Jurnal Ilmu Komputer dan Sistem Informasi (JIKOMSI
Volume 7, Nomor 1, Tahun 2024, Hal 203-207
ISSN 2723- 6129 (media online)
Link Jurnal : https://ejournal.sisfokomtek.org/index.php/jikom

Perbandingan *Algoritma RSA* dengan *Algoritma Blowfish* Pada Perancangan Aplikasi Keamanan Data

Ade Rahayu[1*], Amanda Putri Ardana[2], Chika Pramudhita[3], Dea Syafitri[4], Rianty Zabitha Sirega[5]
[1,2,3,4,5] Sistem Informasi, STMIK KAPUTAMA, BINJAI, Indonesia
Email: [1]aderahayu130602@gmail.com, [2]amandaputrii467@gmail.com, [3]chikapramudhita0056@gmail.com, [4]deasyafitri0202@gmail.com, [5]riantyzabitha78@gmail.com

Abstrak− Pada perkembangan teknologi informasi yang semakin tinggi dan meningkat, serta zaman yang serba canggih seperti ini dibutuhkan alat untuk mengirim pesan sudah banyak termasuk medianya seperti chatting, line atau sejenisnya sehingga kita bisa mengirim pesan dengan cepat begitu juga sebaliknya. Dari semua kemudahan itu tentu akan sangat berpengaruh ketika kita akan mengirim pesan yang isinya hanya orang-orang tertentu saja yang memiliki hak untuk mengetahui isinya. Salah satu yang harus benar-benar diwaspadai dan hati-hati adalah pesan yang bersifat rahasia karena jika pesan itu tersebar maka akan berdampak buruk pada kita sendiri atau orang lain. Beberapacara dapat digunakan, salah satunya dengan cara mengamankan data informasi dengan menggunakan konsep kriptografi berhubungan dengan aspek keamanan informasi, integritas suatu data. Algoritma kripografi yang akan digunakan untuk menyelesaikan masalah pengamanan informasi atau data yaitu dengan menggunakan *metode RSA* dan metode *Algoritma Blowfish*. Hasil *enkripsi* dari kata UPI YPTK didapatlah hasil dengan menggunakan metode *RSA* yaitu didapatlah Hasil Desimal : *98 135 98 0 113 9 9 84 98 34 98 0 98 49 113 135*, dan hasil dari proses *enkripsi* menggunakan *Algoritma Blowfish* yaitu *Ü¬/*œ|9‹*.

Kata Kunci : *Algoritma RSA, Algoritma Blowfish, Keamanan Data*

Abstract− *In the development of information technology that is increasingly high and increasing, as well as sophisticated times like this, tools are needed to send messages, including many media such as chat, line or the like so that we can send messages quickly and vice versa. Of all the conveniences that will certainly be very influential when we will send messages whose contents only certain people who have the right to know the contents. One that must be really vigilant and careful is a message that is confidential because if the message is spread it will have a bad impact on ourselves or others. Beberapacara can be used, one of them by securing information data by using cryptographic concepts related to information security aspects, the integrity of a data. The cryptography algorithm that will be used to solve the problem of securing information or data is by using RSA method and Blowfish algorithm method. Encryption results from the word UPI YPTK obtained results by using the RSA method that is obtained decimal results: 98 135 98 0 113 9 9 84 98 34 98 0 98 49 113 135, and the result of the encryption process using the Blowfish algorithm is Ü|*œ / 9".*

Keywords: *Algoritma RSA, Algoritma Blowfish, Keamanan Data*

1. PENDAHULUAN

Pada perkembangan teknologi informasi yang semakin tinggi dan meningkat, serta zaman yang serba canggih seperti ini dibutuhkan alat untuk mengirim pesan sudah banyak termasuk medianya seperti chatting, line atau sejenisnya sehingga kita bisa mengirim pesan dengan cepat begitu juga sebaliknya. Dari semua kemudahan itu tentu akan sangat berpengaruh ketika kita akan mengirim pesan yang isinya hanya orang-orang tertentu saja yang memiliki hak untuk mengetahui isinya. Salah satu yang harus benar-benar diwaspadai dan hati-hati adalah pesan yang bersifat rahasia karena jika pesan itu tersebar maka akan berdampak buruk pada kita sendiri atau orang lain. Beberapacara dapat digunakan, salah satunya dengan cara mengamankan data informasi dengan menggunakan konsep kriptografi berhubungan dengan aspek keamanan informasi, integritas suatu data. Algoritma kripografi yang akan digunakan untuk menyelesaikan masalah pengamanan informasi atau data yaitu dengan menggunakan *metode RSA* dan metode *Algoritma Blowfish*.

