ANTIVIRUS: Jurnal Ilmiah Teknik Informatika (p – ISSN: 1978 – 5232; e – ISSN: 2527 – 337X)
Vol. 17 No. 1 Mei 2023, pp. 53 – 67

PENGAMANAN DATA FILE DOKUMEN MENGGUNAKAN ALGORITMA ADVANCED ENCRYPTION STANDARD MODE CHIPER BLOCK CHAINING

Sahat Fernando Manullang[1)], Allwine[2)], dan Jakaria Sembiring[3)]
[1, 2, 3)] Teknik Informatika, STMIK Methodist Binjai
[1, 2, 3)]Jl. Gatot Subroto No.255, Bandar Senembah, Binjai Barat, Kota Binjai, Sumatera Utara 20716
e-mail: sahatfernandomanullang@gmail.com[1)], allwineamikmg@gmail.com[2)], jakariasembiring@gmail.com[3)]

Abstrak **:** *Perkembangan teknologi dibidang komputer menjadi suatu kebutuhan bagi pengguna komputer untuk menambah sistem keamanan pada suatu data file dokumen yang mewakili banyak informasi bersifat rahasia atau informasi penting agar tidak dapat dibaca dan diubah oleh pihak pihak tertentu yang dapat menyebabkan kerugian bagi pengguna dan pihak terkait. Dalam meningkatkan aspek kemanan data berupa file dokumen dalam menjaga kerahasiaan informasi dan keaslian data dapat dilakukan dengan sistem keamanan kriptografi. Salah satu kriptografi yang dapat digunakan adalah kriptografi algoritma Advanced Encryption Standard (AES) mode Chiper Block Chaining (CBC), Pembangunan sistem pengamanan kriptografi menggunakan metode Rapid Application Development (RAD) yang menekankan pada siklus pembangunan pendek, singkat, dan cepat dan menggunakan tools Java Development Kit (JDK) untuk implementasi sistem kriptografi kedalam program. Dari hasil sistem pengamanan kriptografi tersebut dapat digunakan dalam enkripsi data dokumen file sehingga isi atau informasi tidak dapat dibaca atau dipahami yang dapat menjaga kerahasiaan informasi dan keaslian pada data file dokumen tersebut.*

Kata Kunci—Kriptografi, Enkripsi, Dekripsi, Advanced Encryption Standard, Chiper Block Chaining

Abstract **:** *Technological developments in the field of computers have become a necessity for computer users to add a security system to document data files containing a lot of confidential or essential information so that certain parties can read and change it which can be detrimental to other parties. users and related parties. Increasing data security aspects in the form of document files in maintaining information confidentiality and data authenticity can be done with a cryptographic security system. One of the cryptography that can be used is the Advanced Encryption Standard cryptographic algorithm in Cipher Block Chaining mode. Developing a cryptographic security system uses the Rapid Application Development (RAD) method which emphasizes short, short, and fast development cycles and uses the Java Development Kit (JDK) tools to implement cryptographic systems into programs. The results of a cryptographic security system can be used to encrypt document file data so that the contents or information cannot be read or understood which can maintain the confidentiality of the information and the authenticity of the document file data..*

Keywords—Cryptography, Encryption, Decryption, Advanced Encryption Standard, Cipher Block Chaining

I. PENDAHULUAN

Seiring dengan perkembangan teknologi pada bidang komputer, memungkinkan banyaknya ilmu teknologi yang di salah gunakan oleh pihak - pihak tertentu sehingga dapat merugikan pengguna komputer secara individu maupun kelompok. Permasalahan yang sering terjadi adalah pencurian suatu data yang mewakili banyak informasi yang bersifat rahasia atau informasi penting yang sangat dijaga kerahasiaannya bagi suatu organisasi yang dapat menyebabkan kerugian banyak pihak.

Bagi pengguna komputer penyimpanan data berupa file text berisi informasi yang saat ini banyak digunakan adalah file dokumen. Jenis dokumen file yang umum digunakan adalah seperti PDF (*Portable Document Format*), format dokumen Microsoft (*docx, xls, pptx*) dan lain sebagainya. Namun format dokumen *file* saat ini tidak memiliki pengamanan yang cukup untuk menjaga kerahasiaan dari sebuah data tersebut, sehingga menjadi kebutuhan pengguna dalam menambah sistem keamanan pada suatu data file dokumen agar informasi yang tersimpan tidak dapat dibuka, dibaca, dikutip, dan diubah oleh pihak yang tidak berhak.

DOI: https://doi.org/10.35457/antivirus.v17i1.2811

Keamanan data merupakan aspek penting dalam melindungi dan menjamin keutuhan maupun kerahasiaan data yang berisi informasi penting. Dalam meningkatkan aspek keamanan dan menjaga kerahasiaan informasi dan kerahasiaan dari suatu data file dokumen dapat dilakukan sistem keamanan kriptografi, yaitu dengan menyandikan isi atau *content file* pada data tersebut menjadi isi yang sulit bahkan tidak dapat dipahami melalui dengan proses enkripsi dan untuk memperoleh kembali informasi yang asli dilakukan proses dekripsi.

Kriptografi juga merupakan studi terhadap teknik matematis yang bertujuan untuk pengamanan suatu sistem informasi seperti kerahasiaan, integritas data, autentikasi dan ketiadaan penyangkalan [1]. Terdapat banyak ilmu kriptografi yang digunakan dalam penyandian pesan atau informasi, salah satunya teknik kriptografi yang saat ini sering digunakan adalah subtitusi dan permutasi [7]. Salah satu kriptografi yang menggunakan kedua teknik tersebut adalah kriptografi algoritma *Advance Encryption Standard* (AES) [2]. Algoritma *Advance Encryption Standard* (AES) memiliki tingkat keamanan yang tinggi berdasarkan variasi panjang kunci yang dimiliki, serta memiliki kompleksitas waktu dan ruang yang baik [17].

Sistem pengamanan data file dokumen yang akan dilakukan adalah dengan proses kriptografi menggunakan algoritma *Advanced Encryption Standard* (AES) dan mengkombinasikan mode operasi *Chiper Blok Chainning* (CBC). Mode *Chiper Block Chaining* (CBC) merupakan salah satu algoritma simetris modern yang beroperasi dalam mode bit [10], sehingga dapat meningkatkan keamanan enkripsi pada sebuah data file dokumen untuk menjaga keaslian dan kerahasiaan informasi pada data tersebut.

