REPOSITOR, Vol. 1, No. 2, Desember 2019, Pp. 117-124
ISSN : 2714-7975
E-ISSN : 2716-1382

Modifikasi Enkripsi Dan Dekripsi AES Menggunakan Polybius Chiper Dalam Pengamanan Data

Shinta Permatasari[*], Aminudin[2], Sofyan Arifianto[3]
[1,2,3]Teknik Informatika/Universitas Muhammadiyah Malang
thashinta@webmail.umm.ac.id[1], aminudin2008@umm.ac.id[2], sofyan.arifianto@gmail.com[3]

Abstrak

Data merupakan file yang dapat bersifat rahasia sehingga membutuhkan sebuah proses pengamanan data untuk menjaga kerahasiaannya. Kripotgrafi yaitu proses pengamanan data yang dapat digunakan berdasarkan penggunaan algoritma salah satunya AES. AES adalah algoritma modern yang dapat dimodifikasi untuk meningkatkan konfusi dan difusi dalam kriptografi. Kombinasi AES dapat dilakukan menggunakan Polybius yang memiliki sifat difusi kriptografi. Penelitian ini melakukan modifikasi AES menggunakan matriks Polybius berukuran 6x6 dan 10x10 yang dilakukan pada plainteks maupun plainteks dan kunci. Analisa dilakukan berdasarkan tingkat perubahan bit tertinggi yang terdapat pada modifikasi II pada plainteks dan kunci matriks 6x6 yaitu sebesar 51,8% menggunakan uji avalanche effect. Hasil dari AE dibandingkan dengan hasil yang diharapkan menggunakan chi square dengan hasil AES modifikasi dapat meningkatkan AE sebesar 5% dengan taraf nyata 0,05 dan derajat kebebasan 4. Waktu eksekusi diuji pada penelitian dengan hasil AES modifikasi lebih tinggi dibandingkan AES standar dikarenakan kompleksitas dari setiap algoritma berbanding lurus dengan proses waktu enkripsi maupun dekripsi.

Kata Kunci: *AES, Polybius, Konfusi, Difusi*

Abstract

Data is a file that can be confidential so it requires a data security process to maintain confidentiality. Kripotgrrafi is a data security process that can be used based on the use of algorithms, one of which is AES. AES is a modern algorithm that can be modified to improve confusion and diffusion in cryptography. AES combination can be done using Polybius which has cryptographic diffusion properties. This study modified the AES using 6x6 and 10x10 polybius matrices that were performed on plaintext and plaintext and keys. Analysis was carried out based on the highest bit change rate found in modification II in the plaintext and 6x6 matrix keys, which amounted to 51.8% using the avalanche effect test. The results of the AE compared to the expected results using chi square with the modified AES results can increase the AE by 5% with the real level is 0,05 and the degree of freedom is 4. Execution time was tested in this study with the results of the AES modification time longer than the standard AES because the complexity of the algorithm affects both encryption and decryption time.

Keywords: *AES, Polybius, Conffusion, Diffusion*

1. Pendahuluan

Teknologi komputer pada era ini memungkinkan manusia dapat berkomunikasi dan saling bertukar informasi atau data. Data yang dimiliki dapat berupa file bersifat rahasia yang memungkinkan pihak lain tidak dapat mengkases sehingga membutuhkan suatu sistem kemanan data untuk menjaga kerahasiannya. Kriptografi merupakan ilmu dan seni untuk menjaga keamanan data. Proses pengamanan data dalam kriptografi dapat dilakukan menggunakan beberapa metode atau algoritma. AES (*Advanced Encryption Standard*) merupakan algoritma kriptografi yang dianggap efisien dalam proses pengamanan data. AES-128 bit memiliki ruang kunci 2^{128} yang merupakan nilai yang sangat besar dan dianggap aman sehingga terhindar dari serangan *brute force attack* [1].

