Perancangan Kriptografi *Block Cipher* Berbasis Pola Tarian Denok Deblong

Yuana Sambadha
Universitas Kristen Satya Wacana
Jl. Diponegoro No.52-60, Salatiga, Kec. Sidorejo, Kota Salatiga, Jawa Tengah
sambadhayuana@gmail.com

Abstract*— *The popularity of cybercrime caused unsafe information. Infrastructure IT made an effort to take in hand the problem, i.e. manipulating information. Cryptography existed as knowledge to protect information. To improve the security level, cryptography needs to be developed. Block Cipher Cryptography based on Dance Denok Deblong is designed to create a new cryptography. This cryptography is designed by using 4 process and 20 rounds. In fourth procesis transformed with S-BOX to get a more random ciphertext. Testing is also done using Avalanche Effect and Correlation value where the character change reaches 49,844%, so it can be used as an alternative in securing data.

Intisari— Maraknya cybercrime membuat informasi menjadi tidak aman. IT infrastruktur berupaya dalam menangani hal tersebut, yang salah satunya adalah dengan memanipulasi informasi. Kriptografi hadir sebagai ilmu dalam mengamankan suatu informasi. Untuk meningkatkan keamanannya, kriptografi perlu dikembangkan. Kriptografi Block Cipher Berbasis Pola Tarian Denok Deblong ini dirancang untuk membuat kriptografi baru. Kriptografi ini dirancang menggunakan 4 proses 20 putaran. Di proses ke 4 ditransfomasikan dengan S-BOX untuk mendapatkan ciphertext yang lebih acak Pengujian juga dilakukan menggunakan Avalanche Effect dan nilai Korelasi dimana terjadi perubahan karakter mencapai 49,844%, sehingga dapat digunakan sebagai alternatif dalam mengamankan data.

Kata Kunci*— *Block Cipher*, Kriptografi, Denok Deblong, S-BOX, *Avalance Effect

I. PENDAHULUAN

Di era Big Data seperti sekarang ini, sikap kehati-hatian yang diiringi dengan langkah check and recheck menjadi sebuah keharusan yang mesti dilakukan para pengguna internet. Pasalnya, ancaman terhadap penggunaan Internet dan semua konektivitasnya meningkat seiring meningkatnya penetrasi penggunaan Internet. Biaya yang timbul dari kerugian akibat pencurian data dan informasi penting baik milik perorangan, pemerintah dan swasta sangat besar, bisa mencapai sepuluh kali lipat. Apalagi saat ini bermunculan tren dimana perusahaan menjadi lebih bergantung pada cloud untuk meningkatkan kolaborasi dan fleksibilitas serta membuat transformasi digital menjadi lebih mudah. Meski demikian, keamanan data tetap menjadi prioritas utama. Karena itu, pentingnya kebijakan dan privasi menjadi dua hal yang tidak bisa dikesampingkan. Bahkan menurut Blue Coat Elastica Shadow Data Report baru-baru ini, 23% dari semua file dalam aplikasi cloud secara luas dibagikan, dan 12% dari file tersebut mengandung data yang sensitif dan yang terkait dengan kepatuhan. Dibutuhkan suatu sistem keamanan yang mampu menjaga kerahasiaan suatu data, sehingga data yang dikirim tetap aman [1].

Dalam hal ini Kriptografi hadir sebagai ilmu untuk menjaga kerahasiaan pesan/mengamankan informasi dengan mengubah informasi tersebut menjadi sandi yang susah dipahami. Pada kriptografi terdapat dua komponen utama yaitu enkripsi dan dekripsi, enkripsi merupakan proses merubah data asli (*Plaintext*) menjadi data acak yang tidak dapat dimengerti (*Ciphertext*) sedangkan dekripsi adalah kebalikan dari enkripsi yaitu merubah *ciphertext* menjadi bentuk semula *plaintext*. Algoritma dalam perancangan kriptografi ini menggunakan algoritma Block *Cipher* 64-¬bit yang di kombinaikan tabel substitusi Advanced Encryption Standard (S-BOX AES) dengan pola tarian Denok Deblong. Pemasukan bit pada blok-blok berjumlah 64-bit, yang dilakukan sebanyak 20 putaran dimana setiap putaran memiliki 4 proses palintext dan juga proses kunci (key). Hasil dari palintext akan di-XOR dengan kunci untuk menghasilkan *Ciphertext* untuk menghasilkan Avalanche *Effect* yang besar.

Denok Deblong merupakan salah satu tarian khas Kota Semarang yang diiringi gamelan Gambang Semarang, Denok yang merupakan sebutan khas remaja putri yang cantik untuk daerah Semarang, sedangkan Deblong adalah timangan (kudangan) dari sesesok ibu atau biyung kepada putrinya yang bermakna kecantikan dan kepandaian. Tari Denok Deblong menceritakan tentang keceriaan masa remaja putri yang cantik rupawan. Selain menceritakan tentang keceriaan seorang remaja putri yang cantik Tari Denok Deblong juga menceritakan agar para remaja putri ini besoknya menjadi putri yang berguna bagi orang tua, agama dan negaranya. Tarian ini dipilih karena keunikan gerakan pola tariannya yang memungkinkan untuk dijadikan menjadi blok blok 64 bit.

Berdasarkan latar belakang masalah yang ada, maka dilakukan penelitian tentang perancangan kriptografi block *cipher* 64-bit yang berbasis Perancangan Kriptografi *Block Cipher* Berbasis Pola Tarian Denok Deblong. Pola gerakan Tarian inilah yang digunakan untuk membuat rancangan kriptografi ini, yang nantinya digunakan sebagai pola pemasukan bit dan pengambilan bit di setiap blok. Dengan harapan agar pola tersebut dapat digunakan untuk menghasilkan korelasi terbaik sebagai proses enkripsi dan dekripsi dari sebuah pesan *plaintext*. Sehingga keamanan

data menjadi lebih kuat dan data dapat digunakan sebagaimana mestinya.

II. TINJAUAN PUSTAKA

Penelitian sebelumnya yang menjadi acuan dalam penelitian yang dilakukan, dijelaskan sebagai berikut, yang pertama adalah "Perancangan Kriptografi *Block Cipher* Berbasis pada Pola Formasi sepak Bola 3-5-2". Penelitian ini membahas tentang perancangan kriptografi *block cipher* berbasis 64-bit dengan proses putaran sebanyak 5(lima) kali pada proses enkripsi dan enkripsi [2].

