JURNAL INFOTEL
Informatika - Telekomunikasi - Elektronika

Website Jurnal : http://ejournal.st3telkom.ac.id/index.php/infotel
ISSN : 2085-3688; e-ISSN : 2460-0997

Inisialisasi Key Generating Kriptografi AES Pada Pendekatan Protokol SMSSec

Muhammad Barja Sanjaya
Program Studi D3 Manajemen Informatika Fakultas Ilmu Terapan, Universitas Telkom
Jl. Telekomunikasi Terusan Buah Batu Bandung 40257 Indonesia
Email korespondensi : mbarja@tass.telkomuniversity.ac.id

Dikirim 25 November 2016, Direvisi 12 Januari 2017, Diterima 19 Januari 2017

Abstrak – Perkembangan teknologi di dunia internet tidak terlepas dari saling keterkaitannya dengan kriptografi yang menyediakan layanan dalam pengamanan data. Salah satu pengamanan data yang diperlukan yakni penyandian data dengan proses komputasi kriptografi yang digunakan saat berlangsungnya proses yang melibatkan data penting seperti pada transaksi perbankan, yakni pertukaran data rahasia yang dilakukan oleh seseorang untuk memudahkan atau memfasilitasi kegiatannya. Kebutuhan untuk kemudahan dalam akses ke data pribadinya di dunia perbankan seperti transaksi cek saldo atau transfer antar rekening sangatlah krusial dan perlu dijaga kerahasiaannya. Metode kriptografi klasik sudah tidak lagi mumpuni jika diterapkan untuk dijadikan salah satu solusi dalam pengamanan data transaksi tersebut. Oleh karena itu dalam penelitian ini, diusulkan suatu rancangan baru mengenai metode pengamanan data yang dapat digunakan pada aplikasi perbankan dengan biaya komputasi yang rendah. Adapun rancangan baru tersebut melibatkan pendekatan protokol SMSSecure yang di dalamnya menyertakan tiga tipe algoritma kriptografi. Pada penelitian ini juga dilakukan pembaharuan proses kriptografi simetrik *Advanced Encryption Standard* (AES) yakni pada inisialisasi pembuatan kunci enkripsi/dekripsi. Dari hasil simulasi pengujian diperoleh hasil bahwa performansi waktu proses komputasi dari usulan rancangan mencapai empat sampai lima belas kali jauh lebih cepat dibanding rancangan awal. Juga ada penghematan dari hasil analisis parameter konsumsi *memory* yang dibutuhkan selama satu kali transaksi yakni sekitar tiga kilo *bytes*. Namun, parameter *avalanche effect* yang dihasilkan jauh dari kriteria baik dan berada di nilai AE sebesar 78,74%.

Kata kunci – kriptografi AES, konsumsi *memory*, protokol SMSSec, *avalanche effect*.

Abstract - The growth of technology in internet is related to role of the cryptography which provides a service to secure the data exchanged each other. One of the methods for securing data needed is to have confidentiality, integrity and non-repudiation which is able to be conducted using computation of cryptography. It is necessary to conduct the applied cryptography while exchanging secret data take places such as in banking, namely the data exchanging that people do to facilitate the activity. The needs of people to get facilitated while doing the private access into their account in banking such as transaction of checking the balance or to do transferring the number of money to other account have to be done in secured secret way. Meanwhile, the conventional of cryptography has to be improved to implement a secure transaction as one of solution to overcome this problem of data exchanging. Thus, in this research is proposed a new design concerning the method for securing the data which can be applied on banking application with minimum computation cost. As for the proposed design involves an approach of SMSSecure protocol which had been studied in previous research and it included three types of cryptography namely asymmetric, symmetric and one-way function (hash). The symmetric cryptography used in proposed design is AES with modification on initialization of key generating either to encrypt or decrypt. Based on the simulation conducted to test the several data, it is achieved that the performance of processing time on proposed design obtains higher performance as big as four even fifteen times faster than the preliminary design. Besides, there is also a retrenchment on memory consumption needed to conduct computation as big as three kilo bytes in each transaction. However, the value of avalanche effect analysis produced gets reduced in its performance as 78.74%..