Serangan terhadap sebuah algoritma dapat menyebabkan proses pengamanan data dinyatakan lemah. Penelitian [2] menyatakan bahwa kemungkinan kelemahan yang dapat timbul dalam algoritma AES dengan fokus penyerangan kunci untuk setiap ronde dan mengaplikasikan pasangan kunci ke ronde berikutnya. Serangan dapat mempengaruhi kompleksitas pada *cipher*

blok sehingga sangat penting untuk dilakukannya peningkatan difusi dalam proses pengamanan data [3]

Konfusi merupakan konsep kriptografi yang membuat pola *statistic* antara plainteks, chiperteks dan kunci menjadi rumit sedangkan difusi merupakan konsep kriptografi yang digunakan dalam proses pengamanan data dengan menyebarkan pengaruh satu bit plainteks atau kunci ke sebanyak mungkin chiperteks [4]. Pengamanan yang bagus dapat dilakukan dengan penerapan konsep kriptografi secara berulang pada sebuah blok tunggal dengan kombinasi yang berbeda [5] .

Kombinasi AES dapat dilakukan dengan melakukan modifikasi pada beberapa proses dalam sebuah algoritma untuk meningkatkan konfusi dan difusi sehingga pola penyerangan susah dilakukan. Peningkatan keamanan data dapat dilakukan menggunakan skema *polybius* yang memiliki kemungkinan fraksionasi yaitu mengarah ke kebingungan dan difusi Claude Shannon [6]. Penelitian [7] melakukan modifikasi menggunakan matriks 6x6 dengan hasil penelitian terfokus pada proses eksekusi enkripsi dan dekripsi yang membutuhkan waktu yang lebih lama. Penelitian [8] melakukan modifikasi AES dan RSA berdasarkan plainteks dan kunci menggunakan *polybius* dengan perluasan matriks 9x9.

Penelitian kali ini melakukan modifikasi algoritma AES menggunakan subtitusi skema *polybius* pada *plaintext* dan modifikasi AES pada *initialkey* dan *plaintext* dengan menggunakan matriks 6x6 dan perluasan matriks 10x10 untuk mengetahui performa eksekusi waktu enkripsi dan dekripsi pada setiap proses modifikasi dan tingkat perubahan bit yang memungkinkan terjadinya peningkatan difusi substitusi karakter pada data. Modifikasi ini dilakukan untuk meningkatkan kekuatan algoritma AES.

2. Metode Penelitian

Kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan pesan dengan cara menyandikan ke dalam bentuk yang tidak dapat dimengerti lagi maknanya. Kerahasiaan pesan dapat dilakukan menggunakan metode dalam kriptografi yaitu enkripsi dan dekripsi. Enkripsi merupakan proses perubahan plainteks menjadi chiperteks sehingga pola dalam pengamanan data tidak diketahui oleh pihak lain, sedangkan dekripsi yaitu proses sebaliknya dengan mengembalikan chiperteks menjadi plainteks awal sehingga data masih dapat di baca.

AES merupakan kriptografi kunci simetrik yang menggunakan satu kunci untuk proses enkripsi dan dekripsi. Penggunaan kunci simetrik terletak pada kerahasiaan kuncinya. AES terbagi menjadi 3 ukuran berdasarkan jumlah putaran dengan penggunaan kunci yang berbeda yaitu AES-128, AES-192, AES 256 dengan jumlah putaran masing-masing 10,12 dan 14. Penelitian ini melakukan uji coba AES-128bit dengan jumlah putaran 10.

2.1 Algoritma AES

AES *(Advanced Encryption Standard)* merupakan algoritma chiper blok yang menggunakan teknik subtitusi, permutasi, dan sejumlah putaran pada setiap blok yang akan di enkripsi. Proses enkripsi AES menggunakan 4 transformasi dasar dengan urutan transformasi *subbytes, shiftrows, mixcolumn* dan *addroundkey*. Proses dekripsi dilakukan sama dengan proses enkipsi dengan melakukan invers untuk setiap *state*. AES bekerja pada matriks berukuran 4 x 4 dengan penggunaan kunci simetrik yaitu hanya menggunakan satu kunci dalam proses enkripsi dan dekripsi.