Algoritma RSA merupakan enkripsi yang termasuk jenis asimetris dengan proses enkripsi yang menggunakan sebuah public key dan proses dekripsi yang membutuhkan sebuah private key. Keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima, pemfaktoran dilakukan untuk memperoleh kunci private. Selama pemfaktoran bilangan besar menjadi bilangan prima belum tentu menemukan algoritma yang benar, maka selama itu pula keamanan algoritma RSA terjamin. Sedangkan *Algoritma Blowfish* merupakan metode enkripsi yang mirip dengan DES (*DES like chiper*) dan diciptakan oleh *Bruce Schneier* yang ditujukan untuk mikroprosessor besar (32 bit ke atas dengan *chace* data yang besar). *Algoritma blowfish* terdiri dari dua bagian yaitu *key expansion* dan *enkripsi* data. *Key expansion* berfungsi untuk mengkonversikan sebuah kunci sampai 448 bit ke dalam beberapa array subkey dengan total 4168 byte. *Enkripsi data* terdiri dari sebuah fungsi yang sederhana dengan iterasi 16 kali. Setiap round mempunyai sebuah permutasi *key-dependent* dan sebuah *subsitusi-key* dan data-*dependent*. Semua operasi, penjumlahan dan XOR pada word 32-bit. Hanya operasi tambahan *diindek empat lookup data array per round*.

Submitted: **16/02/2024**; Accepted: **23/02/2024**; Published: **28/02/2024**

Jurnal Ilmu Komputer dan Sistem Informasi (JIKOMSI
Volume 7, Nomor 1, Tahun 2024, Hal 203-207
ISSN 2723- 6129 (media online)
Link Jurnal : https://ejournal.sisfokomtek.org/index.php/jikom

2. METODOLOGI PENELITIAN

2.1 Metode *Algoritma RSA*

RSA adalah salah satu teknik kriptografi dimana kunci untuk melakukan enkripsi berbeda dengan kunci untuk melakukan dekripsi. Pengirim pesan atau penyimpanan data merupakan hal yang harus dijaga keamanannya sehingga perlu diterapkan suatu teknik pengamanan dalam penyimpanannya (Arief et al., 2016)

Algoritma RSA dibuat oleh tiga orang peneliti dari MIT (*Massachussets Institute of Technology*) pada tahun 1976, yaitu Ron Rivest, Adi Shamir dan Leonard Adleman. RSA adalah salah satu teknik kriptografi dimana kunci untuk melakukan enkripsi berbeda dengan kunci untuk melakukan dekripsi (Aria et al., 2018). Kunci untuk melakukan enkripsi disebut sebagai kunci publik, sedangkan kunci untuk melakukan dekripsi disebut sebagai kunci privat. Mereka yang mempunyai kunci publik hanya dapat melakukan enkripsi dan yang dapat melakukan deskripsi hanya mereka yang memiliki kunci private (Jamaludin, 2018).

2.2 Metode *Algoritma Blowfish*

Algoritma Blowfish merupakan metode enkripsi yang mirip dengan DES (*DES like chiper*) dan diciptakan oleh *Bruce Schneier* yang ditujukan untuk *mikroprosessor* besar (32 bit ke atas dengan chace data yang besar). *Blowfish* sendiri meiliki *chiper* blok 64-bit dengan sebuah kunci yang panjangnya variabel. *Algoritma blowfish* terdiri dari dua bagian yaitu *key expansion* dan enkripsi data(Sitinjak et al., 2014). *Key expansion* berfungsi untuk mengkonversikan sebuah kunci sampai 448 bit ke dalam beberapa *array sub key* dengan total 4168 byte. Enkripsi data terdiri dari sebuah fungsi yang sederhana dengan iterasi 16 kali. Setiap round mempunyai sebuah permutasi *key-dependent* dan sebuah *subsitusi-key* dan data-dependent. Semua operasi, penjumlahan dan XOR pada word 32-bit. Hanya operasi tambahan diindek empat lookup data *array* per *round*. *Blowfish* menggunakan sejumlah *subkey* yang besar. Key ini harus dihitung awal sebelum enkripsi atau dekripsi. (Rachman, 2015).

