Available *online* at : http://bit.ly/InfoTekJar

InfoTekJar : Jurnal Nasional Informatika dan Teknologi Jaringan

ISSN (Print) 2540-7597 | ISSN (Online) 2540-7600

Kombinasi Algoritma Simetri dan ECC untuk Meningkatkan Keamanan Pesan

Adi Widarma

Universitas Asahan

KEYWORDS

Algoritma ECC; Algoritma AES; Keamanan Pesan

CORRESPONDENCE

Phone:

E-mail: adiwidarma10@gmail.com

A B S T R A C T

Pengiriman atau pertukaran data adalah hal yang sering terjadi dalam dunia teknologi informasi. Apalagi pengiriman data dilakukan melalui layanan komunikasi dunia maya, ancaman kejahatan sangat banyak terjadi didalamnya. Untuk menjaga keamanan data, dapat dilakukan dengan menggunakan teknik kriptografi.Banyak algoritma kriptografi yang digunakan untuk melakukan pengamankan pesan, tetapi kekuatan dari keamanan pesan tersebut masih lemah. Peningkatan keamanan pesan dan kunci dilakukan dengan algoritma hybrid. Metode hybrid dilakukan dengan mengkombinasikan algoritma simetri AES dengan OTP dengan algoritma ECC (Elliptic Curve Cryptograpy) yaitu ECDH (Elliptic Curve Diffie-Hellman). Algoritma AES dan OTP digunakan untuk enkripsi dan dekripsi pesan sedangkan ECDH digunakan untuk menentukan dan pertukaran kunci nya sehingga akan semakin meningkatkan keamanan dari pesan.

.

PENDAHULUAN

Dengan semakin maraknya orang menggunakan layanan pengiriman data, ditambah dengan adanya hacker dan cracker, data yang dikirim harus mempunyai keamanan yang kuat. Untuk menjaga keamanan data, dapat dilakukan dengan menggunakan teknik kriptografi.

Kriptografi pada dasarnya terdiri dari dua proses, yaitu proses enkripsi dan proses dekripsi. Proses enkripsi adalah proses penyandian pesan terbuka menjadi pesan rahasia (chiperteks). Proses enkripsi dilakukan menggunakan suatu algoritma dengan beberapa parameter. Kriptografi pada dasarnya terdiri dari dua proses, yaitu proses enkripsi dan proses dekripsi. Proses enkripsi adalah proses penyandian pesan terbuka menjadi pesan rahasia (chiperteks). Proses enkripsi dilakukan menggunakan suatu algoritma dengan beberapa parameter.

Elliptic Curve Cryptografi (ECC) adalah termasuk algoritma asimetri. Beberapa tahun belakangan ini, banyak peneliti dan penulis berpendapat bahwa algoritma ECC merupakan salah satu algoritma yang akan berkembang kedepannya. ECDH (Elliptic Curve Diffie-Hellman) adalah algoritma yang digunakan untuk proses pertukaran kunci.

Untuk memenuhi layanan keamanan kerahasiaan data akan dilakukan kombinasi algoritma simetri *Advanced Encryption Standard* (AES) dan *One Time Pad* (OTP). Penelitian yang dilakukan Shaikh & Kaul (2014) menunjukkan bahwa AES adalah algoritma yang terbaik dari teknologi enkripsi simetri. Secara teoritis, teknik *One Time Pad* merupakan teknik enkripsi yang sempurna asalkan proses pembuatan kunci benar acak (Kromodimoeljo, 2010).

Algoritma ONE TIME-PAD

Algoritma kriptografi One Time Pad (OTP) walaupun sederhana tetapi mempunyai tingkat keamanan yang sangat tinggi. Konsep terpenting dalam algoritma kriptografi OTP adalah meng-xor-kan plainteks dengan key untuk memperoleh cipherteks. Secara sederhana dapat dituliskan dalam persamaan:

$$pXORk = c$$

Persamaan untuk melakukan pendekripsian adalah:

$$cXORk = p \quad (1)$$

Dengan c : cipherteks

p : plainteks

k : kunci rahasia yang digunakan

Operator logika XOR akan menghasilkan T (benar) apabila salah satu dari kedua operand (tetapi tidak keduanya) bernilai T.

