軽量暗号SAND-64に対する差分識別子の探索

杉尾 信行[1,a)]

概要：SAND は Chen らによって提案された AND-RX 型軽量ブロック暗号である．SAND は構造の違いから SAND-64，SAND-128 の 2 種類が存在する．本稿では，SAT ソルバーを用いて SAND-64 に対する差分識別子の探索を行い，確率 2^{-58} で成立する 12 段の差分識別子が存在することを明らかにする．また，発見した 12 段の差分識別子を用いて 16 段 SAND-64 に対する鍵回復攻撃を示す．鍵回復攻撃に必要な選択平文数は 2^{63}，計算量は 2^{108} 回の暗号化計算量，メモリ量は 2^{43} バイトである．

キーワード：軽量暗号，差分攻撃，AND-RX，SAND

Searching for Differential Distinguishers on SAND-64

Nobuyuki Sugio[1,a)]

Abstract: SAND is a lightweight block cipher of the AND-RX type proposed by Chen et al. Due to differences in structure, there are two variants: SAND-64 and SAND-128. In this paper, we investigate differential distinguishers for SAND-64 using a SAT solver, and demonstrate the existence of a 12-round differential distinguisher with probability 2^{-58}. We illustrate a key recovery attack on 16-round SAND. The attack complexities are 2^{63} data, 2^{108} time, and 2^{43} memory, respectively.

Keywords: SAND, AND-RX, lightweight cipher, differential attack

1. はじめに

SAND は Disigns, Codes and Cryptography 2022 で Chen らによって提案された AND-RX 型の軽量ブロック暗号である [1]．ブロック長に応じて SAND-64，SAND-128 の 2 種類が存在する．本稿では，SAND-64 を解析対象とする．SAND-64 はブロック長は 64-bit，秘密鍵長は 128-bit，推奨段数は 48 段である．

1.1 本稿の貢献

本稿では，SAND-64 に対して SAT ソルバーを用いてビット単位の差分識別子を探索し，差分確率 2^{-58} で成立する 12 段差分識別子を発見した．得られた 12 段の差分特性を用いて，16 段 SAND-64 に対する鍵回復攻撃を示す．

[1] 北海道科学大学
Hokkaido University of Science
[a)] sugio-n@hus.ac.jp

1.2 本稿の構成

本稿の構成を以下に示す．2 章にて，SAND-64 に対する関連研究を纏める．3 章にて，差分攻撃について説明する．4 章にて，充足可能問題 (SAT) を用いた差分識別子の探索について紹介する．5 章にて，軽量暗号 SAND-64 の内部構造を説明する．6 章にて，SAND-64 に対して 12 段差分識別子を用いた鍵回復攻撃を行う．7 章にて，まとめと今後の課題を示す．

2. 関連研究

SAND の設計者らは，差分攻撃 [2]，線形攻撃 [3]，不可能差分攻撃 [4]，ゼロ相関線形攻撃 [5], [6]，及び積分攻撃 [7] といった各種攻撃手法に対する SAND の安全性を評価している．彼らは SAT/SMT を用いて，ニブル単位 (4-bit 単位) で差分特性に関する最小アクティブ S-box 数の探索を行い，16 段で最小アクティブ S-box 数が 34 となる事を示した [1]．

表 1　Attack Results on SAND-64

Number of Rounds	Data	Time	Memory	Method
15	2^{63}	2^{105}	2^{52}	Integral by CBDP [10]
15	$2^{63.09}$	2^{56}	2^{59}	Boomerang [11]
16	2^{63}	$2^{109.91}$	2^{85}	Integral by CBDP [10]
16	2^{63}	2^{96}	2^{62}	Impossible Differential [12])
16	2^{63}	2^{108}	2^{43}	Differential (**Ours**)

CBDP: Conventional Bit-based Division Property.

Zhang らは，AND-RX 型暗号に対する不可能差分識別子探索のための体系的な探索フレームワークを開発した．この手法を SAND に適用することで，SAND-64 における 11 ラウンドの不可能差分識別子を発見している [8].

Mirzaie らは，従来のビット単位の Division Property 手法 [9] を SAND-64 に適用し，積分識別子を探索した．その結果，23 ビットがバランスしている 12 ラウンドの積分識別子を発見した．識別子に必要なデータ量は 2^{63} である．この識別子に基づき，Mirzaie らは SAND-64 の 15 ラウンドおよび 16 ラウンドに対して鍵回復攻撃を行い，それぞれ 2^{105} 回および $2^{109.9}$ 回の暗号化計算量が必要であることを示した [10].