II. TINJAUAN PUSTAKA

A. Penelitian Terdahulu Kriptografi Mode CBC (Cipher Block Chaining)

Tinjauan pustaka menjadi landasan dalam melakukan penelitan dan penulis mengambil beberapa contoh penelitian terdahulu yang dapat dijadikan sebagai pertimbangan dan acuan dalam mendukung penulisan, maka dalam tinjauan pustaka peneliti mencantumkan beberapa penelitian terdahulu, sebagai berikut:

1) Penelitian Ahmad Fathurrozi

Dikutip dari artikel yang ditulis oleh Ahmad Fathurrozi pada tahun 2021 dengan judul "Penerapan Algoritma Advanced Encryption Standard (AES - 256) Dengan Mode CBC Dan Secure Hash Algorithm (SHA-256) Untuk Pengamanan Data File". Pada Penelitian tersebut melakukan penerapan kombinasi algoritma AES-256 mode CBC dengan SHA-256 dalam pengamanan data mengenkripsi file dengan berbagai ekstensi seperti file dokumen, file suara (*voice note*/mp3), file video serta file gambar dengan baik dan dapat didekripsikan kembali dengan kunci yang sama pada aplikasi.

Dari penelitian tersebut dapat dilakukan perbandingan panjang kunci Algoritma AES pada penilitian. Ahmad Fathurrozi menggunakan kunci 256 bit dan panjang kunci penelitian yang penulis lakukan menggunakan 128 bit sehingga mempengaruhi proses putaran enkripsi dan deskripsi. Panjang kunci 256 bit memberikan tingkat enkripsi yang lebih kuat dibandingkan panjang kunci 128 bit. kelebihan dari penelitian Ahmad Fathurrozi tersebut menambah fungsi *hash* pada algoritmanya, sehingga menambah tingkat keamanan dalam proses enkripsi. Dan kelemahan pada penelitian tersebut adalah dalam proses mengenkripsi file berukuran besar menggunakan kunci 256 bit akan membutuhkan waktu yang lama saat proses komputasinya.

2) Penelitian Ridha Ismadiah, Muhammad Syahrizal, Putri Ramadhani

Dikutip dari artikel yang ditulis oleh Ridha Ismadiah, Muhammad Syahrizal, Putri Ramadhani pada tahun 2020 dengan judul "Kombinasi Algoritma *Cipher Block Chaining* (CBC) dan Mars Pada Penyandian File PDF". Pada penilitian ini membahas pengamanan file pdf berdasarkan kombinasi algoritma CBC dan Mars meliputi tahap, yaitu proses perluasan kunci (key expansion), proses enkripsi dan dekripsi. Hasil dari proses XOR tersebut yang kemudian di enkripsi. Dari Algoritma Mars ini proses yang akan di jalan kan terdiri dari ekspansi atau pembangkit kunci, pembangkit kunci disini menggunkan modifikasi dari Algoritma DES (*Data Encryption Standard*).

Perbandingan dari penelitian tersebut dengan penelitian yang penulis lakukan adalah proses perluasan kunci yang menggunakan algoritma DES, sehingga pembentukan kunci dan putaran yang jauh berbeda

DOI: https://doi.org/10.35457/antivirus.v17i1.2811

dengan algoritma AES. Kelebihan dari penelitian tersebut adalah kombinasi dari algoritma *Cipher Block Chaining* (CBC) dan Mars membuat proses penyandian enkripsi lebih rumit, dan proses pembangkitan kunci internal dilakukan 16 putaran menggunakan modifikasi algoritma DES. Kekurangan dari penelitian tersebut adalah jika kesalahan satu bit pada proses pembentukan kunci akan mempengaruhi ekspansi kunci dan merambat pada plaintext yang di enkrispsi.

B. Kriptografi

Secara umum kriptografi dapat diartikan sebagai ilmu dan seni penyandian yang bertujuan untuk menjaga keamanan dan kerahasiaan suatu data atau informasi [5]. Dengan cara penyandian data atau informasi tersebut menjadi tidak dapat dimengerti lagi maknanya agar hanya orang yang di tuju dapat melihat isi dari data tersebut.

C. Advanced Encyrption Standard (CBC)

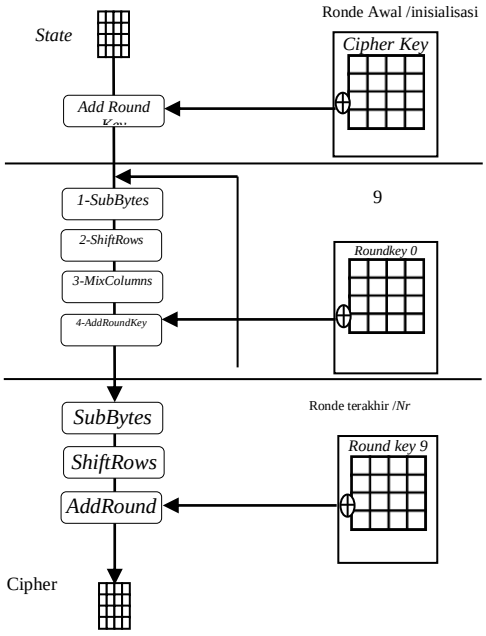
Advanced Encryption Standard (AES) merupakan salah satu algoritma kriptografi, Algortima AES menggunakan substitusi dan permutasi, dan sejumlah putaran (*cipher* berulang) setiap putaran mengunakan kunci internal yang berbeda (kunci setiap putaran disebut *round key*). Panjang kunci Algoritma AES terdiri dari tiga pilihan kunci yaitu 128 *bit*, 192 *bit*, 256 *bit* dengan catatan 1 *word* = 32 *bit*. Pada proses enkripsi algoritma AES terdiri dari beberapa tahapan transformasi *bytes*, yaitu *Key Schedule*, *AddRoundKey, SubBytes*, *ShiftRows*, *Mixcolumns*, dan *AddRoundKey* [5]. Garis besar Algoritma AES yang beroperasi pada blok 128-bit adalah sebagai berikut :

