eISSN : 2621-4970
pISSN : 2621-4962

Volume 8 | No 1
Januari | **2025**

Perbandingan Penggunaan Algoritma Kriptografi *DES, RSA*, Modifikasi *DES* dan Modifikasi *RSA* untuk Penyandian Database

Triloka Mahesti[1], Arum F Ciptaningtyas[2], Agni Astungkara[3]
Jurusan Akuntansi
Politeknik Negeri Semarang
trilokamahesti@gmail.com

Abstrak

Pengembangan ilmu kriptografi mendukung penyimpanan data yang terkomputerisasi karena berhubungan dengan keamanan data perusahaan yang dapat berpengaruh pada operasional perusahaan. Penelitian ini bertujuan melakukan eksplorasi teknik kriptografi *symmetric* dan *asymmetric* yang banyak digunakan yaitu *Data Encryption Standard* (DES) dan *Rivest-Shamir-Adleman* (RSA). Penelitian ini melakukan modifikasi algoritma DES dan algoritma RSA pada proses enkripsinya. Penelitian ini mengimplementasikan empat algoritma kriptografi yaitu DES, modifikasi DES, RSA dan modifikasi RSA. Penelitian menunjukkan, proses enkripsi algoritma RSA memiliki waktu yang lebih cepat daripada algoritma DES, tetapi algoritma DES memiliki pengamanan lebih baik dibanding dengan algoritma RSA karena proses perhitungannya yang cukup rumit dan sulit. Hasil modifikasi algoritma DES menunjukkan hasil yang kurang baik karena hasil enkripsi memiliki nilai ASCII yang rendah, sehingga jika dikonversi akan menghasilkan karakter ASCII yang tidak nampak. Pada modifikasi algoritma RSA ketika dipilih nilai p dan q yang besar yaitu 19 dan 37, akan membantu proses enkripsi agar nilai ASCII yang tetap dapat dikonversi dalam karakter ASCII. Hal ini memperlihatkan bahwa modifikasi algoritma RSA memiliki hasil yang lebih baik dari modifikasi algoritma DES. Penelitian selanjutnya dapat dilakukan penggabungan algoritma DES dan algoritma RSA yang dilakukan secara *sequence* untuk melakukan enkripsi data yang lebih rumit, hal ini untuk menghindari penyalahgunaan data.
Kata kunci: *Data Encryption Standard, Rivest-Shamir-Adleman*, Kriptografi

Abstract

The development of cryptography supports computerized data storage because it relates to the security of company data which can affect company operations. This research aims to explore symmetric and asymmetric cryptography techniques that are widely used, namely Data Encryption Standard (DES) and Rivest-Shamir-Adleman (RSA). This research modified the DES algorithm and RSA algorithm in the encryption process. This research implements four cryptographic algorithms namely DES, modified DES, RSA and modified RSA. The research shows that the RSA algorithm encryption process has a faster time than the DES algorithm, but the DES algorithm has better security than the RSA algorithm because the calculation process is quite complicated and difficult. The results of the DES algorithm modification show poor results because the encryption results have low ASCII values, so if converted will produce ASCII characters that are not visible. In the RSA algorithm modification when a large p and q value is chosen, namely 19 and 37, it will help the encryption process so that the ASCII value can still be converted into ASCII characters. This shows that the RSA algorithm modification has better results than the DES algorithm modification. Future research can be done by combining the DES algorithm and RSA algorithm in sequence to encrypt more complicated data, this is to avoid data misuse.
Keywords: *Data Encryption Standard, Rivest-Shamir-Adleman, Cryptography*

PENDAHULUAN

Era komputerisasi saat ini melakukan pengiriman data menjadi lebih mudah karena adanya jaringan internet, namun keamanan tersebut beriringan dengan kebutuhan keamanan dan kerahasiaan data yang perlu diperhatikan dalam proses pengiriman dan penyimpanan data. Data rahasia yang tersimpan pada komputer atau yang dikirim melalui internet seringkali sangat mudah diakses dan terbaca oleh pihak yang tidak berkepentingan dan tidak bertanggung jawab karena datanya berbentuk *plaintext*. Pencegahan dengan melakukan pengamanan data terlebih dahulu dapat menghindari pembacaan dan pelacakan data oleh pihak yang tidak bertanggung jawab [1].

Database karyawan mencakup berbagai informasi karyawan yaitu NIK, alamat, gaji dan banyak data yang berguna untuk operasional sistem informasi perusahaan. Penggunaan *database* karyawan sering ditampilkan dalam bentuk *plaintext* sehingga memudahkan *cryptanalyst* untuk melakukan pembocoran, pendistribusian dan modifikasi *database* [2]. Hal ini sangat mengkhawatirkan karena dapat mengganggu proses operasional perusahaan.

Kriptografi yang merupakan ilmu pengetahuan untuk menyembunyikan informasi dapat menjaga pesan yang disampaikan terpelihara keaslian dan kerahasiaannya. Proses kriptografi adalah melakukan penyembunyian informasi dengan mengkonversi *plaintext* ke *ciphertext* (enkripsi) kemudian mengembalikan *ciphertext* ke *plaintext* (*decription)* [3]. Terdapat dua jenis kriptografi yang ada saat ini yaitu *symmetric* dan *asymmetric*. Perbedaan mendasar pada kedua jenis kriptografi ini adalah penggunaan kunci pada proses enkripsi dan deskripsi, dimana kiptografi *symmetric* menggunakan kunci yang sama sedangkan kryptografi *asymmetric* menggunakan kunci yang berbeda.

Salah satu algoritma *symmetric* yang paling banyak digunakan adalah kriptografi *Data Encryption Standard* (DES) [4]. *Plaintext* dalam blok 64bit dienkripsi menggunakan *internal key* 56bit menjadi *ciphertext* 64 bit. DES merupakan *block chipper* yang mentransformasikan *input* 64bit menjadi output 64bit dengan beberapa tahap enkripsi. Untuk membalik enkripsi digunakan kriptografi DES dengan menggunakan tahapan dan kunci yang sama yaitu kunci eksternal 64bit dari kunci internal [5]. Algoritma *asymmetric* yang merupakan kriptografi RSA yang memiliki dua kunci pada penerapannya yaitu *public key* untuk proses enkripsi dan *privat key* untuk proses dekripsi. *Privat key* pada algoritma RSA ini perlu dirahasiakan dari pihak manapun selain yang berkepentingan.