さらに，Li と Teh は，AND 専用に設計された新しいブーメラン接続表を用いて，SAND に対するブーメラン型攻撃を提案した．この手法により，SAND-64 に対して 15 ラウンドの鍵回復攻撃が可能であることを示した [11].

杉尾は制約プログラミング (CP) を用いて不可能差分識別子の探索を行い，12 段の不可能差分識別子を発見した．また，12 段の不可能差分識別子を用いて，16 段の鍵回復攻撃を示した [12].

3. 差分攻撃

差分攻撃は, Biham らによって提案された手法である [2]. 平文ペア (P, P^*) に関する排他的論理和差分を $\Delta P = P \oplus P^*$ とする．入力 $X = P \oplus K$ と $X^* = P^* \oplus K$ に関する排他的論理和差分 ΔX は以下の式で表される．

$$\Delta X = X \oplus X^* = (P \oplus K) \oplus (P^* \oplus K) = \Delta P$$

ΔX を入力差分，ΔY を出力差分とする．S-box の差分確率は以下の式で定義される．

$$DP(\Delta X \to \Delta Y) = \frac{\#\{X | S(X) \oplus S(X \oplus \Delta X) = \Delta Y\}}{2^n} \quad (1)$$

式 (1) は S-box に入力される鍵 K とは独立に成立する．平文 P が一様分布に従う場合，出力差分 ΔY は入力差分 ΔP に対して差分確率 DP で期待される．

4. 充足可能問題と暗号解析への応用

4.1 充足可能問題

充足可能問題（Satisfiability Problem，SAT）は，命題論理式が与えられたとき，その論理式を真（true）にするような変数の割り当てが存在するかどうかを判定する問題である．論理式はブール変数と論理演算 AND（∧），OR（∨），NOT（¬）から成る式であり，その論理式を真にする変数の割り当てが存在する場合は「充足可能である」という．

4.2 暗号解析への応用

Sun らは識別子の探索に SAT ソルバーを利用する手法を提案した [13], [14]．以下に，Sun らが提案した差分伝搬の SAT モデリング手法について概説する．詳細は文献 [13], [14] を参照のこと．

記号 α_i $(0 \le i \le n-1)$ を n-bit ベクトル α の i-bit 目を表すものとする．

差分伝搬モデル 1 （分岐） 分岐において $\alpha \in \mathbb{F}_2^n$ を入力差分とし，$\beta \in \mathbb{F}_2^n$ と $\gamma \in \mathbb{F}_2^n$ を出力差分とする．分岐の差分伝搬モデルは以下の式で表される．

$$\left.\begin{array}{l} \alpha_i \vee \overline{\beta_i} = 1 \\ \overline{\alpha_i} \vee \beta_i = 1 \\ \alpha_i \vee \overline{\gamma_i} = 1 \\ \overline{\alpha_i} \vee \gamma_i = 1 \end{array}\right\} (0 \le i \le n-1)$$

差分伝搬モデル 2 （XOR） n-bit の排他的論理和において $\alpha \in \mathbb{F}_2^n$ と $\beta \in \mathbb{F}_2^n$ を入力差分とし，$\gamma \in \mathbb{F}_2^n$ を出力差分とする．排他的論理和の差分伝搬モデルは以下の式で表される．

$$\left.\begin{array}{l} \alpha_i \vee \beta_i \vee \overline{\gamma_i} = 1 \\ \alpha_i \vee \overline{\beta_i} \vee \gamma_i = 1 \\ \overline{\alpha_i} \vee \beta_i \vee \gamma_i = 1 \\ \overline{\alpha_i} \vee \overline{\beta_i} \vee \overline{\gamma_i} = 1 \end{array}\right\} (0 \le i \le n-1)$$

差分伝搬モデル 3 （S-box） S-box において $\alpha \in \mathbb{F}_2^n$ を入力差分とし，$\beta \in \mathbb{F}_2^m$ を出力差分とする．また，入力差分 α を与えた時に出力差分 β が得られる確率を $p(\alpha, \beta)$ と表す．また，0 でない確率の値を表す為の変数 $w \in \mathbb{F}_2^s$, $w_{i+1} \le w_i$ $(0 \le i \le s-2)$ を導入する．S-box の差分伝搬モデルを作成する為，以下のブール関数を考える．

$$g(\alpha, \beta, w) = \begin{cases} 1, \text{ if } p(\alpha, \beta) = 2^{-\sum_{i=0}^{s-1} w_i} \\ 0, \text{ otherwize.} \end{cases}$$