1) *Key Schedule: merupakan proses untuk membentuk kunci yang akan digunakan dalam proses enkripsi dan dekripsi. Pembentukan kunci terdiri dari beberapa tahapan yaitu RotWord, SubWord, XOR dengan nilai R-con, dan XOR dengan word sebelumnya.*
   a. *RotWord* adalah menggeser setiap *byte* pada kolom terakhir dari *cipherkey* secara siklik ke atas satu kali.
   b. *SubWord* adalah tahap pensubtitusian hasil dari *RotWord* dengan tabel subtitusi S-Box.
   c. *R-con* merupakan *Round Constant* yang nilai *R-Con* sudah ditetapkan dan akan dilakukan proses XOR dengan hasil *SubWord.*
2) *AddRoundKey: melakukan XOR antara state awal (plaintext) dengan cipher key. Tahap ini disebut juga initial round.*
3) *Putaran sebanyak Nr – 1 kali. Proses yang dilakukan pada setiap putaran adalah:*
   a. *SubBytes* adalah operasi substitusi *byte nonlinier* yang menukar *byte state* secara *independen* menggunakan tabel S-Box. S-Box dihasilkan dari perkalian *invers polynomial* GF(2^8) [6].
   b. *ShiftRows* adalah pergeseran baris-baris array state secara *wrapping* dimana *bit* yang paling kiri dipindahkan menjadi *bit* yang paling kanan. Jumlah pergeseran tergantung pada nilai bari (r). Baris r = 1 digeser sejauh 1 *byte*, baris r = 2 digeser sejauh 2 *byte*, dan baris r = 3 digeser sejauh 3 *byte*. Baris r = 0 tidak digeser.
   c. *MixColumns* adalah mengoperasikan setiap elemen yang berada dalam satu kolom pada *state*. Perkalian elemen dengan *polynomial* a (x) *mod* (x4 + 1) pada kolom ditetapkan pada persamaan [5] :

$$a(x) = \{03\}x3 + \{01\}\ x2 + \{01\}\ x + \{02\} \qquad (1)$$

4) *Final Round*: proses untuk putaran terakhir tidak mengalami transformasi *MixColumns*:
   a. *SubBytes*
   b. *ShiftRows*
   c. *AddRoundKey*

DOI: https://doi.org/10.35457/antivirus.v17i1.2811



Gambar 1. Diagram Proses Enkripsi AES

D. Chiper Blok Chaining (CBC)

Chiper Block Chaining (CBC) adalah mode operasi yang menerapkan umpan-balik (*feedback*) pada sebuah blok, tiap blok dari *plaintext* dilakukan XOR dengan hasil *ciphertext* dari blok sebelumnya yang kemudian dilakukan enkripsi, Pada mode CBC ciphertext dari masing-masing blok akan tergantung pada seluruh hasil ciphertext dari blok-blok sebelumnya. Pada proses awal enkripsi blok data akan di XOR dengan IV (*Initialization Vector*) untuk membuat tiap *plaintext* menjadi unik [7]. Rumus matematis untuk enkripsi pada mode CBC adalah:

$$C_i = E_k\,(P_i \oplus C_{i-1}),\ C_0 = IV \qquad (2)$$

Enkripsi blok pertama, $C_0 = IV$ (*Initialization Vector*). *IV* dapat diberikan kepada pengguna atau dibangkitkan secara acak oleh program jadi, untuk menghasilkan blok *ciphertext* pertama (C1), IV digunakan untuk menggantikan blok *ciphertext* sebelumnya. Sebaliknya pada dekripsi blok *plaintext* pertama diperoleh dengan cara meng-XOR-kan IV dengan hasil dekripsi terhadap blok *ciphertext* pertama [8]. sedangkan rumus matematis untuk dekripsi pada mode CBC adalah adalah kebalikan dari enkripsi:

$$P_i = D_k\,(C_i) \oplus C_{i-1,\,0} = IV \qquad (3)$$

Keterangan :
C_i = *Chipertext*
E_k = Enkripsi
P_i = *Plaintext*
C_0 = Blok Plaintext Pertama
IV = *initialization vector*
D_k = Dekripsi

E. Unified Modeling Language (UML)

Unifed Modeling Language (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefenisikan requitment, membuat analisis, dan desain serta menggambarkan arsitektur dalam pemograman berorientasi objek. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataan UML paling banyak digunakan pada metodologi berorientasi objek [9]

DOI: https://doi.org/10.35457/antivirus.v17i1.2811

1) Use Case Diagram

Use Case Diagram merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat *use case* untuk mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat.

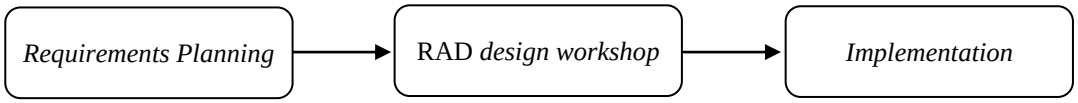
2) Activity Diagram

Activity Diagram adalah tipe khusus dari diagram state yang memperlihatkan aliran dari suatu aktifitas ke aktifitas lainnya dari suatu sistem. *Activity Diagram* penting dalam pemodelan fungsi-fungsi dalam suatu sistem dan member tekanan pada aliran kendali antar objek.

III. METODE PENELITIAN

A. Rapid Application Development (RAD)

Rapid Application Development adalah model proses pembangunan perangkat lunak yang menekankan pada siklus pembangunan pendek, singkat, dan cepat. Dengan metode *Rapid Application Development* ini, akan mempermudah peneliti dalam merancanga aplikasi yang tertata secara terstruktur. Terdapat tiga fase dalam RAD pada tahap perancangan, dan penerapan. Adapun ketiga fase tersebut adalah *requirements planning* (rencana kebutuhan), RAD *design workshop* (proses desain), dan *implementation* (implementasi).



