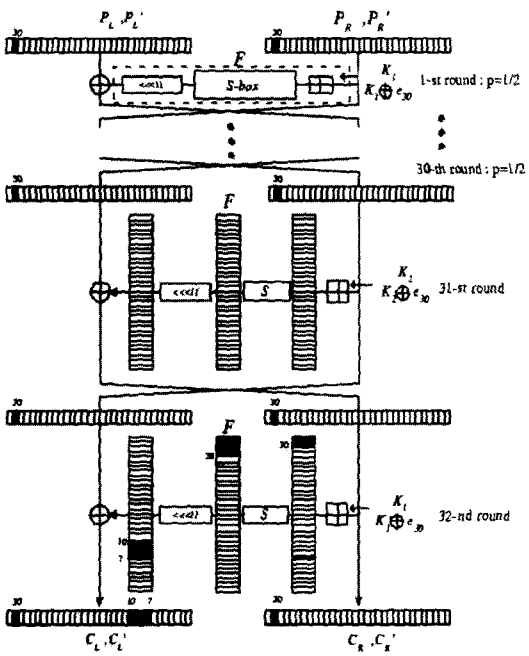


그림 5. Case. 2.를 위한 32-라운드 연관키 차분특성식

$$F_{K_1[0\sim11]}(C_R[0\sim11]) \oplus C_L[11\sim22]$$
$$= F_{K'_1[0\sim11]}(C_R'[0\sim11]) \oplus C_L'[11\sim22]$$

를 검사하여 $K_1[8\sim11]$를 복구할 수 있다. 여기서 $F_{K_1[i\sim j]}(\cdot)$는 부분키 비트 $K_1[i\sim j]$를 이용한 라운드 함수를 뜻한다.

Case. 4.

$$C_R \oplus C_R' \neq e_{30} \& (C_R \oplus C_R')[7] \neq 0.$$

이 경우는 Case. 3.와 거의 비슷하며 다만 암호문의 오른쪽 출력 차분 중 7번째 비트가 0이 아님을 뜻한다. 마찬가지 방법으로 $K_1[4\sim11]$을 복구할 수 있다.

그림 6은 특히 4번째 경우 (Case. 4.)를 묘사하고 있으며, 회색 부분은 고정되지 않은 차분 비트를 가리키고 검은색 부분과 흰색 부분은 차분이 각각 1과 0인 비트를 나타낸다. 주목할 점은 Case. 1과

Case 3.에서 복구할 수 있는 키 비트들은 Case. 2와 Case. 4에서도 복구할 수 있으므로 우리는 부록에 나타나 있는 것과 같이 Case. 2와 Case. 4에서 마지막 라운드의 부분키 12비트를 복구하는 알고리즘 1.을 제시한다. 여기서 올바르지 않은 쌍들이 여과 단계 (filtering step)를 통과할 확률은 그림 6.에서 보는 바와 같이 2^{-32}이고, 3단계에서 키 비트 K_1[30]은 알고리즘 1.에서 언급한 집합 D의 쌍들을 이용하여 확률 $1-2^{-60}$로 찾을 수 있다.

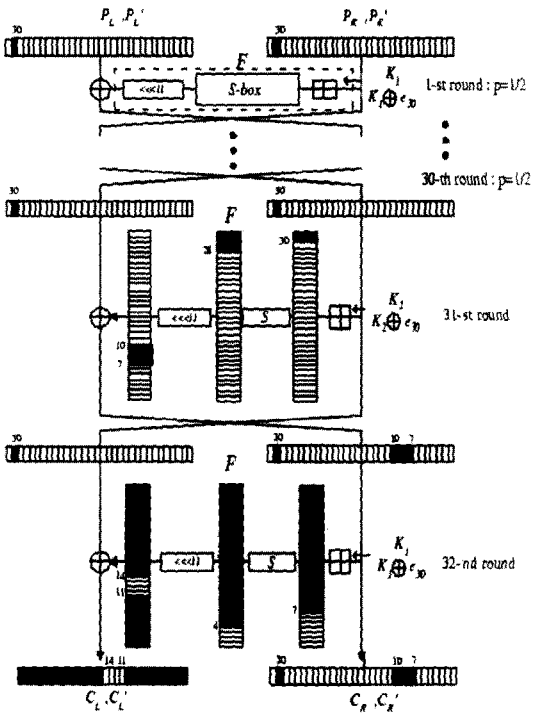


그림 6. Case. 4.를 위한 32-라운드 연관키 차분특성식

우리의 알고리즘 1.에 대한 성공확률을 계산하면 다음과 같다. 만약에 2^{35}의 선택 평문들을 고려하면 Case. 2와 Case. 4를 만족하는 쌍들은 각각 최소한 4개씩 대략 확률 0.96으로 존재한다. 그리고 최대 15개의 올바르지 않은 쌍들이 확률 0.99로 여과 단계 (filtering step)를 통과하므로 단계 3과 단계 4에서 올바르지 않은 키는 거의 확률 1로 최대 3회까지만 카운트 된다. 따라서 알고리즘 1.의 성공확률은 대략 0.917이고 이때 필요한 암호화 시간은 2×2^{35}이다.

