暗号アルゴリズム評価報告書
（SEED）

2002 年 1 月 25 日

日本電信電話株式会社

盛合 志帆

暗号アルゴリズム評価報告書 (SEED)

盛合 志帆

日本電信電話株式会社

2002 年 1 月 25 日

概要　本報告書は、暗号アルゴリズム SEED について安全性の評価を行なった結果を報告する。本報告書では、ブロック暗号に対する汎用的な攻撃方法として、差分解読法、線形解読法、及び代数的解読法として高階差分攻撃と補間攻撃に対する耐性を評価した。さらなる評価の余地はあるものの、現時点では、仕様にある 16 段 SEED を解読する方法は見つからず、これらの攻撃法に対して SEED は適切な安全性マージンを有していると考えられる。

もくじ

1 はじめに

SEED は 1997 年の韓国政府の標準暗号アルゴリズム開発決定を受けて、 Korea Information Security Agency (KISA) により設計された暗号アルゴリズムである [7]。

SEED は韓国国内で標準化が進んでおり、 1999 年に TTA (Telecommunications and Technology Association) 標準 (産業標準) となり (TTA KO-12.0004)、 韓国情報通信省 Ministry of Information and Communication (MIC) により推進されている政府標準 Korean Infomation Communication Standard (KICS) への制定作業も進んでいる [8]。

また、国際標準化に向けて、 ISO/IEC JTC 1/SC27 (18033 Encryption Standards) に提案されている [7]。

以下に暗号アルゴリズム SEED についての安全性評価を示す。本報告書では、ブロック暗号に対する汎用的な攻撃方法として、差分解読法、線形解読法、及び代数的解読法についての安全性評価を行なう。差分解読法、線形解読法については、自己評価書に記載されている評価の妥当性を検討し、さらなる検討方針について述べる。代数的解読法については高階差分攻撃と補間攻撃に対する耐性を一般的な手法で評価する。また、 SEED に関する文献調査も行ない、近日発表される差分解読についての結果も本報告書に含めた。

2 差分解読法

2.1 自己評価書の記述内容の妥当性検証

自己評価書における差分解読法に対する SEED の安全性評価の方針は、以下のようになっている。

1. F 関数内に存在する加算を全て排他的論理和に置き換えた暗号 modified SEED を考え、 modified SEED に対して、高い差分特性確率をもつ差分特性を構成的に[1]発見する。
2. 上記で発見した最も差分特性確率の大きな差分特性に対して、加算演算部分で生じる差分確率を補正し、全体の差分特性確率を導出する。

この方法により、表 1に示すような差分特性が導出されている。

自己評価書では、 G 関数内にある block permutation の diffusion 効果により、 1 段消去型攻撃までしか適用できないと述べられている。一方、上記解析により、差分特性確率が 2^{-128} より大きな 15 段 SEED の差分特性は存在しそうにないことから、フルスペックである 16 段 SEED は差分解読法に対して安全であると結論づけられている。

[1] 自己評価書を読む限り、網羅的に差分経路探索を行なっているようには見えず、試行錯誤で高い確率を持つ差分特性を構成しているように見うけられる。

段数	modified SEED	SEED
6	2^{-102}	2^{-130}
7	2^{-108}	2^{-144}

表 1: 自己評価書に示されている SEED の最良差分特性

自己評価書での差分解析では、加算を排他的論理和に置き換えて差分特性を探索することにより、加算の出力差分値を限定してしまっているという問題点がある。

加算 $x + y = z \pmod{2^{32}}$ の 2 つの入力差分値を $(\Delta x, \Delta y)$ とすると、

$$\Pr_{x,y}[(x+y) \oplus ((x \oplus \Delta x) + (y \oplus \Delta y)) = \Delta z] > 0$$

となるような出力差分値 Δz は $\Delta x \oplus \Delta y$ 以外にも存在することが知られている [4]。よって、上記で探索されなかった差分経路の中に、より高い確率を持つものが存在する可能性がある。

2.2 最大差分特性確率の評価

屋並ら [6] はこの問題点を改良し、計算量の観点から若干、差分値のパターンに制限を加えているものの、計算機によりほぼ網羅的な差分経路探索を行なっている。

屋並らの探索における差分値パターンの制限は、ワード差分値を 0x00 または 0x80 の値を取るものとする、というもので、差分値のハミング重みが小さい方が加算の差分確率が大きい傾向にあるという性質と加算の差分特性での最上位ビットの特殊性を利用したものである。さらに、block permutation は 4 つのワードの同じビットにのみ影響を与えるという性質も考慮されている。詳細は [6] を参照のこと。