Penelitian ini akan melakukan perbandingan dalam penggunaan algoritma kriptografi *symmetric* dan *asymmetric* yaitu DES dan RSA untuk menentukan mana algoritma yang memiliki waktu enkripsi dan deskripsi yang lebih cepat serta keamanan yang lebih baik. Proses modifikasi algoritma RSA dan DES juga akan dilakukan untuk melihat modifikasi mana yang lebih baik dalam mengamankan data.

Rumusan Masalah

Penjabaran latar belakang di atas dapat memberikan simpulan rumusan masalah untuk penelitian sebagai berikut:

1. Bagaimana perbandingan penggunan algoritma kriptografi *symmetric* dan *asymmetric* yaitu DES dan RSA pada enkripsi data karyawan?
2. Bagaimana implementasi algoritma DES dan RSA secara manual?

3. Apakah kelebihan dan kekurangan algoritma DES dan RSA?

Tujuan Penelitian

Tujuan penelitian yang akan dilakukan berdasarkan latar belakang yang telah dipaparkan adalah sebagai berikut:

1. Mengetahui perbandingan penggunan algoritma kriptografi *symmetric* dan *asymmetric* yaitu DES dan RSA pada enkripsi data karyawan.
2. Mengetahui implementasi algoritma DES dan RSA secara manual dan mengetahui kelebihan dam kekurangan dari masing-masing algoritma.

Manfaat Penelitian

Tujuan penelitian di atas memberikan manfaat penelitian agar perusahaan dapat membandingkan penggunaan algoritma kriptografi sesuai kebutuhan serta dapat menjabarkan proses implementasi algoritma DES dan RSA sehingga dapat menjadi referensi bagi penelitian lanjutan atau menjadi perbandingan bagi penggunaan secara professional.

Tinjauan Pustaka

Penelitian sebelumnya yang berjudul "Implementasi DES (*Data Encryption Standard)* untuk Penyandian Data *Bill of Material* (BOM) pada Divisi Produksi PT Siantar TOP, Tbk". Algoritma DES digunakan untuk metode pengamanan dalam penelitian ini untuk meningkatkan pengamanan data BOM divisi produksi PT. Siantar Top, Tbk. Hasil penelitian memperoleh kesimpulan bahwa implementasi DES dapat mengamankan data BOM dengan mengenkripsi per materialnya. Tahapan enkripsi algoritma DES memproses data dalam jaringan feistel sebanyak 16 kali putaran, enkripsi data hanya bisa dalam bentuk teks dan menggunakan kunci simetris [6].

Penelitian lain yang dilakukan oleh Yanti, N. R., Alimah, A., & Ritonga, D. A. (2018) melakukan pengamanan pada *record database* yang ditampilkan dalam teks. Penelitian ini menggunakan metode algoritma DES dan menghasilkan kesumpulan bahwa algoritma DES dapat melakukan enkripsi yang menyulitkan pihak tidak berkepentingan untuk memahami isi *record database*. Penyandian dilakukan dengan menentukan nama *database*, memilih text yang ingin disandikan dan pemilihan kunci yang besar pada algoritma DES dapat meningkatkan keamanan pada *database* yang dienkripsi [4].

Penelitian lain berjudul "Penerapan Enkripsi dan Deskripsi *File* Menggunakan Algoritma *Data Encryption Standard* (DES)" memiliki fokus pada penerapan algoritma DES untuk pengamanan suatu file. Hasil dari penelitian ini memiliki kesimpulan bahwa dengan adanya kriptografi dengan menggunakan algoritma DES dapat mengamankan data penting yang akan dikirim melalui internet. Algoritma DES ini melakukan proses enkripsi dan deskripsi pada file atau teks dengan mekanisme yang sama dan memiliki kecepatan yang relative sama. [7]

Penelitian berjudul "Model Modifikasi Kriptografi Algoritma RSA untuk Keamanan data Pada Database E-Voting" melakukan modifikasi pada proses melakukan enkripsi menggunakan algoritma RSA. Kualitas dan validasi perlu dilakukan pada penelitian, proses ini dapat dilakukan menggunakan metode *black box* dan metode penelitian menggunakan metode eksperimen. Hasil penelitian artikel ini dapat digunakan untuk mengamankan dan merahasiakan database *e-voting* agar

data harus melalui proses deskripsi untuk dapat membaca datanya [11].

Penelitian lain berjudul "Perbandingan Kriptografi Menggunakan Algoritma *Data Encryption Standart* (DES) dan Algoritma *Rivest Shamir Adleman* (RSA) untuk Keamanan Data" bertujuan untuk membandingkan kinerja dari kedua algoritma tersebut dalam melakukan enkripsi dan dekripsi. Pengujian dilakukan dengan memberikan kuesioner kepada responden yang berisi kelebihan dan kekurangan kedua algoritma tersebut dari sistem yang sudah dibangun. Metode yang digunakan adalah *student t-test* dengan menggunakan *paired two sample for means.* Hasil penelitian adalah proses enkripsi dan dekripsi menggunakan algoritma RSA lebih cepat daripada DES, sedangkan untuk keamanan DES lebih baik dari RSA karena proses perhitungannya yang rumit dan sulit. [12]

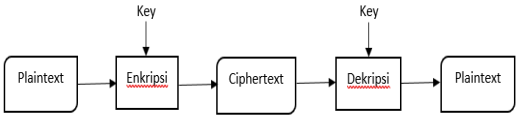
Penelitian lain berjudul "Studi Perbadingan Kriptografi Menggunakan Metode DES, Triple DES dan RSA" melakukan perbandingan untuk mengetahui lama proses dekripsi antara algoritma RSA, DES dan triple DES. Pembuatan perangkat lunak pada penelitian ini menggunakan Visual Basic 6.0. Hasil penelitian ini adalah proses enkripsi menggunakan ketiga algoritma akan menambah kapasitas file karena terdapat perubahan struktur variable dan maximum key dan pada penggunaan algoritma RSA jika kapasitas file semakin besar maka proses enkripsi dan dekripsi akan semakin lama. [13]

Berdasarkan penelitian yang telah dilakukan sebelumnya, telah ada penelitian yang membahas mengenai algoritma kriptografi DES, Triple DES dan RSA dengan metode dan tahap penelitian yang berbeda-beda. Pada penelitian ini untuk membedakan dengan penelitian yang telah ada, akan dilakukan perbandingan penggunaan algoritma DES dan RSA penggunaan kedua algoritma secara normal dan penggunaan dengan modifikasi. Penelitian ini akan menggunakan metode penelitian eksperimen untuk pengujiannya. Penelitian ini diharapkan dapat dijadikan perbandingan pada pengembangan untuk pengembangan keamanan data secara professional, referensi bagi penelitian lanjutan.