ここで，集合 A を以下の様に定義する．

$$A = \{(a,b,c) \in \mathbb{F}_2^{n+m+s} | g(a,b,c) = 0\}.$$

集合 A は差分確率が 0，即ち不能な差分パスを全て含む集合である．また，集合 A に含まれるベクトルの数を $|A|$ と表す．S-box の差分伝搬モデルは $|A|$ 個の節を用いて以下の様に構成される．

$$\bigvee_{i=0}^{n-1}(\alpha_i \oplus a_i^l) \vee \bigvee_{j=0}^{m-1}(\beta_j \oplus b_j^l) \vee \bigvee_{k=0}^{s-1}(w_k \oplus c_k^l) = 1$$

$$(0 \le l \le |A| - 1)$$

上記をブール関数 $h(\alpha,\beta,w)$ で表すと以下の様になる．

$h(\alpha,\beta,w) =$

$$\bigwedge_{l=0}^{|A|-1}\left(\bigvee_{i=0}^{n-1}(\alpha_i \oplus a_i^l) \vee \bigvee_{j=0}^{m-1}(\beta_j \oplus b_j^l) \vee \bigvee_{k=0}^{s-1}(w_k \oplus c_k^l)\right) = 1 \quad (2)$$

また，Abdelkhalek らは式（2）に Quine-McCluskey アルゴリズムや Espresso アルゴリズムを適用し，軽量な差分伝搬モデルが作成できる事を示している [15].

5. 軽量暗号 SAND-64

SAND は Chen らによって提案された軽量ブロック暗号である [1]．内部構造は AND-RX 型であり，ブロック長に応じて SAND-64，SAND-128 の 2 種類が存在する．本稿では SAND-64 を対象とし，ブロック長 64-bit，秘密鍵長 128-bit，推奨段数は 48 段である．また，以下の説明において $n = 32$ とする．なお，以下の説明において，入力と出力にそれぞれ存在する `StateLoding`(置換) は省略する．

5.1 準備

SAND-64 で用いる記号を以下に示す．

- $x = (x_{n-1}, x_{n-2}, \ldots, x_0)$：$n$-bit 変数であり，$x_{n-1}$ が最上位ビット（MSB），x_0 が最下位ビット（LSB）を示す．変数 x において，$4 \times \frac{n}{4}$ の配列を用いて表される．

$$x = \begin{bmatrix} x_{n-1} & \ldots & x_7 & x_3 \\ x_{n-2} & \ldots & x_6 & x_2 \\ x_{n-3} & \ldots & x_5 & x_1 \\ x_{n-4} & \ldots & x_4 & x_0 \end{bmatrix}$$

- $x||y$：変数 x と y の結合
- $x \ll s$：変数 x を s-bit 左シフト
- $x \lll t$：変数 x を t-bit 左巡回シフト
- $x \lll_{\frac{n}{4}} t$：変数 x を 4 つの $\frac{n}{4}$-bit ワード $x = (x_{n-1}, x_{n-2}, \ldots, x_0) = x\{3\}||x\{2\}||x\{1\}||x\{0\}$ に分割し，各ワード $x\{i\}$ 内で t-bit 左巡回シフトを行う．すなわち，$x \lll_{\frac{n}{4}} t = (x\{3\} \lll_{\frac{n}{4}} t)||(x\{2\} \lll_{\frac{n}{4}} t)||(x\{1\} \lll_{\frac{n}{4}} t)||(x\{0\} \lll_{\frac{n}{4}} t)$.
- $x \odot y$：ビット毎の AND 演算
- $x \oplus y$：ビット毎の排他的論理和演算
- $x[i]$：変数 x の i 番目ニブル（4-bit）．
  $x = (x_{n-1}, x_{n-2}, \ldots, x_0)$ において，

$$x[\tfrac{n}{4} - 1] = (x_{n-1}, x_{n-2}, x_{n-3}, x_{n-4}),$$
$$\cdots$$
$$x[1] = (x_7, x_6, x_5, x_4),$$
$$x[0] = (x_3, x_2, x_1, x_0).$$