差分経路探索の結果、6 段 SEED で差分特性確率が 2^{-124} および 2^{-128} の差分特性が見つかっている。自己評価書の評価結果との比較を表 2に示す。

段数	自己評価書の評価結果	屋並らの評価結果
6	2^{-130}	2^{-124}
7	2^{-144}	–

表 2: 自己評価書の評価結果 v.s. 屋並らの評価結果

文献 [6] によると、この 6 段差分特性を用いて、1 段消去型攻撃により、7 段の SEED を 2^{126} 組の選択平文 - 暗号文対と 2^{126} 回の F 関数計算で攻撃可能である。自己評価書の結果からは 6 段までしか攻撃可能でなかったが、これにより、1 段攻撃可能段数が増えた。

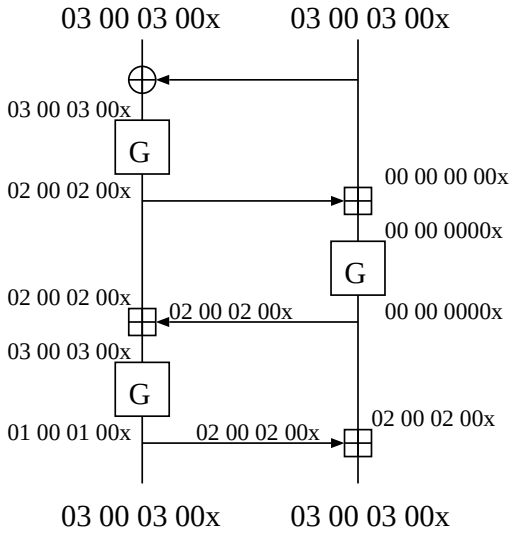

図 1: 自己評価書に示されている F 関数の最良線形特性

さらに高い差分確率を導出できる可能性としては、マルチプルパスの検討 (F 関数内及び段間で複数の入出力差分値を考慮して差分確率を計算する)、差分値の制限パターンをより緩くする、などの方針が考えられる。

3 線形解読法

3.1 自己評価書の記述内容の妥当性検証

自己評価書における線形解読法に対する SEED の安全性評価の方針は、なるべく高い線形特性確率をもつ繰り返し型の線形特性をアドホックに発見して、その線形特性確率から SEED の安全性を評価するというものである。しかし、差分解読法における自己評価と同様に、網羅的に線形経路探索を行なっているようには見えず、試行錯誤で高い確率を持つ線形特性を構成しているように見うけられる。

図 1に、自己評価書に示されている F 関数の線形特性を示す。この線形特性の線形特性確率は $7^8/2^{56} \sim 2^{-33.54}$ である。この確率の導出には 32 ビット加算演算の線形確率を計算する必要があるが、これは文献 [5] で示されているアルゴリズムを用いて計算されている。

自己評価書では、図 1に示した F 関数の線形特性が最良線形特性であると予想されている (p.15 第 3 パラグラフ)。また、この線形特性は繰り返し繋げることで長い段数の線形特性が得られる「繰り返し線形特性」である。

よって、下記の F 関数の線形特性を順に O, A とすると、表 3に示すような線形特性が構成できる。

O : 00 00 00 00_x 00 00 00 00_x $\xleftarrow{F}$ 00 00 00 00_x 00 00 00 00_x (確率 1)
A : 03 00 03 00_x 03 00 03 00_x $\xleftarrow{F}$ 03 00 03 00_x 03 00 03 00_x (確率 $2^{-33.54}$)

段数	線形特性	線形特性確率
4	O-A-A-O	$2^{-67.08}$
5	O-A-A-O-A	$2^{-100.62}$
6	O-A-A-O-A-A	$2^{-134.16}$
⋮	⋮	⋮
15	O-A-A-O-A-A- $\cdots$ -O-A-A	$2^{-335.4}$

表 3: 自己評価書に示されている SEED の最良線形特性

自己評価書では、表 3に示した 15 段の線形特性が最大線形特性確率 ($2^{-335.4}$) をもつと予測できるという主張のもとに、フルスペックの 16 段 SEED は線形解読法に対して安全であると結論づけている。

しかし、線形解読法において 2^{-128} より小さな確率をもつ線形特性は意味を持たないため、このような 15 段線形特性の存在を根拠に安全性を議論するのは適切でない。6 段以上で 2^{-128} より大きな確率をもつ線形特性が見つかっていないことを根拠に、フルスペックの 16 段 SEED は線形解読法に対して十分な安全性のマージンがあるというべきであろう。