Proses transformasi dilakukan sampai ronde ke-n atau ke-10 untuk ukuran kunci 128bit. Sebelum melakukan transformasi pertama, plainteks dan kunci melakukan perkalian *dot product* yang disebut transformasi *AddRoundKey*. Setelah itu melakukan proses transformasi selanjutnya hingga ronde keNr. Hasil enkripsi berupa chiperteks akan menghasilkan bentuk plainteks dengan isi file semula sehingga isi file dapat dibaca kembali. Alur kerja proses enkripsi dan dekripsi dapat dilihat pada flowchart Gambar 1

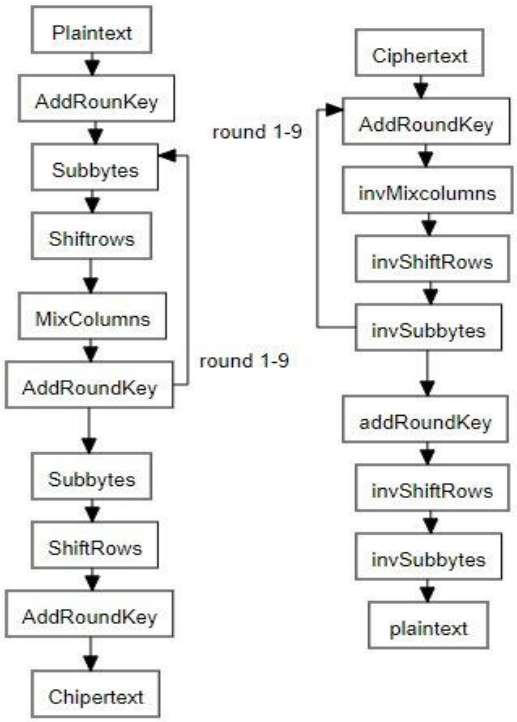
Proses enkripsi memiliki beberapa trasnformasi sesuai pada gambar 1 dapat dijelaskan sebagai berikut:

1. *Addroundkey* yaitu operasi xor dilakukan antara plainteks dan chiperkey yang disebut juga initial key yang akan menghasilkan state baru.
2. *Subbytes* yaitu hasil dari xor disubtitusikan menggunakan tabel s-box yang berisikan *hexadecimal*. Proses subtitusi dilakukan berdasarkan indeks baris dan kolom untuk masing-masing karakter.

3. *ShiftRows* yaitu pergeseran setiap elemen blok yang bekerja pada setiap baris dimana pergeseran dilakukan ke kiri berdasarkan indeks baris tanpa mengubah nilainya. Baris pertama tidak mengalami pergeseran, baris ke-2 melakukan pergeseran 1 byte, baris ke-3 melakukan pergeseran 2 byte, baris ke-4 melakukan pergeseran 3byte.
4. *Mixcolumn* yaitu perkalian matriks antara kontant matriks dengan state sebelumnya. Perkalian matriks menggunakan operasi xor berdasarkan perkalian matriks biasa kemudian akan menghasilkan chiperblok baru. Proses transformasi dilakukan hingga putaran ke-9. Pada putaran terakhir, mixcolumn tidak mengalami proses, sehingga state dari hasil shiftrows langsung melakukan proses xor terhadap kunci terakhir yang akan menghasilkan chiperteks.

Proses dekripsi memiliki proses *Addroundkey* yang sama pada proses enkripsi dengan perbedaan transformasi lainnya yang merupakan invers dari proses transformasi enkripsi sehingga dapat di uraikan sebagai berikut:

1. *Invers Subbytes* yaitu proses subtitusi menggunakan tabel inv s-box. Proses subtitusi dilakukan berdasarkan indeks baris dan kolom pada tabel.
2. *Invers Shiftrows* yaitu melakukan pergeseran ke kanan berdasarkan indeks baris tanpa mengubah nilainya
3. *Invers Mixcolumn* yaitu melakukan perkalian matriks operasi xor antara kontan *invers mixcolumn* dengan *state* sebelumnya. Pada putaran terakhir proses ini tidak dilakukan sehingga state dari proses *inv shiftrows* melakukan proses xor terhadap kunci terakhir yang digunakan yang akan menghasilkan plainteks.



