Digital Transformation Technology (Digitech) | e-ISSN : 2807-9000
Volume 3, **Number** 1, Maret 2023
https://doi.org/10.47709/digitech.v3i1.2293

Mengenal *Advance Encrytion Standard* (AES) Sebagai Algoritma Kriptografi Dalam Mengamankan Data

Nur Wachid Hidayatulloh[1*], Muhlis Tahir[2*], Husnul Amalia[3], Nanda Afdlolul Basyar[4], Ahmad Faizal Prianggara[5], Moh Yasin[6]
[1,2,3,4,5,6]Pendidikan Informatika, Fakultas Ilmu Pendidikan, Universitas Trunojoyo Madura, Indonesia
[1]nwachid5833@gmail.com, [2]muhlis.tahir@trunojoyo.ac.id, [3]husnulamalia661@gmail.com, [4]nandaafd.info@gmail.com, [5]faizalanggara93@gmail.com, [6]yasinefyu127@gmail.com

Histori Artikel:
Diajukan: 5 Mei 2023
Disetujui: 10 Mei 2023
Dipublikasi: 10 Mei 2023

Kata Kunci:
AES; Kriptografi; Rijdnael; Pengamanan Data;

Digital Transformation Technology (Digitech) *is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0).*

Abstrak
Keamanan dan kerahasiaan sebuah data adalah suatu hal yang sangat penting. Keamanan sebuah data ini tidak terlepas dengan adanya pengamanan sebuah data. Pada rumpun ilmu komputer, terdapat banyak wujud pengamanan data salah satunya *Advance Encryption Standard* (AES). AES merupakan salah satu bentuk pengamanan data yang dapat dilakukan untuk mengamankan data dari gangguan atau serangan orang lain. AES sendiri merupakan salah satu bentuk kriptografi yang terdiri atas beberapa proses dalam melakukan enkripsi sebuah data. Proses enkripsi dalam AES sendiri yaitu *Add Round Key*, *Sub Bytes*, *Shift Rows*, dan *Mix Coloumn*. Hasil dari penelitian ini adalah pengenalan algoritma AES dalam kriptografi pada sebuah *plain text* di *round* 0 dan sebesar 128 bit dengan menggunakan perhitungan matematis dari algoritma Rijdnael.

PENDAHULUAN

Era digital saat ini, pertukaran data melalui jaringan komputer telah menjadi hal yang sangat umum. Namun, dengan semakin banyaknya data yang dipertukarkan melalui jaringan, keamanan data menjadi semakin penting dan krusial. Data yang tidak diamankan dengan baik dapat diakses oleh pihak yang tidak berwenang, diubah atau dihapus, atau bahkan digunakan untuk tujuan kriminal.

Teknologi enkripsi data adalah salah satu cara untuk menjaga keamanan dan kerahasiaan data yang dipertukarkan melalui jaringan komputer. Enkripsi adalah sebuah proses mengubah pesan dari pesan yang dapat dipahami (*plaintext*) menjadi pesan yang tidak dapat dipahami (*ciphertext*) dengan menggunakan sebuah kunci (*key*) (Murdowo, 2014). Hanya orang yang memiliki kunci enkripsi yang tepat yang dapat membaca atau mengakses kembali data asli.

Salah satu algoritma enkripsi data yang populer dan sering digunakan pada berbagai aplikasi dan sistem keamanan data adalah *Advance Encryption Standard* (AES). AES adalah sebuah algoritma enkripsi simetris yang digunakan untuk mengamankan data dalam bentuk digital (Pabokory et al., 2016). Algoritma ini terdiri dari beberapa jenis kunci enkripsi yang berbeda, yang masing-masing memiliki panjang bit yang berbeda. AES telah terbukti efektif dalam menjaga kerahasiaan dan keamanan data, sehingga sering digunakan pada sistem keamanan data dan aplikasi (Pabokory et al., 2016).

Pada algoritma AES dalam melakukan enkripsi sebuah data, terdapat perhitungan matematis di dalamnya. Perhitungan tersebut adalah penjumlahan dan perkalian (Murdowo, 2014). Hal ini dikarenakan perhitungan algoritma kriptografi AES berbeda dengan perhitungan matematis pada umumnya. Oleh karena itulah, penelitian ini akan mengenalkan AES sebagai salah satu algoritma kriptografi dalam melakukan pengamanan data dengan perhitungan matematis Rijdnael.

