ANALISIS *UNDISTURBED BIT* PADA KONSTRUKSI S-BOX *QUASIGROUP* 4×4

Caesario Oktanto Kisty[1)], Ryan Setyo Pambudi[2)],Sri Rosdiana[3)]
[1,2,3]Sekolah Tinggi Sandi Negara
Jl. Raya Haji Usa, Desa Putat Nutug, Kec. Ciseeng, Jawa Barat,
E-mail : caesario.oktanto@student.stsn-nci.ac.id[1)]

Abstrak

S-Box merupakan komponen yang sering menjadi sasaran utama ketika melakukan differential cryptanalysis pada algoritma block cipher. Tezcan melakukan impossible differential cryptanalysis pada algoritma PRESENT dengan menggunakan teknik pencarian undisturbed bit pada S-Box. Diketahui adanya undisturbed bit pada sebuah S-Box dapat diperoleh round yang lebih banyak ketika dilakukan impossible differential cryptanalysis dibandingkan jika tidak menggunakan undisturbed bit. Oleh sebab itu, pada penelitian ini akan dilakukan analisis undisturbed bit pada konstruksi S-Box quasigroup 4×4. Leader yang digunakan untuk mengonstruksi S-Box quasigroup 4×4 adalah $l_1 l_2 l_1 l_2$. Penelitian ini dapat menghasilkan S-Box yang dapat mencegah impossible differential cryptanalysis dengan round yang lebih banyak pada sebuah algoritma Block Cipher.

Kata kunci: *Undisturbed bit, S-Box Quasigroup 4×4, Lightweight Block Cipher, Impossible Differential Cryptanalysis.*

Abstract

S-box in the block cioher algorithm usually becomes a main target to the differential cryptanalysis.Tezcan did the impossible differential cryptanalysis on PRESENT algorithm by finding the undisturbed bit against the S-box. Undisturbed bit on the S-box will helps us to find more amount of round at the algorithm that can be exploited by impossible differential cryptanalysis better than not. Therefore, in this paper we present an undisturbed bit analysis on S-box construction based quasigroup 4×4.We use a Leader to construct 4×4 S-Box based quasigroup, there are $l_1 l_2 l_1 l_2$. The result of this research is producting a S-Box that minimize the amount round can be attacked by impossible differential cryptanalysis on Block Cipher algorithm.

Kata kunci: *Undisturbed bit, S-Box Quasigroup 4×4, Lightweight Block Cipher, Impossible Differential Cryptanalysis.*

1. PENDAHULUAN

Dewasa ini, perkembangan perangkat komputer menuntut peningkatan kecepatan dan efisiensi dalam pemrosesan. Selain itu dibutuhkan pula tingkat keamanan yang baik. Salah satu hal utama yang mempengaruhi hal terebut adalah dari algoritma yang digunakan. *Lightweight Cryptography* (LC) merupakan algoritma kriptografi yang didesain agar perangkat memiliki kecepatan pemrosesan yang tinggi, efiien dan memiliki kekuatan kriptografis yang baik. Sampai saat ini standarisasi tentang LC sudah diatur pada ISO/IEC 29192. Salah satunya yang diatur adalah tentang *Lightweight Cryptography: Block Ciphers*. Algoritma *block cipher* PRESENT [1] dan CLEFIA [2] menjadi standar algoritma *lightweight block cipher*.

Dalam algoritma *lightweight block cipher*, salah satu komponen utama dalam menunjang keamanan algoritma tersebut ialah *Substitution Box* (S-Box). S-Box merupakan sebuah fungsi pemetaan yang memetakan n-bit *input* menjadi m-bit *output* [3]. Penggunaan S-Box bertujuan untuk mengubah bit *input* menjadi bit *output* acak sehingga fungsi S-Box tersebut harus sulit dibentuk pendekatan linearnya. Fungsi S-Box juga harus tahan dari berbagai jenis serangan seperti kriptanalsis linear, kriptanalisis diferensial, dan serangan aljabar.