LANDASAN TEORI

Kriptografi

Kriptografi merupakan ilmu pengetahuan yang mempelajari keamanan informasi untuk untuk menjaga data rahasia. Ilmu ini menggunakan pengetahuan matematika untuk mengenkripsi dan dekripsi data. Kriptografi memungkinkan pengguna untuk mengirimkan informasi sensitif melalui jaringan internet yang dinilai merupakan jaringan yang tidak aman tetapi data hanya dapat dibaca oleh pihak yang dituju. Skema dari proses kriptografi dijelaskan pada Gambar 2.



Gambar 1. Skema Proses Kriptografi

Skema proses kriptografi terdiri dari :

a. *Plaintext* atau *cleartext*
   Plaintext merupakan teks biasa yang dapat dibaca dan dimengerti tanpa harus menggunakan tindakan khusus.
b. Enkripsi
   Enkripsi memungkinkan hanya penerima yang dapat membaca teks yang dikirim dengan melakukan transformasi *plaintext* menggunakan suatu algoritma. Penerima yang dapat mengakses teks tersebut jika memiliki akses khusus seperti

cryptanalyst atau penerima yang memiliki kunci.

c. *Ciphertext*

Ciphertext merupakan hasil dari proses transformasi informasi menggunakan algoritma tertentu untuk membuat informasi tersebut tidak terbaca oleh pihak manapun kecuali pihak yang memiliki akses khusus atau memiliki kunci.

d. Dekripsi

Dekripsi merupakan proses decoding data yang telah dienkripsi menggunakan format rahasia. Proses ini merupakan tahap untuk mengembalikan *ciphertext* ke *plaintext* aslinya dengan membutuhkan kunci rahasia.

e. Kunci

Kunci dalam kriptografi merupakan elemen yang penting karena algoritma kriptografi bekerja dengan menggunakan kunci. Keamanan data terenkripsi sepenuhnya bergantung pada dua hal yaitu kekuatan algoritma kriptografi dan kerahasiaan kunci. Kerahasiaan kunci perlu dijaga dengan baik karena jika kunci yang digunakan dapat ditemukan, maka serumit apapun algoritmanya dapat dipecahkan.

Penerapan Teknik kriptografi dilakukan untuk mencapai kerahasiaan, integritas data, autentikasi, dan ketiadaan penyengkalan sehingga data yang dikirimkan terjamin keasliannya [8][9].

Data Encyption Standard (DES)

DES yang merupakan algoritma kriptografi yang banyak digunakan saat ini merupakan sistem kriptografi *symmetric* yang merupakan *block cipher* yaitu skema enkripsi/dekripsi di mana blok *plaintext* digunakan untuk menghasilkan blok *cipher-text* dengan panjang yang sama.

DES beroperasi dengan mengenkripsi blok 64bit menggunakan kunci 56 bit. Algoritma DES mengubah input 64bit dengan serangkaian langkah menjadi 64bit output. Untuk membalikkan enkripsi dilakukan langkah yang sama dan menggunakan kunci yang sama. Penggunaan DES yang dilakukan secara luas saat ini menjadi topik pembicaraan mengenai kontroversi seberapa aman algoritma DES

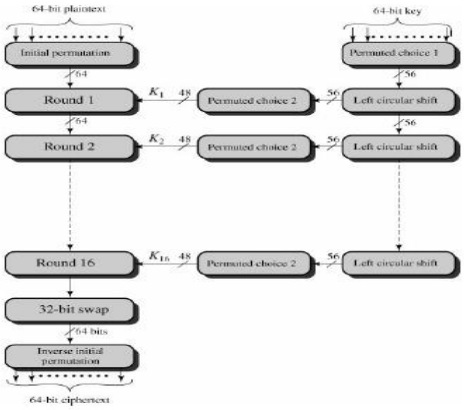
Algoritma *Rivest Shamir Adleman* (RSA)

Algoritma kunci public yang paling popular dan merupakan salah salah satu algoritma kriptografi *assymetric* adalah algoritma RSA yang menggunakan dua kunci berbeda pada proses enkripsi dan deskripsinya. Konsep ini menjawab permasalahan pada penerapan *symmetric key* seperti algoritma DES yaitu pada *key distribution* dan *digital signatures.* Pemfaktoran pada algoritma RSA dilakukan dengan menggunakan bilangan yang sangat besar dengan memilih dua bilangan prima secara acak sehingga mendukung keamanan data. Peneliti dari *Massachussets Institute of Technology* (MIT) yaitu Ron Rivest, Adi Shamir dan Leonard Adleman merupakan penemu dari algoritma RSA ini [14]. Kemampuan melakukan pemfaktoran bilangan besar menjadi faktor bilangan prima merupakan kelebihan dari algoritma RSA [15].

METODE PENELITIAN

A. *Data Encyption Standard* (DES)

Skema enkripsi DES dijabarkan pada Gambar 3 dengan memiliki dua input enkripsi yaitu *plaintext* untuk dienkripsi dan kuncinya.



Gambar 2. Skema Enkripsi DES

Pemrosesan *plaintext* (sisi kiri) berlangsung dalam tiga fase:

a. *Plaintext* 64-bit dipermutasi dengan matriks permutasi awal (initial permutasi (IP)) yang mengatur ulang bit untuk menghasilkan input permutasi.
b. Hasil permutasi awal dienciphering sebanyak 16 putaran menggunakan fungsi yang sama yaitu fungsi permutasi dan substitusi.
c. Output pada putaran terakhir (keenam bela) terdiri dari 64bit yang merupakan fungsi dari input plaintext dan kunci. Output ini dipermutasi dengan matriks permutasi balikan menjadi *ciphertext.*

Pemrosesan kunci (sisi kanan) berlangsung dalam fase:

a. Kunci melalui fungsi permutasi.
b. Pada masing-masing 16 putaran, *subkey* (K_i) dihasilkan menggunakan kombinasi shift lingkaran kiri dan permutasi.
c. Permutasi yang sama dilakukan untuk setiap putaran, tetapi menghasilkan *subkey* berbeda karena adanya pergeseran bit kunci yang berulang.