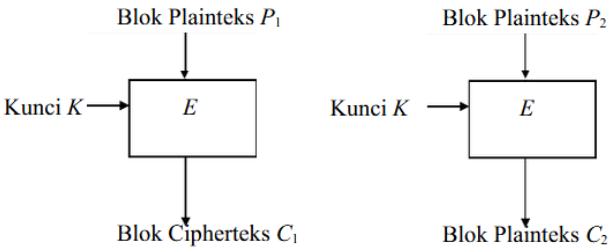
Penelitian kedua berjudul "Perancangan Kriptografi *Block Cipher* Berbasis pada Langkah Kuda". Penelitian ini membahas tentang perancangan kriptografi *block cipher* berbasis 64-bit menggunakan pendekatan pola Langkah kuda sebagai metode pemasukan bit *plaintext* pada blok matriks [3].

Penelitian ketiga "Kriptografi *Block Cipher* 256 Bit berbasis pola Tuangan Air". Perbedaan penelitian ini dengan yang sebelumnya dimana jumlah data yang diproses sebanyak 256bit dengan jumlah putaran sebanyak 20 kali, dan pada setiap putaran dikombinasikan dengan tabel subtitusi S-BOX [4].

Penelitian empat adalah Perancangan Algoritma Super Enkripsi Berbasis Pola 8-Queen of Fitness Chess, Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa perancangan algoritma super enkripsi berbasis pola 8-Queen of fitness chess ini dikatakan sebagai sistem kriptografi yang baik. Algoritma dalam proses enkripsi dan dekripsi ini menggunakan permutasi, kombinasi, transposisi, substitusi, random integer, dan optimalisasi korelasi terendah, sehingga dalam proses enkripsi menghasilkan *ciphertext* yang acak dan secara statistik memiliki korelasi mendekati atau sama dengan nol sehingga *ciphertext* tidak ada hubungannya dengan *plaintext*, yang mana hal tersebut dapat dibuktikan bahwa delapan data masukan pada *plaintext*, ada tida data masukan *plaintext* yang memiliki korelasi nol (0,00000) terhadap *ciphertext*, serta rata-rata hasil korelasi dari delapan data masukan *plaintext* terhadap *ciphertext* yaitu sebesar 0,0008859. [5].

Penelitian ke Lima Perancangan Kriptografi *Block Cipher* 256bit Berbasis Pola Rumah Adat Souraja, Berdasarkan penelitian yang telah dilakukan maka dapat diambil kesimpulan, bahwa Perancangan kriptografi simetris berbasis pola rumah adat Souraja dengan menambahkan S-BOX AES, dapat melakukan enkripsi dan dekripsi, dan juga memenuhi konsep 5-tuple sehingga dapat dikatakan sebagai sebuah sistem kriptografi. Pola rumah adat Souraja ini dapat menghasilkan output enkripsi yang random. Dalam perancangan ini didapatkan hasil nilai korelasi terendah mencapai 0.005183892 dan nilai *avalanche effect* tertinggi yang mencapai 51.171875% [6].

Berdasarkan penelitian-penelitian sebelumnya terkait perancangan kriptografi block *cipher*, maka dilakukan penelitian tentang perancangan kriptografi Block *Cipher* dengan memanfaatkan pola tarian Denok Deblong. Block *cipher* merupakan algoritma simetris yang mempunyai input dan output yang berupa blok dan setiap bloknya biasanya terdiri dari 64-bit atau lebih. Pada block *cipher*, hasil enkripsi berupa blok *ciphertext* biasanya mempunyai ukuran yang sama dengan blok *plaintext*. Dekripsi pada block *cipher* dilakukan dengan cara yang sama seperti pada proses enkripsi. Secara umum dapat dilihat pada Gambar 1.



Gambar. 1. Skema Proses Enkripsi-Dekripsi Pada *Block Cipher* [7]

Misalkan *block plaintext* (P) yang berukuran m bit dinyatakan sebagai vektor

$$P= (P_1, P_2, \ldots, P_m) \quad (1)$$

yang dalam hal ini pi adalah 0 atau 1 untuk i = 1, 2, …, m, dan *block ciphertext* (C) adalah

$$C = (C_1, C_2, \ldots, C_m) \quad (2)$$

yang dalam hal ini ci adalah 0 atau 1 untuk i = 1, 2, …, m. Bila plaintext dibagi menjadi n buah blok, barisan blok-blok plaintext dinyatakan sebagai

$$(P_1, P_2, \ldots, P_n) \quad (3)$$

Untuk setiap *block plaintext* Pi, bit-bit penyusunnya dapat dinyatakan sebagai vektor

$$Pi = (P_{i1}, P_{i2}, \ldots, P_{im}) \quad (4)$$

Enkripsi dan dekripsi dengan kunci K dinyatakan berturut-turut dengan persamaan

$$E_K (P) = C \quad (5)$$

untuk enkripsi, dan

$$D_K (C)= P \quad (6)$$

Fungsi E haruslah fungsi yang berkoresponden satu-ke-satu, sehingga

$$E^{-1}= D \quad (7)$$

Sebuah kriptografi dapat dikatakan sebagai suatu teknik kriptografi, harus memenuhi lima-tuple (Five tuple) [8]:

1. P adalah himpunan berhingga dari *plaintext*,
2. C adalah himpunan berhingga dari *ciphertext*,
3. K merupakan ruang kunci (keyspace), adalah himpunan berhingga dari kunci,
4. Untuk setiap $d_k \in \boldsymbol{K}$ terdapat aturan enkripsi $e_k \in \boldsymbol{E}$ berkorespondensi dengan aturan dekripsi $d_k \in \boldsymbol{D}$. Setiap $e_k : \mathbf{P} \rightarrow \mathbf{C}$ dan $e_k : \mathbf{C} \rightarrow \mathbf{P}$ adalah fungsi sedemikian hingga $d_k (e_k (x)) = x$ untuk setiap *plaintext* $x \in \mathbf{P}$.