Untuk membentuk sebuah S-Box, terdapat beberapa cara yang dapat dilakukan. Adapun beberapa cara tersebut adalah menggunakan *pseudo random number generator* (PRNG), *random* berdasarkan uji, manual berdasarkan metode matematika sederhana dan pendekatan matematika [4]. Salah satu metode konstruksi S-Box dengan pendekatan matematika adalah dengan penerapan aljabar *quasigroup* [5]. Pada [5], dilakukan pembentukan S-Box dengan menggunakan 4 buah *round e-transformation* dan 2 buah *leader* dengan pola masukkan *leader* $l_1 l_2 l_1 l_2$. Dari proses tersebut dihasilkan S-Box 4×4 *quasigroup* sebanyak 6912 buah dan terdapat 1152 buah S-Box yang dikatakan sebagai S-Box optimal. Kritera S-Box optimal adalah bijektif serta nilai linearitas dan diferensial sebesar ¼.

Algorima *block cipher* PRESENT [1] merupakan salah satu algoritma *lightweight block cipher*. PRESENT berkerja dengan mengenkripsi 64 bit teks terang menjadi 64 bit teks sandi. Ukuran kunci yang digunakan terdapat dua buah jenis yaitu 80 dan 128 bit. Struktur dalam PRESENT terdiri dari *addRoundkey*, *SubstitutionLayer*, dan *PermutationLayer*. *AddRoundKey* merupakan proses XOR dengan *subkey* pada setiap *round*. *SubstitutionLayer* direpresentasikan dalam bentuk S-Box berukuran 4×4 . *PermutationLayer* merupakan permutasi berbasis bit.

Berbagai serangan telah banyak dilakukan terhadap PRESENT. Diantaranya adalah *multidimensional linear cryptanalysis* [6], *differential cryptanalysis* [7], *truncated differential cryptanalysis* [8], dan *improbable differential cryptanalysis* [9]. *Improbable differential cryptanalysis* pada PRESENT dilakukan dengan menggabungkan *differential characteristic* dan *impossible differential characteristic*. Pencarian karakteristik *impossible differential* dilakukan dengan teknik pencarian *undisturbed bit* pada S-Box PRESENT. Sehingga didapatkan karakteristik sepanjang 6 *round*. Sebelumnya dilakukan pencarian karakteristik tanpa menggunakan *undisturbed bit*, tetapi jumlah *round* paling panjang yang dapat ditemukan hanya sepanjang 4 *round*.

Ketika spesifik nilai *difference* diberikan pada *input* dan nilai *difference* yang setidaknya terdapat satu bit *output* pada S-Box dapat ditebak dengan probabilitas 1, dan berlaku sebaliknya, maka hal ini disebut *undisturbed bit* [9]. Keberadaan *undisturbed bit* pada S-Box membantu penyerang untuk mengonstruksi *truncated* atau *impossible differential* dengan *round* yang lebih panjang. Sehingga hal tersebut harus diperhatikan oleh pendesain S-Box agar tahan terhadap serangan tersebut. Pada S-Box PRESENT terdapat 6 buah *undisturbed bit* sehingga dapat dimanfaatkan pada serangan *impossible differential*, seperti yang telah dijelaskan pada paragraf sebelumnya.

Oleh karena itu, adanya *undisturbed bit* pada suatu S-Box mempengaruhi ketahanan S-Box itu sendiri. Salah satu kasusnya adalah S-Box pada algoritma PRESENT. Sehingga pada penelitian ini, dilakukan konstruksi S-Box melalui metode *quasigroup*. Kemudian, hasil dari S-Box yang telah dibangkitkan tersebut diperiksa keberadaan *undisturbed bit*. S-Box yang memiliki *undisturbed bit* akan dibuang. S-Box yang tidak memiliki *undisturbed bit* kemudian diseleksi dan diambil yang memenuhi kriteria S-Box optimal. Tujuan penelitian ini adalah memberikan rekomendasi S-Box yang tahan terhadap *impossible differential cryptanalysis* berdasarkan *undisturbed bit*.

2. LANDASAN TEORI

2.1 *Quasigroup*

Nyatakan $(Q,*)$ sebagai grupoid dengan sebuah operasi biner $*$ pada sebuah himpunan tak kosong Q dan $a, b \in Q$.

Definisi 1: Grupoid merupakan sebuah himpunan berhingga Q yang memiliki sebuah operasi biner $*$, sehingga untuk seluruh nilai $a, b \in Q$, $a * b \in Q$. Dengan kata lain, nilai pada Q hanya memiliki sifat tertutup pada operasi biner $*$ [4].