Gambar 2. Model RAD

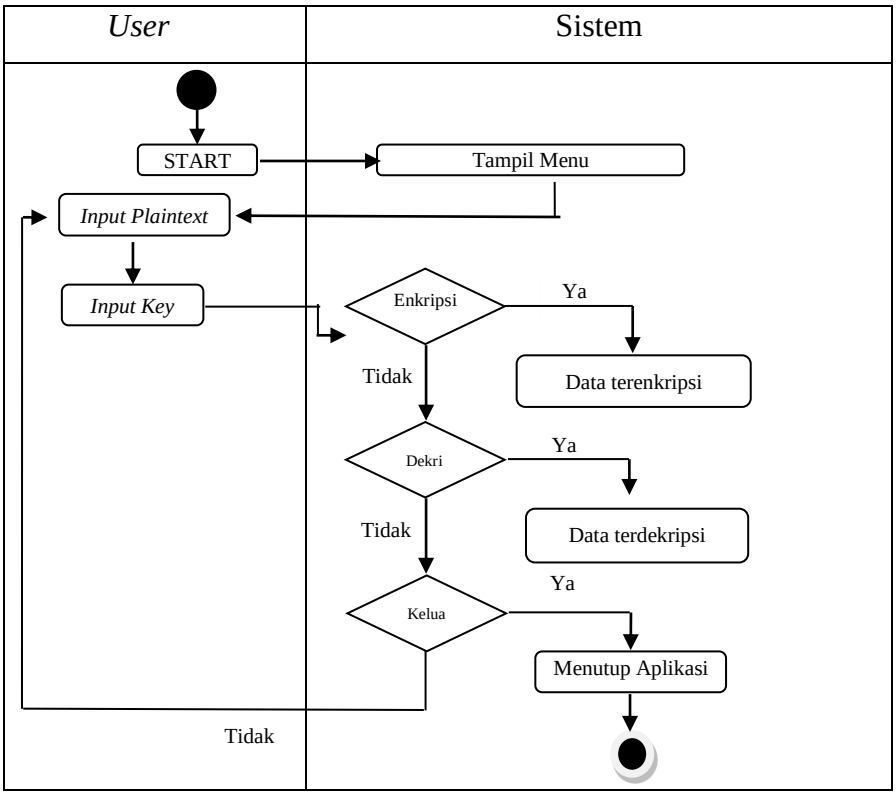
B. Perancangan Sistem

Perancangan terhadap sistem diperlukan untuk mengetahui bagaimana sistem akan dibangun dengan menggunakan *Unified Modelling Language* sebagai alat bantu dalam pemodelan sistem. *Unified Modelling Language* yang digunakan adalah sebagai berikut:

1) Activity Diagram

Activity diagram berupa *flowchart* yang digunakan untuk memperlihatkan aliran kerja dari aplikasi. pada Gambar 4. *Activity Diagram* memperlihatkan aliran dari suatu interaksi *user* terhadap aktifitas sistem kriptografi melalui pemodelan rangkaian *state* awal (*start*) hingga rangkaian *state* terakhir (*menutup aplikasi*) dengan aliran proses keputusan "*ya"* atau "*tidak*" untuk melakukan "*enkripsi"* dan "*dekripsi"* sehingga menghasilkan "*data terenkripsi*" dan "*data terdeskripsi"* .

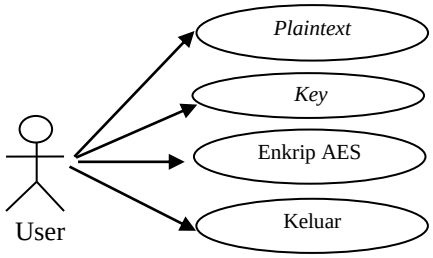
DOI: https://doi.org/10.35457/antivirus.v17i1.2811



Gambar 3. Activity Diagram

2) Use Case Diagram

Use case diagram mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Interaksi aktor dengan sistem memperlihatkan user dapat memasukkan *plaintext*, *key*, *dan* melakukan proses *enkripsi*.

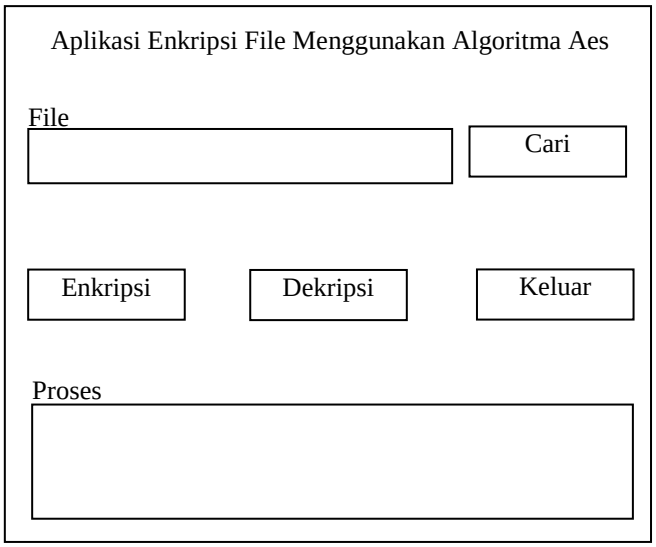


Gambar 4. Use Case Diagram

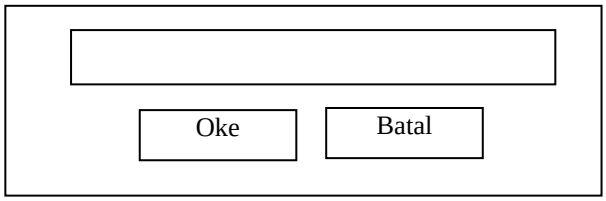
3) Tampilan Sistem

Tampilan antarmuka sistem aplikasi penyandian *file* dokumen menggunakan algoritma AES memiliki form utama yang memiliki tombol-tombol untuk proses enkripsi dan dekripsi serta *textbox* untuk meng-*input*-kan file dan *textbox* proses, form kedua memiliki tombol-tombol dan *textbox* untuk menginputkan kunci.

DOI: https://doi.org/10.35457/antivirus.v17i1.2811



Gambar 5. Tampilan Form Utama



Gambar 6. Tampilan Input Key

C. Implementasi Algoritma

Untuk mengetahui bagaimana langkah-langkah algoritma *Advanced Encryption Standard* dalam mengenkripsi dan mendekripsi file dokumen pada sistem, maka dibutuhkan algoritma untuk memecahkan masalah tersebut, yaitu:

1) Algoritma Enkripsi Advanced Encryption Standard

Header : Algoritma enkripsi *advanced encryption standard*
Deklarasi : P, C : *String*, IV, K : *Byte*
Deskripsi :
Input :
IV ← *Initialization Vector*
P ← *Plaintext*
K ← *Key*
Output :
C ← *Chipertext*
Proses :
Mulai
Memasukkan Ekspansi(K)
XOR (IV)
AddRoundkey
Putaran Transformasi
Subbytes;
Shiftrows;
Mixcolumns(P);
Addroundkey;
Selesai.