3.2 最大線形特性確率の評価

さらに高い線形確率を導出できる可能性としては、マルチプルパスの検討 (F 関数内及び段間で複数の入出力差分値を考慮して線形確率を計算する) などの方針が考えられる。

4 高階差分攻撃

提出された自己評価書には高階差分攻撃に対する安全性評価は記述されていない。よって、本報告書では文献 [2] に基づき、暗号文の各ビットを平文ビットのブール多項式で表現した時の代数次数を見積もることで安全性評価を行なう。

4.1 G 関数の解析

まず、G 関数に含まれる S-box S_1,S_2 の次数を調べる。

S_1,S_2 は以下のように定義されている。

$$S_1 \quad : \quad Z_{2^8} \to Z_{2^8}, \quad S_1(x) = A^{(1)} \cdot x^{247} \oplus b_1$$

$$S_2 \quad : \quad Z_{2^8} \to Z_{2^8}, \quad S_2(x) = A^{(2)} \cdot x^{251} \oplus b_2$$

但し、$A^{(i)} \cdot x^{n_i} \oplus b_i,\ (i = 1, 2)$ は x^{n_i} のアフィン変換である。

$\mathrm{GF}(2^8)$ 上の冪乗関数 $x^{247},\ x^{251}$ は、$\mathrm{GF}(2^8)$ 上の逆関数 $x^{-1} = x^{254}$ と非退化な $\mathrm{GF}(2)^8$ 上線形関数との合成関数であり、各出力ビットを入力のブール多項式で表現した時の次数はどの出力ビットについても等しく、7 であることが知られている。

S_1,S_2 の定義のように、アフィン変換 $A^{(i)} \cdot x^{n_i} \oplus b_i,\ (i = 1, 2)$ を加えても、各出力のブール多項式表現には 7 次項が存在することを確認した。よって、S-box S_1,S_2 の次数は 7 である。

次に G 関数の diffusion 層 (block permutation) の影響を検討する。出力各ビット単位で見ると、block permutation では 3 つの S-box からの出力結果の排他的論理和を計算している。もし、攻撃者が 1 段目の F 関数への入力をコントロールでき、2 つの S_1,S_2 へそれぞれ同じ変数を入力できたとしても、2 つの S-box からの出力された 7 次項がキャンセルされることはあっても、残り 1 つの S-box からの出力された 7 次項は残るため、block permutation により 7 次項が消滅することはない。この結果、block permutation によって次数が下がることはなく、G 関数のどの出力においても、入力について 7 次のブール多項式で表されることが分かる。

4.2 F 関数の解析

F 関数には 3 つの G 関数と 3 つの 32 ビット加算が含まれる。加算では、出力ビットのブール多項式は上位になるほどキャリー伝播の影響により入力に関する次数が高くなる。最初の G 関数を通過した直後では次数は全て 7 であるが、次の加算の通過後の次数は、最下位ビットで 7 次、最上位ビットでは 63 次になる。その後の G 関数で多くの出力ビットに 63 次項が現れ、その後、加算、G 関数、加算を通過すると F 関数の全ての出力ビットの次数が 63 になると考えられる。(F 関数は 64 ビット入出力の全単射関数であるので最高 63 次までとりうる。)

4.3 SEED 全体の解析

文献 [2] 等に見られる典型的な高階差分攻撃の適用法は、平文の右 64 ビット (1 段目の F 関数への入力) を定数とし、暗号文を平文の左 64 ビットに対応する 64 変数についてのブール多項式で表すという方法である。これを SEED に適用した場合、3 段目で 64 次項が現れ、高階差分攻撃は不可能となる。

平文の右 64 ビットを定数とし、左 64 ビットを $(x_{32}, \ldots, x_1,\ x_{32}, \ldots, x_1)$ のように左右 32 ビットずつ同じ値をとるように変数を設定した場合は (この場合、2 段目の最初の G 関数への入力が定数となる)、2 段目で 32 次項が現れ、攻撃不可能となる。

以上の考察で全ての場合が尽くされているわけではないが、上記で挙げた事例により、16 段 SEED は高階差分攻撃に対し十分な耐性を持っていると考えられる。

5 補間攻撃

補間攻撃は、暗号文と平文の関係を (GF(2^8) などの) ある体上での多項式または有利表現として表した時に、その多項式に含まれる項数が攻撃者によって得られる平文-暗号文の組数より少ない場合に攻撃可能となる。

SEED への適用を考えた場合、G 関数に含まれる S-box は GF(2^8) 上の関数であり、F 関数に含まれる加算は法を 2^{32} とする加算であるというように、異なる代数体上の演算が組み合わせられている。よって、どのような体上での多項式または有利表現として表しても、非常に複雑な (= 項数の多い) 表現になることが予想され、補間攻撃の適用は困難と思われる。