Algoritma DES memiliki langkah kerja sebagai berikut [10]:

a. Proses awal dilakukan enkripsi blok *plaintext* dengan cara membagi menjadi dua bagian yaitu L[i] dengan panjang 32bit dan R[i] dengan Panjang 32bit dan dilakukan dalam 16 putaran.
b. Setiap putaran yang dilakukan dengan *I* dan pada blok R dimasukan fungsi transformasi (f).
c. Fungsi transformasi pada blok R selanjutnya dikombinasikan dengan kunci Ki dengan meng-XOR-kan fungsi f dengan blok L. hal ini dilakukan aggar blok R terbaru didapatkan.
d. Blok L yang didapatkan dapat diambil dari blok R. hal ini merupakan salah satu putaran DES.

Algoritma DES memiliki proses kerja yang dibagi menjadi dua bagian yaitu:

a. *Enkripsi*

Plaintext dilakukan enkripsi setelah dilakukan permutasi awal. Jaringan Feistel digunakan untuk mengenkrispi setiap blok *plaintext.* Enkripri tersebut dilakukan sebanyak 16 kali putaran. Pernyataan matematis enkripsi ini dinyatakan dengan formula sebagai berikut:

$$L_i = R_i - 1 \quad (1)$$

$$R_i = L_i - 1 \oplus f(R_i - 1, K_i) \quad (2)$$

Fungsi ekspansi dinyatakan dengan menggunakan symbol E, perluasan blok $R_i - 1$ dengan Panjang 32bit diperpanjang menjadi 48bit. Vector A merupakan hasil ekspansi $E(R_i - 1)$ yang diXOR.

$$E(R_i - 1) \oplus K_i = A \quad (3)$$

Pengolahan vector A dilakukan dengan membagi vector A menjadi delapan bagian, dimana setiap bagian terdiri dari 6bit yang selanjutnya digunakan sebagai input proses substitusi. Vector A yang dibagi menjadi 8 bagian tersebut merupakan kotak S-box yang digunakan untuk substitusi. Proses substitusi pada kotak S-Box terdiri dari 6bit yang selanjutnya menghasilkan keluaran 4bit. S-Box-1

merupakan kelompok 6bit pertama, dan seterusnya hingga S-Box-8.

b. Dekripsi

Jika sebelumnya adalah proses enkripsi, kal ini dilakukan proses deskripsi pada *ciphertext* yang merupakan inverse dari proses enkripsi. Proses deskripsi ini dilakukan dari K[16] menuju ke K[1], dengan keluaran pada tiap putarannya adalah *deciphering* [$HYPERLINK \l "Roh121" 5$], adalah:

$$L_i = R_i - 1 \quad (4)$$

$$R_i = L_i - 1 \; \boldsymbol{Error}!^{\oplus}\} f(R_i - 1, K_i) \quad (5)$$

B. Modifikasi DES

Modifikasi yang dilakukan pada algoritma DES adalah saat proses enkripsi sedangkan untuk pembangkitan kunci internal dilakukan dengan cara yang sama pada algoritma DES pada umumnya. Proses encode data pada tahap awal enkripsi dilakukan dengan algoritma DES pada umumnya, sedangkan untuk fungsi f pada data dilakukan modifikasi sebagai berikut :

a. Expansi Nilai R[0] akan dilakukan dengan algoritma DES pada umunya.
b. Pada proses ini E(R[0]) di-XOR dengan K[1], tetapi akan dimodifikasi dengan E(R[0]) di-AND dengan K[1].
c. A[1] disubsitusikan ke dalam S-Box DES akan dilakukan dengan algoritma DES pada umumnya.
d. Permutasi B[1], berdasarkan tabel P-Box DES.
e. Nilai R[1] dan L[1] didapatkan dengan persamaan $R[1] = P[1] \oplus L[0]$ akan dimodifikasi dengan $R[1] = P[1]\ AND\ L[0]$.
f. Proses di atas selanjutnya dilakukan dengan round 2-16 dengan menggunakan proses yang sama. Pada round ke-16 akan didapatkan R16 dan L16 yang akan dipermutasi menggunakan tabel IP^{-1}.

C. Algoritma *Rivest Shamir Adleman* (RSA)

Proses pada algoritma RSA dilakukan dengan membangkitkan kunci public dan privat, melakukan enkripsi selanjutnya deskripsi. Pada proses pembangkitan kunci public dan kunci privat dilakukan dengan proses sebagai berikut [11]:

a. Lakukan pemilihan dua bilangan prima secara acak dengan bilangan prima pertama p dan bilangan prima kedua q yang berbeda p ≠ q. Kedua bilangan prima yang dipilih merupakan bilangan yang bersifat rahasia dan semakin besar bilangan yang dipilih maka semakin baik.
b. Lakukan penghitungan modulus *(n)* pada kunci public dan dan kunci privat dengan cara mengalikan bilangan prima pertama *(p)* dan bilangan prima kedua *(q)*. Nilai modulus yang dihitung dengan formula $n = p * q$ bukan merupakan bilangan rahasia.
c. Untuk mencari kunci prifat dilakukan dengan mengunakan formula $\varphi(n) = (p - 1) * (q - 1)$. Sesuai Namanya kunci privat dengan nilai φ(n) merupakan bilangan yang rahasia.
d. Selanjutnya dilakukan penghitungan nilai e dengan syarat $1 < e < \varphi(n)\ dan\ GCD(\varphi(n), e) = 1$. Nilai e ini bukan merupakan bilangan rahasia.
e. Nilai d dicari dengan menggunakan syarat (d * e) mod φ(n) = 1 atau $d = (1 + k * \varphi(n)) / e$. Nilai k yang merupakan nilai yang dapat digunakan untuk menghitung nilai d dengan cara mencoba-coba nilai k agar mendapat nilai bulat d. Nilai d merupakan nilai yang rahasia.

Proses enkripsi *plaintext* (P) pada algoritma RSA dilakukan dengan menggunakan persamaan matematis $C = Me \bmod n$, persamaan ini menggunakan kunci public (e,n) yang akan menghasilkan

ciphertext (C). Hasil dari *ciphertext* (C) akan dikirimkan kepada penerima dan hanya penerima yang memiliki kunci privat (d,n) untuk mendekripsi *ciphertext* dengan menggunakan formula $P = C^d \bmod n$.

D. Modifikasi Algoritma RSA

Modifikasi algoritma RSA akan dilakukan pada proses enkripsi dengan $Modifikasi\ Ci = Ci\ mod\ 256$. Modifikasi ini dibutuhkan karena pemilihan p dan q yang besar akan menghasilkan nilai ASCII yang tinggi sehingga tidak bisa dikonversi ke dalam karakter ASCII yang memiliki nilai maksimum 128.