Gambar 1. Enkripsi dan Dekripsi AES

2.2 Algoritma Modifikasi

Polybius square merupakan matriks yang berisikan karakter yang digunakan untuk proses modifikasi pada tahap awal sebelum melakukan transformasi pada AES standar menggunakan matriks 6x6 dan 10x10. Proses modifikasi dilakukan dengan melakukan subtitusi menggunakan matriks berdasarkan indeks baris dan kolom.

Modifikasi 1 dilakukan berdasarkan perubahan plainteks menggunakan masing-masing matriks Polybius. Modifikasi 2 dilakukan berdasarkan perubahan plainteks dan kunci yang disubtitusikan menggunakan masing-masing matriks polybius sehingga menghasilkan 4 modifikasi dengan penggunaan matriks yang berbeda. Penelitian ini menggunakan matriks yang berbeda untuk menganalisa apakah terdapat perbedaan hasil yang didapatkan. Matriks yang

digunakan dapat dilihat pada Gambar 2 untuk matriks ukuran 6x6 dan Gambar 3 untuk ukuran matriks 10x10.

	0	**1**	**2**	**3**	**4**	**5**
0	a	b	c	d	e	f
1	g	h	i	j	k	l
2	m	n	o	p	q	r
3	s	t	u	v	w	x
4	y	z	1	2	3	4
5	5	6	7	8	9	

Gambar 2. Matriks 6x6

Proses subtitusi dilakukan berdasarkan karakter yang digunakan pada plainteks maupun kunci yang kemudian dilihat berdasarkan indeks baris kemudian kolom. Hasil subtitusi yang dihasilkan diproses selanjutnya dengan melakukan proses transformasi pada AES yaitu *addroundkey, subbytes, shitrows, mixcolumn.*

	0	1	2	3	4	5	6	7	8	9
0	a	b	c	d	e	f	G	h	i	j
1	k	l	m	n	o	p	q	r	s	t
2	u	v	w	x	y	z	0	1	2	3
3	4	5	6	7	8	9		!	@	#
4	$	%	^	&	*	(	)	-	+	{
5	}	_	=	[	]	;	“	.	,	?
6	<	>	:	‘	/	\	α	β	γ	δ
7	ϵ	√	η	θ	λ	μ	л	σ	ϕ	
8	∫	Ä	È	÷	≠	¥	≤	≥	↓	→
9	←	↑	↔	¶	∑	Ω	|	ø	Δ	~

Gambar 3. Matriks 10x10

Proses subtitusi dilakukan sama menggunakan matriks 10x10 namun menghasilkan indeks bilangan yang berbeda. Hasil subtitusi akan melakukan proses transformasi pada AES standar baik enkripsi maupun dekripsi.

2.3 *Avalanche Effect*

Avalanche Effect merupakan salah satu karakteristik yang menjadi acuan untuk menentukan baik atau tidaknya sebuah algoritma kriptografi [9]. Metode yang dilakukan dengan menghitung perubahan bit pada perubahan plainteks maupun kunci menjadi chiperteks [10]. Semakin banyak perubahan bit maka semakin baik pula algoritma yang digunakan dengan nilai *Avalanche Effect* yang tinggi. Pengujian AE dilakukan berdasarkan Persamaan 1 [11].

$$Avalanche\ Effect\ (AE) = \frac{\sum \text{bit\_berubah}}{\sum bit\_total} * 100\% \quad (1)$$

Keterangan:
∑bit_berubah = Jumlah bit yang berubah
∑bit_total = Jumlah bit total

2.4 *Chi Square*

Chi square merupakan pengujian yang digunakan untuk membandingkan proporsi hasil pengujian dalam penelitian dengan hasil yang diharapkan sehingga dapat menentukan apakah

kedua hasil tersebut berbeda secara signifikan [12]. Hasil perbandingan didapatkan berdasarkan *chi square* hitung dan *chi square table* sehingga dapat mengetahui apakah hasil yang diharapkan dapat dipenuhi dengan hasil pengujian.