3. HASIL DAN PEMBAHASAN

3.1 Proses *Enkripsi* menggunakan *Algoritma RSA*

Langkah-langkah untuk *enkripsi algoritma RSA*

1. Menentukan 2 buah bilangan prima untuk *p* dan *q*
   p = 11
   q = 13
2. Mendapatkan nilai *n* dimana rumurnya adalah sebagai berkut:
   *n = p * q*
   dan akan menjadi seperti dibawah ini:
   *n = 11 * 13*
   n = 143
3. Mendapatkan nilai *m* dimana rumusnya adalah sebagai berikut:
   *m = (p-1) * (q-1)*
   dan akan menjadi seperti dibawah ini:
   *m = (11-1) * (13-1)*
   *m = (10) * (12)*
   m = 120
4. Menentukan nilai *e* dengan syarat :
   e = e > 1 and GCD (m,e) = 1
   dimana "17" adalah nilai yang memenuhi syarat untuk nilai *e*
   maka didapatlah nilai *e = GCD (120,17) = 1*
5. Menentukan nilai *d* dengan syarat sebagai berikut:
   *d = (d * e) mod m = 1*
   dimana "473" adalah nilai yang memenuhi syarat untuk nilai *d*
   maka didapatlah nilai *d = (473 * 17) mod 120 = 1*
6. Dari proses diatas, maka akan didapatkan kunci public dan kunci privat menggunakan rumus sebagai berikut:
   Public Key = (e,n)
   Private Key = (d,n)
   Dan hasil kunci yang didapat seperti berikut:
   Public Key = (17,143)

Submitted: **16/02/2024**; Accepted: **23/02/2024**; Published: **28/02/2024**

Jurnal Ilmu Komputer dan Sistem Informasi (JIKOMSI
Volume 7, Nomor 1, Tahun 2024, Hal 203-207
ISSN 2723- 6129 (media online)
Link Jurnal : https://ejournal.sisfokomtek.org/index.php/jikom

Private Key = (473,143)

7. Setelah mendapatkan *public key* dan *private key*, selanjutnya melakukan *enkripsi* dan *deskripsi*, yaitu kata "UPI YPTK":
   Dibawah ini merupakan proses penyelesaiannya:
8. Setelah mendapatkan *public key* dan *private key*, selanjutnya melakukan *enkripsi* dan *deskripsi*, yaitu kata "UPI YPTK":

Tabel 1. Proses penyelesaiannya

Text	***ASCII (A)***	***Proses Enkripsi (X)*** ***C = A ^ e mod n***	***Proses Dekripsi (Y)*** ***Y = C ^ d mod n***
U	*85*	*= (8 ^ 17) mod 143 = 98* *= (5 ^ 17) mod 143 = 135* *= 98,135*	*= (98 ^ 473) mod 143 = 7* *= (135 ^ 473) mod 143 = 2* *= 85 → U*
P	*80*	*= (8 ^ 17) mod 143 = 98* *= (0 ^ 17) mod 143 = 0* *= 98,0*	*= (98 ^ 473) mod 143 = 8* *= (0 ^ 473) mod 143 = 0* *= 80 → P*
I	*73*	*= (7 ^ 17) mod 143 = 113* *= (3 ^ 17) mod 143 = 9* *= 113,9*	*= (113 ^ 473) mod 143 = 7* *= (9 ^ 473) mod 143 = 3* *= 73 → I*
<space>	*32*	*= (3 ^ 17) mod 143 = 9* *= (2 ^ 17) mod 143 = 84* *= 9,84*	*= (9 ^ 473) mod 143 = 3* *= (84 ^ 473) mod 143 = 2* *= 32 → <space>*
Y	*89*	*= (8 ^ 17) mod 143 = 98* *= (9 ^ 17) mod 143 = 34* *= 98,34*	*= (98 ^ 473) mod 143 = 8* *= (34 ^ 473) mod 143 = 9* *= 89 → Y*
P	*80*	*= (8 ^ 17) mod 143 = 98* *= (0 ^ 17) mod 143 = 0* *= 98,0*	*= (98 ^ 473) mod 143 = 8* *= (0 ^ 473) mod 143 = 0* *= 80 → P*
T	*84*	*= (8 ^ 17) mod 143 = 98* *= (4 ^ 17) mod 143 = 49* *= 98,49*	*= (98 ^ 473) mod 143 = 8* *= (49 ^ 473) mod 143 = 4* *= 84 → T*
K	*75*	*= (7 ^ 17) mod 143 = 113* *= (5 ^ 17) mod 143 = 135* *= 113,135*	*= (113 ^ 473) mod 143 = 7* *= (135 ^ 473) mod 143 = 5* *= 75 → K*