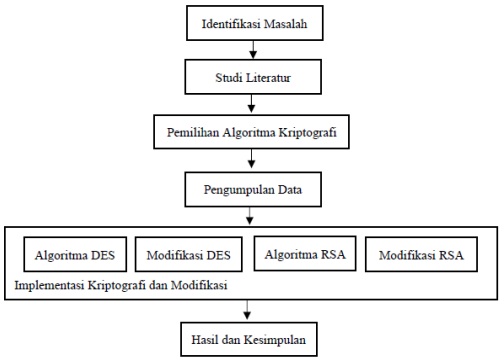
ANALISIS DAN PERANCANGAN

A. Data Penelitian

Penelitian ini menggunakan data karyawan PT Mitra Edukasi Nusantara (Kompas Gramedia Grup of Manufacture) yang berisi data pribadi karyawan. Data ini merupakan sumber informasi yang digunakan dalam proses operasional perusahaan. Data ini perlu diamankan dengan baik agar tidak disalahgunakan karena saat ini data ditampilkan dalam bentuk text.

B. Tahap Penelitian

Tahapan penelitian yang dilakukan pada penelitian ini antara lain adalah identifikasi masalah, studi literatur, pemilihan algoritma kriptografi, pengumpulan data, implementasi algoritma kriptografi dan modifikasinya serta hasil dan kesimpulan. Tahap penelitian disajikan pada Gambar 3.



Gambar 3. Tahapan Penelitian

Tahap identifikasi masalah dilakukan untuk mengetahui masalah yang terjadi dan menentukan metode kriptografi yang akan digunakan. Tahap kedua yaitu studi literatur dilakukan untuk mempelajari jurnal-jurnal terdahulu sebagai acuan dan sebagai pengembangan penelitian. Tahap ketiga adalah menentukan algoritma kriptografi yang digunakan, pada tahap ini dilakukan eksplorasi pada algoritma DES dan algoritma RSA agar didapatkan modifikasi dari kedua algoritma kriptografi tersebut. Tahap selanjutnya dilakukan pengumpulan data yang akan diamankan. Tahap kelima adalah proses implementasi algortima kriptografi dan modifikasinya, implementasi algoritma DES, RSA, modifikasi DES dan modifikasi RSA dilakukan secara terpisah untuk mengetahui hasil dari masing-masing algoritma sehingga dapat dibandingkan. Tahap terakhir adalah hasil dan kesimpulan yang dapat digunakan sebagai rekomendasi pengamanan data kedepan.

IMPLEMENTASI DAN PEMBAHASAN

Pada penelitian ini data yang akan dipakai adalah data karyawan PT Mitra Edukasi Nusantara (Kompas Gramedia Group of Manufacture). Data yang akan digunakan sebagai contoh untuk penyandian adalah NIK karyawan dengan *plaintext* adalah GRAMEDIA dan *key* adalah GRA52223.

1. Algoritmas DES

1.1. Pembangkitan kunci internal

Pembangkitan kunci internal melibatkan PC1 dan PC2 dan menghasilkan 16 subkunci internal untuk kemudian diproses dalam fungsi f dengan data yang yang sudah di *encode*.

1.1.1. Langkah awal adalah mengkonversi kunci menjadi biner

Key : GRA52223

Biner : 01000111 01010010 01000001 00110101 00110010 00110010 00110010 00110011

1.1.2. Kunci sebesar 64bit akan diubah urutannya dengan permutasi menggunakan tabel PC1 menjadi 56bit. Hasil permutasi dinotasikan sebagai K+ dan dibagi menjadi dua bagian yang dinotasikan dengan C_0 dan D_0.

K^+ : 00000000000000111111100001111111110011000010010000000001010

C_0 : 00000000000001111111 10001111

D_0 : 111100110000100100 0000001010

1.1.3. Pergeseran bit dengan tabel *shift schedule*

Notasi C_0 dan D_0 digeser menggunakan tabel *shift schedule* hingga menghasilkan C_n dan D_n dengan n = 1,2,…..,16.

C0 = 00000000000001111111110001111

D0 = 11110011000010010000000001010

C1 = 00000000000011111111100011110

D1 = 11100110000100100000000010101

C2 = 00000000000111111110011110

D2 = 11001100001001000000001010111

C3 = 00000000011111111100011110000

D3 = 00110000100100000000101011111

C4 = 00000001111111100011110000000

D4 = 11000010010000000010101111100

C5 = 00000111111110001111000000000

D5 = 00001001000000001010111110011

C6 = 00011111111000111100000000000

D6 = 00100100000000101011110011000

C7 = 01111111100011111000000000000

D7 = 10010000000010101111001100000

C8 = 11111110001111000000000000001

D8 = 01000000001010111100110000010

C9 = 11111100011110000000000000011

D9 = 10000000010101111001100000100

C10 = 11110001111000000000000001111

D10= 00000010101111001100001001000

C11 = 11000111100000000000000111111

D11= 00001010111100110000100100000

C12 = 00011110000000000000011111111

D12= 00101011110011000010010000000

C13 = 01111000000000000001111111100

D13= 10101111001100001001000000000

C14 = 11100000000000000111111110001

D14= 10111100110000100100000000001

C15 = 10000000000000011111111000111

D15= 11110011000010010000000000101

C16 = 00000000000000111111110001111

D16= 11100110000100100000000001010

1.1.4. Substitusi Cn dan Dn pada matrik PC2

Matrik PC2 akan mengubah 56 bit CnDn menjadi 48 bit yang akan dinotasikan menjadi Kn, n=1,2,….,16.

K1 :110100 001000 010010 101110 011000 110000 000110 000010

K2 :110100 001000 111010 100010 000001 101010 001111 010010

K3 :111100 001011 101000 100010 001101 011000 010101 000001

K4 :101000 001011 011001 000110 010010 101000 010001 000010

K5 :011000 000101 011001 010110 010011 001110 010100 001100

K6 :011001 001101 000101 110000 001010 000101 010011 001000

K7 :100001 101100 000101 110011 110010 001101 000000 100011

K8 :101011 110100 001100 010011 100001 100100 111000 101000

K9 :001011 110101 001100 001011 000000 010101 110000 010010

K10: 001010 110001 000111 001001
110011 010000 000000 110100
K11: 000110 010100 100011 011001
100000 010100 101011 001100
K12: 000101 010110 100110 011000
000100 001001 001010 010101
K13: 000101 100010 110110 000101
100100 110000 010010 100101
K14: 010110 110010 110000 000101
000010 100010 101110 000001
K15: 010010 011010 010010 101100
001100 100110 000100 010101
K16: 110000 011010 010010 101100
101001 100000 100001 000111

1.2. Enkripsi Data

1.2.1. Encode data

Kata yang akan disandikan adalah GRAMEDIA dikonversikan menjadi bilangan biner. Hasil konversi dinotasikan sebagai M dan M dipermutasi dengan tabel IP.

