ANALISIS SISTEM KEAMANAN INFORMASI CHAT ONLINE DENGAN MENGGUNAKAN ALGORITMA AES

Elvaret[1], Noviandi[2]*, Ananda Kurnia Mahesa Naibaho[3], Amanda Putri[4], Yunan Fikri Husaini[5]
[1,2,3,4,5]Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Esa Unggul Jakarta
Jl. Arjuna Utara No. 9 Kb. Jeruk, Jakarta Barat, 11510
Email*: noviandi@esaunggul.ac.id

Abstract

Online chat is an activity between two or more people who communicate with each other by exchanging messages through the use of devices in chat applications and the internet. However, this online chat application has vulnerabilities in the form of wiretapping, modification, and data leakage. One of the currently viral data leak cases is Bjorka which resulted in a lack of user and public trust in information security system services in Indonesia. Therefore, a solution is needed to secure the online chat information security system. The Advanced Encryption Standard (AES) algorithm is a modern cryptography that is symmetrical and has various key lengths such as 128, 192, 256 with a different number of rounds depending on the key length, therefore the AES algorithm is the right solution in maintaining the confidentiality and authenticity of data. AES uses a key that is divided into the same byte blocks on the sending and receiving sides in the encryption and decryption process. The results of the study show that the Advanced Encryption (AES) algorithm can encode the contents of online chats so that they can secure the chats. By applying the AES algorithm to online chat application encryption, unauthorized parties will not get message information because the sent chat is guaranteed that the data will remain confidential.

.

Keywords : *AES, Chat, Cryptography*

Abstrak

Chat online merupakan kegiatan antara dua orang atau lebih yang saling berkomunikasi dengan saling bertukar pesan melalui pemanfaatan gawai pada aplikasi chat dan internet. Akan tetapi aplikasi chat online ini memiliki kerentanan berupa penyadapan, modifikasi, dan kebocoran data. Salah satu kasus kebocoran data yang sedang viral saat ini adalah Bjorka yang mengakibatkan kurangnya tingkat kepercayaan pengguna dan masyarakat terhadap layanan sistem keamanan informasi di Indonesia. Maka dari itu, dibutuhkan solusi untuk mengamankan sistem keamanan informasi chat online. Algortima Advanced Encryption Standard (AES) adalah kriptografi modern yang bersifat simetris dan memiliki panjang kunci yang bervariasi seperti 128, 192, 256 dengan jumlah ronde yang berbeda tergantung panjang kuncinya maka dari itu algoritma AES merupakan solusi yang tepat dalam menjaga kerahasiaan dan keaslian data. AES menggunakan kunci yang terbagi ke dalam blok-blok byte yang sama pada sisi pengirim dan penerima pada proses enkripsi dan deskripsi. Hasil penelitian menunjukan bahwa algoritma Advanced Encryption (AES) dapat menyandikan isi chat online sehingga dapat mengamankan chat tersebut. Dengan menerapkan algoritma AES pada enkripsi aplikasi chat online, pihak yang tidak berwenang tidak akan mendapatkan informasi pesan karena chat yang terkirim terjamin datanya akan tetap rahasia.

Kata Kunci *:* AES, *Chat*, Kriptografi

Pendahuluan

Undang-Undang tentang informasi dan transaksi elektronik (ITE) Nomor 11 Tahun 2008 menyatakan bahwa setiap pengendali data wajib menjaga keamanan dan juga kerahasiannya (Pidana & Bareskrim, 2022). Akan tetapi, di Indonesia banyak terjadi kasus kebocoran data meskipun ada Undang-Undang yang mengatur dan menjamin tentang hal tersebut. Salah satu kasus kebocoran data yang sedang viral saat ini adalah Bjorka. Dampak dari kasus ini mengakibatkan kurangnya tingkat kepercayaan pengguna dan masyarakat terhadap layanan sistem keamanan informasi di Indonesia (It, n.d.).