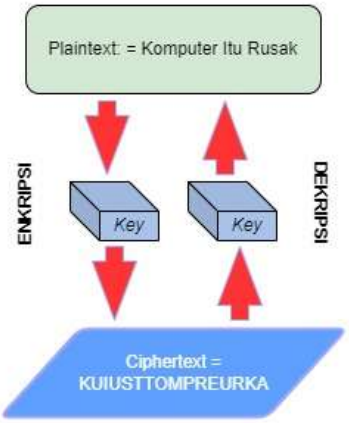
STUDI LITERATUR

1. Kriptografi

a. Definisi Kriptografi

Menurut (Cristy & Riandari, 2021) kriptografi berasal dari bahasa Yunani yaitu *crypto* yang memiliki arti rahasia dan *graphia* yang berarti tulisan. Kriptografi juga didefinisikan oleh (Sitepu et al., 2022) bahwa kriptografi adalah ilmu yang mempelajari seni mengamankan pesan atau data dan informasi sehingga pesan dan data tersebut aman untuk dikirimkan ke tujuannya. Selain itu, terdapat definisi lain menurut Munir dalam (Ziliwu et al., 2022) yaitu cabang ilmu yang mempelajari teknik komputasi yang berkaitan dengan masalah keamanan informasi seperti kerahasiaan, integritas data,

dan otentikasi. Berdasarkan beberapa definisi tersebut, maka dapat disimpulkan bahwa kriptografi adalah cabang ilmu yang mempelajari beberapa teknik komputasi yang berkaitan dengan aspek keamanan informasi guna menjaga kerahasiaan pesan. Menurut (Azhari et al., 2022) terdiris atas 2 proses yakni enkripsi dan dekripsi. Adapun contoh ilustrasi dari kriptografi pada gambar 1 di bawah ini.



Gambar 1. Ilustrasi Kriptografi

b. Sejarah Kriptografi
Pada artikel yang ditulis oleh (Azhari et al., 2022) dan berjudul "Implementasi Pengamanan Data pada Dokumen Menggunakan Algoritma Kriptografi *Advanced Encryption Standard* (AES)" menjelaskan sejarah dari kriptografi sebagai berikut.

"Sebagian besar sejarah kriptografi berawal dari kriptografi klasik, yakni metode enkripsi menggunakan kertas dan pensil atau mungkin dengan bantuan alat mekanis sederhana. Secara umum, algoritma kriptografi klasik terbagi dalam dua kategori, yaitu algoritma transposisi dan algoritma substitusi. Algoritma transposisi adalah algoritma yang mengubah urutan huruf dalam pesan, sedangkan algoritma substitusi adalah algoritma yang mengganti setiap huruf atau kelompok huruf dengan huruf atau kelompok huruf yang berbeda."

c. Tujuan Kriptografi
Ilmu kriptografi tercipta tidak hanya sekedar dalam menjaga keamanan sebuah pesan yang akan dikirimkan. Adanya ilmu kriptografi pada keamanan jaringan juga memiliki tujuan bagi para pengguna. Adapun dasar-dasar tujuan adanya ilmu kriptografi ini yaitu (Ziliwu et al., 2022)
1) Kerahasiaan: Ilmu kriptografi bertujuan agar isi pesan tidak diketahui orang lain selama proses pengiriman.
2) Integritas Data: pesan yang diterima utuh dan dan tidak mengalami modifikasi apapun selama proses pengiriman.
3) Otentikasi: layanan kriptografi mengenai identifikasi terlebih dahulu antara pengirim dan penerima pesan.
4) Anti Penyangkalan: menghindari pihak yang menyampaikan suatu penyangkalan dimana pengirim pesan menyangkal telah mengirim pesan dan sebaliknya penerima pesan tidak mengakui telah menerima pesan tersebut.

d. Elemen Sistem Kriptografi
Elemen-elemen pada ilmu kriptografi diantaranya (Condro, 2015):
1) *Plain text*: pesan atau sumber yang pertama dibuat oleh *user* (pengguna); pesan yang dapat dibaca oleh semua orang.
2) *Cipher text*: pesan *plain text* yang telah diubah bentuknya menjadi lebih aman sehingga tidak dapat dibaca.
3) *Cryptographic algorithm*: langkah-langkah yang digunakan berdasarkan operasi matematika untuk mengubah *plain text* menjadi *cipher text*.
4) *Key*: kunci yang digunakan didasarkan pada *cryptographic algorithm* untuk melakukan proses enkripsi dan dekripsi terhadap pesan yang dikirim. Hal ini berarti hanya pengguna yang memiliki kunci yang dapat melakukan dekripsi pesan dalam bentuk *ciphertext*.