ALGORITMA AES (ADVANCED ENCRYPTION STANDARD)

AES ini merupakan algoritma block cipher dengan menggunakan sistem permutasi dan substitusi (P-Box dan S-Box) bukan dengan jaringan Feistel sebagaiman block cipher pada umumnya. Setiap putaran mengunakan kunci internal yang

berbeda (disebut round key). Enciphering melibatkan operasi substitusi dan permutasi. Jenis AES terbagi 3, yaitu :

1. AES-128
2. AES-192
3. AES-256

ALGORITMA ECC (ELLIPTIC CURVE CRYPTOGRAPHY)

Kriptosistem kurva eliptik ini dapat digunakan pada beberapa keperluan seperti :

- Skema enkripsi (ElGamal ECC)
- Tanda tangan digital (ECDSA – Elliptic Curves Digital Signature)
- Protokol pertukaran kunci (Diffie Hellman ECC)

Salah satu sistem kriptografi kunci publik yang aman dan efisien berdasarkan permasalahan matematis, yaitu sistem kripto Kurva Eliptik (Elliptic Curves Cryptosystem). Pada sistem ini digunakan masalah logaritma diskrit kurva eliptik dengan menggunakan grup kurva eliptik. Struktur kurva eliptik digunakan sebagai grup operasi matematis untuk melangsungkan proses enkripsi dan deskripsi. Cara ini menyebabkan kesulitan menghitung k jika diketahui Q dan P dimana Q = k P.

PERTUKARAN KUNCI ECDH (ELLIPTIC CURVE DIFFIE-HELLMAN)

Implementasi sistem kripto kurva eliptik untuk protokol pertukaran kunci dapat dilakukan dengan menggunakan tahapan pada Diffie-Helman dengan mengubah parameter yang disesuaikan dengan persamaan kurva elips.

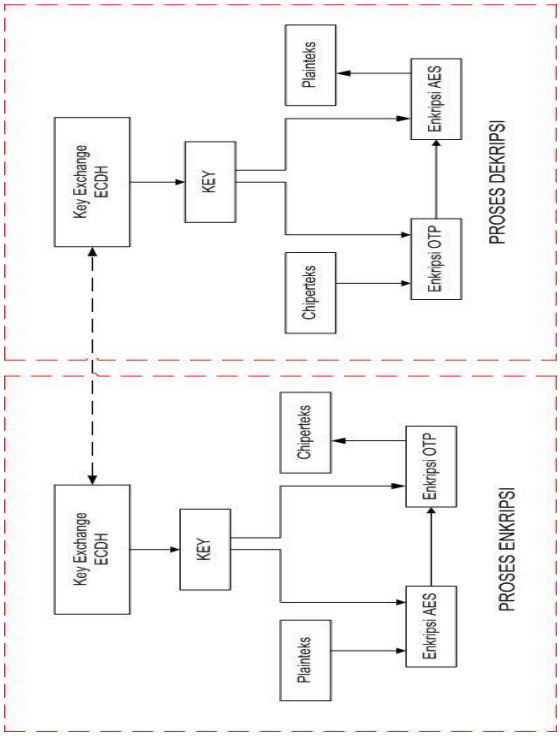
Misal diberikan kurva E : y2= x3+ 2x + 1 dengan bilangan prima yang dipilih p=5, maka himpunan titik-titik pada kurva E(Z5) = {(0,1), (1,3), (3,3), (3,2), (1,2), (0,4)} yang berturut-turut diberi notasi P1, P2, P3, P4, P5, P6

1. Alice memilih kunci rahasia a, misalkan a = 2 dan menghitung kunci publiknya yaitu :PAlice = a P1 = 2 P1 = P1 + P1 = P2 = (1,3)
   Hasil kunci publik ini dikirm ke Bob
2. Bob memilih kunci rahasia b, misalkan b = 3 dan menghitung kunci publiknya yaitu :PBob = a P1 = 3 P1 = P1 + P1 + P1 = P3 = (3,3)
   Hasil kunci publik ini dikirm ke Alice
3. Alice dan Bob secara terpisah menghitung dari kunci publik yang diterima untuk menghasilkan kunci rahasia yang sama, yaitu :
   Alice : SAlice_Bob = a PBob = 2 P3 = P6 = (0,4)
   Bob : SAlice_Bob = a PAlice = 3 P2 = P6 = (0,4)

RANCANGAN METODE

Pada penelitian ini dilakukan kombinasi antara algoritma simetri menggunakan AES dan OTP dengan kunci publik ECDH dalam skema hybrid. Dimana ECDH digunakan untuk proses pertukaran dan penentuan kuncinya. Sedangkan AES dan OTP digunakan untuk proses enkripsi dan dekripsi pesan.