Dari penyelesaian diatas maka didapatlah
Hasil Desimal : *98 135 98 0 113 9 9 84 98 34 98 0 98 49 113 135*

3.2 Proses *Enkripsi* menggunakan *Algoritma Blowfish*

Dalam hal ini langkah awal yang dilakukan adalah sebagai berikut:
Plaintext = UPI YPTK
Password = 2905

Tabel 2. Konversi *Plaintext* Ke *Binner*

Karakter	ASCII (Hexa)	Biner
U	55	01010101
P	50	01010000
I	49	01001001
<spasce>	20	00100000
Y	59	01011001
P	50	01010000
T	54	01010100
K	4B	01001011

1. Kemudian Plaintext dibagi menjadi 2 bagian XL dan XR menjadi :
   XL = 01010101 01010000 01001001 00100000

Submitted: **16/02/2024**; Accepted: **23/02/2024**; Published: **28/02/2024**

Jurnal Ilmu Komputer dan Sistem Informasi (JIKOMSI
Volume 7, Nomor 1, Tahun 2024, Hal 203-207
ISSN 2723- 6129 (media online)
Link Jurnal : https://ejournal.sisfokomtek.org/index.php/jikom

XR = 01011001 01010000 01010100 01001011

2. Pembangkitan Sub Kunci :

Tabel 3. Kunci : 2905

Karakter	ASCII (Hexa)	Biner
2	32	00110010
0	30	00110000
2	32	00110010
4	34	00110100

Biner : 00110010 00111001 00110000 00110101

a. *SubKunci untuk Iterasi Pertama:*
P_0 = P_0 XOR Kunci
P_0 = 00100100 00111111 01101010 10001000 XOR
00110010 00111001 00110000 00110101
P_0 = 00010110 00000110 01011010 10111111

b. *SubKunci untuk Iterasi Kedua:*
P_1 = P_1 XOR P_0
P_1 = 10000101 10100011 00001000 11010011 XOR
00010110 00000110 01011010 10111111
P_1 = 10010011 10100101 01010010 01101100

3. Langkah selanjutnya yaitu melakukan satu iterasi, dikarenakan total iterasi proses enkripsi adalah 16 putaran.a.Untuk iterasi pertama i= 0 yaitu :

a. Untuk terasi pertama $i = 0$ yaitu:
XL = XL XOR P_0
XL = 01010101 01010000 01001001 00100000 XOR
00010110 00000110 01011010 10111111
XL = 01000011 01010110 00010011 10011111
Fungsi F didapat dari :
XL dibagi menjadi 4 (a, b, c, d) masing-masing 8 bit =
a = 01000011
b = 01010110
c = 00010011
d = 10011111
Fungsi F:
F(XL) = (((S$_0$.a+ S$_1$.b mod 2^{32}) XOR S$_2$,c) + S$_3$,dmod 2^{32})S$_0$.a + S$_1$.b mod 2^{32}
= (11010001 00110001 00001011 10100110 . 01000011) +
(01001011 01111010 01110000 11101001. 01010110) mod 2^{32}
= (11011010111111111010110000011000111001 0 +
11001010110110010000111101110011000110) mod 2^{32}
= 00011010 11110111 11111010 10111000