Dalam pengujian menggunakan korelasi yang merupakan teknik statistik untuk mengukur kekuatan hubungan antar dua variabel dan untuk mengetahui bentuk hubungan antara dua variabel tersebut dengan hasil yang bersifat kuantitatif. Kekuatan hubungan antar dua variabel itu disebut dengan koefisien korelasi. Nilai koefisien akan selalu berada diantara -1 sampai +1. Untuk menentukan kuat atau lemahnya hubungan antara variabel yang diuji, dapat

JUTEI Edisi Volume.3 No.1 April 2019
ISSN 2579-3675, e-ISSN 2579-5538
DOI 10.21460/jutei.2018.31.146

digunakan Tabel 1 [9].

TABEL 1
KLASIFIKASI KOEFISIENSI KORELASI

Interval Koefisien	Tingkat Hubungan
0,00 – 0,199	Sangat Rendah
0,20 – 0,399	Rendah
0,40 – 0,599	Sedang
0,60 – 0,799	Kuat
0,80 – 1,000	Sangat Kuat

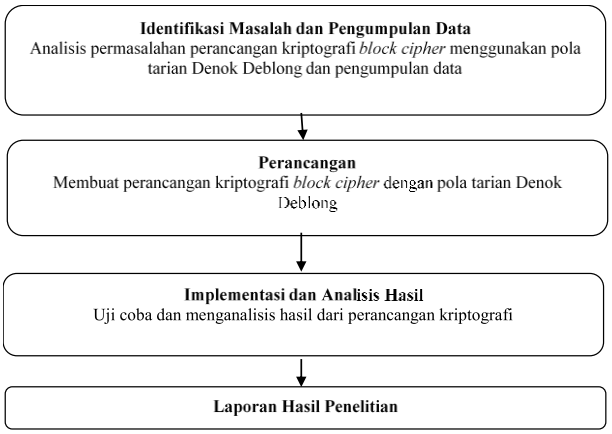
Selain itu proses *Block cipher* ini menggunakan operasi XOR dimana output yang dihasilkan dari proses enkripsi akan susah ditebak, karena apabila melihat dasar dari XOR seperti berikut:

- 0 XOR 0 = 0
- 0 XOR 1 = 1
- 1 XOR 0 = 1
- 1 XOR 1 = 0

Maka apabila hasil output adalah 0 maka untuk mendapatkan input nya tidak tahu, bisa jadi input yang dihasilkan adalah 1 atau 0. Dasar tersebut digunakan untuk melakukan kriptografi *block cipher*.

III. METODE PERANCANGAN

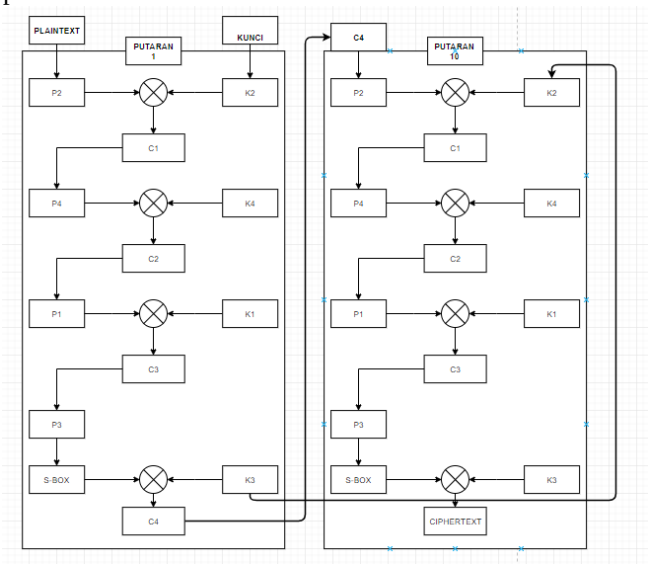
Perancangan kriptografi ini akan diselesaikan melalui beberapa tahapan Penelitian yaitu: (1) Identifikasi Masalah, (2) Pengumpulan Data, (3) Perancangan Kriptografi, (4) Uji Kriptografi dan, (5) Penulisan Laporan.



Gambar. 2.Tahapan Penelitian

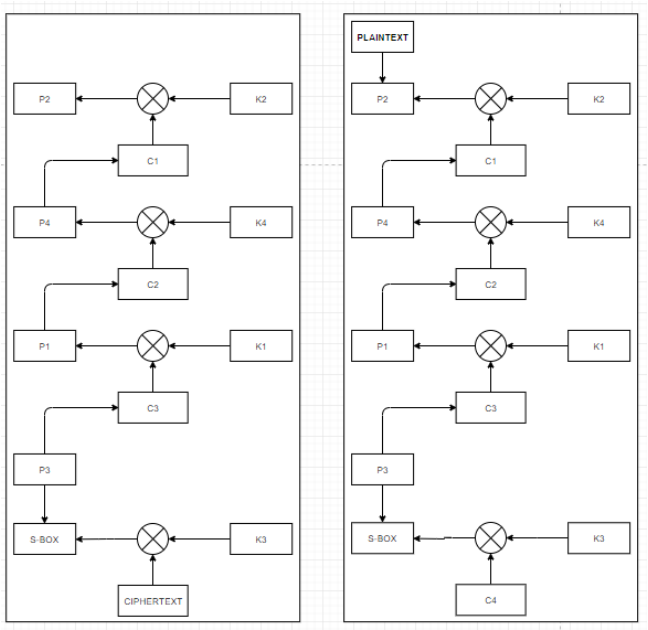
Tahapan penelitian dari Gambar 2, dapat dijelaskan sebagai berikut: Tahap pertama: Identifikasi masalah, yaitu mencari dan melihat kekurangan dari segi keamanan algoritma kriptografi sebelumnya, serta efisiensi putaran yang digunakan dalam proses enkripsi yang nantinya akan digunakan sebagai rumusan masalah serta tujuan dari penelitian ini. Dalam tahap ini dilakukan pengumpulan data terkait pola tarian Denok Deblong dengan membaca jurnal-jurnal terdahulu. Tahap kedua: Perancangan: Pada tahap ini akan dilakukan perancangan kriptografi *block cipher* 64-bit Berbasis Pola tarian Denok Deblong dengan menggunakan 4 pola yang telah dibuat dan menggunakan Tabel S-BOX sebagai tambahan agar terbentuk ciphertext yang lebih acak. Untuk pembuatan kunci, proses enkripsi dan proses dekripsi dikombinasikan dengan XOR. Tahap keempat: Pengujian kriptografi dilakukan dengan cara manual dari proses input *plaintext*, mengubah *plaintext* ke dalam bit dan melakukan enkripsi. Tahap kelima: Menulis laporan dari hasil penelitian yang sudah dilakukan dari tahap awal hingga tahap akhir. Batasan masalah dalam penelitian ini yaitu: 1) Proses enkripsi hanya dilakukan pada data berupa teks, 2) Pola tarian Denok Deblong digunakan dalam proses transposisi *plaintext*, 3) Jumlah kapasistas *plaintext* dan kunci dibatasi, maksimal 32 karakter serta proses putaran terdiri dari 4 putaran, 4) Panjang *block* adalah 64-bit.