Untuk menjaga dan mengamankan sistem keamanan informasi diperlukan ilmu kriptografi dalam mengenkripsi dan deskripsi data tersebut (Pratama & Desyani, 2022). Kriptografi modern memiliki tiga jenis yaitu simetris, asimetris, dan hibrida (Sumandri, 2017). Algoritma simetris seperti *Data Encryption Standard* (DES) (Adhar, 2019), *Advance Encryption Standard* (AES) (Putra et al., 2021), dan International *Data Encryption* Algoritma (IDEA) (Nizatsary et al., 2022). Kekurangan dari algoritma DES yaitu hanya bisa mengenkripsikan 64 bit plainteks dengan menggunakan 56 bit internal key atau subkey menjadi 64 bit cipherteks (Donzilio Antonio Meko, 2018). Sedangkan Algoritma IDEA yaitu menggunakan algoritma yang

sama untuk proses enkripsi dan dekripsi dalam beroperasi pada blok pesan dengan lebar 64 bit dan panjang 128 bit (Saragih & Hasugian, 2021).

Kelebihan AES (*Advanced Encryption Standard*) yaitu kecepatan operasi yang lebih tinggi pada jenis kunci simetri (Suhandinata et al., 2019). Algoritma AES menggunakan kunci yang lebih banyak dari algoritma DES dan IDEA yaitu berukuran 128, 192, dan 256 bit pada proses enkripsi dan dekripsi data (Siringoringo, 2020). Serangan *exhaustive key search* mampu tahan pada Algoritma AES (Andini & Mariami, 2020) dan memilki kelebihan di karakteristik sifatnya dari medan GF (2^8), pada sebuah medan tunggal terbatas selalu terdapat bilangan prima sehingga memiliki sifat polinomail dan isomorphic bersifat irreducible yakni tidak dapat dibagi yang berderajat 8 m(x) (Sodikin & Hidayat, 2020).

Graciella Valeska Liander pada penelitian terdahulu dalam Implementasi End-to-End Encryption WhatsApp dengan Metode Penggunaan Algoritma *Elliptic Curve Diffie Hellman* dan AES 256 (Liander & Review, 2022). Pengujian yang dilakukan pada penelitian ini berupa end-to-end encryption pada metode Algoritma Elliptic Curve Diffie Hellman dan AES 256. Penelitian ini menekankan bahwa kombinasi dari penggunaan Algoritma Elliptic Curve Diffie Hellman dan AES 256 dapat meningkatkan keamanan suatu data. Selain itu, pada penelitian yang dilakukan oleh Laila Mustika dalam e-commerce, Untuk Pengamanan Login Dan Data Customer Pada E-Commerce Berbasis Web pada Implementasi Algoritma AES (Mustika, 2020). Mengimplementasikan algoritma AES dalam e-commerce berbasis web yang berfungsi sebagai pengaman data customer dan login.

Meskipun pesan yang telah disimpan kedalam database server dapat berupa pesan yang sudah di enkripsi, masih ada peluang bagi penyerang untuk mencuri informasi selama perjalanan pesan dari client ke server (Ajhari & Windarto, 2018). Alasan ini lah yang membuat metode untuk mengamankan komunikasi digital dengan melakukan enkripsi pesan di client. Dengan menggunakan metode ini, pesan hanya dapat dibaca di device client yang mengirim pesan dan device client yang menerima pesan (Informasi et al., 2021).

.