Keywords- AES cryptography, memory consumption, SMSSec protocol, avalanche effect.

I. PENDAHULUAN

Dewasa ini perkembangan teknologi makin berkembang, hal ini dibuktikan dengan meningkatnya penggunaan teknologi di berbagai aspek kehidupan. Aspek-aspek ini didukung penuh dengan teknologi. Salah satu teknologi yang berkembang yakni aplikasi pesan singkat yang digunakan sebagai sarana komunikasi antar sistem untuk memfasilitasi kebutuhan suatu instansi terhadap *client*-nya, misal di dunia perbankan. Pesan singkat yang dilakukan antar pihak tersebut memang dilakukan secara langsung, namun terdapat tambahan proses yakni berupa pengamanan data untuk menghindari adanya kebocoran pada informasi yang seharusnya hanya diketahui oleh pihak yang diinginkan [1]. Informasi-informasi yang dilewatkan secara bebas, bisa berakibat dicuri oleh pihak tidak berkepentingan, terlebih informasi tersebut adalah sangat rahasia.

Salah satu teknologi yang menjadi sorotan di zaman sekarang memang sudah dilengkapi dengan fitur keamanan data yakni dengan mengenkripsi pesan yang akan disampaikan ke pihak penerima. Namun, algoritma yang digunakan tersebut sudah tersedia di berbagai media internet [2]. Bahkan dengan mudahnya bisa diperoleh metode-metode pengamanan data serta bisa dipelajari lebih dalam. Hal ini memunculkan sisi kerentanan pada konten informasi yang bertebaran di media. Padahal, pesan informasi tersebut hanya untuk orang atau pihak yang dituju.

Adapun protokol yang mengatur lalu lintas peredaran informasi yang dikomunikasikan yakni seperti protokol SMSSec. Seperti dijelaskan di penelitian [1][3] bahwa terdapat tiga metode pengamanan yang dilakukan saat proses komunikasi data. Algoritma kriptografi asimetrik Rivest Shamir Adleman (RSA) dengan komputasi pembangkit bilangan primanya yang tangguh, komputasi kriptografi simetrik AES dengan kesederhanaan sistem dengan hasil *cipher* yang tangguh, dan kriptografi fungsi satu arah (*hash*) Secure Hash Algorithm (SHA) untuk dilibatkan dipertukaran kunci yang dilakukan [4]. Rancangan yang diusulkan memang termasuk kategori sangat super untuk ditembus oleh pihak *attacker*. Namun, semakin diekspos nya metode-metode tersebut ke media mengakibatkan metode tersebut makin menurun dari sisi ketahanannya [4]. Selain itu, penelitian ini juga harus memperhatikan dari segi kehandalan sistem saat diimplementasikan [5]. Hal terkait implementasi suatu protokol meliputi beberapa parameter yakni *bandwith* dan *delay* yang yang menjadi parameter evaluasi seperti yang dijelaskan di penelitian [6].

Adapun penelitian lebih dalam mengenai perluasan pada komputasi kriptografi AES telah dilakukan, yakni pada [8], dipaparkan bahwa terjadi peningkatan ketahanan yang cukup signifikan jika komputasi AES disandingkan dengan komputasi *Chaotic Function* saat menghasilkan kunci yang digunakan untuk komputasi. Berdasarkan pemaparan tersebut, dilakukan penelitian yang menggabungkan perluasan kriptografi AES dengan metode *Chaotic Function* yang telah dilakukan pada penelitian [9].

II. METODE PENELITIAN

A. *Gambaran Umum Kinerja Sistem*

Berikut gambaran umum yang terjadi pada saat transaksi berlangsung.