5.2 段関数

SAND-64 の段関数を図 1 に示す．r 段の入力を (x^r, y^r)，段鍵を sk^r，出力を (x^{r+1}, y^{r+1}) とする．段関数には 2 種類の非線形関数 G_0 と G_1，および線形関数 P_n が存在する．また，$\alpha = 0$，$\beta = 1$ である．

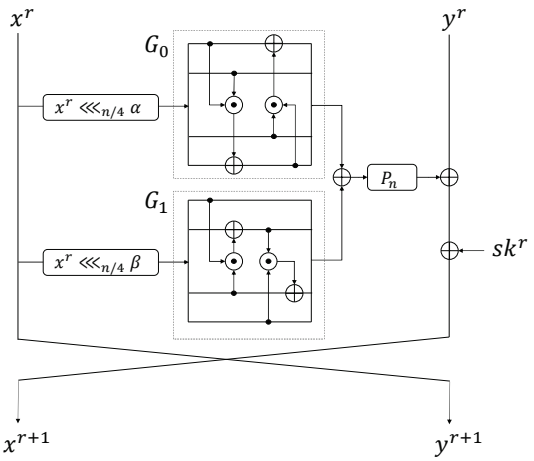

図 1 SAND-64 の段関数

G_0 と G_1 の入力を n-bit 変数 $x = x\{3\}||x\{2\}||x\{1\}||x\{0\}$ とし，出力を $y = y\{3\}||y\{2\}||y\{1\}||y\{0\}$ とする．G_0 において以下の式が成立する．

$$y\{3\} = y\{0\} \odot x\{1\} \oplus x\{3\},$$
$$y\{2\} = x\{2\},$$
$$y\{1\} = x\{1\},$$
$$y\{0\} = x\{3\} \odot x\{2\} \oplus x\{0\}.$$

同様に，G_1 において以下の式が成立する．

$$y\{3\} = x\{3\},$$
$$y\{2\} = x\{3\} \odot x\{1\} \oplus x\{2\},$$
$$y\{1\} = y\{2\} \odot x\{0\} \oplus x\{1\},$$
$$y\{0\} = x\{0\}.$$

線形関数 P_n の i 番目の入力ワード $x\{i\} = (x_{\frac{n}{4}\cdot i+\frac{n}{4}-1}, \ldots, x_{\frac{n}{4}\cdot i+1}, x_{\frac{n}{4}\cdot i})$ に対し，出力ワード $y\{i\}$ は以下の式で定義される．

$$y_{\frac{n}{4}\cdot i+p\frac{n}{4}(j)} = x_{\frac{n}{4}\cdot i+j}, \text{ for } 0 \le j < \frac{n}{4},\ 0 \le i < 4.$$

線形関数 P_n は $\frac{n}{4}$-bit ワードに対する置換 p_8 を 4 つ並列に適用したものと見なすことができる．置換 p_8 を表 2 に示す．

表 2　SAND-64's Permutation p_8

j	0	1	2	3	4	5	6	7
$P_8(j)$	7	4	1	6	3	0	5	2

5.3　鍵生成

SAND-64 は 128-bit の秘密鍵から段鍵を生成する．秘密鍵を 32-bit 毎のワードの結合 $K = K^3||K^2||K^1||K^0$ と見なす．SAND-64 の鍵生成部を図 3，図 4 に示す．

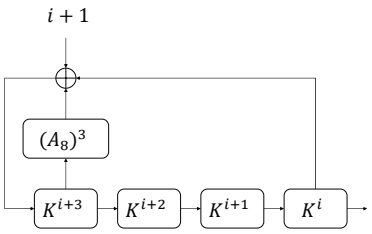

図 2　SAND-64 の鍵生成部

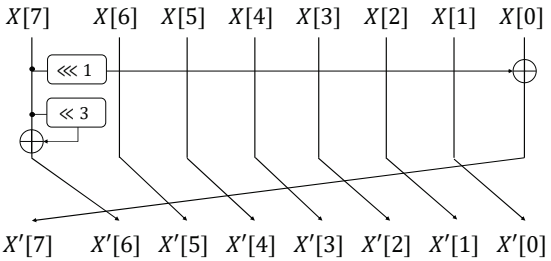

図 3　A_8

$i+1$, $0 \le i \le R-4$ は段毎の定数である．LFSR の更新は以下の式で定義される．

$$K^{i+4} \leftarrow (A_8)^3(K^{i+3}) \oplus K^i \oplus (i+1).$$

A_8 は 4-bit 単位で処理を行なう関数であり，K^{i+3} に対して 3 回繰り返して適用される．段鍵 sk^r, $(0 \le r < R)$ は K^r が以下の通り設定される．