Perancangan kriptografi ini dilakukan dalam empat proses enkripsi yang diputar sebanyak sepuluh kali seperti pada Gambar 3.



Gambar. 3. Proses alur Enkripsi

Langkah-langkah proses enkripsi dapat dijabarkan sebagai berikut: a) Mempersiapkan *plaintext*; b) Mengubah *plaintext* menjadi biner sesuai dalam tabel ASCII; c) Dalam proses enkripsi, *plaintext* dan kunci akan melewati empat proses pada setiap putaran, yaitu : 1) Putaran pertama *Plaintext* 2 (P2) melakukan transformasi dengan pola tarian Denok Deblong dan di XOR dengan Kunci 2 (K2) menghasilkan *Ciphertext* 1 (C1); 2) *Plaintext* 4 (P4) melakukan transformasi dengan pola tarian Denok Deblong dan di XOR dengan Kunci 4 (K4) menghasilkan *Ciphertext* 2 (C2), dan tahapan tersebut akan berlanjut sampai proses empat dimana *Plaintext* 3 (P3) melakukan transformasi dengan pola tarian denok Deblong kemudian dilakukan proses subtitusi dengan S-BOX untuk menghasilkan bilangan biner baru, kemudian di XOR dengan Kunci 3 (K3) yang menghasilkan *Ciphertext* 4 (C4) ; 3) *Ciphertext* 4 (4) masuk pada putaran kedua dengan alur proses yang sama dengan putaran pertama, dan tahapan tersebut akan berlanjut sampai putaran ke-10 yang menghasilkan *Ciphertext* Akhir.

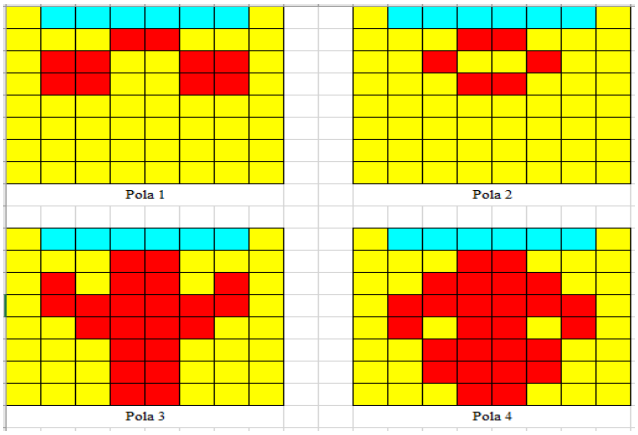


Gambar. 4. Proses Alur Dekripsi

Gambar 4 menunjukan alur proses dekripsi, langkah-langkah proses dekripsi tersebut dijelaskan sebagai berikut: a) Mempersiapkan *ciphertext* dan kunci; b) Mengubah *ciphertext* dan kunci menjadi biner sesuai dalam tabel ASCII; c) dalam perancangan dekripsi, *ciphertext* dan kunci akan melewati empat proses pada setiap putaran secara terbalik; d) Putaran pertama *Ciphertext* (C) diproses dengan pola dan di XOR dengan Kunci 3 (K3) dari putaran 10, menghasilkan P4; e) P4 tersebut melakukan transformasi dengan pola tarian denok Deblong kemudian dilakukan proses subtitusi dengan S-BOX untuk menghasilkan bilangan biner baru menjadi C3 di putaran 10; f) Masuk pada putaran dua, C3 diproses dengan pola dan di XOR dengan Kunci 1 (K1) dari putaran 10, menghasilkan P3; Proses tersebut berlanjut sampai ke putaran 1 sehingga menghasilkan *Plaintext* akhir P2.

IV. Hasil Dan Pembahasan

Dalam bagian ini akan membahas tentang algoritma perancangan kriptografi *block cipher* 64-bit berbasis pola tarian Denok Deblong secara lebih rinci. Dalam hal ini pola Tarian Denok delong diginakan sebagai proses pemasukan dan pengambilan bit.



Gambar. 5. Pola Tarian Denok Deblong

Pada Gambar 5 menunjukan empat pola yang berbeda, dimana pola-pola tersebut menunjukan pola-pola yang terdapat pada tarian Denok Deblong. Berdasarkan pola-pola yang sudah dirancang, dilakukan pengujian korelasi dengan mengkombinasikan urutan pola untuk menemukan nilai korelasi terbaik. Pengujian dilakukan menggunakan contoh *plaintext* "DIESUKSW" menggunakan kunci "BUDAYAKU". Berdasarkan hasil pengujian korelasi, maka hasil terkecil yang akan digunakan sebagai acuan perancangan dalam proses enkripsi dan dekripsi yang ditunjukan Tabel 2.

Tabel II
Hasil Korelasi Setiap Kombinasi Pola Tarian Denok Deblong

	RATA-RATA NILAI KORELASI		
POLA	**RATA-RATA**	**POLA**	**RATA-RATA**
1-2-3-4	0,36147491	**3-1-2-4**	0,213294507
1-2-4-3	0,146338465	**3-1-4-2**	0,623014281
1-3-2-4	0,197018645	**3-2-1-4**	0,316562897
1-3-4-2	0,739404365	**3-2-4-1**	0,064966156
1-4-2-3	0,158868154	**3-4-1-2**	0,183189577
1-4-3-2	0,285426929	**3-4-2-1**	0,092698866
2-1-3-4	0,13064667	**4-1-2-3**	0,063752557
2-3-1-4	0,392613531	**4-1-3-2**	0,78259488
2-3-1-4	0,343321394	**4-2-1-3**	0,554969406
2-3-4-1	0,052668036	**4-2-3-1**	0,423383466
2-4-1-3	0,046976586	**4-3-1-2**	0,399893928
2-4-3-1	0,159081999	**4-3-2-1**	0,312028935

Hasil kombinasi pola dan mendapatkan nilai korelasi terbaik pada kombinasi pola 2-4-1-3. Kombinasi ini lah yang akan digunkan untuk melanjutkan proses enkripsi hingga putaran ke-10 untuk menghasilkan *ciphertext*.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Gambar. 6.Tabel Subtitusi S-BOX

Gambar 6 merupakan tabel substitusi S-box yang digunakan dalam proses enkripsi. Cara pensubstitusian adalah sebagai berikut: untuk setiap byte pada array state, misalkan S [0, 0] = 11, maka S'[0, 0] = 82. nilai substitusinya, dinyatakan dengan elemen di dalam S-BOX yang merupakan perpotongan antara baris x dengan kolom y.