Definisi 2: Sebuah grupoid $(Q,*)$ dikatakan *quasigroup* jika untuk semua pasangan $(a, b) \in Q^2$ terdapat solusi berupa $x, y \in Q$ pada persamaan berikut [5]:

$$(x * a = b \; dan \; a * y = b). \quad (1)$$

Setiap *quasigroup* dapat direpresentasikan sebagai sebuah tabel perkalian atau tabel Cayley. Order pada sebuah *quasigroup* $(Q,*)$ merupakan kardinalitas dari $|Q|$ pada himpunan tak kosong Q. Pada penelitian ini menggunakan *quasigroup* berorder 4. Berikut ini adalah contoh sebuah *quasigroup* $(Q,*)$ berorder 4. Misal $Q = \{0,1,2,3\}$, sehingga *quasigroup* memiliki bentuk tabel cayley sebagai berikut:

Tabel 1. Tabel cayley *quasigroup* berorder 4

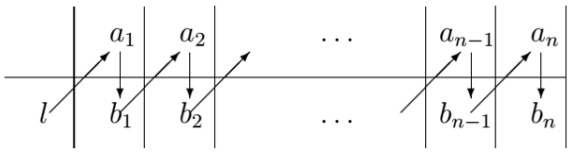
*	0	1	2	3
0	0	1	3	2
1	1	0	2	3
2	2	3	0	0
3	3	2	1	1

Pada konstruksi S-Box yang akan dijelaskan pada subbab 2.2, metode yang digunakan adalah menggunakan transformasi *quasigroup* dengan jenis *e-transformation*. Tentang jenis-jenis transformasi *quasigroup* dapat dilihat pada [10].

Misal Q merupakan himpunan elemen-elemen ($|Q| \geq 2$) dan kita nyatakan bahwa $Q^r = \{a_1, a_2, \ldots, a_r | a_i \in Q, r \geq 2\}$ himpunan dari seluruh *string* berhingga dengan elemen-elemen dari Q. Asumsikan bahwa $(Q,*)$ merupakan sebuah *quasigroup* yang diberikan, dengan elemen tetap $l \in Q$ adalah *leader*, maka *e-transformation* $e_l: Q^r \rightarrow Q^r$ dapat dinyatakan sebagai berikut:

$$e_l(a_1, \ldots, a_r) = (b_1, \ldots, b_r) \Leftrightarrow \begin{cases} b_1 = l * a_1 \\ b_i = b_{i-1} * a_i, 2 \leq i \leq r \end{cases} \quad (2)$$

Dari persamaan (2), dapat direpresentasikan ke dalam bentuk grafik yang dapat dilihat pada Gambar 1 berikut ini:



Gambar 1. Skema e-transformation [5]

Apabila terdapat beberapa rangkaian inisial *leader* $l_1, l_2, \ldots, l_k$ maka penerapan *e-transformation* bisa dilakukan secara beruntun. Komposisi dari *e-transformation* disebut transformasi gabungan *quasigroup*. Transformasi gabungan ini diperoleh hanya dari komposisi *e-transformation* saja yang dinotasikan sebagai E. Berikut adalah definisi dari transformasi gabungsn *e-transformation*:

$$E = E^{(k)}_{l_k,\ldots,l_1} = e_{l_k} \circ e_{l_{k-1}} \circ \ldots \circ e_{l_1} \quad (3)$$

2.2 Konstruksi S-Box *Quasigroup* 4×4

Berikut ini adalah contoh konstruksi S-Box *quasigroup* 4×4 menggunakan *e-transformation*. Misalkan dua buah *leader* pada *quasigroup* berorder 4, yaitu $l_1 = 1 = 01$ dan $l_2 = 3 = 11$. Dengan menggunakan *leader pattern* $l_1 l_2 l_1 l_2$ dan *quasigroup* berorder 4 seperti pada tabel 2, maka S-Box *quasigroup* 4×4 dapat dibentuk.