$$K^r = \begin{bmatrix} K^r_{31} & \dots & K^r_7 & K^r_3 \\ K^r_{30} & \dots & K^r_6 & K^r_2 \\ K^r_{29} & \dots & K^r_5 & K^r_1 \\ K^r_{28} & \dots & K^r_4 & K^r_0 \end{bmatrix},$$

$$sk^r = K^r_{31} \dots K^r_3 || K^r_{30} \dots K^r_2 || K^r_{29} \dots K^r_1 || K^r_{28} \dots K^r_0.$$

6.　SAND-64 に対する差分攻撃

6.1　SAT モデルの構築

4 章に示す SAT モデルを用いて，SAND-64 の差分識別子を探索する為の SAT モデルを構築する．以下に示す変数の定義域は $\{0,1\}$ とし，それぞれ差分値が 0，又は 1 を表すものとする．

図 1 に示す段関数から，32-bit の変数

$$\Delta X^r = \Delta x^r\{3\}||\Delta x^r\{2\}||\Delta x^r\{1\}||\Delta x^r\{0\}$$
$$\Delta Y^r = \Delta y^r\{3\}||\Delta y^r\{2\}||\Delta y^r\{1\}||\Delta y^r\{0\}$$

$(0 \le r \le R)$ を定義する．これらの変数は各段における内部状態の差分を表す変数である．ΔX^0，ΔY^0 は入力差分を表し，ΔX^R，ΔY^R は R 段目の出力差分を表すものとする．

変数 ΔX^r，ΔY^r を用いて SAT モデルの構築を以下に示す．非線形関数 G_0 と G_1 の出力差分を表す 32-bit の変数 ΔG^r_0，ΔG^r_1 $(1 \le r \le R)$ を定義し，以下の制約式を構成する．

$$\Delta G^r_0 = G_0(\Delta X^r),$$
$$\Delta G^r_1 = G_1(\Delta X^r \lll_{\frac{n}{4}} 1).$$

G_0 と G_1 の制約式は差分分布表（differential distribution table，DDT）から導出する．なお，本稿では文献 [16] の付録 N を参考に，sbox analyzer[*1]を用いて G_0 と G_1 の制約式を導出した．

線形関数 P_n の出力差分を表す 32-bit の変数 ΔP^r $(1 \le r \le R)$ を定義し，以下の制約式を構成する．

$$\Delta P^r = P_n(\mathtt{XOR}(\Delta G^r_0, \Delta G^r_1)).$$

段関数の出力に関して，以下の制約式を構成する．

$$\Delta X^{r+1} = \mathtt{XOR}(\Delta Y^r, \Delta P^r),$$
$$\Delta Y^{r+1} = \Delta X^r.$$

上記の制約式を r 段 $(0 \le r \le R)$ 作成し，SAT モデルの構築が完成する．また，以下に示す制約条件を追加し，自明な解をすべて除外する．

$$\sum_{i=0}^{n-1} \Delta X^0_i + \sum_{i=0}^{n-1} \Delta Y^0_i \neq 0$$

上記において，ΔX^0_i と ΔY^0_i は入力差分 ΔX^0 と ΔY^0 の i-th bit を表すものとする．最後に，r 段差分識別子の確率が $p = 2^{-m}$ となることを保証するため，Sequential Encoding Method [14] を用いて以下の制約を追加する．

$$\mathtt{Sequential\_Encoding\_Method}(\text{bound} = m)$$

上記は，差分が通過する複数の S-box(アクティブ S-box) を含む差分特性確率 DCP の上限が 2^{-m} となることを表す．構築した SAT モデルが SAT ソルバーにて充足可能である場合，r 段の差分識別子が存在することを意味する．

6.2　SAND-64 の差分識別子探索

本稿では，前節で構築した SAT モデルを Python で実装し，SAT ソルバーは Kissat[17][*2]を用いた．本稿で使用した計算機環境を表 3 に示す．

[*1] https://github.com/hadipourh/sboxanalyzer
[*2] https://github.com/arminbiere/kissat