Telah dijelaskan sebelumnya bahwa perancangan kriptografi ini dilakukan sebanyak 10 putaran, dan disetiap putaran memiliki 4 proses untuk mendapatkan hasil akhir yaitu *ciphertext*. Proses pertama *plaintext* dan kunci diubah kedalam bentuk ASCII kemudian diubah lagi kedalam biner.

JUTEI Edisi Volume.3 No.1 April 2019
ISSN 2579-3675, e-ISSN 2579-5538
DOI 10.21460/jutei.2018.31.146

Kemudian bit-bit *plaintext* diproses dengan pola pemasukan dan pengambilan kedalam kolom matriks 8x8 menggunakan bagian dari pola tarian yang berbeda-beda pada setiap proses. Kemudian di setiap proses dilakukan X-OR dari *Plaintext* (P) dan kunci (K) menghasilkan *ciphertext* (C) sampai proses keempat di setiap putaran. Kemudian diulang terus sampai putaran ke-10 dan hingga menghasilkan *Ciphertext* akhir.
Untuk menjelaskan secara detail proses pemasukan bit dalam matriks maka diambil proses 1 pada putaran 1 sebagai contoh. Misalkan angka 1 merupakan inisialisasi setiap bit yang merupakan hasil konversi *plaintext* maka urutan bit adalah sebagai berikut 1, 2, 3, 4, ….64.

1	9	17	25	33	41	49	57
2	10	18	26	34	42	50	58
3	11	19	27	35	43	51	59
4	12	20	28	36	44	52	60
5	13	21	29	37	45	53	61
6	14	22	30	38	46	54	62
7	15	23	31	39	47	55	63
8	16	24	32	40	48	56	64

Gambar. 7. Pola Ambil Semua Kunci

1	2	3	4	5	6	7	8
16	15	14	13	12	11	10	9
17	18	19	20	21	22	23	24
32	31	30	29	28	27	26	25
33	34	35	36	37	38	39	40
48	47	46	45	44	43	42	41
49	50	51	52	53	54	55	56
64	63	62	61	60	59	58	57

Gambar. 8. Pola Pemasukan Semua Kunci

13	12	11	10	9	8	7	57
14	21	28	3	2	44	50	58
15	22	4	34	39	1	51	59
16	23	29	5	6	45	52	60
17	24	30	35	40	46	53	61
18	25	31	36	41	47	54	62
19	26	32	37	42	48	55	63
20	27	33	38	43	49	56	64

Gambar. 9. Pola Pemasukan *Plaintext* dari pola 2 Untuk Proses 1

Gambar 9 merupakan posisi penari yang memutar yang digunakan sebagai pola masuk dari pola 1 yang digunakan untuk memasukkan setiap 8-bit dari karakter *plaintext*, kemudian pola tersebut diimplementasikan ke dalam excel. Dari pola tersebut kemudian diambil bitnya dari pola pemasukan *plaintext* sesuai Gambar 7 dan dimasukan ke dalam kolom matriks lagi sehingga menghasilkan P2 yang nantinya akan di XOR dengan kunci K2 yang sebelumnya sudah dimasukkan ke pola pemasukan kunci seperti yang ditunjukkan pada Gambar 8, sehingga menghasilkan *Ciphertext* 1.

33	27	28	29	30	31	32	57
34	41	46	13	14	49	52	58
35	42	12	1	26	15	53	59
36	10	11	2	25	16	17	60
37	9	47	3	24	50	18	61
38	43	8	4	23	19	54	62
39	44	7	5	22	20	55	63
40	45	48	6	21	51	56	64

Gambar. 10.Pola Pemasukan *Plaintext* dari pola 4 Untuk Proses 2

Gambar 10 merupakan bentuk badan dan tangan penari yang digunakan sebagai pola masuk dari pola 2 yang digunakan untuk memasukkan setiap 8-bit dari karakter *plaintext*, kemudian pola tersebut diimplementasikan ke dalam excel. Dari pola tersebut kemudian diambil bitnya dari pola pemasukan *plaintext* sesuai Gambar 7, dimana C1 pada proses 1 digunakan sebagai P4 dan K2 sebagai K4, sehingga P4 di XOR dengan K4 yang sebelumnya sudah dimasukkan ke pola pemasukan kunci seperti yang ditunjukkan pada Gambar 8 akan menghasilkan *Ciphertext* 2.

17	11	12	13	14	15	16	57
18	25	30	6	5	47	52	58
19	9	7	35	41	3	1	59
20	10	8	36	42	4	2	60
21	26	31	37	43	48	53	61
22	27	32	38	44	49	54	62
23	28	33	39	45	50	55	63
24	29	34	40	46	51	56	64

Gambar. 11.Pola Pemasukan *Plaintext* dari pola 1 Untuk Proses 3

Gambar 11 merupakan pola masuk dari pola 1 yang digunakan untuk memasukkan setiap 8-bit dari karakter *plaintext*, kemudian pola tersebut diimplementasikan ke dalam excel. Dari pola tersebut kemudian diambil bitnya dari pola pemasukan *plaintext* sesuai Gambar 7, sehingga menghasilkan P1 yang nantinya akan di XOR kan dengan K1 yang sebelumnya sudah dimasukkan ke pola pemasukan

kunci seperti yang ditunjukkan pada Gambar 8, sehingga menghasilkan *Ciphertext*

29	28	27	26	25	24	23	57
30	37	42	1	22	47	52	58
31	5	43	2	21	48	18	59
32	6	4	3	20	19	17	60
33	38	7	8	15	16	53	61
34	39	44	9	14	49	54	62
35	40	45	10	13	50	55	63
36	41	46	11	12	51	56	64

Gambar. 12. Pola Pemasukan Plaintext dari pola 3 Untuk Proses 4

Gambar 12 merupakan pola masuk dari pola yang digunakan untuk memasukkan setiap 8-bit dari karakter *plaintext*, kemudian pola tersebut diimplementasikan ke dalam excel. Dari pola tersebut kemudian diambil bitnya sesuai pola pemasukan *plaintext* sesuai Gambar 7 sehingga menghasilkan P3 dan dikombinasikan dengan S-BOX yang nantinya akan di XOR dengan K3 yang sebelumnya sudah dimasukkan ke pola pemasukan kunci seperti yang ditunjukkan pada Gambar 8, sehingga menghasilkan *Ciphertext* 4.