Metode Penelitian

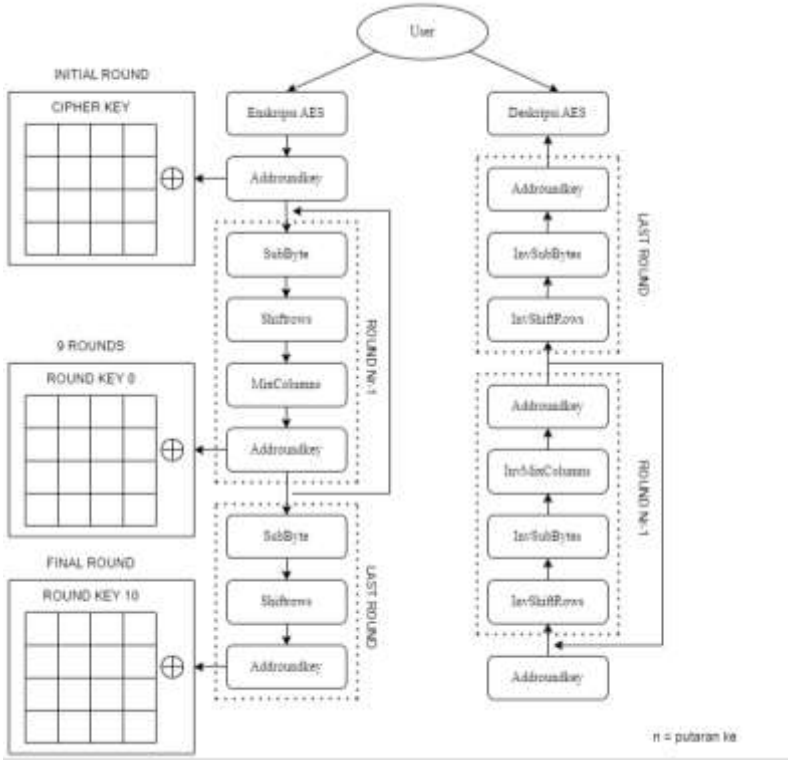
Pada penelitian ini menggunakan jenis penelitian kualitatif dan kuantitatif dengan menggunakan metodologi AES.

Studi Literatur

Tahapan pertama yang dilakukan yaitu melakukan studi literatur untuk mencari berbagai informasi dari sumber digital berupa jurnal, paper, maupun materi perkuliahan mengenai AES 256, end-to-end encryption serta penerapannya pada aplikasi chat online.

Eksperimen

Eksperimen ini dilakukan dengan mengenkripsi dan deskripsi suatu pesan. Terdapat 2 bagian yang akan dilakukan, yaitu sebelum melakukan pertukaran key dan setelah melakukan pertukaran key. Sebelum melakukan pertukaran key, seharusnya tidak terdapat perubahan pesan yang dikirim. Sedangkan, setelah melakukan pertukaran key, pesan akan berubah menjadi kumpulan string yang tidak memiliki makna yang berarti. Algoritma AES memiliki panjang kunci 128, 192 dan 256 bit dan memiliki sistem penyandian blok yang bersifat non-Feistel karena menggunakan panjang blok 128 bit pada komponen yang memiliki invers (Siswanto et al., 2018). Proses yang berulang disebut dengan round pada penyandian AES. Jumlah round yang digunakan bergantung pada panjangnya suatu kunci. Pada setiap round memerlukan kunci dan masukan dari round berikutnya. Kunci round ini di bangkitkan berdasarkan kunci yang di berikan (Muharram et al., 2018).

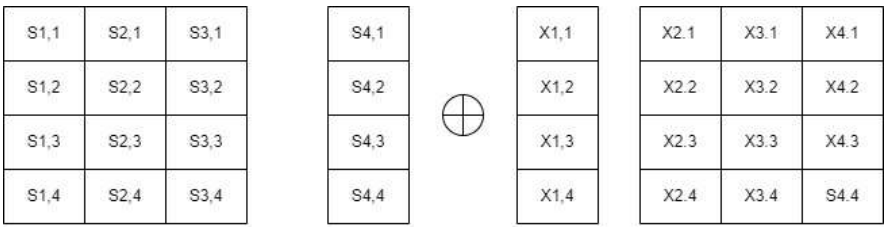


Gambar 1. Flowchart Algoritma AES

Tahapan atau Fase AES dalam Penelitian ini:

1) Addroundkey

Tahap pertama ini dilakukan dengan kombinasi *chipertext* yang sudah ada bersama chiperkey kemudian disambungkan dengan XOR.