表 3　Computer Environment

Environment	Details
OS	Desktop 24.04.2 LTS
Platform	VS Code
Solver	Kissat
CPU	AMD Ryzen 7 5800X
Memory	128 GB
Disk	9 TB

SAND-64 に対する差分特性を探索した結果，65 秒で解が得られ，差分確率 $p = 2^{-58}$ で成立する 12 段差分識別子を発見した．詳細を表 4 に示す．

表 4　12-round differential distinguisher of SAND-64

ΔX^r		ΔY^r		$log_2(p)$
ΔX^0	00450000	ΔY^0	00F04400	
ΔX^1	00840000	ΔY^1	00450000	−6
ΔX^2	00800000	ΔY^2	00840000	−10
ΔX^3	00000000	ΔY^3	00800000	−12
ΔX^4	00800000	ΔY^4	00000000	−12
ΔX^5	00840000	ΔY^5	00800000	−14
ΔX^6	00450000	ΔY^6	00840000	−18
ΔX^7	00F04000	ΔY^7	00450000	−24
ΔX^8	00CA0400	ΔY^8	00F04000	−33
ΔX^9	000A8000	ΔY^9	00CA0400	−43
ΔX^{10}	00100000	ΔY^{10}	000A8000	−50
ΔX^{11}	00038000	ΔY^{11}	00100000	−52
ΔX^{12}	04929500	ΔY^{12}	00038000	−58

また，13 段差分識別子を探索した結果，116 秒で解が得られ，差分確率 $p = 2^{-64}$ で成立する識別子を発見した．詳細を表 5 に示す．

表 5　13-round differential distinguisher of SAND-64

ΔX^r		ΔY^r		$log_2(p)$
ΔX^0	20000000	ΔY^0	21000000	
ΔX^1	00000000	ΔY^1	20000000	−2
ΔX^2	20000000	ΔY^2	00000000	−2
ΔX^3	21000000	ΔY^3	20000000	−4
ΔX^4	91000000	ΔY^4	21000000	−8
ΔX^5	3C000008	ΔY^5	91000000	−15
ΔX^6	B2000009	ΔY^6	3C000008	−25
ΔX^7	420000C1	ΔY^7	B2000009	−39
ΔX^8	00000019	ΔY^8	420000C1	−48
ΔX^9	02000002	ΔY^9	00000019	−54
ΔX^{10}	02000000	ΔY^{10}	02000010	−59
ΔX^{11}	00000000	ΔY^{11}	02000000	−62
ΔX^{12}	02000000	ΔY^{12}	00000000	−62
ΔX^{13}	12000000	ΔY^{13}	02000000	−64

6.3　16 段 SAND-64 に対する鍵回復攻撃

表 4 に示す 12 段の差分識別子を用いて，図 4 に示す 16 段 SAND-64 に対する鍵回復攻撃を行う．段関数を F と略記する．図 4 において，差分が存在する箇所を赤，又は灰色で示す．