Proses enkripsi putaran 1 telah selesai, kemudian dilakukan proses yang sama secara terus-menerus hingga putaran ke-10. Sebelumnya pada setiap proses P4 disubtitusikan dengan tabel S-BOX, Dalam Proses S-BOX sendiri pertama harus mengubah bit biner menjadi hexadecimal terlebih dahulu, perubahan sebelum di subtitusi S-BOX dan sesudah di subtitusi S-BOX ditunjukan pada Tabel 3.

S-BOX yang dimasukan setiap putaran menunjukan perubahan Hexadecimal *Plaintext* 3 (P3) disetiap proses enkripsi sebelum disubtitusikan dengan tabel S-BOX dan setelah disubtisusikan dengan tabel S-BOX. Penggunaan S-BOX sebenarnya tidak harus pada P3 saja, dalam kasus ini penempatan S-BOX dilakukan pada proses 4. Dari Tabel tersebut hasil dari Hexadecimal terlihat sangat berbeda dari sebelumnya, sehingga pola menjadi lebih acak. Sehingga semua proses sudah mendapatkan S-BOX dan mendapatkan *ciphertext* akhir.

Untuk pengujian algoritma dilakukan dengan mengambil contoh *plaintext* DIESUKSW dan kunci adalah BUDAYAKU. Kemudian dilakukan proses enkripsi sebanyak 10 putaran, dan disetiap putaran enkripsi akan mendapatkan *ciphertext* (C) dan dikonversi ke dalam bentuk desimal dan Character. Hasil enkripsi dari putaran ke-10 adalah final *ciphertext* yang ditunjukan pada Tabel 4.

Kemudian masuk ke proses dekripsi. Proses dekripsi adalah proses merubah *ciphertext* menjadi *plaintext* awal. Dekripsi dilakukan sama seperti enkripsi, tetapi dekripsi dimulai dari putaran ke-10 menuju putaran ke-1 untuk mendapatkan *plaintext* awal.

TABEL III
Hasil Hexadecimal Setelah S-BOX Pada Proses Enkripsi P4

Putaran	Proses 4	Hexadecimal		Hexadecimal
1	P3	17B4B3CA4C0FA715	Setelah S-*BOX*	87C64B105DFB892F
2		35D0323137FD3533		D960A12EB221D966
3		5BEB3067B05EDBDC		573C080AFC9D9F93
4		9770DEFE64F7C46B		85D09C0C8C268805
5		D536D03264CE2F22		B52460A18CEC4E94
6		4C45C1D777F1F7EE		5D68DD0D022B2699
7		383E1C7CA504627C		76D1C4012930AB01
8		5638524F20DEA4E6		B9764892549C1DF5
9		0525C055194580E4		36C21FED8E683AAE
10		280522C89645D4D4		EE3694B135681919

TABEL IV
HASIL *CIPHERTEXT* SETIAP PUTARAN PADA PROSES ENKRIPSI

Putaran	*Plaintext*	Hasil Desimal	*Character*
1	DIESUKSW	133941217114679108215	• ãÿ'jq
2	• ãÿ'jq	5414372496222453140	"uð5ITj }"
3	"uð5ITj }"	2042925224667819587	›-ù,¦?
4	›-ù,¦?	175951641501421522122 13	öþu³ãCæJ
5	öþu³ãCæJ	19615323219917733113224	• ôá~sæZ
6	• ôá~sæZ	1981647486822115193	ÅÍäã˜cRŠ
7	ÅÍäã˜cRŠ	1711241785113119212338	"q%³‰¿ È¥"
8	"q%³‰¿ È¥"	16520215231130935236	¹?O¯è]
9	¹?O¯è]	2411849132164316154	¼£v«
10	¼£v«	1792062344751144156190	ºÞ 6ýóÍ

JUTEI Edisi Volume.3 No.1 April 2019
ISSN 2579-3675, e-ISSN 2579-5538
DOI 10.21460/jutei.2018.31.146