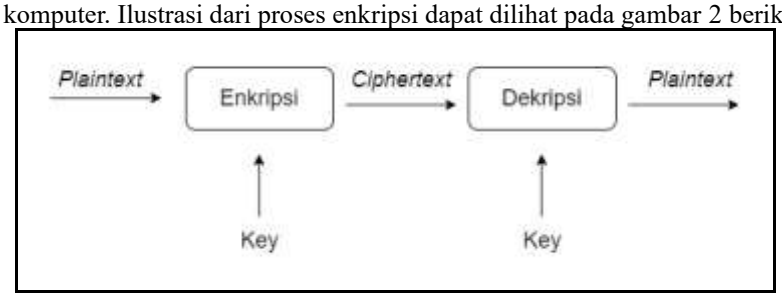
2. Enkripsi

a. Definisi Enkripsi
Enkripsi merupakan Proses yang dilakukan untuk mengamankan pesan (*plain text*) hingga menjadi pesan tersembunyi (Kusumo, 2019). Selain itu enkripsi juga didefinisikan sebagai keamanan data

yang dikirimkan agar terjaga kerahasiaannya (Abdul et al., 2019). Pada sebuah keamanan jaringan, enkripsi sering disebut sebagai *ciphertext* atau kode. *Ciphertext* memiliki definisi yakni sebuah pesan yang tidak dapat dibaca dengan mudah (Kusumo, 2019).

b. Proses Enkripsi

Pada sebuah keamanan jaringan, enkripsi memiliki proses sehingga data dari pengguna tidak dapat terbaca dengan mudah. Proses tersebut juga masuk ke dalam cara kerja enkripsi pada sebuah keamanan komputer. Ilustrasi dari proses enkripsi dapat dilihat pada gambar 2 berikut.



Gambar 2. Proses Enkripsi

3. *Advance Encryption Standard (AES)*

a. Sejarah AES

Pada artikel yang ditulis oleh (Murdowo, 2014) dan berjudul "Mengenal Proses Perhitungan Enkripsi Menggunakan Algoritma Kriptografi Advance Encryption Standard (AES) Rijdnael" menjelaskan sejarah dari AES sebagai berikut.

"Pada tahun 1997, *National Institute of Standard and Technology* (NIST) mengeluarkan *Advance Encryption Standard* (AES) untuk menggantikan *Data Encryption Standard* (DES). AES dikembangkan dengan tujuan memastikan tata kelola di berbagai bidang. Algoritma AES dirancang untuk menggunakan minimum blok input enkripsi 128-bit dan mendukung 3 ukuran kunci yaitu 128-bit, 192-bit, dan 256-bit. Di Agustus 1998, NIST mengumumkan bahwa 15 proposal AES telah diterima dan dievaluasi setelah melalui proses seleksi algoritma yang masuk. Di tahun 1999, NIST mengumumkan bahwa hanya 5 algoritma yang diterima. Algoritma tersebut yaitu RC6, MARS, Snake, Rijdnael dan Twofish. 5 algoritma ini akan menjalankan berbagai pengujian. Di bulan Oktober 2000, Rijndael diumumkan sebagai algoritma pilihan untuk standar AES yang baru."

b. Pengantar Matematis

Bit dalam algoritma AES diinterpretasikan sebagai elemen *finite field* (Murdowo, 2014). Elemen-elemen pada AES dapat dikalikan dan dijumlahkan, akan tetapi hasil penjumlahan dan perkalian elemen ini sangat berbeda dengan hasil penjumlahan dan perkalian bilangan biasa. Adapun penjelasan dari tiap perhitungan matematis ini sebagai berikut.

1) Penjumlahan

Menurut (Murdowo, 2014) Penjumlahan dua elemen dalam *finite field* dilakukan dengan menjumlahkan koefisien pangkat polinomial masing-masing dari kedua elemen. Penjumlahan dalam AES dengan dilakukannya operasi XOR dan dinotasikan dengan "Å". Dengan operasi ini, maka "1Å1 = 0, 1Å0 = 1, 0Å1 = 1, dan 0Å0 = 1". Adapun contoh operasi penjumlahan pada AES sebagai berikut.

$$(x^5 + x^4 + x^2 + x) Å\ (x^4 + x^3 + x + 1) = x^5 + x^3 + x^2 + 1 \text{ (notasi polinomial)}$$
$$\{00110110\} Å\ \{00011010\} = \{00101101\} \text{ (notasi biner)}$$
$$\{36\} Å\ \{1A\} = \{2D\} \text{ (notasi heksadesimal)}$$

2) Perkalian

Menurut (Murdowo, 2014) perkalian dalam AES merupakan representasi dari polynomial yang dinotasikan dengan • mengacu pada perkalian polinomial modulo dari *irreducible polynomial* derajat 8. *Irreducible polynomial* pada AES yang digunakan sebagai berikut.