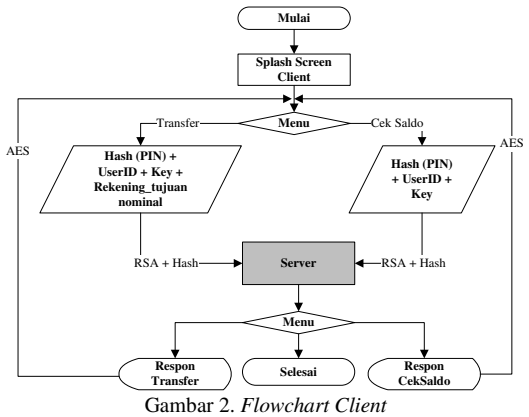
Gambar 1. Gambaran Umum Kinerja Sistem

Uraian proses yang digambarkan pada Gambar 1, sebagai berikut.

a) Client mengirim data berupa transaksi terhadap server. Data tersebut sudah dienkripsi dengan AES dan RSA.
b) Server menerima data cipher dan memproses dekripsi serta memproses query transaksi.
c) Server meneruskan ke database untuk meminta validasi akun dengan cara memproses data akun di database dengan fungsi *hash*. Setelah itu, salah satu hasil dekripsi dari *cipher* yang dikirimkan *client* berupa data akun juga diproses dengan fungsi hash selanjutnya dicocokkan kedua data hasil komputasi *hash* tersebut.
d) Jika kedua data hash pada poin ketiga sama maka *server* akan memproses transaksi, jika tidak sama maka *server* akan mengirim informasi ke *client* bahwa terjadi kesalahan.

B. *Flowchart client*

Berikut adalah gambar *flowchart* pada *client*.



Gambar 2. *Flowchart Client*

Pada Gambar 2 diperlihatkan diagram aliran proses ketika *client* akan melakukan transaksi ke *server*. Misal, *client* C melakukan transaksi cek saldo ke *server* dengan mengirimkan variabel nomor_rekening, PIN, dan *key*. Nilai *key* diperoleh dari hasil operasi komputasi kriptografi *hash* terhadap PIN. Isi *query* transaksi diproses terlebih dahulu yakni menggabungkan Hash (PIN), PIN beserta nomor rekening dienkripsi dengan kritografi asimetrik RSA

(kunci *public*). Setelah terkirim ke *server*, maka *ciphertext* dari *query* didekripsi dengan menggunakan kunci *private* lalu dicek untuk validasi terhadap hasil *hash* PIN di *server* dengan isi *query*. Bila hasil validasi adalah sama maka *query* diproses dan akan direspon "Berhasil" ke *client*. Sebaliknya bila ditemukan ketidaksamaan pada pencocokan hasil *hash* PIN maka *query* tidak akan diproses dan dikirim respon pesan "Gagal" ke *client*.

C. *Entity Relational Diagram (ER-D) yang Akan Dibangun*

Tabel 1 adalah struktur database sederhana yang diimplementasikan untuk mendukung simulasi dan pengujian

Tabel 1. Struktur ER-D

Account
userID_rekening
account_name
tabungan
pin

III. HASIL PENELITIAN

Adapun pada bagian ini dipaparkan mengenai implementasi pengujian dan analisis pada sistem yang telah dibangun.

A. *Response time*

Tabel 2 berikut menyajikan waktu proses komputasi yang dihasilkan dari percobaan yang diujikan terhadap sistem.

Tabel 2. Respon *Time Result*

No. Percobaan	***Response Time* (*second*)**	
	Transaksi Cek Saldo	**Transaksi Transfer**
1	4.210,97	4.822,15
2	7.120,99	3.510,30
3	2.946,04	5.685,30
4	3.734,14	3.491,91
5	5.269,54	4.046,35
6	5.876,83	3.828,06
7	6.842,38	4.065,69
8	2.920,99	5.593,19
9	6.733,93	3.067,12
10	6.246,45	6.278,29
11	6.935,85	3.550,35

B. *Resource memory*

Tabel 3 berikut menyajikan pemakaian atau konsumsi *memory* yang diperlukan saat melangsungkan proses komputasi.