Tabel 2. Q*uasigroup* berorder 4

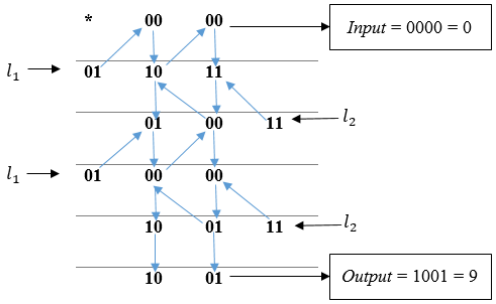
*	0	1	2	3
0	0	2	1	3
1	2	0	3	1
2	3	1	0	2
3	1	3	2	0

Dikarenakan S-Box yang akan dihasilkan berukuran 4×4 maka terdapat 16 buah nilai *input* yang digunakan, yaitu 0000,0001, …,1111 yang merupakan representasi bit dari $0_x, 1_x, \ldots, F_x$. Dengan menerapkan persamaan (2), maka untuk setiap *input* dapat diselesaikan dengan:

$$e_{01}(00,00) = (10,11) \Leftrightarrow \begin{cases} 10 = 01 * 00 \\ 11 = 10 * 00 \end{cases}$$
$$e_{11}(11,10) = (00,01) \Leftrightarrow \begin{cases} 00 = 11 * 11 \\ 01 = 00 * 10 \end{cases}$$

$$e_{01}(01,00) = (00,00) \Leftrightarrow \begin{cases} 00 = 01 * 01 \\ 00 = 00 * 00 \end{cases}$$
$$e_{11}(00,00) = (01,10) \Leftrightarrow \begin{cases} 01 = 11 * 00 \\ 10 = 01 * 00 \end{cases}$$

Dari hasil tersebut, proses dapat direpresentasikan pada sebuah grafik yang dapat dilihat pada Gambar 2.



Gambar 2. Representasi proses e-transformation

Sehingga setelah setiap *input* dilakukan proses yang sama seperti proses diatas, akan diperoleh sebuah S-Box *quasigroup* 4×4 yang dapat dilihat pada Tabel 3. Pada *leader pattern* $l_1 l_2 l_1 l_2$ akan dihasilkan S-Box *quasigroup* 4×4 sebanyak 6912 buah.

Tabel 3. S-Box yang dihasilkan

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$S(x)$	9	5	1	D	C	8	0	4	2	6	A	E	7	B	3	F

2.2 *Undisturbed Bit*

Ketika spesifik nilai *difference* diberikan pada *input* dan nilai *difference* yang setidaknya terdapat satu bit *output* pada S-Box dapat ditebak dengan probabilitas 1, dan berlaku sebaliknya, maka hal ini disebut *undisturbed bit* [9]. Keberadaan *undisturbed bit* pada S-Box membantu penyerang untuk mengonstruksi *truncated* atau *impossible differential* dengan *round* yang lebih panjang.

Definisi 3: (*Undisturbed bit*) Misalkan $\overline{\alpha} \in F_2^n$ sebuah *input difference* tidak nol pada S-Box S (untuk *s-box* berukuran $n \times m$ $S: F_2^n \to F_2^m$) dan,

$$\Omega_{\overline{\alpha}} = \{\overline{\beta} = (\beta_{m-1}, \dots, \beta_0) \epsilon F_2^m \left| \mathbf{Pr}_S[\overline{\alpha} \to \overline{\beta}] > 0 \right.\} \quad (4)$$

menjadi himpunan dari semua kemungkinan *output difference* dari S yang berkorespondensi dengan $\overline{\alpha}$. Untuk nilai c yang tetap sehingga $c \in F_2$ dan untuk semua $\overline{\beta} \in \Omega_{\overline{\alpha}}$, jika $\beta_i = c$ dengan nilai $i \in \{0, \dots, m-1\}$, maka S-Box S memiliki *undisturbed bit*. Untuk *input difference* $\overline{\alpha}$, bit ke-i dari *output difference* S adalah *undisturbed* (dan nilainya adalah c).

Pada S-Box PRESENT terdapat 6 buah *undisturbed bit* sebagai berikut [9]:

1) Jika *input difference* dari *s-box* bernilai 9, maka LSB dari *output difference* adalah *undisturbed* yang bernilai 0.
2) Jika *input difference* dari *s-box* bernilai 1 atau 8, maka LSB dari *output difference* adalah *undisturbed* yang bernilai 1.
3) Jika *output difference* dari *s-box* bernilai 1 atau 4, maka LSB dari *input difference* adalah *undisturbed* yang bernilai 1.
4) Jika *output difference* dari *s-box* bernilai 5, maka LSB dari *input difference* adalah *undisturbed* yang bernilai 0.