$$m(x) = (x^8 + x^4 + x^3 + x + 1)$$

Hasil dari modular oleh $m(x)$ nantinya akan berupa polinomial biner dengan derajat kurang dari 8, sehingga dapat dipresentasikan dengan 1 *byte* saja. Sebagai contoh, pada notasi $\{42\} \bullet \{91\} = \{62\}$ karena

$$(x^6 + x) \bullet (x^7 + x^4 + 1)$$
$$= (x^{13} + x^{10} + x^6 + x^8 + x^5 + x)$$

dan

$$(x^{13} + x^{10} + x^6 + x^8 + x^5 + x)\ modulo\ (x^8 + x^4 + x^3 + x + 1)$$
$$= x^6 + x^5 + x$$

c. Kelebihan AES
Berdasarkan artikel yang ditulis oleh (Asriyanik, 2017) menyatakan bahwa terdapat beberapa kelebihan dari AES yang ditemukan, diantaranya:
1) Panjang kunci minimal pada AES adalah 128 bit. Sehingga dengan teknologi yang sekarang, AES tahan terhadap serangan *exhaustive key lookup*. Dengan panjang kunci 128 bit adalah $2^{128} \approx 3{,}4 \times 1038$ kemungkinan larangan.
2) Kekuatan AES terletak pada sifat karakteristik bidang GF(2^8), di mana untuk setiap bilangan prima selalu ada satu bidang unik hingga sehingga semua representasi GF(2^8) adalah isomorfik dan pemilihan polynomial biner derajat 8.

d. Kelemahan AES
Berdasarkan artikel yang ditulis oleh (Asriyanik, 2017) menyatakan bahwa terdapat beberapa kelemahan dari AES yang ditemukan, diantaranya:
1) Kesulitan dalam manajemen kunci muncul dengan jenis kunci simetris. Ini karena diperlukan kunci yang berbeda untuk setiap pengiriman dan penerimaan data dengan pengguna yang berbeda.
2) AES merupakan salah satu algoritma kriptografi dengan tipe kunci simetris dalam proses pengiriman dan penerimaan data. Hal ini menyebabkan kunci simetris mudah bocor meski dalam jangka waktu yang lama.

METODE

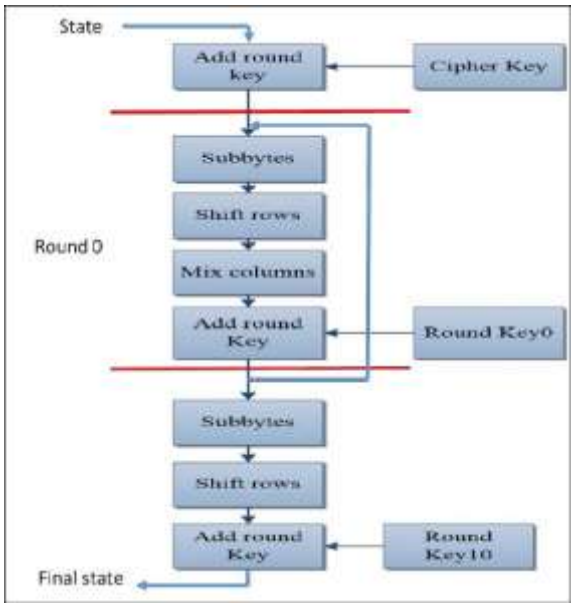
Penelitian ini menggunakan metode penelitian terapan, yaitu jenis penelitian yang bertujuan memberikan solusi praktis terhadap masalah tertentu (Widyastuti et al., 2019). Menurut (Guritno dalam Widyastuti et al., 2019) metode ini tidak berfokus pada pengembangan suatu gagasan, teori atau gagasan, tetapi lebih fokus pada penerapan dalam kehidupan sehari-hari, ciri khas dari penelitian ini adalah rendahnya tingkat abstraksi dan manfaat atau efek yang dapat diperoleh dan dirasakan secara langsung. Pemilihan algoritma AES didasarkan pada asumsi dasar bahwa AES merupakan kelanjutan dari algoritma DES (*Data Encryption Standard*) yang masa berlakunya telah habis karena alasan keamanan (Munawar dalam Widyastuti et al., 2019).

Algoritma AES adalah blok *ciphertext* simetris yang dapat mengenkripsi (enkrip) dan mendekripsi (dekrip) informasi (Pabokory et al., 2016). Algoritma AES menggunakan kunci kriptografi 128, 192, dan 256-bit untuk mengenkripsi dan mendekripsi data dalam blok 128 bit (Asriyanik, 2017). Setiap blok dienkripsi dalam sejumlah putaran tertentu. Sehingga terdapat beberapa variasi AES yang dikenal sebagai AES 128, AES 192, dan AES 256 (Munawar dalam Zailani & Alwan, 2017). Jumlah putaran tiap blok yang ditunjukkan dengan tabel 1 (Munir dalam Prameshwari & Sastra, 2018) di bawah ini.