TABEL V
ALGORITMA ENKRIPSI DAN DEKRIPSI

NO	Proses Enkripsi	NO	Proses Dekripsi
	Mulai		Mulai
1.	Masukkan *plaintext*	1.	Masukkan *ciphertext*
2.	*Plaintext* diubah ke DECIMAL	2.	*Ciphertext* diubah ke DECIMAL
3.	DECIMAL diubah ke BINER	3.	DECIMAL diubah ke BINER
4.	*Bit* BINER dimasukan ke kolom matriks 8x8 P2 dengan pola pemasukan *plaintext*	4.	*Bit* BINER dimasukan ke kolom matriks 8x8 C4 dengan pola pemasukan *plaintext*
5.	*Bit* pada kolom matriks diambil menggunakan pola pengambilan pola 2	5.	C4 di-XOR dengan K3 menghasilkan *plaintext* baru atau dalam *hexadecimal* menghasilkan deret angka 21;C0;23;7D;01;61;A8;AF
6.	*Bit* pengambilan dimasukan lagi kedalam matrik mendapatkan hasil akhir P2 dalam bentuk *bit*	6.	Hasil XOR selanjutnya dilakukan subtitusi menggunakan tabel S-*BOX* sehingga menghasilkan P3 atau dalam hexa*decimal* menghasilkan deret angka FA;FD;F5;F5;59;94;AE;F9 pada deskripsi 1 proses 1
7.	P2 di-XOR dengan K2 menghasilkan C1 atau dalam hexa*decimal* menghasilkan deret angka 2E;AE;0D;B0;D5;37;98;FD pada enkripsi 1 proses 1	7.	P3 diproses dengan pola pemasukan *plaintext*
8.	C1 menjadi P4 untuk proses selanjutnya	8.	Hasil proses P3 dimasukkan kedalam matriks 8x8 lagi dengan pola pengambilan pola 3
9.	*Bit* pada kolom matrik diambil menggunakan pola pengambilan pola 4	9.	P3 menjadi C3 untuk proses selanjutnya
10.	*Bit* pengambilan dimasukan lagi ke dalam matriks mendapatkan hasil akhir P4	10.	C3 di-XOR dengan K1 menghasilkan P1 atau dalam hexa*decimal* menghasilkan deret angka F3;4E;3F;F6;31;9F;66;3F pada deskripsi 1 proses 2
11.	P4 di-XOR dengan K4 menghasilkan C2 atau dalam hexa*decimal* menghasilkan deret angka F3;4E;3F;F6;31;9F;66;3F pada enkripsi 1 proses 2	11.	P1 diproses dengan pola pemasukan *plaintext*
12.	C2 menjadi P1 untuk proses selanjutnya	12.	Hasil proses P1 dimasukkan kedalam matriks 8x8 lagi dengan pola pengambilan pola 1
13.	*Bit* pada kolom matrik diambil menggunakan pola pengambilan pola 1	13.	P1 menjadi C2 untuk proses selanjutnya
14.	*Bit* pengambilan dimasukan lagi kedalam matrik mendapatkan hasil akhir P1	14.	C2 di-XOR dengan K4 menghasilkan P4 atau dalam hexa*decimal* menghasilkan deret angka 2E;AE;0D;B0 D5;37;98;FD pada deskripsi 1 proses 3
15.	P1 di-XOR dengan K1 menghasilkan C3 atau dalam hexa*decimal* menghasilkan deret angka FA;FD;F5;F5;59;94;AE;F9 pada enkripsi 1 proses 3	15.	P4 diproses dengan pola pemasukan *plaintext*
16.	C3 menjadi P3 untuk proses selanjutnya	16.	Hasil proses P4 dimasukkan ke dalam matriks 8x8 lagi dengan pola pengambilan pola 4
17.	*Bit* pada kolom matrik diambil menggunakan pola pengambilan pola 3	17.	P4 menjadi C1 untuk proses selanjutnya
18.	*Bit* pengambilan dimasukan lagi kedalam matrik mendapatkan hasil akhir P3	18.	C1 di-XOR dengan K2 menghasilkan P2 atau dalam hexa*decimal* menghasilkan deret angka 44;7F;E3;1B;FF;27;6A;71 pada deskripsi 1 proses 4
19.	P3 dilakukan subtitusi menggunakan tabel S-*BOX* sehingga bentuk *hexadecimal* P3 berganti menjadi 21;C0;23;7D;01;61;A8;AF	19.	P2 diproses dengan pola pemasukan *plaintext*

20.	Hasil P3 dan S-*BOX* selanjutnya di-XOR dengan K3 menghasilkan C4 atau dalam hexa*decimal* menghasilkan deret angka 05;7F;E3;1B;FF;27;6A;71 pada enkripsi 1 proses 4	20.	Hasil proses P2 dimasukkan kedalam matriks 8x8 lagi dengan pola pengambilan pola 2
21.	C4 diubah ke DECIMAL	21.	P2 diubah ke DECIMAL
22.	DECIMAL diubah ke CHAR untuk mendapatkan *Ciphertext* akhir • ãÿ'jq pada enkripsi 1.	22.	DECIMAL diubah ke CHAR untuk mendapatkan *Plaintext* awal DIESUKSW pada hasil akhir deskripsi 1.
	Selesai		Selesai

Tabel 5 merupakan algoritma proses enkripsi dan dekripsi secara menyeluruh. Proses enkripsi menghasilkan *Ciphertext* akhir, dan proses dekripsi menghasilkan *Plaintext* awal. Nilai korelasi antara *plaintext* dan *ciphertext* dapat digunakan untuk mengukur seberapa acak hasil enkripsi (*ciphertext*) dengan *plaintext*. Nilai korelasi sendiri berkisar 1 sampai -1, dimana jika nilai kolerasi mendekati 1 maka *plaintext* dan *ciphertext* memiliki nilai yang sangat berhubung, tetapi jika mendekati 0 maka *plaintext* dan *ciphertext* tidak memiliki nilai yang berhubungan.