Pencarian *undisturbed bit* pada S-Box dilakukan dengan mengamati *Differential Distribution Table* (DDT) pada S-Box tersebut. Berikut adalah lemma, teorema dan *corollary* yang berkaitan dengan pencarian *undisturbed bit*.

Lemma 1: *Untuk suatu nonzero input difference* $\bar{\alpha} \in F_2^n$, *bit ke-i dari output difference S itu undisturbed jika dan hanya jika* $r_{h_i}(\bar{\alpha}) = \pm 2^n$ *untuk* $i \in \{0, \dots, m-1\}$.

Teorema 1: *Hubungan antara DDT dan fungsi komponen dari autocorrelation* S, $r_{\bar{j}.S}(\bar{\alpha}) = \sum_{\bar{v}\in F_2^m} DDT(\alpha, v)\ (-1)^{\bar{j}\cdot\bar{v}}$ *untuk* $\bar{\alpha} \in F_2^n$ dan $\bar{j} \in F_2^m$.

Corollary 1: (DDT dan *Undisturbed* bit) *Untuk suatu nonzero input difference* $\bar{\alpha} \in F_2^n$, *bit ke-*i *dari output difference* S *itu undisturbed jika dan hanya jika,*

$$\sum_{\bar{v}\in F_2^m} DDT(\alpha, v)\ (-1)^{\overline{e_i}\cdot\bar{v}} = \pm 2^n \quad (5)$$

untuk $i \in \{0, \dots, m-1\}$ *dan* $\overline{e_i}$ *adalah basis standar ke-* i *dari* F_2^m.

2.3 *Optimum* S-Box

Definisi 4: Misalkan $S: F_2^n \to F_2^n$ merupakan sebuah S-Box berukuran 4×4 dengan *input* sebanyak 2^4. S dapat dikatakan sebagai S-Box optimal jika memenuhi kriteria berikut ini [5]:

1) S merupakan fungsi bijektif;
2) $Lin(S) = \frac{1}{4}$;
3) $Diff(S) = \frac{1}{4}$.

Untuk menghitung nilai linearitas pada S-Box dapat dengan menggunakan persamaan berikut,

$$Lin(S) = max\{\tfrac{1}{2^{2n}} S^W(u,v)^2 | u \in F_2^n, v \in F_2^n, (u,v) \neq 0\}, \quad (6)$$

dimana $S^W(u,v) = \sum_{x\in F_2^n} (-1)^{u.v+v.S(x)}$.

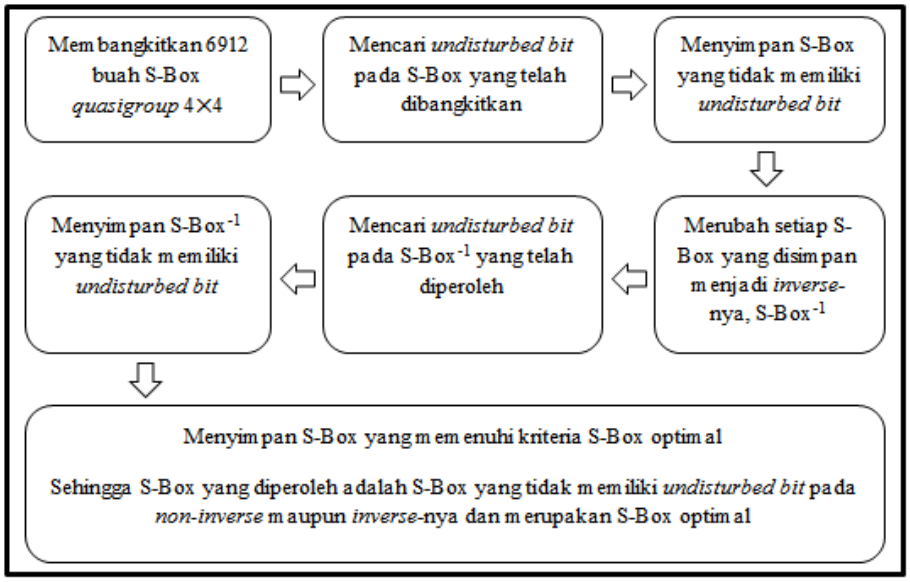
Kemudian, untuk menghitung nilai diferensial pada S-Box dapat dengan menggunakan persamaan berikut,

$$Diff(S) = \max\{\tfrac{1}{2^n} \Delta S(u,v) |\ u \in F_2^n, v \in F_2^n, (u,v) \neq 0\} \quad (7)$$

dimana $\Delta S(u,v) = |\{x \in F_2^n | S(x \oplus u) = S(x) \oplus v\}|$.