Gambar 2. *Process Addroundkey*

2) SubByte

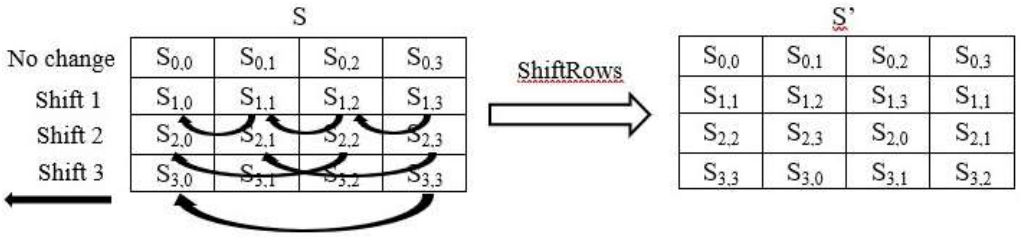
Tahap SubByte ialah akan dilakukan petakan menggunakan sebuah *table* S-Box pada setiap elemen state. Cara pensubstitusian pada array state untuk setiap byte, misalkan S[r,c] = xy, xy adalah digit heksadesimal dari nilai S'[r,c] dan nilai substitusi yang dinyatakan dengan S'[r,c] yaitu perpotongan baris x dengan kolom y pada elemen yang di dalam S-Box.

x \ y	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 3. *S-Box* Tabel

3) Shiftrows

Tahap Ketiga ini melakukan pergeseran pada tiap baris dari tabel. Baris pertama tidak dilakukan pergeseran, Pada baris kedua dilakukan pergeseran 1-byte, Lalu baris ketiga dilakukan pergeseran 2-byte dan terakhir pada baris keempat dilakukan pergeseran 3-byte.



Gambar 4. Ilustrasi *ShiftRows*

4) Mix Columns

Pada tiap proses elemen dari block chipper dengan matriks dilakukan perkalian. Pengalian dengan menggunakan dot *product* lalu hasil perkalian akan dimasukkan kedalam block chipper yang baru. Data yang akan didekripsi memerlukan waktu yang relatif. Beberapa data membutuhkan ada yang membutuhkan tambahan waktu dan ada pula yang membutuhkan waktu yang relatif lebih singkat dari proses enkripsi.

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} S_{0,c} \\ S_{1,c} \\ S_{2,c} \\ S_{3,c} \end{bmatrix}$$

Gambar 5. Matrik *Mix Column*

Hasil substitusi dikalikan dengan matrik *Mix Column*:

$$\begin{bmatrix} S'_{0,c} \\ S'_{1,c} \\ S'_{2,c} \\ S'_{3,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} S_{0,c} \\ S_{1,c} \\ S_{2,c} \\ S_{3,c} \end{bmatrix}$$

Gambar 6. Ilustrasi Perkalian Matrik

Perkalian hasil matriks diatas dianggap sama seperti perkalian yang dibawah ini:

$$S'_{0,c} = (\{02\} \bullet S_{0,c}) \oplus (\{03\} \bullet S_{1,c}) \oplus S_{2,c} \oplus S_{3,c} \quad (1)$$

$$S'_{1,c} = S_{0,c} \oplus (\{02\} \bullet S_{1,c}) \oplus (\{03\} \bullet S_{2,c}) \oplus S_{3,c} \quad (2)$$

$$S'_{2,c} = S_{0,c} \oplus S_{1,c} \oplus (\{02\} \bullet S_{2,c}) \oplus (\{03\} \bullet S_{3,c}) \quad (3)$$

$$S'_{3,c} = (\{03\} \bullet S_{0,c}) \oplus S_{1,c} \oplus S_{2,c} \oplus (\{02\} \bullet S_{3,c}) \quad (4)$$

Hasil dan Pembahasan

Dibawah ini adalah contoh kasus algoritma *Advanced Encryption Standard* (AES) berupa perhitungan manual:

Plaintext : JURNALKULIAHKSI1
ChiperKey : KRIPTOGRAFIAESKU

Addroundkey

Plaintext merupakan pesan yang dikirim dalam bentuk aslinya atau format yang mudah dibaca. Sedangkan CipherKey merupakan sebuah kunci yang digunakan dalam kombinasi algoritma untuk mentransformasi plaintext menjadi ciphertext. XOR adalah algoritma sederhana yang menggunakan prinsip operator logika XOR.