Plaintext: GRAMEDIA

M: 01000111 01010010 01000001 01001101
01000101 01000100 01001001
01000001

IP: 11111111 00000010 00111001 11011101
00000000 00000000 01001000
00000011

Hasil IP biner plain selanjutnya dibagi menjadi 2 kelompok yang masing-masing terdiri dari 32bit.

L0: 11111111 00000010 00111001
11011101

R0: 00000000 00000000 01001000
00000011

1.2.2. Fungsi f pada data

Bagian ini akan melibatkan blok Rn dan Kn, blok Rn sebelum masuk ke fungsi f akan diturunkan langsung menjadi L_{n+1}.

Round 1 (n=1)

i. Expansi Nilai R[0]

E(R0):100000 000000 000000 000000
001001 010000 000000 000110

ii. E(R[0]) di-XOR dengan K[1]

E(R0)100000 000000 000000 000000
001001 010000 000000 000110

K1 110100 001000 010010 101110
011000 110000 000110 000010

XOR

A1 010100 001000 010010 101110
010001 100000 000110 000100

iii. A[1] disubsitusikan ke dalam S-Box DES

A1	S-Box	BARIS	KOLOM	DESIMAL	[1]
010100	S1	00	1010	6	0110
001000	S2	00	0100	6	0110
010010	S3	00	1001	13	1101
101110	S4	10	0111	13	1101
010001	S5	01	1000	5	0101
100000	S6	10	0000	9	1001
000110	S7	00	0011	14	1110
000100	S8	00	0010	8	1000

iv. Permutasikan B[1], berdasarkan tabel P-Box DES

P(B1) 11111100 00010101
10110111 01010001

v. Mendapatkan Nilai R[1] dan L[1] dengan $R[1] = P[1] \oplus L[0]$

P(B1)	11111100	00010101	10110111	01010001
L0	11111111	00000010	00111001	11011101 XOR
R1	00000011	00010111	10001110	10001100
L1=R0	00000000	00000000	01001000	00000011

Lakukan proses ini untuk round 2-16 dengan proses yang sama. Pada round ke-16 akan didapatkan R16 dan L16 yang akan dipermutasi menggunakan tabel IP^{-1}.

R16: 00000110 11100010
10100110 01000101

L16=R15: 11011000 00011010
10000011 01101010

Hasil dari *ciphertext* adalah:

Biner: 00001001 01111110 01000101
10100010 10100000 00010110
10010011 10011100

HEXA: 09 7E 45 A2 A0 16 93 9C

ASCII: HT~E¢ SYN]ST

2. Modifikasi Algoritma DES

2.1. Pembangkitan kunci internal akan dilakukan dengan cara yang sama pada algoritma DES pada umumnya.

2.2. Enkripsi Data

2.2.1. Encode data akan dilakukan dengan menggunakan algoritma DES pada umumnya.

2.2.2. Fungsi f pada data

Pada fungsi f ini akan dilakukan modifikasi pada beberapa tahap.

a. Expansi Nilai R[0] akan dilakukan dengan algoritma DES pada umunya.
b. Pada proses ini seharusnya E(R[0]) di-AND dengan K[1], tetapi akan dimodifikasi dengan meng-XOR-kan E(R[0]) dengan K[1].
c. S-Box DESakan disubstitusi dengan A[1] akan dilakukan dengan algoritma DES pada umumnya.
d. Permutasi B[1], berdasarkan tabel P-Box DES.
e. Mendapatkan Nilai R[1] dan L[1] dengan $R[1] = P[1] \oplus L[0]$ akan dimodifikasi dengan $R[1] = P[1]\ AND\ L[0]$.
f. Proses di atas akan dilakukan pada round 2-16 menggunakan proses yang sama. Pada round ke-16 akan didapatkan R16 dan L16 yang akan dipermutasi menggunakan tabel IP^{-1}.

Hasil dari *ciphertext* adalah:

Biner: 00001000 00000000 00000010 00000000 00001010 00000000 10000100 1000000

HEXA: 8 0 2 0 A 0 84 80

ASCII: BS NUL STX NUL LF NUL IND Dicadangkan

3. Algoritma RSA

3.1. Pemilihan bilangan prima kecil

Pemilihan bilangan prima kecil akan dilakukan pada tahap ini untuk membangkitkan kunci public (e,n) dan kunci privat (d,n). Langkah-langkah yang dilakukan adalah sebagai berikut:

1. Nilai bilangan p = 3 dan q = 7.
2. Nilai n dihitung dengan mengalikan p dan q dengan hasil n = 21.
3. Nilai φ(n) dihitung dengan persamaan φ(n) = (p – 1) * (q – 1) sehingga menghasilkan nilai φ(n) = 12
4. Nilai e dihitung dengan interval 1 < e < φ(n) dan GCD(φ(n), e) = 1.

Tabel 1. Hitung Nilai e

e	Nilai GCD (12, e)
2	2
3	3
4	4
5	1
6	6
7	1
8	4
9	3

Tabel di atas merupakan hasil dari perhitungan e. Jika dilihat dari hasil tabel 1, maka dapat disimpulkan bahwa nilai e yang dapat digunakan adalah angka 5 atau angka 7. Karena kebutuhan bilangannya hanya satu maka yang akan digunakan adalah angka 5.

5. Persamaan d= (1 + k * φ(n)) / e selanjutnya digunakan untuk menghitung nilai d, dengan hasil sebagai berikut:

Tabel 2. *Hitung Nilai d*

Nilai k	d = (1 + k * φ(n)) / e	Hasil
1	d = (1 + 1 * 12) /5	2,6
2	d = (1 + 2 * 12) / 5	5
3	d = (1 + 3 * 12) / 5	7,4
4	d = (1 + 4 * 12) /5	9,8

Hasil perhitungan nilai d ditampilkan pada tabel di atas, dengan nilai d adalah 5.

6. Dapat disimpulkan kunci publik (5, 21) dan kunci privat (5, 21). Pemilihan nilai p dan q yang kecil dapat mengakibatkan kunci public dan kunci privat memiliki nilai yang sama seperti pada contoh di atas.