6 まとめ

本報告書では、暗号アルゴリズム SEED の安全性評価として、ブロック暗号に対する汎用的な攻撃方法として、差分解読法、線形解読法、及び代数的解読法として高階差分攻撃と補間攻撃に対する耐性を評価して報告した。以上の解析により、SEED の攻撃可能な最長段数は 7 段で、2^{126} 組の選択平文・暗号文対と 2^{126} 回の F 関数を用いる差分攻撃である。

さらなる評価の余地はあるものの、現時点では、仕様通りの 16 段 SEED を解読する方法は存在せず、SEED は既知の代表的な攻撃法に対して適切な安全性マージンを有していると考えられる。

参考文献

[1] Eli Biham, Adi Shamir, "Differential Cryptanalysis of the Data Encryption Standard," Springer-Verlag, 1993.

[2] Thomas Jakobsen, Lars Ramkilde Knudsen, "The Interpolation Attack on Block Cipher," Fast Software Encryption — 4th International Workshop, FSE'97, Lecture Notes in Computer Science 1267, pp. 28–40, Springer-Verlag, 1997.

[3] Lars Ramkilde Knudsen, "Truncated and Higher Order Differentials," Fast Software Encryption — Second International Workshop, Lecture Notes in Computer Science 1008, pp.196–211, Springer-Verlag, 1995.

[4] Helger Lipmaa, Shiho Moriai, "Efficient Algorithms for Computing Differential Properties of Addition," Preproceedings of Fast Software Encryption Workshop 2001, pp.347-361, 2001.

[5] Sangwoo Park, Seongtack Chee, and Soo Hak Sung, "Linear approximation of integer additions," Electronics Letters, Vol.35, No.24, pp.2105–2106, November 1999.

[6] 屋並仁史, 下山武司, 「SEED の差分攻撃」 2002 年暗号と情報セキュリティシンポジウム, 2002 年 1 月 29 日 –2 月 1 日. (Private communications, 2001 年 12 月)

[7] Korean National Body, "Contribution for Korean Candidates of Encryption Algorithm (SEED)", related to ISO/IEC JTC1 SC27 N2563, 2000. http://www.kisa.or.kr/seed/data/algorithm/seed_english.doc.

[8] Korean Cryptography Standards, http://dosan.skku.ac.kr/~sjkim/kg_std.html.

A　自己評価書で修正を推奨する箇所

本評価作業中に気がついた自己評価書の誤植等を挙げておく。自己評価書はほぼそのままの内容が [7] で公開されているため、提案者により修正されることをお勧めする。

技術的コメント

- 1.1.2 章 (modified SEED の差分解読) p.2 最終行から始まる段落で、差分攻撃に利用できそうな差分特性が示されているが、その根拠や導出過程が明記されておらず、論理が追いにくい。
- 上記段落中に which makes the number of S-boxes become 4 とあるが、 which makes the number of **active** S-boxes become 4 の意味か？
- 1.1.3 章 (SEED の差分解読) p.8 中に addition mod 256 という表現が 3 箇所あるが、 32 ビット加算であるので addition mod 2^{32} の誤りと思われる。
- 1.2.2 章 (線形解読) p.11 中に at least more than two G functions should be active and at least more than two additions should be active という記載があるが、 at least two G functions should be active and at least two additions should be active が正しいと思われる。

自己評価書全体を通して

- 英文技術論文として、改行位置が適切でないと思われるので修正してはどうか。例えば、単語の途中 (通常のハイフネーション位置と異なる位置) での改行や、数式において指数の直前での改行などが頻出しているため、読みづらい。

誤植と思われる箇所

- p.3 下から 3 行目の of の直後にピリオドがある。
- p.8 上から 3 行目のピリオドは 2 行目末に。
- 1.1.3 章 第 4 パラグラフの 1 行目と 5 行目のカンマの後にスペースが入ると読み易くなると思われる。
- p.13 上から 6 行目の $00_x \xleftarrow{S_2} 00_x$ は $00_x \xleftarrow{S_1} 00_x$ の誤植ではないか。
- 同様に p.13 上から 12 行目の $00_x \xleftarrow{S_2} 00_x$ も $00_x \xleftarrow{S_1} 00_x$ の誤植ではないか。
- p.18 表 1.3-1 の 9 段目から 16 段目の $K_{i,0}$ の最初のバイトが大文字で表されているが、小文字の誤植と思われる。