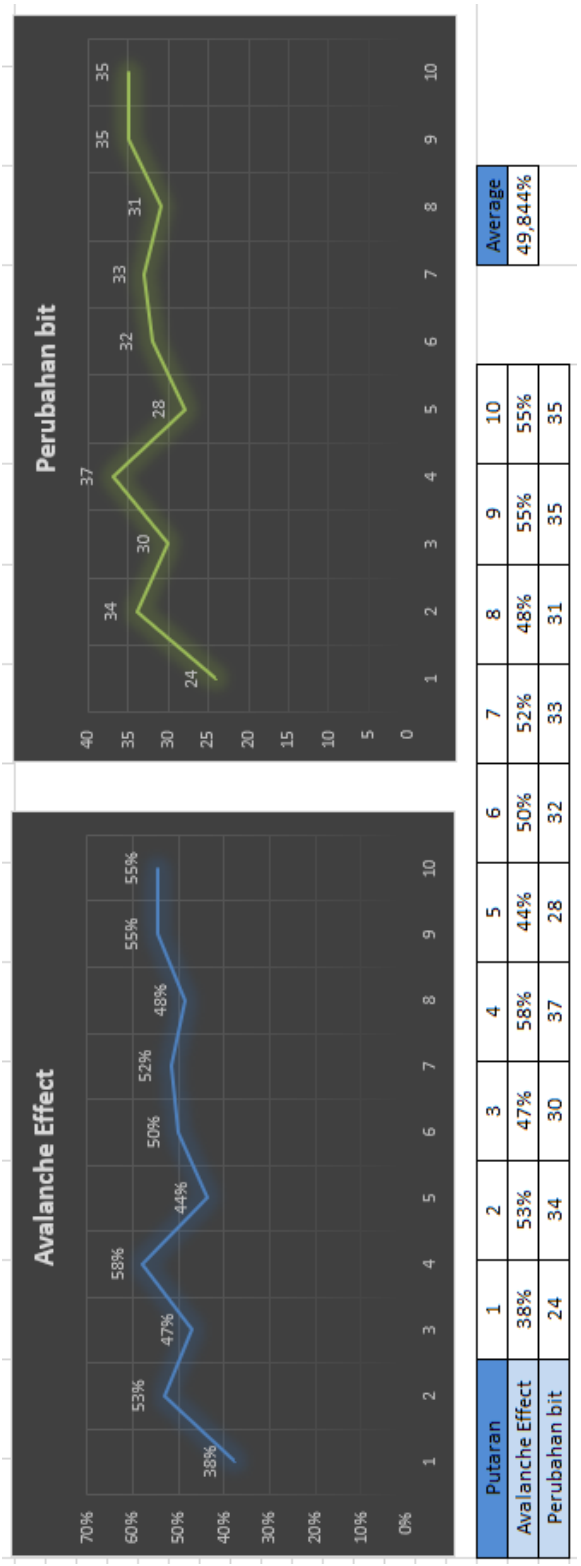
TABEL VI
NILAI KORELASI SETIAP PUTARAN

Putaran	Nilai Korelasi
1	-0,330215757
2	-0,259113959
3	-0,045521389
4	0,195618675
5	0,441842066
6	-0,658043826
7	0,626068584
8	-0,207952789
9	0,278442536
10	-0,256063877

Tabel 6 menunjukan nilai korelasi setiap putaran, dan dapat disimpulkan bahwa algoritma kriptografi *block cipher* 64-bit berbasis pola tarian Denok Deblong memiliki nilai korelasi lemah dan menghasilkan nilai korelasi yang acak.

Kemudian pengujian *Avalanche Effect* dilakukan untuk mengetahui perubahan bit yang ada ketika *plaintext* diubah. Pengujian dilakukan dengan merubah karakter yang terdapat pada *plaintext* awal, sehingga akan menghasilkan perbedaan pada setiap putarannya.

Pada Gambar 13 adalah hasil dari pengujian *Avalanche Effect*, pada kasus ini *plaintext* awal adalah DIESUKSW yang kemudian diubah menjadi Yuana123. Terjadi perubahan bit yang besar disetiap putarannya, seperti yang terjadi pada putaran ke-4 perubahan bit melebihi angka 58% dengan arti pada putaran ini terjadi perubahan bit yang baik, dan putaran ke-1 perubahan bitnya berada di bawah 38% yang berarti perubahan bitnya kurang baik. Suatu *avalanche effect* dikatakan baik jika perubahan bit yang dihasilkan berkisar antara 45-60% (sekitar separuhnya, 50 % adalah hasil yang sangat baik). Hal ini dikarenakan perubahan tersebut berarti membuat perbedaan yang cukup sulit untuk kriptanalis melakukan serangan. Dan dapat disimpulkan bahwa pengujian *Avalanche Effect* pada pola tarian Denok Deblong ini mendapatkan hasil yang baik, dengan rata-rata dari 10 putaran adalah 49,844%.



JUTEI Edisi Volume.3 No.1 April 2019
ISSN 2579-3675, e-ISSN 2579-5538
DOI 10.21460/jutei.2018.31.146

V. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, dapat diambil kesimpulan bahwa perancangan kriptografi *Block Cipher* berbasis pola tarian Denok Deblong ini dikatakan sebagai sistem kriptografi yang baik. Enkripsi *Block Cipher* memiliki kelemahan yaitu apabila data yang sama di enkripsi menggunakan kunci yang sama maka akan menghasilkan *cipherteks* yang sama, namun hal tersebut dapat diatasi dengan menggunakan ukuran blok yang lebih besar, misalnya 256 bit, sehingga walaupun blok data yang sama di enkripsi menggunakan kunci yang sama maka cipherteks yang dihasilkan akan berbeda. Dalam proses enkripsi, rancangan Kriptografi *Block Cipher* berbasis pola Tarian Denok Deblong ini menghasilkan output yang sangat acak sehingga memungkinkan untuk digunakan sebagai alternatif dalam pengamanan data. Berdasarkan pengujian terlihat bahwa dengan menambahkan S-BOX Pola tarian Denok Deblong ini dapat menghasilkan output enkripsi yang random. Selain itu, Dalam pengujian *Avalance Effect* yang dilakukan dengan menggunakan *plaintext* Yuana123 mendapatkan hasil yang baik, dengan rata-rata dari 10 putaran adalah 49,844%.

DAFTAR PUSTAKA

[1]. Keamanan Data Harus Jadi Prioritas di Era *Big Data Official Web Page*, http://www.beritasatu.com/iptek/403822-keamanan-data-harus-jadi-prioritas-di-era-big-data.html (Diakses pada 26 November 2018 jam 18:16:23).

[2]. Karinda, Tryanto. (2017). Perancangan Kriptografi *Block Cipher* 64 Bit Berbasis Pada Pola Formasi Sepak Bola 3-5-2. Jurusan Teknik Informatika, FTI UKSW Salatiga.

[3]. Bili, D., Dairo. (2015). Implementasi Kriptografi *Block Cipher* dengan Langkah Kuda. Jurusan Teknik Informatika, FTI UKSW Salatiga.

[4]. Tuhumury, Frellian. (2015). Perancangan Kriptografi *Block Cipher* 256 Bit Berbasis pada Pola Tuangan Air. Jurusan Teknik Informatika, FTI UKSW Salatiga.

[5]. Siswanto, Eko. (2018). Perancangan Algoritma Super Enkripsi Berbasis Pola 8-Queen of Fitness Chess. Jurusan Teknik Informatika, FTI UKSW Salatiga.

[6]. Mardika, W. Wayan, I., 2018, Perancangan Kriptografi *Block Cipher 256 bit* Berbasis Pola Rumah Adat Souraja. Jurusan Teknik Informatika, FTI UKSW Salatiga.

[7]. M Munir, R., 2006. Kriptografi. Bandung: Informatika.

[8]. Stinson, D. R., 1995, Cryptography: Theory and Practice. CRC Press, Boca Raton, London, Tokyo.

[9]. Sugiyono. 2009. "Metode Penelitian Bisnis (Pendekatan Kuantitatif, Kualitatif, dan R&D)". Bandung: Alfabeta.

[10]. D.A. de Vaus, Survey in Social Research, 5th Edition (New South Wales: Allen and Unwin, 2002) p. 259.