V. 결 론

지금까지 연관키 차분 특성을 이용한 GOST의 다양한 공격 방법에 대하여 설명하였다. [10]에 의하면 일반적인 선형공격이나 차분공격에서는 전체 32-라운드 GOST에 대해서 안전하다고 언급하였지만 연관키 개념을 적용하면 32-라운드의 부분키를 찾을 수 있다는 것을 보였다.

제안한 공격 방법에 의하면 블록 암호 알고리즘인 GOST를 단지 2쌍의 선택 평문과 연관키에 대한 암호화 과정으로 랜덤 oracle과 확률 $1-2^{-64}$로써 구별할 수 있었다. 또한, H. Seki등이 사용했던 차분 특성 집합은 우리가 제시한 연관키 차분 특성식을 사용했을 경우 31-라운드 GOST에 대한 공격으로 확장됨을 보였다. 이 공격은 성공확률 97.9%로 2^{26}의 선택 평문과 2^{39}의 암호화 시간을 이용하여 31-라운드 GOST의 마지막 라운드 키 32비트를 복구할 수 있다. 마지막으로, 이 논문은 전체 32-라운드 GOST에 대한 연관키 차분 공격이 2^{35}개의 선택 평문과 2^{36}의 암호화 과정을 이용하면 91.7%로 성공할 수 있음을 보여준다.

참 고 문 헌

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

[2] E. Biham, "New Types of Cryptanalytic Attacks Using Related Keys", Advances in Cryptology-EUROCRYPT'93, Springer-Verlag, 1994, pp. 398-409.

[3] GOST, Gosudarstvennyi Standard 28147-89, "Cryptographic Protection for Data Processing Systems", Government Committee of the USSR for Standards, 1989.

[4] J. Kelsey, B. Schneier, and D. Wagner, "Key Schedule Cryptanalysis of IDEA, G-DES, GOST, SAFER, and Triple-DES" Advances in CryptologyCRYPTO '96, volume 1109 of Lecture Notes of Computer Science, Springer-Verlag, 1996, pp. 237-251.

[5] L. Knudsen, "Truncated and Higher Order Differential", Fast Software Encryption,

Second International Workshop Proceedings (FSE '94), volume 1008 of Lecture Notes of Computer Science, Springer-Verlag, 1995, pp. 229-236.

[6] National Bureau of Standards, "Data Encryption Standard", FIPS Pub. 46, 1977.

[7] B. Schneier, "Applied Cryptography", John Wiley & Sons, pp. 331-334.

[8] H. Seki and T. Kaneko, "Differential Cryptanalysis of Reduced Rounds of GOST", Seventh Annual Workshop on Selected Areas in Cryptography (SAC '00), volume 2012 of Lecture Notes of Computer Science, Springer-Verlag, 2001, pp. 315-323.

[9] A. Selcuk and A. Bicak "On Probability of Success in Linear and Differential Cryptanalysis", Third International Conference, SCN 2002, volume 2365 of Lecture Notes of Computer Science, Springer-verlag, 2002, pp. 174-185.

[10] Vitaly V. Shorin and Vadim V. Jelezniakov and Ernst M. Gabidulin, "Linear and Differential Cryptanalysis of Russian GOST", Electronic Notes in Discrete Mathematics, Vol. 6, 2001

6. 부 록

- 가 정 : 공격자는 $K \oplus K' = (e_{30}, e_{30}, \cdots, e_{30})$임을 알고 있다
- 입 력 : (P_i, P_i'), $(i = 1, \cdots, 2^{35})$ where $P_i \oplus P_i' = (e_{30}, e_{30})$
- 출 력 : K_1의 12-비트 부분키 ; $K_1[4 \sim 11]$과 $K_1[28 \sim 31]$

1.구성 단계 (Setup step)

- $K = \{k_1, k_2, \cdots, k_{2^4}\}$, $K' = \{k_1', k_2', \cdots, k_{2^8}'\}$: $K_1[4 \sim 11]$과 $K_1[28 \sim 31]$의 키 후보.
- D, D': 공 집합.
- $ctr[1] = 0, \cdots, ctr[2^4] = 0$, $ctr'[1] = 0, \cdots, ctr'[2^8] = 0$.