Tabel 1
Putaran Tiap Blok

Varian AES	**Panjang Kunci (*Nk* words)**	**Ukuran Blok (*Nb words*)**	**Jumlah Putaran (*Nr*)**
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14

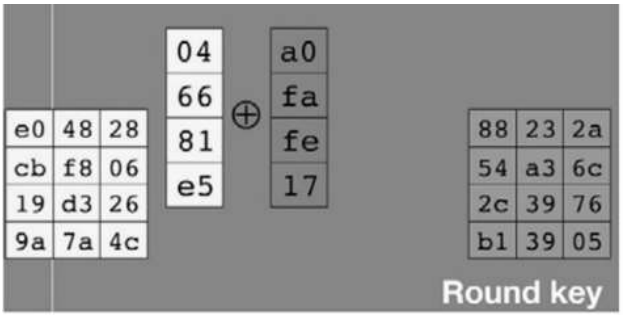
Pemilihan ukuran blok data dan kunci tersebut akan menentukan jumlah proses yang harus dilalui untuk proses enkripsi. Ilustrasi algoritma AES pada kunci 128 bit ditunjukkan pada gambar 3 di bawah ini.



Gambar 3. Algoritma AES-128 bit (Munir dalam Prameshwari & Sastra, 2018)

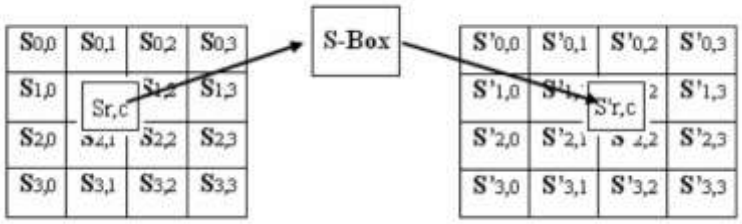
Proses enskripsi algoritma AES ini mengikuti tahapan yang tertera pada gambar 3. Adapun beberapa penjelasan tahapan dalam algoritma ini yaitu:

a. *Add Round Key*
Pada dasarnya *add round key* terdiri dari kombinasi *ciphertext* yang ada dan kunci enkripsi menggunakan operasi XOR. Adapun ilustrasi tahap *add round key* ditunjukkan pada gambar 4.



Gambar 4. Ilustrasi *Add Round Key* (Murdowo, 2014)

b. *Sub Bytes*
Sub Bytes adalah menukar isi matriks yang disebut dengan *Rijndael S-Box*. Ilustrasi tahap *sub bytes* dapat dilihat pada gambar 5 berikut.



Gambar 5. Ilustrasi *Sub Bytes* (Ibrahim, 2017)

Pada artikel yang ditulis oleh (Jayana et al., 2022) mengatakan bahwa terdapat prosedur dari tahapan ini sebagai berikut.

1) Mengambil salah satu isi kotak matriks dan membandingkannya dengan angka di sebelah kiri sebagai baris dan angka di sebelah kanan sebagai kolom.

2) Setelah mengetahui baris dan kolom, selanjutnya kita mendapatkan isi tabel dari *Rijdnael S-Box*.
3) Mengubah seluruh blok cipher menjadi blok baru dari hasil pertukaran semua isi blok.

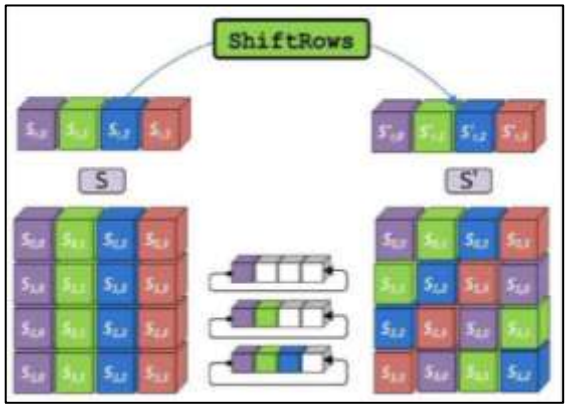
Adapun tabel *S-Box* yang dapat dilihat pada tabel 2 di bawah ini.

Tabel 2
S-Box

	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

c. *Shift Rows*

Shift Rows adalah proses menggeser atau memindahkan setiap blok atau elemen tabel baris demi baris (Jayana et al., 2022). Artinya, baris pertama tidak digeser, baris kedua digeser 1 *byte*, baris ketiga digeser 2 *byte*, dan baris keempat digeser 3 *byte*. Pergeseran dalam blok adalah pergeseran kiri dari setiap elemen sesuai dengan jumlah *byte* yang digeser. Setiap pergeseran 1 *byte* berarti pergeseran ke kiri. Gambar 6 merupakan ilstrasi pada tahap *shift rows*.



Gambar 6. Ilustrasi *Shift Rows*

d. *Mix Coloumn*

Mix Column terdiri dari mengalikan setiap elemen dari blok cipher (Jayana et al., 2022). Perkalian dilakukan seperti perkalian matriks normal dengan perkalian titik, dan kedua perkalian tersebut dimasukkan ke dalam cipher blok baru. Blok matriks yang digunakan dalam proses perkalian sebagai berikut.