Plaintext

4A	41	4C	4B
55	4C	49	53
52	4B	41	49
4E	55	48	31

ChiperKey

4B	54	41	45
52	4F	46	53
49	47	49	4B
50	52	41	55

Mengkombinasikan plantext yang telah ada dengan ChiperKey kemudian disambungkan dengan XOR.

4A	41	4C	4B	XOR	4B	54	41	45	=	01	15	0D	0E
55	4C	49	53		52	4F	46	53		07	03	0F	00
52	4B	41	49		49	47	49	4B		1B	0C	08	02
4E	55	48	31		50	52	41	55		1E	07	09	65

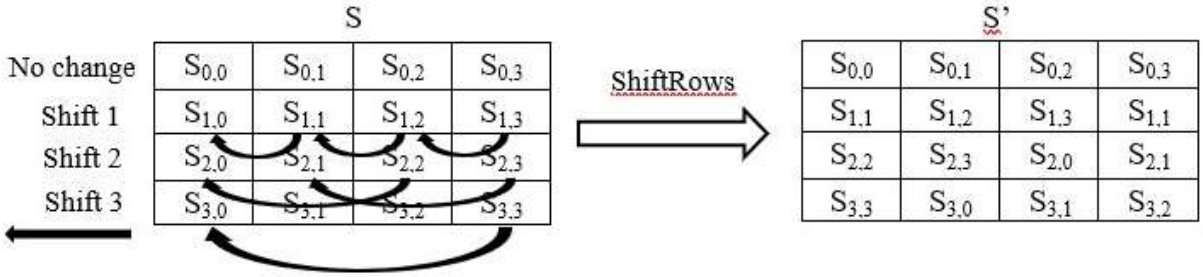
SubByte

Tahap SubByte ini akan dilakukan petakan menggunakan sebuah tabel S-Box pada setiap elemen state. Proses ini bertujuan untuk menyebarkan pengaruh transformasi non-linear pada baris-baris matriks state untuk putaran selanjutnya yang disebut diffusion.

01	15	0D	0E	= Subyte	7C	59	D7	AB
07	03	0F	00		C5	7B	76	63
1B	0C	08	02		AF	FE	30	77
1E	07	09	65		72	C5	01	4D

Shiftrows

Proses ini melakukan shift atau pergeseran disetiap elemen block yang dilakukan pada perbarisnya.



01	15	0D	0E		01	15	0D	0E
07	03	0F	00		03	0F	00	07
1B	0C	08	02	⟹	08	02	1B	0C
1E	07	09	65		65	1E	07	09

Mix Column

Pada proses ini setiap kolom dilakukan operasi perkalian dari matriks state. Proses ini bertujuan untuk menyebarkan pengaruh yang dihasilkan pada arah kolom matriks state pada setiap bit *plaintext* dan *cipherkey* terhadap ciphertext.

01	15	0D	0E
03	0F	00	07
08	02	1B	0C
65	1E	07	09

02	03	01	01		01		66
01	02	03	01	X	03	=	7A
01	01	02	03		08		BD
03	01	01	02		65		C2

02	03	01	01		15		1B
01	02	03	01	X	0F	=	13
01	01	02	03		02		44
03	01	01	02		1E		E

02	03	01	01		0D		2
01	02	03	01	X	00	=	5B
01	01	02	03		1B		2E
03	01	01	02		07		22

02	03	01	01		0E		C
01	02	03	01	X	07	=	2D
01	01	02	03		0C		A
03	01	01	02		09		33

66	1B	2	C
7A	13	5B	2D
BD	44	2E	A
C2	E	22	33

State MixColumns XOR RoundKey:

66	1B	2	C	XOR	01	15	0D	0E	=	67	E	F	2
7A	13	5B	2D		07	03	0F	00		7D	10	54	2D
BD	44	2E	A		1B	0C	08	02		A6	48	26	8
C2	E	22	33		1E	07	09	65		DC	9	2B	56

Round 1

67	E	F	2
7D	10	54	2D
A6	48	26	8
DC	9	2B	56

Demikian seterusnya langkah tersebut diulang hingga mendapatkan round ke 10. Hasil dari keseluruhan dapat dilihat pada tabel-tabel dibawah ini:

Round 1

67	E	F	2
7D	10	54	2D
A6	48	26	8
DC	9	2B	56

Round 2

91	D3	45	4C
ED	2A	38	BC
BC	41	C3	14
72	A6	ED	6E

Round 3

E3	84	7B	CA
68	43	51	22
4F	F9	13	BD
C6	12	47	58

Round 4

3B	DE	67	30
F6	19	FA	44
21	82	AF	F7
47	37	6C	6F

Round 5

32	9D	A9	6C
7C	27	54	B3
39	C2	BC	77
AA	98	B4	FA

Round 6

AC	72	11	A5
14	5D	22	64
D2	9A	B6	47
E6	43	D2	3F

Round 7

97	62	68	C3
49	7C	B3	F1
F3	B6	31	8D
9A	36	62	56

Round 8

30	B4	41	E5
5D	52	F1	98
BF	AE	11	1E
D4	E0	B8	27

Round 9

FA	54	A3	76
17	2C	39	26
6C	88	FE	4C
A0	B1	05	D3

Round 10

97	19	E1	F1
6C	39	F7	96
C2	A8	19	5E
25	BE	C2	DE

Ciphertext: (97 6C C2 25 19 39 A8 BE E1 F7 19 C2 F1 96 5E DE)
Ciphertext merupakan pesan (plaintext) yang telah diubah bentuknya menjadi kode yang lebih aman dan tidak ada yang dapat membukanya kecuali memliki key.

Contoh penerapan algoritma Advanced Encryption Standard (AES) dengan menggunakan Bahasa pemograman python:

Source code:

```
from Cryptodome.Cipher import AES
data = b'JURNALKULIAHKSI1'
key  = b'KRIPTOGRAFIAESKU'
 #ENKRIPSI chiper = AES.new(key, AES.MODE_EAX) nonce = chiper.nonce chiperText, tag = chiper.encrypt_and_digest(data)
 # Print print(chiperText, '\n', tag, '\n', nonce, '\n')
 #DESKRIPSI
 key = b'KRIPTOGRAFIAESKU'
chiper = AES.new (key, AES.MODE_EAX, nonce)
 plaintext = chiper.decrypt(chiperText)
try:
 chiper.verify(tag)
print(plaintext.decode())
 except ValueError:
print("Chiperkey yang dimasukkan salah")
```

Penjelasan tahapan enkripsi:

1. Pertama, import library Cryptodome.Cipher
2. Selanjutnya, definisikan data yang ingin dienkripsi sebagai plaintext. Dalam hal ini, data harus dalam bentuk byte (b"JURNALKULIAHKSI1"). Byte biasanya selalu diawali dengan 'b' atau 'B'.
3. Lakukan hal yang sama pada key (b"KRIPTOGRAFIAESKU"). Dalam hal ini, data dan key berukuran 16-byte (16 huruf) karena dalam penelitian ini menggunakan algoritma AES yang panjang kuncinya 128 bit.
4. AES.new () digunakan untuk membuat sandi. Dibutuhkan 2 argumen yaitu kunci dalam byte (yang didefinisikan dengan pernyataan sebelumnya) dan mode yang merupakan konstanta. Dalam hal ini, digunakan MODE_EAX. EAX berarti mengenkripsi lalu mengautentikasi lalu menerjemahkan dan merupakan mode opersi untuk cipher blok kriptografi.
5. encrypt_and_digest () melakukan enkripsi dan digest. Enkripsi menyembunyikan konten data, sementara hash adalah representasi numerik ukuran tetap yang bertindak sebagai pengidentifikasi untuk konten data. Metode encrypt_and_digest menerima data dan mengembalikan tupel dengan ciphertext dan kode otentikasi pesan (MAC). MAC dikenal juga sebagai tag, yang menegaskan keaslian dan otoritas data.
6. Cipher.nonce adalah nilai arbitrer yang hanya digunakan sekali untuk memastikan bahwa data tersebut asli. Jika cipher.nonce digunakan lebih dari sekali untuk potongan data yang berbeda, maka dapat diketahui bahwa keamanan telah disusupi.