Apabila hasil nilai hitung < nilai tabel chi square maka hipotesa 1 diterima dan hipotesa 2 ditolak begitu pula sebaliknya. H0 merupakan hipotesis awal yang digunakan yaitu AES modifikasi dapat meningkatkan avalanche effect sebesar 5% dan H1 merupakan hipotesis alternative yang menolak hipotesis awal yaitu AES modifikasi TIDAK dapat meningkatkan avalanche effect sebesar 5%.

Penelitian ini menggunakan taraf nyata 0,05. Taraf nyata yang digunakan berdasarkan taraf nyata yang pada umumnya diterapkan dan dapat diterima pada banyak situasi[12]. Hasil *chi square* hitung didapatkan berdasarkan Persamaan 2 [13].

$$X^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \quad (2)$$

Keterangan:
O = Frekuensi Observasi
E = Frekuensi Harapan

2.5 Waktu

Waktu pada proses enkripsi dan dekripsi pengamanan data merupakan hal yang sangat penting, karena akan mempengaruhi sebuah kualitas dari kecepatan eksekusi pada sebuah algoritma. Efisiensi waktu perlu dipertimbangkan sehingga tidak berdampak pada proses enkripsi dengan ukuran file dalam skala yang besar[14] Pengujian waktu ekseskusi dilakukan untuk mengetahui sistem yang dibangun lebih baik atau tidak.

3. Hasil Penelitian dan Pembahasan

Implementasi dilakukan berdasarkan metode algoritma AES dan modifikasi yang digunakan sehingga dapat dilakukan uji coba dan menganalisa hasil. Setiap algoritma dibandingkan berdasarkan tiga metode pengujian yang digunakan yaitu *avalanche effect, chi square* dan eksekusi waktu.

3.1 *Avalanche effect*

Pengujian dilakukan berdasarkan indeks Persamaan 1 sehingga mendapatkan hasil sesuai dengan Tabel 1.

Tabel 1. Hasil Pengujian Avalanche Effect

No	Algoritma	Avalanche Effect (%)
1	AES standar	46,2%
2	AES modifikasi I pada plain (matriks 6x6)	51,2%
3	AES modifikasi I pada plain (matriks 10x10)	50,6%
4	AES modifikasi II pada plain dan kunci (matriks 6x6)	51,8%
5	AES modifikasi II pada plain dan kunci (matriks 10x10)	50,8%

Berdasarkan Tabel 1 dapat diketahui persentase *avalanche effect* rendah terdapat pada algoritma AES standar dengan persentase 46,2% sedangkan persentase tertinggi terdapat pada AES modifikasi II pada plainteks dan kunci untuk ukuran matriks 6x6 dengan sebesar 51,8%. Perubahan bit tertinggi didapatkan dari hasil modifikasi sehingga subtitusi karakter pada penggunaan matriks dalam modifikasi mempengaruhi tingkat perubahan bit yang mempengaruhi tingkat konfusi dan difusi pada kriptografi. Hasil didapatkan berdasarkan rata-rata dari 5 file uji dengan ukuran yang berbeda.

3.2 *Chi Square*

Pengujian ini dilakukan mengacu pada data hasil pengujian sebelumnya. Pengujian chi hitung dilakukan berdasarkan persamaan 2. Hipotesa pertama yang digunakan yaitu AES modifikasi dapat meningkatkan *avalanche effect* sebesar 5% dan hipotesa kedua yaitu AES

modifikasi TIDAK dapat meningkatkan *avalanche effect* sebesar 5%. Hasil didapatkan berdasarkan Tabel 2.