Adapun skema alurnya dapat di lihat pada gambar 1 dibawah ini.



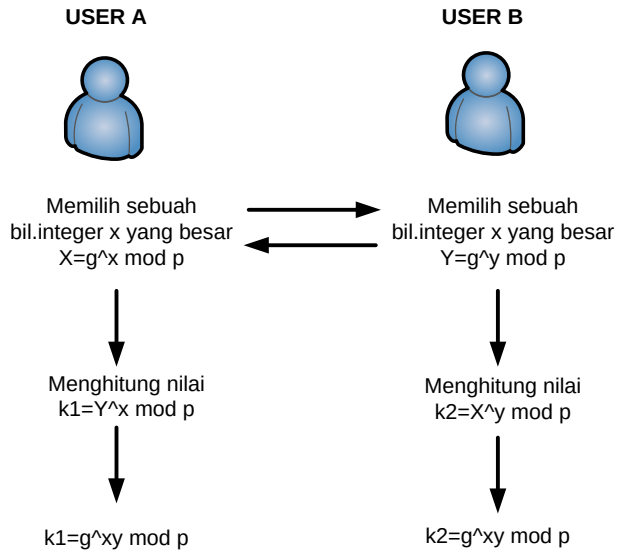
HASIL DAN PEMBAHASAN

Untuk memudahkan dalam hal pembahasan, gambar 1 diatas dapat dibagi menjadi beberapa bagian, diantaranya adalah :

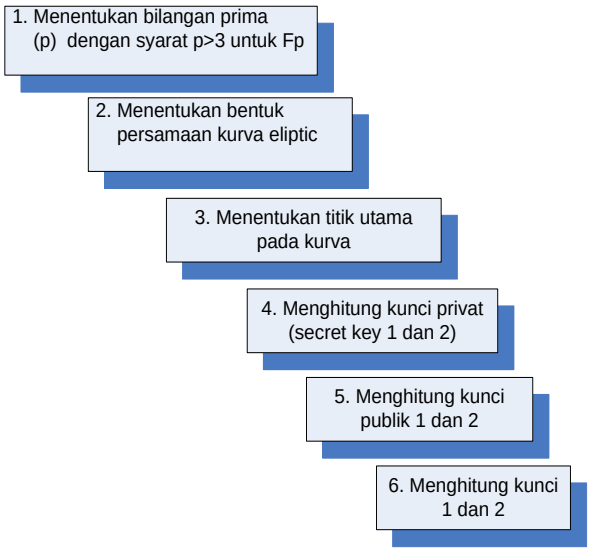
1. Proses pertukaran dan pembentukan kunci dengan ECDH.
2. Proses enkripsi pesan dengan AES.
3. Proses enkripsi pesan hasil enkripsi AES dengan OTP.
4. Proses dekripsi pesan dengan OTP.
5. Proses dekripsi pesan hasil dekripsi OTP dengan AES.

1. Proses pertukaran dan pembentukan kunci dengan ECDH

Implementasi pertukaran kunci dengan sistem ECDH dilakukan dengan menggunakan tahapan pada diffie-hellman dengan mengubah parameter yang disesuaikan dengan persamaan kurva elips. Algoritma pertukaran kunci Diffie-Hellman yang diilustrasikan oleh 2 user seperti gambar 2 berikut ini.



Tahapan untuk menentukan kunci privat (secret key) dalam prosedur pertukaran kunci algoritma ECDH seperti gambar 3 berikut ini.



2. Proses enkripsi pesan dengan AES

Proses enkripsi AES diawali proses AddRoundKey diikuti sembilan round dengan arsitektur yang tersusun atas empat proses dan urutan identik yaitu:

1. SubBytes
2. ShiftRows
3. MixColumns
4. AddRoundKey

Akhir proses enkripsi digunakan round kesepuluh yang tersusun atas tiga proses terurut yaitu: SubBytes, ShiftRows, AddRoundKey. Yang keseluruhan proses tersebut diringi proses penjadwalan key bagi setiap round.