Penjelasan tahapan deskripsi:

1. AES.new() digunakan untuk membuat sandi, seperti yang dilakukan sebelumnya. Tetapi kali ini, menyertakan nonce. Biasanya hanya orang yang mendekripsi pesan yang memiliki akses ke kunci.
2. Gunakan decrypt_and_verify untuk mendeskripsikan data dan memverifikasikan data.
3. Data yang dikembalikan masih dalam bentuk byte sehingga harus menggunakan *print* (*plaintext.decode*) untuk menunjukkan pesan di konsol.

Kesimpulan

Berdasarkan penelitian diatas, maka dapat disimpulkan kriptografi akan menjadi lebih aman dan sangat membantu agar terhindar dari pencurian dan modifikasi data pada system keamanan isi chat online karena isi data tersebut akan diamankan saat sampai ke tujuan penerima pesan melalui proses enkripsi yang menggunakan algoritma AES dan algoritma XoR. Proses dekripsi yang telah dienkripsi dengan kunci yang sesuai akan mengembalikan pesan semula tanpa mengalami perubahan sedikitpun. Hal ini sangat berguna bagi user saat melakukan chatting online tanpa khawatir pesan akan diketahui orang lain, disebar, dan dimodifikasi.

Daftar Pustaka

Adhar, D. (2019). ImpleAdhar, D. (2019). Implementasi Algoritma Des (Data Encryption Standard) Pada Enkripsi Dan Deskripsi Sms Berbasis Android. Jurnal Teknik Informatika Kaputama (JTIK), 3(2), 53–

60. https://jurnal.kaputama.ac.id/index.php/JTIK/article/view/185mentasi Alg. *Jurnal Teknik Informatika Kaputama (JTIK)*, *3*(2), 53–60.

Ajhari, A. A., & Windarto. (2018). *Implementasi Algoritma Affine Cipher Dan Aes-128 Untuk Pengamanan Pesan Dan One Time Password Registrasi Akun Pada Aplikasi Chatting Berbasis Android Di Sma Hang Tuah 1 Jakarta*. *1*(1), 323–334.

Andini, L., & Mariami, I. (2020). *E-Security Untuk Member Data Collection Administration System Menggunakan Metode Advanced Encryption Standard (AES)*. *3*(7), 1264–1275.

Donzilio Antonio Meko. (2018). Jurnal Teknologi Terpadu Perbandingan Algoritma DES , AES , IDEA Dan Blowfish dalam Enkripsi dan Dekripsi Data Donzilio Antonio Meko Program Studi Teknik Informatika , STIMIK Kupang Jurnal Teknologi Terpadu. *Jurnal Teknologi Terpadu*, *4*(1), 8–15.

Informasi, S., Komputer, I., Gunadarma, U., Margonda, J., No, R., Cina, P., & Barat, J. (2021). Analisis Perbandingan Performasi QoS VPN Encryption Protocol Pada Jaringan Berbasis Hybrid Cloud. *Jurnal Ilmiah Komputasi*, *20*(1), 69–82. https://doi.org/10.32409/jikstik.20.1.2695

It, W. (n.d.). *Kebocoran Data Punya Dampak Luas untuk Berbagai Lapisan Masyarakat*.