Tabel 3. *Resource Memory Consumption*

No. Percobaan	***Resource Memory* (*Bytes*)**	
	Transaksi Cek Saldo	**Transaksi Transfer**
1	3.216.656	3.256.632
2	3.252.048	3.093.416
3	3.281.944	3.341.696
4	3.094.384	4.296.240
5	3.095.152	4.062.832
6	3.102.648	3.017.488
7	3.008.616	3.178.192
8	3.145.624	3.263.808
9	4.223.776	3.520.456
10	4.391.322	3.036.784
11	4.229.216	3.385.480

C. *Panjang Data Ciphertext*

Untuk hasil panjang data *output* (*length of ciphertext*) dari pengujian yang telah dilakukan, diperoleh sebagai berikut.

a) Hasil skenario pengujian dengan kriptografi RSA_OAEP (*the Optimal Asymmetric Encryption Padding*)

Pada Tabel 4 disajikan hasil pengujian yang telah dilakukan untuk skenario (a).

Tabel 4. Panjang Data *Ciphertext* Dengan Kriptografi RSA_OAEP

Komputasi Kriptografi RSA_OEAP		
No. Percobaan	**Panjang Data Cipher Hasil Transaksi (digit)**	
	Cek Saldo	**Transfer**
1	616	616
2	617	617
3	616	616
4	616	616
5	617	615
6	617	616
7	616	617
8	616	616
9	616	616
10	616	617
11	617	616

b) Hasil skenario pengujian dengan kriptografi AES

Pada Tabel 5 disajikan hasil pengujian yang telah dilakukan untuk skenario (b).

Tabel 5. Panjang Data *Digest* Dengan Kriptografi SHA_256

Komputasi Kriptografi SHA_256		
No. Percobaan	**Panjang Data Digest Hasil Transaksi (hexa)**	
	Cek Saldo	**Transfer**
1	10	10
2	10	10
3	10	10
4	10	10
5	10	10
6	10	10
7	10	10
8	10	10
9	10	10
10	10	10
11	10	10

c) Hasil Skenario Pengujian Dengan SHA_256

Pada Tabel 6 ditunjukkan hasil pengujian yang telah dilakukan untuk skenario (c).

Tabel 6. Panjang Data *Ciphertext* Dengan Kriptografi AES

Komputasi Kriptografi AES_256		
No. Percobaan	**Panjang Data (digit) Cipher Hasil Transaksi (hexa)**	
	Cek Saldo	**Transfer**
1	16	16
2	16	16
3	16	16
4	16	16
5	16	16
6	16	16
7	16	16
8	16	16
9	16	16
10	16	16
11	16	16

D. *Avalenche effect (AE)*

Hasil pengujian pada parameter AE dilakukan dengan dua kasus, yakni.

a) Perubahan satu *bit* pada data *plaintext*.

Perhatikan contoh pengujian berikut.

Tabel 7. Perubahan Satu *Bit* Data Plainteks

Data	Pesan	Pesan (dalam hexa)
Plaint ext 1	"aaaaaaaaaaaaaaa**b**"	6161616161616161616161616161616162
Plaint ext 2	"aaaaaaaaaaaaaaa**c**"	6161616161616161616161616161616163
Key	"00000000000000000"	00000000000000000000000000000000

b) Perubahan Satu *Bit* Pada *Key*

Perhatikan contoh pengujian berikut.