2) Algoritma Dekripsi Advanced Encryption Standard

Header : Algoritma dekripsi Advanced Encryption Standard
Deklarasi : C, P : String, K : Byte

DOI: https://doi.org/10.35457/antivirus.v17i1.2811

Deskripsi :

Input :
C ← *Chipertext*
K ← *Key*
IV ← *Initialization Vector*
Output :
P ← *Plaintext*
Proses :

Mulai

Memasukkan Ekspansi(K)
AddRoundkey
InvSubBytes
InvShiftRows
AddRoundKey
InvMixColumns
XOR (IV)

Selesai.

IV. HASIL DAN PEMBAHASAN

A. Pembahasan

Proses enkrispi dari langkah-langkah algoritma *Advanced Encryption Standard* tersebut dilakukan sampai 10 kali putaran, pada proses terkahir yaitu tanpa *MixColumns* [2]. Proses terakhir yang dihasilkan adalah sebagai *chipertext*. Sebagai contoh dari sebuah *plaintext* yang dilakukan proses enkripsi menggunakan algoritma *Advanced Encryption Standard* metode *Chiper Block Chaining* dapat dilihat pada table 1.

Tabel 1. Plaintext dan Chipertext AES

Input	***Byte***
Plaintext	25 50 44 46 2D 31 2E 37 0D 0A 25 B5 B5 B5 B5 0D
IV	69 76 73 74 6D 69 6B 6D 65 74 68 6F 64 69 73 74
Key	70 72 6F 64 69 69 6E 66 6F 72 6F 61 74 69 6B 61
Output	***Byte***
Chipertext	1B 81 88 79 C1 FE 94 E7 A3 8B 7A E8 90 07 04 13

Key Schedule terdiri dari beberapa tahapan yaitu *RotWord*, *SubWord*, *XOR* dengan nilai *R-con*, dan XOR dengan *word* sebelumnya.

Tabel 2. Kunci (ChiperKey) dalam Heksadesimal

Kunci	**Heksadesimal**
prodiinformatika	70 72 6F 64 69 69 6e 66 6F 72 6F 61 74 69 6B 61

Tabel 3. Kunci dibagi 4 byte

W_{i-4}	W_{i-3}	W_{i-2}	W_{i-1}
70	69	6F	74
72	69	72	69
6F	6E	6F	6B
64	66	61	61

Tahapan selanjutnya adalah *RotWord*, yang dilakukan pada tahap *RotWord* adalah menggeser setiap *byte* pada kolom terakhir dari *cipherkey* secara siklik ke atas satu kali.

DOI: https://doi.org/10.35457/antivirus.v17i1.2811

Tabel 4. RotWord Wi_{-1}

W_{i-1}	W_{i-1}
74	69
69	6B
6B	61
61	74

Hasil dari *RotWord* Wi-1 kemudian dilakukan pensubtitusian dengan tabel S-Box yang sudah ditetapkan

Tabel 5. Subtitusi Wi_{-1}

W_{i-1}	Hasil SubtitusiW_{i-1}
74	F6
69	7F
6B	EF
61	92

Tahap terakhir untuk mendapatkan kunci kolom ke (W1) yaitu proses XOR yang dilakukan terhadap hasil *subword* dengan nilai R-con yang bersesuaian, lalu XOR lagi dengan kolom (Wi-4), berikut adalah tabel *R-con* (*Round Constant*).

Tabel 6. *R-con* (*Round Constant*).

1	2	3	4	5	6	7	8	9	10
01	02	04	08	10	20	40	80	1B	36
00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00

Sebelum mendapatkan nilai matrik W1 dengan melakukan Wi-1 ⊕ Wi-4 ⊕ Rcon, bilangan heksadesimal dirubah dulu menjadi bilangan biner untuk dapat dilakukan operasi XOR. Prosesnya adalah sebagai berikut:

$$\begin{bmatrix}70\\72\\6F\\64\end{bmatrix} \oplus \begin{bmatrix}F6\\7F\\EF\\92\end{bmatrix} \oplus \begin{bmatrix}01\\00\\00\\00\end{bmatrix}$$

Berikut bilangan heksadesimal yang diatas telah dirubah ke dalam bilang biner:

$$\begin{bmatrix}01110000\\01110010\\01101111\\01100100\end{bmatrix} \oplus \begin{bmatrix}11110110\\01111111\\11101111\\10010010\end{bmatrix} \oplus \begin{bmatrix}00000001\\00000000\\00000000\\00000000\end{bmatrix} = \begin{bmatrix}10000111\\00001101\\10000000\\11110110\end{bmatrix}$$

Selanjutnya untuk mendapatkan sub kunci pertama pada kolom kedua hingga keempat dilakukan operasi XOR antara Wi dengan kolom Wi-1-3, tanpa proses XOR R-con, Sehingga pada *round* pertama didapatkan

Key Schedul Round Pertama yaitu = $\begin{bmatrix}87 & EE & 81 & F5\\0D & 64 & 16 & 7F\\80 & EE & 81 & EA\\F6 & 90 & F1 & 90\end{bmatrix}$

Proses pengolahan nilai matrik W1 dengan melakukan Wi-4 ⊕ Wi-1 ⊕ *Rcon* dilakukan berulang sebanyak 10 iterasi sehingga menghasilkan 10 *Round Key Schedule*.

DOI: https://doi.org/10.35457/antivirus.v17i1.2811

Operasi XOR IV adalah proses pertama dalam metode CBC dilakukan operasi XOR antara IV dengan *plaintext* blok pertama yang kemudian hasil operasi XOR tersebut dilakukan proses enkripsi AES, untuk proses operasi nya sebagai berikut:

$$\begin{bmatrix} 25 & 2D & 0D & B5 \\ 50 & 31 & 0A & B5 \\ 44 & 2E & 25 & B5 \\ 46 & 37 & B5 & 0D \end{bmatrix} \oplus \begin{bmatrix} 69 & 6D & 65 & 64 \\ 76 & 69 & 74 & 69 \\ 73 & 6B & 68 & 73 \\ 74 & 6D & 6F & 74 \end{bmatrix} = \begin{bmatrix} 4C & 40 & 68 & D1 \\ 26 & 58 & 7E & DC \\ 37 & 45 & 4D & C6 \\ 32 & 5A & DA & 79 \end{bmatrix}$$