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$$ (Blok Matriks AES)

HASIL

Proses pengamanan data pada algoritma AES bergantung pada ukuran kunci yang dipilih. Penelitian ini menggunakan ukuran kunci sebesar 128 bit. Sehingga jumlah *round* yang dilakukan sebanyak 10 *round* sebagai berikut.

Plain Text : KJKKELOMPOK06PIF

Chiper Key : 2019202300000000

Dari *plain text* dan *chipper text* tersebut, maka langkah awal adalah memasukkan ke dalam kolom 4 ×4 sebagai berikut.

Plain Text

K	J	K	K
E	L	O	M
P	O	K	0
6	P	I	F

Chiper Key

2	0	1	9
2	0	2	3
0	0	0	0
0	0	0	0

Kemudian keduanya dikonversikan ke dalam bilangan heksadesimal menggunakan tabel ASCII, dan hasil konversi sebagai berikut.

Hasil Konversi *Plain Text*

4B	4A	4B	4B
45	4C	4F	4D
50	4F	4B	30
36	50	49	46

Hasil Konversi *Chiper Key*

32	30	31	39
32	30	32	33
30	30	30	30
30	30	30	30

Selanjutnya dilakukan konversi ke dalam bilangan biner, dan hasil konversi sebagai berikut.

Konversi Biner *Plain Text*

01001011	01001010	01001011	01001011
01000101	01001100	01001111	01001101
01010000	01001111	01001011	00110000
00110110	01010000	01001001	01000110

Konversi Biner *Chiper Key*

00110010	00110000	00110001	00111001
00110010	00110000	00110010	00110011
00110000	00110000	00110000	00110000
00110000	00110000	00110000	00110000

Initial Round

Berdasarkan algoritma yang ada, proses enkripsi pada AES dimulai pada *initial round* sebagai proses awal data akan dilakukan enskripsi. *Initial round* yaitu tahap *add round key* yang dilakukan dengan operasi XOR (*Exlusive* OR) antara *state* awal dari data masukan (*plaintext*) dengan *cipher key* (kunci cipher) (Asriyanik, 2017). Adapun hasil dan proses *add round key* dari tahap ini sebagai berikut.

Add Round Key
Tahap pertama ini sebagai berikut.
4B XOR 32 = 01001011 XOR 00110010 = 01111001
45 XOR 32 = 01000101 XOR 00110010 = 01110111
50 XOR 30 = 01010000 XOR 00110000 = 01100000
36 XOR 30 = 00110110 XOR 00110000 = 00000110

Dengan cara yang sama, maka didapatkan hasil biner sebagai berikut.

01111001	01111010	01111010	01110010
01110111	01111100	01111101	01111110
01100000	01111111	01111011	00000000
00000110	01100000	01111001	01110110

Dari hasil biner tersebut, dilakukan konversi ke bentuk heksadesimal sehingga hasil pada tahap *initial round* sebagai berikut.

79	7A	7A	72
77	7C	7D	7E
60	7F	7B	00
06	60	79	76

Round 0

Pada *round* ini hasil akhir *initial round* akan dilanjutkan ke tahap selanjutnya sesuai dengan algoritma yang ada yakni *sub bytes, shift rows, mix coloum,* dan *add round key*. Hasil dari tiap tahap sebagai berikut.

Sub Bytes

Tahap ini, melakukan transformasi dari hasil yang didapatkan dengan tabel S-Box. Salah satu contoh transformasi pada tahap ini ditunjukkan pada gambar 7 di bawah ini.

	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 7. Penentuan *State* dengan *S-Box*

Sehingga hasil pada tahap ini sebagai berikut.

79	7A	7A	72
77	7C	7D	7E
60	7F	7B	00
06	60	79	76

Proses *Sub Bytes* →

B6	DA	DA	40
F5	10	FF	F3
D0	D2	21	63
6F	3C	B6	38

Shift Rows

Tahap ini melakukan pergeseran pada tiap kolom sesuai dengan ketentuan yang ada. Sehingga hasil dari tahap ini sebagai berikut.

B6	DA	DA	40
F5	10	FF	F3
D0	D2	21	63
6F	3C	B6	38

Proses *Shift Rows* →

B6	DA	DA	40
10	FF	F3	F5
21	63	D0	D2
38	6F	3C	B6

Mix Coloumn

Tahap ini akan dilakukan perkalian dari hasil akhir tahap sebelumnya dengan blok matriks polinomial tetap. Sebagai contoh perhitungan, berikut contoh perhitungan pada bit pada baris 1 kolom 1 ($S^1_{0,0}$).