Tabel 2. Hasil Pengujian Chi Square

No	Parameter Uji	Taraf Nyata	Derajat Kebebasan	Chi Square Table	Chi Square Hitung	Hasil
1	*Avalanche Effect* Modifikasi 6x6 pada Plain	0,05	4	9,4877	0,5859	H0 diterima, H1 ditolak
2	*Avalanche Effect* Modifikasi 10X10 pada Plain	0,05	4	9,4877	1,0647	H0 diterima, H1 ditolak
3	*Avalanche Effect* Modifikasi 6x6 pada Plain dan Key	0,05	4	9,4877	1,3407	H0 diterima, H1 ditolak
4	*Avalanche Effect* Modifikasi 10x10 pada Plain dan Key	0,05	4	9,4877	1,022	H0 diterima, H1 ditolak

Tabel 2 menunjukkan hasil pengujian dengan taraf nyata 0,05 dan derajat kebebasan yang didapatkan dari jumlah file yang di uji dikurangi 1 sehingga db = 4. Nilai dari chi square table dilihat berdasarkan taraf nyata dan db yang digunakan. Hasil menunjukkan bahwa nilai hitung chi lebih rendah dibandingkan nilai chi table yang berarti hipotesa pertama diterima dan hipotesa kedua ditolak sehingga didapatkan kesimpulan bahwa AES modifikasi dapat meningkatkan *avalanche effect* sebesar 5%.

3.3 Performa Waktu

Implementasi perfoma waktu dilakukan berdasarkan proses enkripsi dan dekripsi. Hasil akan menunjukan waktu yang dibutuhkan untuk melakukan proses enkripsi dan dekripsi berdasarkan 5 ukuran file yang berbeda. Proses enkripsi lebih cepat akan lebih baik karena membutuhkan waktu yang relative singkat dan proses derkipsi lebih lama akan lebih baik karena membutuhkan waktu untuk mengetahui kunci yang digunakan dalam proses pengacakan. Hasil dapat dilihat bersarkan Tabel 3.

Tabel 3. Hasil Pengujian Waktu Enkripsi

Size File	AES Standar (detik)	AES Modifikasi 6x6 pada Plain (detik)	AES Modifikasi 10x10 pada Plain (detik)	AES Modifikasi 6x6 pada Plain dan Key (detik)	AES Modifikasi 10x10 pada Plain dan Key (detik)
(1)	(2)	(3)	(4)	(5)	(6)
11.003 *bytes*	195	465	534	409	658
11.993 *bytes*	790	1458	2251	1439	1884
12.598 *bytes*	3991	7620	7667	8639	10124
13.827 *bytes*	5530	9886	12628	9951	12473
14.587 *bytes*	6450	12006	14733	12118	13782
Rata-rata	3391,2	6287	7562,6	6511,2	7784,2

Pada Tabel 3 menunjukkan hasil dengan rata-rata eksekusi waktu enkripsi tercepat pada algoritma AES standar, yaitu 3391,2/detik. Perbedaan waktu eksekusi yang didapatkan antar algoritma dikarenakan adanya penambahan proses subtitusi *polybius* untuk algoritma modifikasi sehingga mempengaruhi waktu eksekusi pada proses enkripsi yang berarti kompleksitas algoritma berbanding lurus dengan proses waktu komputasi.

Tabel 4. Hasil Pengujian Waktu Dekripsi

Size File	AES Standar (detik)	AES Modifikasi 6x6 pada Plain (detik)	AES Modifikasi 10x10 pada Plain (detik)	AES Modifikasi 6x6 pada Plain dan Key (detik)	AES Modifikasi 10x10 pada Plain dan Key (detik)
11.003 *bytes*	260	608	1726	542	863
11.993 *bytes*	1145	2404	2961	2569	3208
12.598 *bytes*	6399	10992	14564	11544	16851
13.827 *bytes*	8913	15194	19539	15144	18183
14.587 *bytes*	10000	20174	22400	18862	22400
Rata-rata	5502,6	9874,4	12238	9732,2	12301

Pada Tabel 4, hasil menunjukkan bahwa AES modifikasi plainteks dengan matriks 10x10 yaitu 12301/detik memiliki performa waktu yang baik. Secara keseluruhan untuk setiap algoritma dapat dikatakan baik dari segi performa waktu karena hasil performa waktu enkripsi lebih rendah dibandingkan waktu proses dekripsi untuk masing-masing algoritma.