Liander, G. V., & Review, A. L. (2022). *Penggunaan Algoritma Elliptic Curve Diffie Hellman dan AES 256 pada Implementasi End-to-End Encryption WhatsApp*. *18219075*.

Muharram, F., Azis, H., & Manga, A. R. (2018). Analisis Algoritma pada Proses Enkripsi dan Dekripsi File Menggunakan Advanced Encryption Standard (AES). *Proc. of the Seminar Nasional Ilmu Komputer Dan Teknologi Informasi*, *3*(2), 112–115.

Mustika, L. (2020). Implementasi Algoritma AES Untuk Pengamanan Login Dan Data Customer Pada E-Commerce Berbasis Web. *JURIKOM (Jurnal Riset Komputer)*, *7*(1), 148. https://doi.org/10.30865/jurikom.v7i1.1943

Nizatsary, R. N., Seta, H. B., & Wahyono, B. T. (2022). Penerapan Keamanan Data Siswa Menggunakan International Data Encryption Algorithm (Idea) Dan Rivest Shamir Adleman (Rsa). *Informatik : Jurnal Ilmu Komputer*, *18*(2), 152. https://doi.org/10.52958/iftk.v18i2.4665

Pidana, T., & Bareskrim, S. (2022). *Ulah Bjorka dan Rentannya Kebocoran Data di Indonesia*. *September*.

Pratama, R. W., & Desyani, T. (2022). *Analisa dan Implementasi Kriptografi File Dokumen Dengan Metode Algoritma Advanced Encryption Standard (AES) Berbasis Web*. *1*(06), 758–762.

Putra, Y., Yuhandri, Y., & Sumijan, S. (2021). Meningkatkan Keamanan Web Menggunakan Algoritma Advanced Encryption Standard (AES) terhadap Seragan Cross Site Scripting. *Jurnal Sistim Informasi Dan Teknologi*, *3*, 56–63. https://doi.org/10.37034/jsisfotek.v3i2.44

Saragih, D. I., & Hasugian, P. M. (2021). Enkripsi Database Sekolah SMK Pembangunan Dengan Algoritma IDEA. *Jurnal Nasional Komputasi Dan Teknologi Informasi (JNKTI)*, *4*(1), 50–56. https://doi.org/10.32672/jnkti.v4i1.2704

Siringoringo, R. (2020). Analisis dan Implementasi Algoritma Rijndael (AES) dan Kriptografi RSA pada Pengamanan File 31 Oleh : Rinmar Siringoringo Analisis dan Implementasi Algoritma Rijndael (AES) dan Kriptografi RSA pada Pengamanan File ARTICLE INFORMATION A B S T R A K. *Jurnal Optimasi Sistem Industri*, *02*(01), 31–42.

Siswanto, A., Syukur, A., & Husna, I. (2018). Perbandingan Metode Data Encryption Standard (Des) Dan Advanced Encryption Standard (Aes) Pada Steganografi File Citra. *Seminar Nasional Teknologi Informasi Dan Komunikasi*, *October*, 229–236.

Sodikin, L., & Hidayat, T. (2020). Analisa Keamanan E-Commerce Menggunakan Metode Aes Algoritma. *Teknokom*, *3*(2), 8–13. https://doi.org/10.31943/teknokom.v3i2.46

Suhandinata, S., Rizal, R. A., Wijaya, D. O., Warren, P., & Srinjiwi, S. (2019). Analisis Performa Kriptografi Hybrid Algoritma Blowfish Dan Algoritma Rsa. *JURTEKSI (Jurnal Teknologi Dan Sistem Informasi)*, *6*(1), 1–10. https://doi.org/10.33330/jurteksi.v6i1.395

Sumandri. (2017). Studi Model Algoritma Kriptografi Klasik dan Modern. *Seminar Matematika Dan Pendidikan Matematika UNY*, 265–272.