Enkripsi pada algoritma RSA dengan *plaintext* GRAMEDIA dan kunci public (5,21) menghasilkan ciphertext "".

Pemilihan p dan q yang terlalu kecil akan menghasilkan karakter ASCII dengan nilai decimal yang kecil sehingga menyebabkan karakter ASCII tidak nampak karena beberapa karakter ASCII pada desimal awal tidak nampak.

Tabel 3. Contoh Proses Enkripsi

Pi	Nilai ASCII Pi	Ci = P^e mod n	Karakter ASCII Ci
G	71	8	
R	82	10	"
A	65	11	"
M	77	14	
E	69	6	
D	68	17	
I	73	19	
A	65	11	

Contoh proses dekripsi untuk algoritma RSA dengan *ciphertext* "" dengan menggunakan kunci privat (5, 21) dapat dilihat pada tabel 4.

Tabel 4. Contoh Proses Dekripsi

Ci	Nilai ASCII Ci	Pi = C^d mod n	Karakter ASCII Pi
	8	71	G
"	10	82	R
"	11	65	A
	14	77	M
	6	69	E
	17	68	D
	19	73	I
	11	65	A

3.2. Pemilihan bilangan prima sedang

Bilangan prima sedang akan dicoba untuk dipilih pada tahap ini untuk membangkitkan kunci public (e,n) dan kunci privat (d,n).

1. Nilai bilangan p = 17 dan q = 11.
2. Nilai n = 187 dengan cara penghitungan n = p * q
3. Nilai φ(n) = 160 dengan cara penghitungan φ(n) = (p – 1) * (q – 1)
4. Selanjutnya menghitung nilai e sehingga 1 < e < φ(n) dan GCD(φ(n), e) = 1

Tabel 5. Hitung Nilai e

e	Nilai GCD (160, e)
2	2
3	1
4	4
5	5
6	2
7	1
8	8
9	1

Setelah melakukan penghitungan nilai e, didapatkan hasil seperti yang terlihat pada Tabel 5. Nilai e yang dapat digunakan adalah 3, 7 atau 9. Untuk nilai yang dapat digunakan adalah satu saja sehingga ditentukan menggunakan nilai 3.

5. Nilai d dihitung dengan persamaan d = (1 + k * φ(n)) / e.

Tabel 6. *Hitung Nilai d*

Nilai k	d = (1 + k * φ(n)) / e	Hasil
1	d = (1 + 1 * 160) /3	53,66666667
2	d = (1 + 2 * 160) / 3	107
3	d = (1 + 3 * 160) / 3	160,3333333
4	d = (1 + 4 * 160) /3	213,6666667

Berdasarkan tabel perhitungan nilai d diatas, maka nilai d adalah 107.

6. Pada perhitungan yang telah dilakukan didapatkan nilai kunci publik (3, 187) dan kunci privat (107, 187).

Langkah-langkah pada proses enkripsi pada algoritma RSA pada plaintext GRAMEDIA dengan kunci public (3, 187) menghasilkan ciphertext ´\ m B‰ U9 m. Pemilihan p dan q sedang akan menghasilkan karakter ASCII yang dapat terlihat/nampak.

Tabel 7. Contoh Proses Enkripsi

Pi	Nilai ASCII Pi	Ci = P^e mod n	Karakter ASCII Ci
G	71	180	´
R	82	92	\
A	65	109	m
M	77	66	B
E	69	137	‰
D	68	85	U
I	73	57	9
A	65	109	m

Contoh proses dekripsi untuk algoritma RSA dengan *ciphertext* ´\mB‰U9m menggunakan kunci privat (107, 187) dapat dilihat pada tabel 8.

Tabel 8. Contoh Proses Dekripsi

Ci	Nilai ASCII Ci	Pi = C^d mod n	Karakter ASCII Pi
´	180	71	G
\	92	82	R
m	109	65	A
B	66	77	M
‰	137	69	E
U	85	68	D
9	57	73	I
m	109	65	A

4. Modifikasi Algoritma RSA

Pada modifikasi algoritma RSA ini dilakukan proses pembangkitan kunci

public (e,n) dan kunci privat (d,n) dengan langkah sebagai berikut:

1. Nilai bilangan p = 19 dan q = 37.
2. Penghitungan nilai n dengan menggunakan n = p * q, sehingga nilai n = 703
3. Penghitungan nilai φ(n) = (p – 1) * (q – 1), sehingga didapatkan nilai φ(n) = 648
4. Penghitungan nilai e dengan interval 1 < e < φ(n) dan GCD(φ(n), e) = 1.

Tabel 9. Hitung Nilai e

e	Nilai GCD (160, e)
2	2
3	3
4	4
5	1
6	6
7	1
8	8
9	9

Tabel 9 memperlihatkan nilai e yang dapat digunakan adalah 5 atau 7. Pada penelitian ini dipilih angka 5 sebagai nilai e yang akan digunakan.

5. Nilai d dapat dihitung dengan menggunakan formula d = (1 + k * φ(n)) / e.

Tabel 10. *Hitung Nilai d*

Nilai k	d = (1 + k * φ(n)) / e	Hasil
1	d = (1 + 1 * 648) / 5	129,8
2	d = (1 + 2 * 648) / 5	259,4
3	d = (1 + 3 * 648) / 5	389
4	d = (1 + 4 * 648) / 5	518,6

Tabel 10 menunjukkan hasil perhitungan nilai d, dari tabel di atas dapat disimpulkan nilai d yang dipilih adalah 389.

6. Jadi didapatkan kunci publik (5, 703) dan kunci privat (389, 703).

Proses modifikasi algoritma RSA akan dilakukan pada proses enkripsi dengan $Modifikasi\ Ci = Ci\ mod\ 256$. Modifikasi ini dibutuhkan karena pemilihan p dan q yang besar akan menghasilkan nilai ASCII yang tinggi sehingga tidak bisa dikonversi ke dalam karakter ASCII yang memiliki nilai maksimum 128. Plaintext GRAMEDIA dengan kunci public (5, 703) menghasilkan ciphertext î± u: Ð Ø u. Pemilihan p dan q sedang akan menghasilkan karakter ASCII yang dapat terlihat/nampak.