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \times \begin{bmatrix} B6 \\ 10 \\ 21 \\ 38 \end{bmatrix} =$$

$02 \times B6 = 00000010\ XOR\ 10110110$
$= x \bullet x^7 + x^5 + x^4 + x^2 + x$
$= x^8 + x^6 + x^5 + x^3 + x^2\ mod\ x^8 + x^4 + x^3 + x + 1$
$= x^6 + x^5 + x^4 + x^2 + x + 1$
$= 01110111$
$03 \times 10 = 00000011\ XOR\ 00010000$
$= (x + 1) \bullet (x^4)$
$= x^5 + x^4 = 00110000$
$01 \times 21 = 00000001\ XOR\ 00100001$
$= 1 \bullet x^5 + 1$
$= 00100001$
$01 \times 38 = 00000001\ XOR\ 00111000$
$= 1 \bullet x^5 + x^4 + x^3$
$= 00111000$

Kemudian hasil perkalian tersebut dijumlahkan sebagai berikut.
01110111
00110000
00100001
00111000
01011110 = 5F
Dengan cara yang sama, maka hasil dari *mix coloumn* sebagai berikut.

5F	B9	4D	E0
CD	33	70	4A
AC	12	D6	CB
80	37	2E	90

Add Round Key
Tahap akhir pada *round* 0 ini adalah melakukan *add round key* dari hasil *mix coloumn* dengan hasil pada *initial round*. Sehingga hasil pada tahap ini sebagai berikut.

5F	B9	4D	E0
CD	33	70	4A
AC	12	D6	CB
80	37	2E	90

$\oplus$

79	7A	7A	72
77	7C	7D	7E
60	7F	7B	00
06	60	79	76

Hasil dari proses XOR di atas sebagai berikut.

26	C3	37	92
BA	4F	0D	34
CC	6D	AD	CB
86	57	57	E6

PEMBAHASAN

Berdasarkan hasil dari proses enkripsi dengan AES, didapatkan bahwa penelitian ini menggunakan ukuran *key* 128 bit atau dikenal dengan AES-128. Sehingga AES ini akan melakukan proses enkripsi sebanyak 10 *round.* Namun, penelitian ini hanya menyelesaikan proses enkripsi pada *round 0.* Pada *round* ini memiliki 2 proses yakni *initial round* dan *round 0* yang mencakup proses *sub bytes, shift rows, mix coloumn,* dan *add round key*.

Pada tahap *initial round, plaintext* KJKKELOMPOK06PIF akan dilakukan proses *add round key* dengan *cipher key* yakni 2019202300000000. Sehingga pada tahap ini memiliki hasil yaitu 797760067A7C7F607A7D7B79727E0076. Hasil ini didapatkan dengan melakukan operasi XOR pada setiap *state*-nya. Kemudian hasil ini diteruskan pada *round 0.* Sehingga hasil pada *round* ke-0 yakni 26BACC86C34F6D57370DAD579234CB90. Hasil ini didapatkan setelah melakukan seluruh algoritma AES dari Rijdnael yaitu melalui *sub bytes, shift rows, mix coloumn,* dan *add round key* pada setiap state-nya. Proses enkripsi pada penelitian ini belum selesai. Dalam memenuhi algoritma AES secara keseluruhan, maka hasil akhir *round 0* diteruskan ke *round 1* hingga ke *round* 10 sesuai algoritma AES Rijdnael di setiap *state*-nya. Perhitungan setiap *state* pada *round* 1 hingga *round* terakhir dilanjutkan dengan cara yang sama hingga didapatkan hasil akhir dari AES yang diharapkan yaitu sebuah pesan yang tidak dapat dibaca.

KESIMPULAN

Berdasarkan hasil dan pembahasan yang telah dijelaskan, maka dapat disimpulkan bahwa algoritma kriptografi Rijdnael yang ditetapkan oleh NIST sebagai AES memiliki ciri khas yang menjadikannya istimewa. Seperti halnya kriptografi yang melindungi sebuah informasi, AES juga memiliki tujuan melindungi informasi dari orang yang tidak menerima dengan serangkaian ronde yang dilakukan menggunakan kunci simteris. Algoritma Rijdnael memiliki ciri khas istimewa, dikarenakan pada setiap ronde proses enkripsi melalui beberapa tahapan yaitu *add round key, sub bytes, shift rows,* dan *mix coloumn*.

Penelitian ini memiliki hasil enkripsi dari AES dengan ukuran kunci yang dipilih sebesar 128 bit. Ukuran ini yang menentukan jumlah putaran dalam proses enkripsi. Jumlah putaran pada penelitian ini sebanyak 10 *round*. Hasil penelitian ini adalah enkripsi pada ronde ke-0 melalui proses perhitungan dari algoritma Rijdnael. Dalam melanjutkan proses enkripsi ini, hasil akhir dari ronde ke-0 dilanjutkan sesuai dengan algoritma AES yang melalui proses *round key* hingga *mix coloumn* pada setiap *state* dengan melakukan XOR pada *expand key* dari kunci pertama hingga seterusnya pada ronde ke-10 dan didapatkannya hasil enkripsi.