3. METODOLOGI PENELITIAN

Berikut ini adalah tahapan penelitian yang dilakukan:



Gambar 3. Tahapan penelitian

4. HASIL PENELITIAN

Pada bab ini akan ditunjukkan hasil penelitian yang dilakukan oleh penulis. Berikut ini adalah hasil S-Box *quasigroup* 4×4 yang telah dikonstruksi.

Tabel 4. Hasil konstruksi S-Box

Input	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S_1	0	4	D	9	5	1	8	C	B	F	2	6	E	A	7	3
S_2	1	5	C	8	4	0	9	D	A	E	3	7	F	B	6	2
S_3	3	7	E	A	6	2	B	F	9	D	0	4	C	8	5	1
⋮	⋮															
S_{6912}	B	F	7	3	4	D	8	9	E	A	C	5	1	6	0	2

Selanjutnya S-Box yang telah dikonstruksi tersebut diseleksi sesuai dengan tahapan penelitian yang telah dijabarkan pada bab 3. Setiap dari S-Box dicari keberadaan *undisturbed bit* nya. S-Box yang memiliki *undisturbed bit* akan dibuang. Sedangkan yang tidak memiliki *undisturbed bit* digunakan untuk proses selanjutnya. Kemudian hasil dari penyaringan yang pertama, setiap S-Box dicari keberadaan *undisturbed bit* pada posisi *inverse*-nya. Hal ini bertujuan untuk ketika mencari karakteristik *impossible differential* pada proses dari bawah ke atas (dekripsi). Sama seperti sebelumnya, S-Box *inverse* yang tidak memiliki *undisturbed bit* akan disimpan untuk proses selanjutnya. Terakhir adalah menyeleksi S-Box yang memenuhi kriteria sebagai S-Box optimal. Sehingga diperoleh sebanyak 689 buah S-Box *quasigroup* 4×4 yang tidak memiliki *undisturbed bit*. Namun semua S-Box tersebut tidak ada yang memenuhi kriteria S-Box optimal. Berikut ini adalah hasil S-Box *quasigroup* 4×4 yang tidak memiliki *undisturbed bit*.

Tabel 5. Hasil S-Box yang tidak memiliki *undisturbed bit*.

Input	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S_1	0	3	7	8	C	B	D	1	9	4	5	F	E	2	A	6
S_2	A	1	5	2	E	F	9	3	D	6	7	B	C	8	0	4
S_3	F	C	4	B	6	A	0	7	8	E	3	2	1	D	9	5
⋮	⋮															
S_{689}	B	F	7	3	4	D	8	9	E	A	C	5	1	6	0	2

Berikut ini adalah contoh pencarian *undisturbed bit* pada S-Box S_1 berdasarkan *input difference* dan *output difference*.

Tabel 6. Pencarian *undisturbed bit* pada S-Box S_1 berdasarkan *input difference.*

		Input Difference														
		1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Bit ke-i	0	0	0	8	0	-8	-8	0	-8	0	0	-8	0	8	0	0
	1	0	0	8	0	-8	-8	0	-8	0	0	-8	0	8	0	0
	2	-8	0	0	0	0	8	-8	0	0	-8	0	-8	8	0	0
	3	-8	0	0	-8	0	0	0	8	-8	0	0	-8	8	0	0

Tabel 7. Pencarian *undisturbed bit* pada S-Box S_1 berdasarkan *output difference*.

		Output Difference														
		1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Bit ke-i	0	-8	0	0	0	0	-8	8	-8	8	0	0	0	0	0	-8
	1	-8	8	-8	-8	0	-8	8	0	0	0	0	0	0	0	0
	2	0	-8	0	0	-8	0	8	0	-8	0	0	8	0	-8	0
	3	0	-8	0	0	-8	0	0	0	-8	0	8	8	0	-8	0

Berdasarkan Tabel 6 dan 7, dapat dilihat bahwa tidak ditemukan nilai ± 16. Sehingga dapat dikatakan bahwa S-Box S_1 tidak memiliki *undisturbed bit*. Namun S-Box S_1 bukan lah termasuk S-Box optimal. Hal tersebut dikarenakan S-Box tersebut memiliki nilai $Lin(S_1) = \frac{9}{16}$ dan $Diff(S_1) = \frac{1}{2}$.