AddRoundKey melakukan XOR antara Hasil XOR Blok Pertama yang dinotasikan sebagai *state* awal (P_{i-1}) dengan *key* sehingga menjadi sebuah *state* 4x4.

$$\begin{bmatrix} 4C & 40 & 68 & D1 \\ 26 & 58 & 7E & DC \\ 37 & 45 & 4D & C6 \\ 32 & 5A & DA & 79 \end{bmatrix} \oplus \begin{bmatrix} 70 & 69 & 6F & 74 \\ 72 & 69 & 72 & 11 \\ 6F & 6E & 6F & 6B \\ 64 & 66 & 61 & 61 \end{bmatrix} = \begin{bmatrix} 3C & 29 & 07 & A5 \\ 54 & 00 & 0C & CD \\ 6F & 2B & 22 & AD \\ 56 & 3C & BB & 18 \end{bmatrix}$$

Proses selanjutnya melakukan subtitusi Addroundkey sebelumnya dengan menggunakan *Subtitution Box* (S-Box) AES

$$\begin{bmatrix} 3C & 29 & 07 & A5 \\ 54 & 00 & 0C & CD \\ 6F & 2B & 22 & AD \\ 56 & 3C & BB & 18 \end{bmatrix} \mathrm{S-Box} \begin{bmatrix} EB & A5 & C5 & 06 \\ 20 & 63 & FE & BD \\ A8 & F1 & 93 & 95 \\ B1 & EB & EA & AD \end{bmatrix}$$

Transformasi *ShiftRows* melakukan rotasi setiap baris. Baris ke 1 dirotasi 0 kali, baris ke 2 dirotasi 1 kali, baris ke 3 dirotasi 2 kali, dan baris ke 4 dirotasi 3 kali.

Tabel 7. Hasil Rotasi ShiftRows

SubBytes				Rotasi
EB	A5	C5	06	0 kali
63	FE	BD	20	1 kali
93	95	A8	F1	2 kali
AD	B1	EB	EA	3 kali

Kemudian p roses *MixColumns* melakukan perkalian setiap kolom dari *array state polynomial* a(x) mod (x4 + 1)

$$\begin{bmatrix} EB & A5 & C5 & 06 \\ 63 & FE & BD & 20 \\ 93 & 95 & A8 & F1 \\ AD & B1 & EB & EA \end{bmatrix} \oplus \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} = \begin{bmatrix} 56 \\ 2E \\ 59 \\ 97 \end{bmatrix}$$

Berikut hasil keseluruhan dalam tahapan *MixColoumns* pada *Round* ke 1:

Tabel 8. Transformasi MixColoumns Round 1

56	7B	0E	77
2E	4D	AC	A4
59	A2	15	FA
97	E6	65	0C

Proses *AddRoundKey, SubBytes, ShiftRows, dan MixColoums* yang dilakukan untuk Round ke 1 hingga *round* ke 9 menggunakan cara yang sama dengan proses *round* sebelumnya. Pada proses *round* 10 yang terakhir tanpa transformasi *MixColoumns*, tahapan yang dilakukan adalah adalah *SubBytes, ShiftRows* dan *AddRoundKey* yang dihasilkan adalah sebagai *chipertext* 16 byte yang pertama, sehingga

DOI: https://doi.org/10.35457/antivirus.v17i1.2811

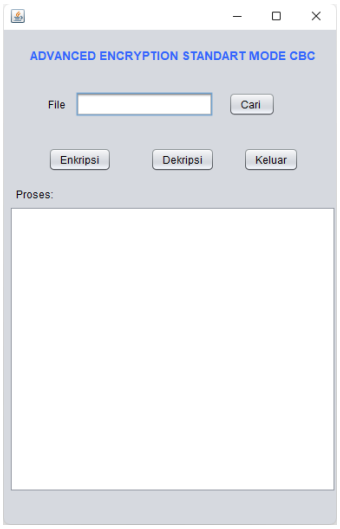
didapatkan hasil *AddRoundKey* terakhir sebagai *chipertext* adalah (*1B 81 88 79 C1 FE 94 E7 A3 8B 7A E8 90 07 04 13*).

Tabel 9. Hasil Tranformasi Round 10

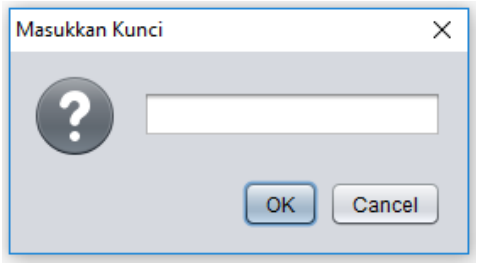
Key Schedule				***SubBytes***				***ShiftRows***				***AddRoundKey***			
07	B3	1E	D4	1C	72	BD	44	1C	72	BD	44	1B	C1	A3	90
9B	10	DA	8F	88	1A	EE	51	1A	EE	51	88	81	FE	8B	07
64	C6	76	C9	0C	CD	EC	52	EC	52	0C	CD	88	94	7A	04
C0	C1	71	7C	26	99	6F	B9	B9	26	99	6F	79	E7	E8	13

B. *Tampilan Program*

Program aplikasi yang dibuat menggunakan bahasa pemgroman java menggunakan perangkat lunak *open-source* Java Netbeans IDE 8.2 berupa *Windows Application*. Sistem yang telah dirancang mencakup dengan proses enkripsi dan dekripsi dengan menggunakan algoritma AES mode CBC. Tampilan program aplikasi pengamanan file yang dibangun terbagi atas dua form yaitu form menu utama, dan *form input* kunci. Form utama menampilkan kolom *input file*, tombol enkripsi, dekripsi dan hasil prosesnya.