REFERENSI

Abdul, D. F., Budiman, M. I., & Kurniawan, T. (2019). *Analisis Sistem Keamanan Sistem Operasi (Windows , Linux , MacOS)*. https://www.researchgate.net/publication/331562729_Analisis_Sistem_Keamanan_Sistem_Operasi_Windows_Linux_MacOS

Asriyanik. (2017). Studi Terhadap Advanced Encrytion Standard (AES) dan Algoritma Knapscak dalam Pengamanan Data. *Jurnal Ilmiah Sains Dan Teknologi*, *7*(1), 553–561.

Azhari, M., Mulyana, D. I., Perwitosari, F. J., & Ali, F. (2022). Implementasi Pengamanan Data pada Dokumen Menggunakan Algoritma Kriptografi Advanced Encryption Standard (AES). *Jurnal Pendidikan Sains Dan Komputer*, *2*(1), 163–171. https://doi.org/10.47709/jpsk.v2i01.1390

Condro, P. (2015). *Enkripsi dan Kriptografi*. https://slideplayer.info/slide/2889451/

Cristy, N., & Riandari, F. (2021). Implementasi Metode Advanced Encryption Standard (AES 128 Bit) untuk Mengamankan Data Keuangan. *JIKOMSI [Jurnal Ilmu Komputer Dan Sistem Informasi]*, *4*(2), 75–85. https://ejournal.sisfokomtek.org/index.php/jikom/article/view/181%0A

Ibrahim, A. A. (2017). Perancangan Pengamanan Data Menggunakan Algoritma AES (Advanced Encyption Standard). *Jurnal Teknik Informatika STMIK Antar Bangsa*, *3*(1), 53–60. https://ejournal.antarbangsa.ac.id/index.php/jti/article/view/131

Jayana, M. A., Rafael, D., & Rahman, A. A. (2022). Implementasi Pengamanan Data Pengarsipan Dengan Metode Algoritma Kriptografi AES Studi Kasus pada Bank BJB KCP Pasteur Bandung. *Prosiding Seminar Sosial Politik, Bisnis, Akuntansi Dan Teknik*, *4*, 184. https://doi.org/10.32897/sobat.2022.4.0.1922

Kusumo, D. S. (2019). *Kriptografi, Enkripsi dan Dekripsi*. https://slideplayer.info/slide/13350602/

Murdowo, S. (2014). Mengenal Proses Perhitungan Enkripsi Menggunakan Algoritma Kriptografi Advance Encryption Standard (Aes) Rijdnael. *INFOKAM Nomor I / Th. X/ Maret / 14*, *10*(1), 32–40. http://jurnal.amikjtc.com/index.php/jurnal/article/view/55

Pabokory, F. N., Astuti, I. F., & Kridalaksana, A. H. (2016). Implementasi Kriptografi Pengamanan Data Pada Pesan Teks, Isi File Dokumen, dan File Dokumen Menggunakan Algoritma Advanced Encryption Standard. *Informatika Mulawarman : Jurnal Ilmiah Ilmu Komputer*, *10*(1), 20. https://doi.org/10.30872/jim.v10i1.23

Prameshwari, A., & Sastra, N. P. (2018). Implementasi Algoritma Advanced Encryption Standard (AES) 128 untuk Enkripsi dan Dekripsi File Dokumen. *Eksplora Informatika*, *8*(1), 52. https://doi.org/10.30864/eksplora.v8i1.139

Sitepu, D. A., Nurhayati, & Khair, H. (2022). Implementasi Pengamanan Data Koperasi Menggunakan Algoritma Advanced Encryption Standard (AES). *Jurnal Ilmiah Kaputama (JIKA)*, *6*(1), 49–58. https://citisee.amikompurwokerto.ac.id/assets/proceedings/paper/8__Amikom_Purwokerto_Implementasi_Pengamanan_Data_Koperasi_Menggunakan_Algoritma_Advanced_Encryption_Standard_(Aes).pdf

Widyastuti, S., Ariandi, W., & Sulistiono, V. (2019). Implementasi Kriptografi AES Dalam Pengamanan Data Seleksi Peserta JAMKESMAS. *Jurnal Ilmiah Intech : Information Technology Journal of UMUS*, *1*(2), 13–22. https://doi.org/10.46772/intech.v1i02.66

Zailani, A. U., & Alwan, K. (2017). Penerapan Algoritma AES untuk Keamanan Data (Studi Kasus : CV . Ranger Reload). *Seminar Nasional Teknologi Informaai, Bisnis, Dan Desain 2017*, 6–9.

Ziliwu, K. B., Maslan, A., & Kremer, H. (2022). Implementasi Caesar Cipher pada Algoritma Kriptografi dalam Penyandian Pesan Whatsapp. *Jurnal Comasie*, *7*(2), 117–125.