4. Kesimpulan

Algoritma dilakukan pengujian *avalanche* effect dengan persentase tertinggi terdapat pada algoirtma modifikasi menggunakan matriks 6x6 pada plainteks dan kunci yaitu sebesar 51,8% yang berarti perubahan bit terbanyak terdapat pada algoritma modifikasi dengan hasil pengujian chi square yang menunjukkan AES modifikasi yang digunakan dapat meningkatkan *avalanche effecti* sebesar 5%. Performa waktu enkripsi dan dekripsi yang dimiliki AES modifikasi membutuhkan waktu yang relative lama dibandingkan AES standar dikarenakan kompleksitas algoritma yang digunakan mempengaruhi waktu komputasi.

Referensi

[1] A. R. Tulloh, Y. Permanasari, and E. Harahap, “Kriptografi Advanced Encryption Standard (AES) Untuk Penyandian File Dokumen,” *J. Mat. UNISBA*, vol. 2, no. 1, pp. 118–125, 2016.
[2] T. Des, D. A. N. Algoritma, A. E. S. Dalam, and P. File, “Konferensi Nasional Ilmu Komputer (KONIK) 2014 Kombinasi Algoritma Triple Des Dan Algoritma AES DALAM Pengamanan File,” no. July 2016, 2014.
[3] P. Studi, T. Informatika, and U. Muhammadiyah, “Studi Terhadap Advanced Encryption Standard (Aes) Dan.”
[4] R. Munir, *Kriptografi*. Bandung: INFORMATIKA, 2006.
[5] A. Ilmiah, “Pemenuhan Prinsip Shannon (Confussoin dan Diffusion) pada Block Cipher dengan Pola Anyaman Rambut Papua (ARAP) menggunakan Constanta Bilangan Prima Pemenuhan Prinsip Shannon (Confussoin dan Diffusion) pada Block Cipher dengan Pola Anyaman Rambut Pap,” no. April, 2016.
[6] T. S. Kondo and L. J. Mselle, “An Extended Version of the Polybius Cipher,” vol. 79, no. 13, pp. 30–33, 2013.
[7] P. Kumar and S. B. Rana, “Development of modified AES algorithm for data security,” *Optik (Stuttg).*, vol. 127, no. 4, pp. 2341–2345, 2016.
[8] Z. Rahman, M. B. Science, A. D. Corraya, M. B. Science, M. A. Sumi, and M. B. Science, “A Novel Structure of Advance Encryption Standard (AES) with 3-Dimensional S- box , RSA

based Key Scheduling and modified 3-Dimensional Polybius Cube Encipherment A Novel Structure of Advance Encryption Standard (AES) with," no. February 2017, pp. 0–8, 2016.
[9] A. Karima, M. N. Diyatan, T. Informatika, F. Ilmu, K. Universitas, and D. Nuswantoro, "Algoritma Kriptografi Gost Dengan Implementasi," vol. 15, no. 4, pp. 292–302, 2016.
[10] R. Rumani, "Desain Dan Implementasi Aplikasi Sms (Short Message Service) Pada Android Menggunakan Algoritma Aes," *e-Proceeding Eng.*, vol. 2, no. 2, pp. 3318–3326, 2015.
[11] E. B. Perolihin, "Penerapan Transpose Matriks pada Algoritma Kriptografi Aes untuk Content Database," 2018.
[12] S. Onchiri, "Conceptual model on application of chi-square test in education and social sciences," vol. 8, no. 15, pp. 1231–1241, 2013.
[13] M. A. Salim, "Analisa Algoritma AES Modifikasi dengan Teknik Blum Blum Shub - Chaotic Function dan Modifikasi ShiftRows," pp. 1–79, 2017.
[14] E. H. A. Mendrofa and M. Zarlis, "Implementasi Algoritma RSA dengan Kunci EM2B dalam Mengenkripsi Pesan," pp. 155–163, 2017.