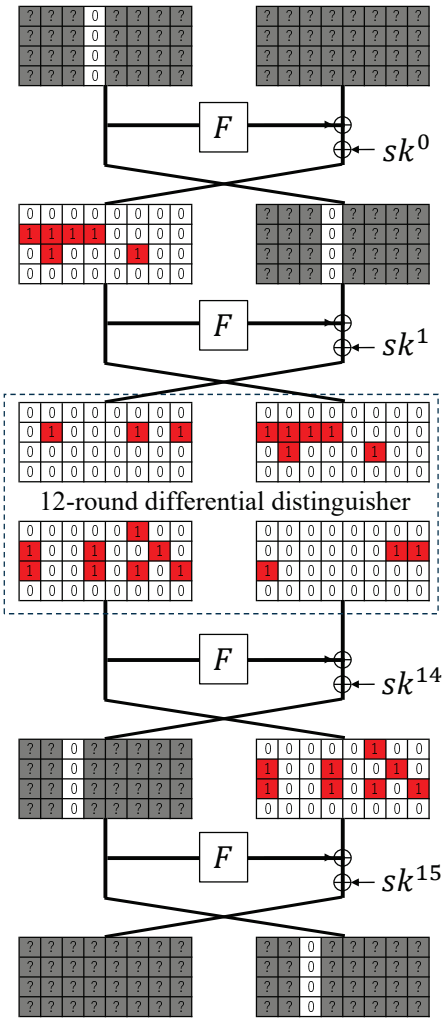

図 4　16 段 SAND-64 に対する鍵回復攻撃

鍵回復攻撃の手順を以下に示す．

(1) 図 4 に示す入力差分 $(\Delta X^0, \Delta Y^0)$ を 2^{62} 組用意する．

(2) 2^{62} 組の入力差分 $(\Delta X^0, \Delta Y^0)$ を暗号化し，対応する暗号差分 $(\Delta X^{16}, \Delta Y^{16})$ を入手する．図 4 に示す暗号文差分の条件を満たす暗号差分 $(\Delta X^{16}, \Delta Y^{16})$ は $2^{62} \times 2^{-4} = 2^{58}$ 組入手できる．

(3) 段鍵 $sk^0 = 20$ bits と $sk^{15} = 20$ bits の合計 40 bits に対応したカウンター c_{sk} を用意し，カウンター c_{sk} を 0 で初期化する．

(4) 段鍵 sk^0 と sk^{15} をそれぞれ推定する．2^{58} の差分組を用いて，平文側と暗号文側から部分的に暗号化 (又は復号) を行い，2 段目出力差分 $(\Delta X^2, \Delta Y^2)$ と 14 段目出力差分 $(\Delta X^{14}, \Delta Y^{14})$ が図 4 に示す差分と一致した場合，カウンター c_{sk} に 1 を加算する．

(5) 上記 (4) を段鍵 sk^0 = 20 bits と sk^{15} = 20 bits の合計 40 bits に対して繰り返し行い，カウンター c_{sk} の値が最も大きい値を正しい鍵として出力する．

(6) 鍵生成部を用いて残りの秘密鍵 K=128-20=108 bits を総当たりで回復する．

上記を行う為に必要な選択平文数は $D = 2^{63}$ である．2^{62} 組の入力差分 $(\Delta X^0, \Delta Y^0)$ を暗号化する計算量は $T_1 = 2^{63}$ 回の暗号化計算量である．また，段鍵 sk^0 と sk^{15} の推定に必要な計算量は $T_2 = 2^{40} \times 2 \times 2^{58} \times \frac{4}{16} = 2^{96}$ 回の暗号化計算量である．最後に，残りの秘密鍵 K=108 bits を総当たりする計算量は $T_3 = 2^{108} + 2^{44} \approx 2^{108}$ 回の暗号化計算量である．以上より，16 段 SAND-64 の鍵回復攻撃に必要な計算量は $T = T_1 + T_2 + T_3 \approx 2^{108}$ 回の暗号化計算量である．また，カウンター c_{sk} に必要なメモリ量は $M = 2^{40} \times 64 \times \frac{1}{8} = 2^{43}$ バイトである．攻撃結果を表 1 に示す．

7. まとめと今後の課題

軽量暗号 SAND-64 の差分識別子を探索し，確率 2^{-58} で成立する 12 段差分識別子が存在することを明らかにした．また，発見した 12 段差分識別子を用いて 16 段の SAND-64 に対する鍵回復攻撃が可能であることを示した．

今後の課題を以下に示す．1 点目は，鍵回復攻撃に優位な差分識別子の探索である．今回は差分確率 p に着目した探索を行ったが，Zong らが提案した探索手法 [18] の適用が考えられる．2 点目は，SAND-64 に対する鍵回復攻撃の改良である．今回は素朴に鍵回復攻撃を行ったが，Boura らの差分中間一致攻撃 [19] 等の適用が考えられる．3 点目は，差分識別子の探索を SAND-128 に適用し，差分攻撃に対する安全性評価を行うことである．

参考文献

[1] Chen, S., Fan, Y., Sun, L., Fu, Y., Zhou, H., Li, Y., Wang, M., Wang W., and Guo, C.: SAND: an AND-RX Feistel lightweight block cipher supporting S-box-based security evaluations, Designs, Codes and Cryptography, Vol. 90, pp. 155–198 (2022).

[2] Biham, E., and Shamir, A.: Differential Cryptanalysis of the Data Encryption Standard", Springer-Verlag, New York, pp. 79-88 (1993).

[3] Matsui, M.: Linear Cryptanalysis Method for DES Cipher, Proc. Workshop on the Theory and Application of Cryptographic Techniques, EUROCRYPT ' 93, Vol.765 of LNCS, pp.386–397, Springer-Verlag (1993).