Gambar 7. Tampilan Input File

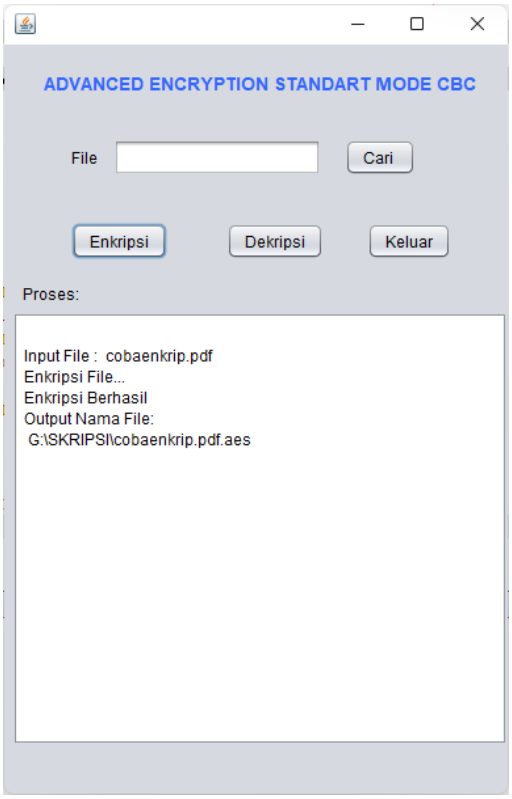


Gambar 8. Tampilan Input Key

C. *Pengujian*

Dari program yang telah dibuat dilakukan pengujian enkripsi terhadap objek file dokumen "cobaenkrip.pdf" dilakukan proses enkripsi dengan pembentukan IV (*Initialization Vector*) secara acak pada program dan memberikan kunci sepanjang 16 byte "prodiinformatika".

DOI: https://doi.org/10.35457/antivirus.v17i1.2811

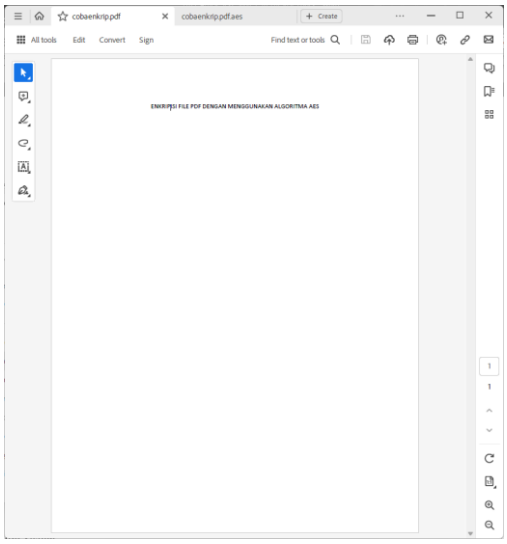


Gambar 9. Tampilan Proses Enkripsi

Tabel 10. Data Pengujian Enkripsi

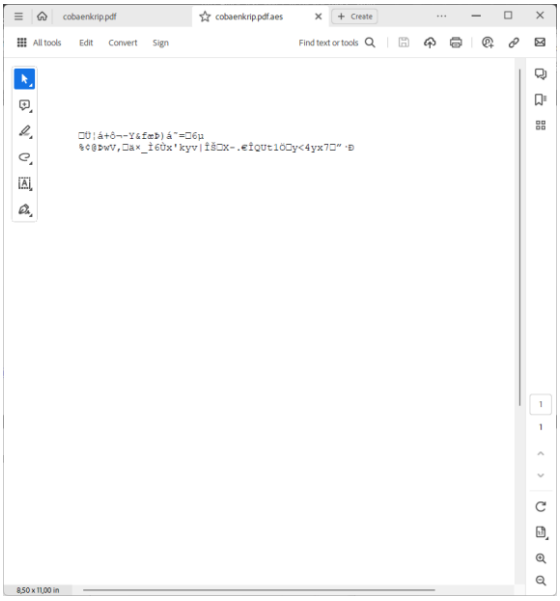
Input	***Data***
Plaintext	cobaenkrip.pdf
IV (*Initialization Vector*)	*Random*
Key	*prodiinformatika*
Output	***Byte***
Chipertext	cobaenkrip.pdf.aes

Dari hasil pengujian untuk proses enkripsi berhasil dilakukan, dan lokasi output file yang telah di enkripsi tersimpan di folder yang sama dengan file asli. Untuk file enkripsi diberikan penambahan ekstensi ".aes" dari hasi file pengujian.

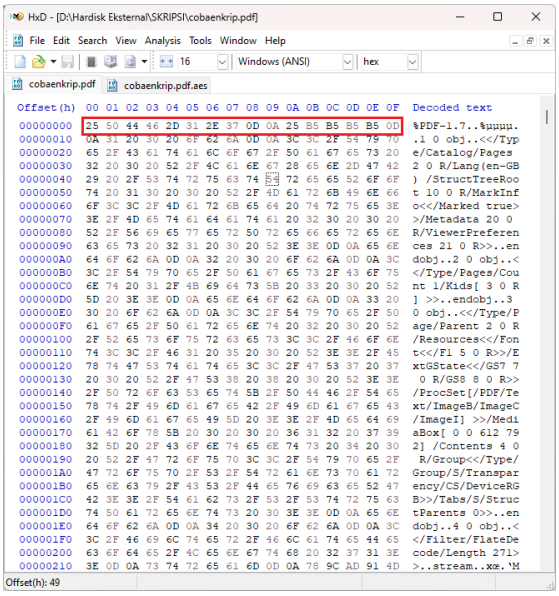


Gambar 10. Isi Dokumen File Asli

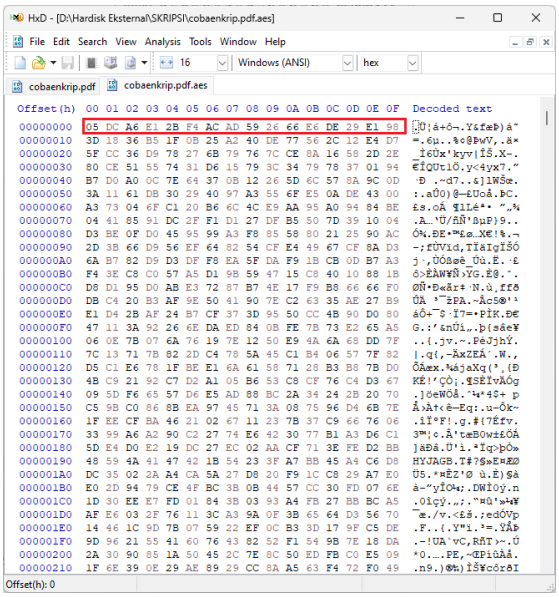
DOI: https://doi.org/10.35457/antivirus.v17i1.2811