2.여과 단계 (Filtering step)

1. $i = 1, \cdots, 2^{35}$ 에 대해서,
   $C_i = E_K(P_i)$와 $C_i' = E_{K'}(P_i')$을 얻는다.
   만약 $C_i \oplus C_i'$이 Case. 2를 만족하면 $D = D \cup (C_i, C_i')$,
   만약 $C_i \oplus C_i'$이 Case. 4를 만족하면 $D' = D' \cup (C_i, C_i')$.

3. $K_1[28 \sim 31]$ 찾기 (Finding key $K_1[28 \sim 31]$)

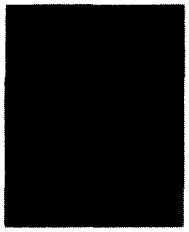
1) $j = 1, \cdots, 2^4$ 에 대해서,
   각각의 $(C_i, C_i') \in D'$ 에 대해서 ,
   만약
   $F_{K_j}(C_R[28\text{-}31]) \oplus C_L[7\text{-}10] = F_{K'_j}(C_R'[28\text{-}31]) \oplus C_L'[7\text{-}10]$, 이면
   $ctr[j] += 1$.
2) $j = 1, \cdots, 2^4$ 에 대해서,
   만약 $ctr[j] \geq 4$이면, k_j를 $K_1[28 \sim 31]$로 출력, GOTO 4.

4. $K_1[4 \sim 11]$ 찾기 (Finding key $K_1[4 \sim 11]$)

1) $j = 1, \cdots, 2^8$ 에 대해서,
   1.1) $i = 0, \cdots, 2^4 - 1$ 에 대해서,
   각각의 $(C_i, C_i') \in D'$ 에 대해서,
   만약
   $F_{K_j \| i}(C_R[0\text{-}11]) \oplus C_L[11\text{-}22] = F_{K'_j \| i}(C_R'[0\text{-}11]) \oplus C_L'[11\text{-}22]$, 이면
   $ctr'[j] += 1$.
   ($k_j \mid i$ 와 $k_j' \mid i$ 는 $K_1[0 \sim 11]$과 $K_1[0 \sim 11]$를 각각 가리킨다.)
2) $j = 1, \cdots, 2^8$에 대해서,
   만약 $ctr'[j] \geq 4$ 이면, k_j'를 $K_1[4 \sim 11]$로 출력, 종료.

알고리즘 1. 32-라운드 GOST 연관키 차분 공격

〈著者紹介〉

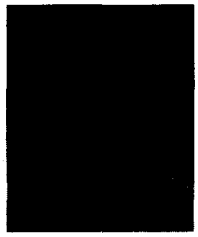


이 태 건 (Tae-keon Lee)
2002년 8월 : 한신대학교 수학과 학사
2003년 3월~현재 : 고려대학교 정보보호학과 석사 과정
〈관심분야〉 블록 암호 및 운영모드 분석, 설계

고 영 대 (Young-dai Ko)
2002년 8월 : 서울시립대학교 수학과 학사
2002년 9월~현재 : 고려대학교 정보보호학과 석사 과정
〈관심분야〉 블록 암호 및 스트림 암호 분석, 설계

이 원 일 (Wonil Lee) 정회원
1998년 2월 : 고려대학교 수학과 학사
2000년 2월 : 고려대학교 수학과 석사
2003년 8월 : 고려대학교 수학과 박사
2000년 8월~현재 : 고려대학교 정보보호기술연구센터 연구원
〈 관심분야〉 해쉬 함수, 블록 암호, 스트림 암호, 암호 프로토콜

홍 석 희 (Seok-hie Hong) 정회원
1995년 2월 : 고려대학교 수학과 학사
1997년 2월 : 고려대학교 수학과 석사
2001년 2월 : 고려대학교 수학과 박사
2000년 8월~현재 : 고려대학교 정보보호기술연구센터 연구원
〈관심분야〉 정보보호 암호 알고리즘, 비밀키 암호 설계 및 분석, 패스워드 기반 프로토콜



이 상 진 (Samg-jin Lee) 정회원
1987년 2월 : 고려대학교 수학과 학사
1989년 2월 : 고려대학교 수학과 석사
1994년 2월 : 고려대학교 수학과 박사
1989년 2월~1999년 2월 : 한국전자통신연구원 선임 연구원
1999년 2월~2001년 8월 : 고려대학교 자연과학대학 조교수
2001년 9월~ 현재 : 고려대학교 정보보호대학원 부교수
〈관심분야〉 대칭키 암호의 분석 및 설계, 정보은닉이론, 컴퓨터 포렌식스.