Tabel 11. Contoh Proses Enkripsi

Pi	Nilai ASCII Pi	Ci = P e mod n	Modifikasi Ci	Karakter ASCII Ci
G	71	238	238	î
R	82	689	177	±
A	65	373	117	u
M	77	58	58	:
E	69	464	208	Ð
D	68	216	216	Ø
I	73	517	5	
A	65	373	117	u

Proses dekripsi untuk modifikasi algoritma RSA terlebih dahulu dilakukan dengan melakukan inverse modulo pada nilai ASCII Ci. Setelah itu dilakukan proses dekripsi dengan cara yang sama pada algoritma RSA pada umumnya. Hasil dekripsi *ciphertext* î± u: Ð Øu menggunakan kunci privat (389, 703) dapat dilihat pada tabel 12.

Tabel 12. Contoh Proses Dekripsi

Ci	Nilai ASCII Ci	Modifikasi Ci	Pi = C^d mod n	Karakter ASCII Pi
´	238	71	238	G
\	177	82	689	R
m	117	65	373	A
B	58	77	58	M
‰	208	69	464	E
U	216	68	216	D
9	5	73	517	I
m	117	65	373	A

KESIMPULAN

Hasil eksplorasi yang dilakukan pada penelitian ini mendapatkan hasil sebagai berikut:

1. Waktu untuk melakukan enkripsi dan deskripsi dapat dilakukan lebih cepat dengan menggunakan algoritma RSA disbanding dengan menggunakan algoritma DES.
2. Algoritma DES memiliki keamanan yang lebih baik dibanding dengan algoritma RSA karena proses perhitungannya yang cukup rumit dan sulit.
3. Modifikasi algoritma DES dengan mengubah rumus pada proses enkripsi

data menunjukkan hasil yang kurang baik karena hasil enkripsi memiliki nilai ASCII yang rendah sehingga ketika dikonversi menghasilkan karakter ASCII yang tidak nampak.

4. Modifikasi algoritma RSA memiliki hasil yang baik karena dapat membantu proses enkripsi. Hal ini bisa didapatkan ketika memilih nilai p dan q yang besar karena dapat menghasilkan nilai ASCII yang tinggi sehingga tidak bisa dikonversi ke dalam karakter ASCII yang memiliki nilai maksimum 128.
5. Pada penelitian selanjutnya dapat dilakukan penggabungan algoritma DES dan algoritma RSA untuk melakukan enkripsi data yang lebih rumit untuk menghindari penyalahgunaan data perusahaan pada orang yang kurang berkepentingan.

Saran

Saran untuk penelitian yang akan datang dapat dilakukan:

1. Mempertimbangkan kualifikasi waktu enkripsi dengan lebih detail.
2. Menggunakan metrik keamanan untuk memperlihatkan tingkat keamanan yang didapatkan.
3. Membahas tentang potensi penggabungan algoritma RES dan RSA termasuk tantangannya.

DAFTAR PUSTAKA

[1] S.H. Suryawan, Hamdani, "*Pengamanan Data File Dengan Menggunakan Algoritma Enkripsi Rivest Code 5*," Jurnal Informatika Mulawarman, vol. Vol. 8, No. 2, pp. 44-49, 2013.

[2] E. Setyaningsih, *Kriptografi & Implementasinya Menggunakan Matlab*, Yogyakarta: Andi, 2015.

[3] 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

[4] Yanti, N. R., Alimah, A., & Ritonga, D. A. (2018). *Implementasi Algoritma Data Encryption Standard Pada Penyandian Record Database*. J-SAKTI (Jurnal Sains Komputer Dan Informatika), *2*(1), 23. https://doi.org/10.30645/j-sakti.v2i1.53

[5] Hamdani, S.H. Suryawan, A. Septiarini,"*Pengujian Algoritma Rivest Code 5 Untuk Enkripsi Struktur File Dokumen*," Jurnal Informatika Mulawarman, vol. Vol. 8 No. 2, pp. 44-49, Juni 2013.

[6] Top, P. T. S., Ginting, E. F., Ibnutama, K., & Suryanata, M. G. (2019). *Implementasi DES (Data Encryption Standard) Untuk Penyandian Data Bill Of Material pada Divisi Produksi*. 18(2), 161–166.

[7] Primartha, R. (2011). *Penerapan Enkripsi Dan Dekripsi File Menggunakan Algoritma Data Encryption Standard (DES)*. Sriwijaya Journal of Information Systems, *3*(2), 371–387.

[8] H. Pandiangan and S. Sijabat "*Perancangan Media Pengiriman Pesan Teks Dengan Penyandian Pesan Menggunakan Algoritma Rc4 Berbasis Web* ," Jurnal Matik Penusa , vol. Volume XIX, No. 1 , no. ISSN 2088-3943 , pp. 63-71, Juni 2016

[9] U. R. S. Lubis, Mesran, and T. Zebua, "*Implementasi Algoritma Chua Chaotic Noise Pada Enkripsi Citra RGB*," in KOMIK *(*Konferensi Nasional Teknologi Informasi dan Komputer*)*, 2017, vol. I, no. 1, pp. 220–224.

[10] Hamdani, S.H. Suryawan, A. Septiarini,"*Pengujian Algoritma Rivest Code 5 Untuk Enkripsi Struktur File Dokumen*," Jurnal Informatika

Mulawarman, vol. Vol. 8 No. 2, pp. 44-49, Juni 2013.

[11] Martono. (2017). *Model Modifikasi Kriptografi Algoritma Rsa Untuk Keamanan Data Pada Database E-Voting*. Jurnal Ilmiah Media Sisfo, *11*(2), 896–910. *http://ejournal.stikom-db.ac.id/index.php/mediasisfo/article/view/245*

[12] Hidayat, A. Dan Faizin, A. *Perbandingan Kriptografi Menggunakan Algoritma Data Encryption Standart (Des) Dan Algoritma Rivest Shamir Adleman (Rsa) Untuk Keamanan Data*. JASIEK, Vol.1, No.2, Desember 2019, Pp. 143~148. DOI: 10.12928/JASIEK.V13i2.Xxxx.

[13] Suroso, A. *Studi Perbandingan Kriptograpi Menggunakan Metode Des, Triple Des Dan Rsa*. SIGMA – Jurnal Teknologi Pelita Bangsa. Volume 8, Nomor I, Maret 2018, ISSN : 2407-3903.

[14] Dony Ariyus. "*Pengantar Ilmu Kriptografi*". 2nd ed. Yogyakarta : Andi. 2012.

[15] Muhammad Safri Lubis, et.al. "*Penggunaan Algoritma RSA dengan Metode The Sieve of Eratosthenes dalam Enkripsi dan Deskripsi Pengiriman Email*". Seminar Nasional Aplikasi Teknologi Informasi (SNATI), (Juni, 2013) : 28-33.