[4] Biham, E., Biryukov, A., and Shamir, A.: Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials", Advances in Cryptology-EUROCRYPT'99, vol. 1592 of LNCS, pp. 12–23 (1999).

[5] Bogdanov, A., Wang, M.: Zero Correlation Linear Cryptanalysis with Reduced Data Complexity, Proc. of the 19th International Workshop on Fast Software Encryption, FSE 2012, Vol. 7549 of LNCS, pp. 29–48, (2012).

[6] Bogdanov, A., and Rijmen, V.: Linear hulls with correlation zero and linear cryptanalysis of block ciphers, Designs, Codes and Cryptography, Vol. 70, pp. 369-383, (2014).

[7] Knudsen, L. R., and Wagner, D.: Integral cryptanalysis, Proc. of Fast Software Encryption, FSE2002, Vol.2365 of LNCS, pp.112-127. Springer-Verlag, (2002).

[8] Zhang, K., Wang, S., Lai, X., Wang, L., Guan, J., Hu, B.: Impossible Differential Cryptanalysis and a Security Evaluation Framework for AND-RX Ciphers, IEEE Transactions on Information Theory, vol. 70, no. 8, pp. 6025–6040, (2024).

[9] Xiang, Z., Zhang, W., Bao, Z., and Lin, D.: Applying MILP Method to Searching Integral Distinguishers Based on Division Property for 6 Lightweight Block Ciphers, Proc. 22nd International Conference on the Theory and Application of Cryptology and Information Security, ASIACRYPT2016, Vol.10031 of LNCS, pp.648-678, Springer-Verlag (2016).

[10] Mirzaie, A., Ahmadi, S., Aref, R. M.: Integral Cryptanalysis of Reduced-Round SAND-64 Based on Bit-Based Division Property, ISC International Journal of Information Security (ISeCure), vol. 15, no. 3, (2023).

[11] Li, Y. and Teh, J. S.: Boomerang Cryptanalysis of SAND, Journal of Information Security and Applications, Vol. 92, pp. 104086, (2025).

[12] Sugio, N.: Impossible Differential Attack on SAND-64, Proc. 26th International Conference on Information Security Applications, WISA 2025, (to be appeared.)

[13] Sun, L., Wang, W., and Wang, M.: More Accurate Differential Properties of LED64 and Midori64, IACR Transactions on Symmetric Cryptology, Vol.2018, Issue.3, no.3, pp.93-123 (2018).

[14] Sun, L., Wang, W., and Wang, M.: Accelerating the Search of Differential and Linear Characteristics with the SAT Method, IACR Transactions on Symmetric Cryptology, Vol.2021, Issue.1, no.1, pp.269-315 (2021).

[15] Abdelkhalek, A., Sasaki, Y., Todo, Y., Tolba, M., and Youssef, A. M.: MILP Modeling for (Large) S-boxes to Optimize Probability of Differential Characteristics, IACR Transactions on Symmetric Cryptology, Vol. 2017, No. 4, pp. 99-129 (2017).

[16] Hadipour, H., Gerhalter, S., Sadeghi, S. and Eichlseder, M.: Improved Search for Integral, Impossible Differential and Zero-Correlation Attacks, Application to Ascon, ForkSKINNY, SKINNY, MANTIS, PRESENT and QARMAv2, IACR Transactions on Symmetric Cryptology, Vol. 2024, No. 1, pp. 234–325 (2024).

[17] Biere, A., Faller, T., Fazekas, K., Fleury, M., Froleyks, N., and Pollitt, F..: CaDiCaL, Gimsatul, IsaSAT and Kissat Entering the SAT Competition 2024, Proc. SAT Competition 2024: Solver, Benchmark and Proof Checker Descriptions, Vol. B-2024-1, pp.8-10, (2024)

[18] Zong, R., Dong, X., Chen, H., Luo, Y., Wang, S., and Li, Z.: Towards Key-recovery-attack Friendly Distinguishers: Application to GIFT-128, IACR Transactions on Symmetric Cryptology, Vol. 2021, No. 1, pp.156-184, (2021).

[19] Boura, C., David, N., Derbez, P., Leander, G., Naya-Plasencia, M.: Differential Meet-In-The-Middle Cryptanalysis, Proc. of 43rd Annual International Cryptology Conference, CRYPTO 2023, Vol. 14083 of LNCS, pp.240 - 272, (2023).