Gambar 11. Isi Dokumen Hasil Enkripsi



Gambar 12. Data Byte File Dokumen Asli



Gambar 13. Data Byte Dari Hasil Enkripsi

DOI: https://doi.org/10.35457/antivirus.v17i1.2811

Tabel 11. Data 16 Byte Input dan Output Pengujian

Input	***Plaintext***
Byte	25 50 44 46 2D 31 2E 37 0D 0A 25 B5 B5 B5 B5 0D
Text	ENKRIPISI FILE PDF DENGAN MENGGUNAKAN ALGORITMA AES
Output	***Chipertext***
Byte	05 DC A6 E1 2B F4 AC AD 59 26 66 E6 DE 29 E1 98
Text	·Ü¦á+ô¬Y&fæÞ)á˜= 6µ %¢@ÞwV, ä×_Ì6Ùx'kyv|ÎŠ X-.€ÎQUt1Ö y<4yx7 ”·Ð

V. KESIMPULAN

Dari hasil pengujian enkripsi menggunakan algoritma Advanced Encryption Standard mode Chiper Block Chaining terhadap sebuah dokumen file pdf yang berisi plaintext atau pesan asli dapat disimpulkan bahwa proses penyandian menggunakan kunci 128 bit atau 16 karakter menghasilkan chipertext atau hasil dari enkripsi yang isi nya menampilkan karakter atau simbol-simbol yang unik sehingga pesan dan informasi sulit untuk dibaca atau dipahami, dan untuk mengembalikan chipertext ke semula dilakukan proses dekripsi menggunakan kunci yang sama pada saat proses enkripsi.

Lama waktu proses enkripsi berpengaruh dari ukuran data file dokumen yang di proses, semakin besar data dokumen file yang di enkrispi atau dekripsi maka akan menambah waktu proses komputasinya. Output yang dihasilkan dari proses enkripsi membuat file baru dari hasil penyandian file dokumen pdf tersebut di dalam satu folder yang sama, sehingga tidak mempengaruhi atau merubah file dokumen yang asli.

DAFTAR PUSTAKA

[1] R. Ismadiah, M. Syahrizal, and P. Ramadhani, “Kombinasi Algoritma Cipher Block Chaining (CBC) dan Mars Pada Penyandian File PDF,” *J. Comput. Syst. Informatics*, vol. 1, no. 4, pp. 337–345, 2020.

[2] G. Bhaudhayana and I. Widiartha, “Implementasi Algoritma Kriptografi Aes 256 Dan Metode Steganografi Lsb Pada Gambar Bitmap,” *J. Ilmu Komput.*, vol. 8, no. 2, pp. 15–25, 2015.

[3] A. Fathurrozi, “Penerapan Algoritma Advanced Encryption Standard (AES- 256) Dengan Mode CBC DanSecure Hash Algorithm (SHA-256) Untuk Pengamanan Data File,” vol. 2, no. 2, pp. 227–238, 2021.

[4] S. M. Wadi and N. Zainal, “High Definition Image Encryption Algorithm Based on AES Modification,” *Wirel. Pers. Commun.*, vol. 79, no. 2, pp. 811–829, 2014, doi: 10.1007/s11277-014-1888-7.

[5] A. Sudrajat, Y. H. Prasetyo, and M. Kusumawardani, “Implementasi Enkripsi Advanced Encryption Standard (AES-128) Mode Cipher Block Chaining (CBC) sebagai Keamanan Komunikasi Pergerakan Robot Humanoid KRSBI,” *J. Jartel J. Jar. Telekomun.*, vol. 11, no. 1, pp. 6–11, 2021, doi: 10.33795/jartel.v11i1.16.

[6] Rosa A.S dan M.Shalaludin, “Modul Pembelajaran Rekayasa Perangkat Lunak (Terstuktur dan Beorientasi Objek),” *Modul. Bandung*, 2011.

[7] R. Munir, “Kriptografi,” *Bandung Inform. Bandung.*, 2006.

[8] S. A. B. R. Kristoforus JB, “Implementasi Algoritma Rijndael untuk Enkripsi dan Dekripsi pada Citra Digital,” *Semin. Nas. Apl. Teknol. Inf. 2012*, vol. 2012, no. Snati, pp. 15–16, 2012.

[9] J. Aliyah, “Aplikasi Mobile Untuk Enkripsi Data Gambar Menggunakan Kombinasi Fungsi Xor Dan Mode Operasi Cbc,” *J. Inform. Teknol. dan Sains*, vol. 2, no. 4, pp. 214–222, 2020, doi: 10.51401/jinteks.v2i4.824.

[10] Henry, A. H. Kridalaksana, and Z. Arifin, “Kriptografi Aes Mode Cbc Pada Citra Digital Berbasis Android,” *Semin. Ilmu Komput. dan Teknol. Inf.*, vol. 1, no. 1, pp. 45–52, 2016.

[11] Dony Ariyus, “Pengantar Ilmu Kriptografi Teori Analisis dan Implementasi,” *Yogyakarta: Penerbit Andi.*, 2008.

[12] M. A. Maricar and N. P. Sastra, “Efektivitas Pesan Teks Dengan Cipher Substitusi, Vigenere

DOI: https://doi.org/10.35457/antivirus.v17i1.2811

Cipher, dan Cipher Transposisi," *Maj. Ilm. Teknol. Elektro*, vol. 17, no. 1, p. 59, 2018, doi: 10.24843/mite.2018.v17i01.p08.

[13] Jubilee Enterprise, "Belajar Java, Database, dan Netbeans dari Nol," *PT Elex Media Komputindo, Jakarta*, 2016.

[14] Y. Supardi, "Pemrograman Bahasa Java Bagi Pemula," *Bandung Inform. Bandung.*, 2010.

[15] Zuraidah, E., & Akbar, S. (2019). *Perancangan aplikasi absensi siswa berbasis Java netbeans*. Prosisko,6(1), 53–59.

[16] Maricar, M. A., & Sastra, N. P. (2018). *Efektivitas Pesan Teks Dengan Cipher Substitusi, Vigenere Cipher, dan Cipher Transposisi*. Majalah Ilmiah Teknologi Elektro, 17(1), 59.

[17] A. . Putra, Herfina, S. Maryana, and A. Setiawan, "Implementasi Algoritma AES (Advance Encryption Standard) Rijndael Pada Aplikasi Keamanan Data," Jurnal Ilmiah Penelitian Teknologi informasi & Komputer, vol. 1, no. 2. pp. 46–51, 2020.

DOI: https://doi.org/10.35457/antivirus.v17i1.2811