Misalkan plainteknya : JURNAL SENOPATI
Key : 1234

Plainteks di ubah ke dalam bentuk hexadecimal sehingga menjadi chiperteks

J	U	R	N	A	L		S	E	N	O
4a	55	52	4e	41	4c	20	53	45	4e	4f

P	A	T	I
50	41	54	49

Key diubah ke dalam bentuk hexadesimal sehingga menjadi chiperkey

1	2	3	4
31	32	33	34

State 1

4a	41	45	41
55	4c	4e	54
52	20	4f	49
4e	53	50	20

Add Round Key (state 1 XOR chiperkey)

4a = 01001010
31 = 00110001 xor
01111011 = 7b

Dst.....

State 2

7b	70	74	70
64	7d	7f	65
63	11	7e	78
7f	62	61	11

SubBytes (Subtitusi state 2 dengan S-BOX Rijndael)

State 3

21	51	92	51
43	ff	d2	4d
fb	82	f3	bc
d2	aa	ef	82

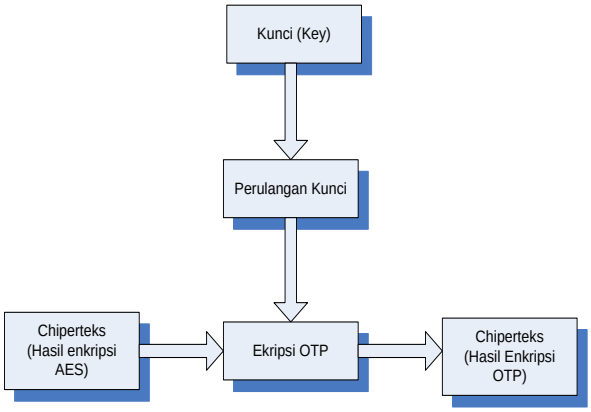
ShiftRow (Pergeseran elemen state 3)

State 4

21	51	92	51
ff	d2	4d	43
f3	bc	fb	82
82	d2	aa	ef

2. Proses enkripsi pesan hasil enkripsi AES dengan OTP

Hasil enkripsi AES berupa chiperteks kemudian dienkripsi kembali dengan menggunakan algoritma OTP. Bagan alur untuk proses ini dijelaskan pada gambar 4 dibawah ini.``

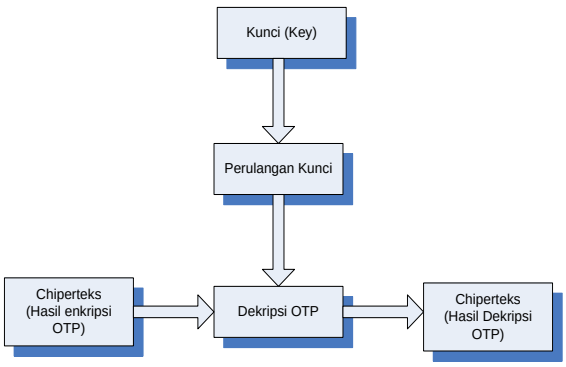


Kunci yang dihasilkan oleh pembangkit bilangan acak tidak akan langsung dipakai sebagai kunci OTP, tetapi akan di lakukan perulangan terlebih dahulu. Pada penelitian ini, perulangan yang digunakan penulis adalah sebagai berikut :

- Perulangan ke 1 akan ditambahkan 1
- Perulangan ke 2 akan ditambahkan 2 dari hasil perulangan ke 1
- Perulangan ke 3 akan di tambahkan 3 dari hasil perulangan ke 2 dan seterusnya sampai panjang kunci sama dengan panjang plainteks.

3. Proses dekripsi pesan dengan OTP

Chiperteks dari hasil enkripsi dengan OTP di dekripsi dengan algoritma OTP oleh si penerima pesan. Bagan alur dapat dilihat pada gambar 5 dibawah ini.



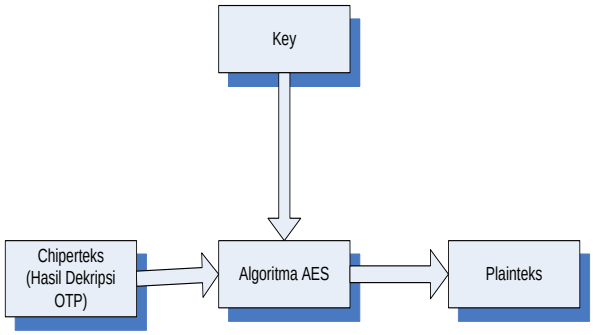
Sama seperti pada proses enkripsi, kunci tidak akan langsung dipakai sebagai kunci untuk dekripsi dengan OTP. Tetapi harus di lakukan perulangan terlebih dahulu, yakni:

- Perulangan ke 1 akan ditambahkan 1
- Perulangan ke 2 akan ditambahkan 2 dari hasil perulangan ke 1

Perulangan ke 3 akan di tambahkan 3 dari hasil perulangan ke 2 dan seterusnya sampai panjang kunci sama dengan panjang plainteks.