XOR S2.c = 00011010 11110111 11111010 10111000 XOR (11101001 00111101 01011010 01101000. 11100100)
= 00011010 11110111 11111010 10111000 XOR
11001111101110101010010010000010010100000
= 11001111 10100000 01010011 01111110 00011000 + S3.d mod 2^{32}
= (11001111 10100000 01010011 01111110 00011000 + (00111010 00111001 11001110 00110111. 10011111)) mod 2^{32}
= (11001111 10100000 01010011 01111110 00011000 +
10010000101001111001110001010000101001) mod 2^{32}
= 11001010 00111010 10010010 01000001

F(XL) = 11001010 00111010 10010010 01000001

Submitted: **16/02/2024**; Accepted: **23/02/2024**; Published: **28/02/2024**

XR= F(XL)XOR XR
XR= 11001010 00111010 10010010 01000001 XOR
10010001 01111100 00111001 00111100
XR= 01011011 01000110 10101011 01111101
Menukar Nilai XL dan XR:
XL= XR; XR = XL
XL= 01011011 01000110 10101011 01111101;
XR= 01000011 01010110 00010011 100111117.
Setelah melakukan 16 iterasi, maka akan menghasilkan nilai baru XL dan X Rmasing-masing 32 bit.
Tukar kembali XL dan XR.
Setelah itu XOR-kan nilai XL dan XR: XR= XR XOR P_{16} dan XL= XL XOR P_{17}
Kemudian XLdan XR digabungkan sehingga menjadi 64 bit.

Nilai biner tersebut di konversikan ke dalam kode ASCII sehingga menghasilkan
ciphertext yaitu : Ü/*œ|9‹
Password = 2905

4. KESIMPULAN

Dari analisa yang dilakukan dengan membandingkan *Algoritma RAS* dengan *Algoritma Blowfish* bahwa metode tersebut ternyata dapat mengubah pesan asli menjadi pesan terenkripsi menjadi kode-kode yang tidak dapat dibaca dan mengembalikannya kembali menjadi pesan aslinya tanpa merubah dan merusak pesan. Hasil yang didapat tidaklah sama karena nilai yang diambil pada kunci masing-masing *algoritma* tidak lah sama melainkan secara acak. Akan tetapi langkah yang sangat mudah digunakan saat ini ialah *Algoritma RAS* dibandingkan menggunakan *Algoritma Blowfish.* Hasil *enkripsi* dari kata UPI YPTK didapatlah hasil dengan menggunakan metode *RSA* yaitu didapatlah Hasil Desimal : *98 135 98 0 113 9 9 84 98 34 98 0 98 49 113 135,* dan hasil dari proses *enkripsi* menggunakan *Algoritma Blowfish* yaitu *Ü¬/*œ|9‹*.

REFERENCES

[1] Arief, A., & Saputra, D. R. (2016). Implementasi Kriptografi Kunci Publik dengan Algoritma RSA-CRT pada Aplikasi Instant Messaging. *Scientific Journal of Informatics*, *3*(1). http://journal.unnes.ac.id/nju/index.php/sji
[2] Jamaludin. (2018). Rancang Bangun Kombinasi Hill Cipher dan RSA Menggunakan Metode Hybrid Cryptosystem. *Sinkron*.
[3] Rachman, T. (2015). Sistem Keamanan Data Menggunakan Algoritma Blowfish Dengan Kunci Simetrik. *STT STIKMA Internasional Malang*, *1*(1).
[4] RAHAJOENINGROEM, T., & Aria, M. (2018). Studi Dan Implementasi Algoritma RSA Untuk Pengamanan Data Transkrip Akademik Mahasiswa. *Majalah Ilmiah Unicom*.
[5] Sitinjak, S., Fauziah, Y., & Juwairiah. (2014). Aplikasi Kriptografi File Menggunakan Algoritma Blowfish. *Seminar Informatika*, *1*(1), 78–86.

Submitted: **16/02/2024**; Accepted: **23/02/2024**; Published: **28/02/2024**