4. SIMPULAN DAN SARAN

4.1 Simpulan

Berdasarkan tahapan dan hasil penelitian yang telah dilakukan, terdapat beberapa simpulan yang dapat diambil. Konstruksi S-Box dengan *quasigroup* menggunakan *leader pattern* $l_1 l_2 l_1 l_2$ menghasilkan 6912 S-Box *quasigroup* 4×4. Kemudian setelah dilakukan pencarian *undisturbed bit* pada setiap S-Box, terdapat 689 S-Box yang tidak memiliki *undisturbed bit*. Namun S-Box tersebut tidak ada yang memenuhi sebagai S-Box optimal. Sehingga dapat dikatakan S-Box tersebut dapat menjadi alternative untuk algoritma PRESENT agar tahan terhadap *impossible differential cryptanalysis* tidak lebih dari 6 *round*. Namun dibutuhkan penelitian lebih lanjut dengan langsung menerapkan *impossible differential cryptanalysis* agar dapat terbukti dengan pasti ketahanannya. Walaupun dapat dikatakan tahan terhadap *impossible differential cryptanalysis*, namun S-Box tersebut rentan terhadap *linear cryptanalysis* dan *differential cryptanalysis* karena tidak ada yang memenuhi kriteria S-Box optimal.

4.2 Saran

Berikut ini adalah saran-saran yang dapat digunakan untuk penelitian selanjutnya:

1) Dapat dilakukan *impossible differential cryptanalysis* pada penerapan S-Box yang dihasilkan di algoritma PRESENT, untuk mengetahui ketahanannya secara pasti.
2) Perlu dilakukan menggunakan *leader pattern* lainnya untuk mencari kemungkinan adanya S-Box yang tidak memiliki *undisturbed bit* dan memenuhi kriteria S-Box optimal.
3) Dapat menggunakan konstruksi S-Box yang lainnya untuk mencari S-Box yang tidak memiliki *undisturbed bit* dan memenuhi kriteria S-Box optimal.

5. REFERENSI

[1] Bogdanov, A. *et al*. 2007. *PRESENT: An Ultra-Lightweight Block Cipher*. Springer Berlin Heidelberg, Volume 4727, pp. 450-466.

[2] Sony Corporation: The 128-bit Blockcipher CLEFIA, Security and Performance Evaluations, Revision 1.0, June 1 (2007)

[3] Schneier, B., 1996. *Applied Cryptography, second edition.* New York: John Wiley and Sons, Inc.

[4] Nyberg, K., 1991. *Perfect Nonlinear S-Boxes*. *Springer-Verlag*, pp. 378-385.

[5] Mihajloska, H. & Gligoroski, D., 2012. *Construction of Optimal 4-bit S-Boxes by Quasigroups of Order 4*. The Sixth International Conference on Emerging Security Information, Systems and Technologies.

[6] Cho, J. Y. 2010. *Linear cryptanalysis of reduced-round PRESENT*. In Topics in Cryptology-CT-RSA 2010 (pp. 302-317).

[7] Wang, M. 2008. *Differential Cryptanalysis of PRESENT*. AFRICACRYPT 2008, Lecture Notes in Computer Science, vol. 5023, pp 40-49.

[8] Blondeau, C. & Nyberg, K. 2014. *Links Between Truncated Differential and Multidimensional Linear Properties of Block Ciphers and Underlying Attack Complexities*. In Advances in Cryptology—EUROCRYPT'14 (To Appear). Springer Berlin Heidelberg.

[9] Tezcan, C. 2014. *Improbable Differential Attack on PRESENT using Undisturbed Bits.* ELSEVIER, Volume 259, Part B, pp. 503-511.

[10] S. Markovski. 2003. *Quasigroup String Processing and Applications in Cryptography*. in The Proceedings of the 1st MII Conference, pp. 278–290.

Halaman ini sengaja dikosongkan