4. Proses dekripsi pesan hasil dekripsi OTP dengan AES

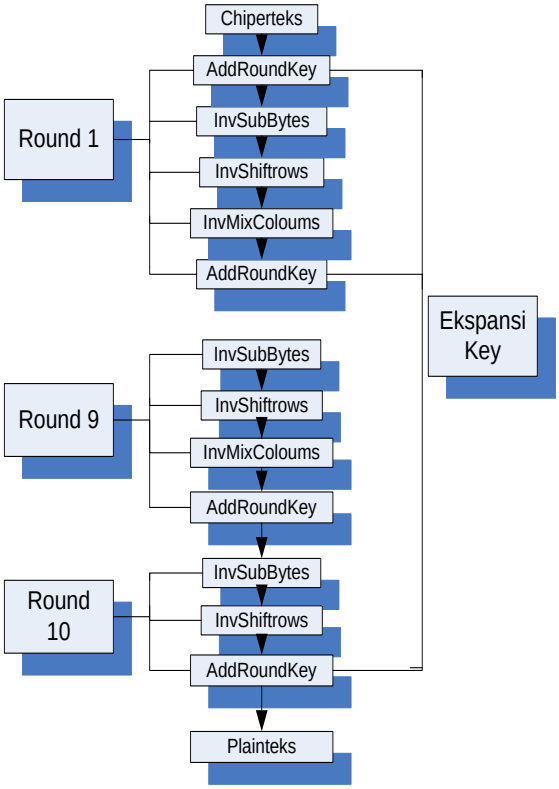
Pada tahapan proses ini hasil dekripsi OTP akan didekripsi kembali dengan algoritma AES yang dapat dilihat pada gambar 6 dibawah ini.



Proses dekripsi dengan algoritma AES atau disebut invers chiper, dimana pada Round terakhir proses MixColoums tidak disertakan. Untuk prosesnya seperti berikut ini:

1. InvShiftRows yaitu dengan menggeser secara cyclic state 1 menjadi state2.
2. InvSubBytes yaitu mensubtitusikan state 2 dalam bentuk hexadesimal kedalam tabel S-Box
3. AddRoundKey yaitu langkah terakhir dengan mengoperasikan XOR antara state 3dan state 4 sehingga menghasilkan state 5 sebagai plainteks

Alur prosesnya dapat dilihat pada gambar dibawah ini 7 berikut ini.



KESIMPULAN

Berdasarkan hasil pembahasan yang telah dilakukan, dapat diambil kesimpulan sebagai berikut:

1. Pembentukan dan pertukaran kunci dengan Algoritma ECDH (Elliptic Curve Diffie-Hellman) membuat kemanan kunci semakin terjaga.
2. Kunci yang dihasilkan dengan algoritma ECDH harus berjumlah matirk 4x4=16 artinya jumlah kata nya 16.
3. Karena kunci untuk proses enkripsi berjumlah 16 maka sulitnya untuk menentukan kunci dengan ECDH.

REFERENSI

[1] Agustanti,S.P. 2010. Pengamanan kunci enkripsi *One-Ttime Pad* (OTP) menggunakan enkripsi RSA, *Jurnal Media Teknik***1**Volume 7: 95-100

[2] Jain, M. &Agrawal, A. 2014. Implementation of hybrid cryptography algorithm. *International Journal Of Core Engineering & Management(IJCEM)***1**(3):126-142..

[3] Kromodimoeljo, S. 2010. *Teori dan aplikasi kriptografi*. SPK IT Consulting

[4] Mollin, R.A. 2002. *RSA and Public Key Cryptography*. CRC Press: Boca Raton.

[5] Mollin, R.A. 2007. *And Introduction to Cryptography. 2nd Edition* .Taylor & Francis Group: Boca Raton.

[6] Munir, R. 2006. *Kriptografi*. Informatika: Bandung

[7] Saikh, A.P & Kaul, P. 2014. Enhanced security algorithm using hybrid encryption and ECC. *IOSR Journal of Computer Engineering (IOSR-JCE)*16(3): 80 – 85