Tabel 8. Perubahan Satu *Bit* Kunci

Data	Pesan	Pesan (dalam hexa)
Plaint ext	"00000000000000000"	00000000000000000000000000000000
***Key* 1**	"aaaaaaaaaaaaaaa**b**"	6161616161616161616161616161616162
***Key* 2**	"aaaaaaaaaaaaaaa**c**"	6161616161616161616161616161616163

IV. PEMBAHASAN

Pada pembahasan analisis akan dilaporkan hasil data setelah dilakukan uji coba aplikasi simulasi dengan jenis masukan tertentu. Adapun yang akan dilakukan yakni menganalisis *response time*, *resource memory* yang terpakai, serta panjang data *output* hasil enkripsi dengan RSA dan AES serta ketahanan terhadap serangan *Brute Force*. Berikut pemaparan mengenai parameter–parameter yang dianalisis, diantaranya sebagai berikut.

A. *Respon Time*

Berdasarkan Tabel 2 terlihat bahwa waktu rata-rata untuk melakukan transaksi cek saldo adalah hampir lima detik yakni tepatnya 4,8652 detik, sedangkan untuk melakukan transaksi *transfer* adalah 4,3803 detik atau sekira 4,5 detik lebih cepat dibandingkan dengan waktu proses transaksi cek saldo. Adanya tambahan waktu tersebut dikarenakan pada simulasi pengujian yang dilakukan, skenario pada cek saldo disertakan pencocokan nilai *hash*. Hal tersebut dilakukan untuk menghitung waktu proses pencocokan nilai *hash* yang dihasilkan yakni sekitar lima detik. Data *respon time* pada simulasi diperoleh hasil yang lebih singkat dibanding dengan *respon time* dari sistem SMSSec yang disebutkan di [1] yang berkisar antara 20 detik < waktu (detik) < 60 detik setiap satu kali transaksi. Hal ini dikarenakan pada [1] menggunakan *handshake* yang lebih dari satu kali, yakni *handshake* pertama sampai ke-*n* dan disetiap *handshake* dilakukan komputasi ketiga kriptografi sehingga total waktu respon yang dibutuhkan pun bertambah.

Berbeda halnya dengan usulan rancangan di penelitian ini yang menerapkan pendekatan SMSSec yakni hanya satu kali *handshake* namun, pada kriptografi simetrik AES dilakukan penambahan proses komputasi yakni pada saat menghasilkan *key* yang melibatkan *Chaotic Function* untuk mempercepat komputasinya [9].

B. *Resource Memory*

Berdasarkan Tabel 3 terlihat bahwa pemakaian *memory* pada proses transaksi yang dilakukan sekitar 3.149.634 *bytes* dari total *memory* yang tersedia sebesar 4GB atau sekitar 0,769% sedangkan untuk transaksi *transfer* dibutuhkan *memory* sebesar 3.438.788 *bytes* atau sekitar 0,840% dari total *memory*. Selisih konsumsi *memory* yang dibutuhkan pada saat

proses komputasi berlangsung yakni 0,0075%. Jumlah *memory* yang dibutuhkan pada hasil pengujian ini menandakan suatu ukuran yang relatif sangat minimum.

C. *Panjang Data Ciphertext*

Pada parameter panjang data *ciphertext*, terdiri dari tiga sub bagian dengan masing-masing proses kriptografi yang berbeda di sistem, yakni.

a) Panjang data *ciphertext* pada proses kriptografi RSA_OAEP

Berdarkan data hasil pada Tabel 4, panjang data *cipher* yang dihasilkan oleh proses enkripsi dengan RSA_OAEP pada transaksi cek saldo atau pun transfer sebesar 615-617 digit.

b) Panjang data *digest* pada proses kriptografi SHA

Dari Tabel 5 diperlihatkan bahwa *message digest* yang dihasilkan dari proses SHA_256 yakni sebesar 10 karakter hexa-desimal.

c) Panjang data *ciphertext* pada proses kriptografi AES

Sedangkan dari Tabel 6 ditunjukkan bahwa cipher yang dihasilkan pada tiap proses yakni berjumlah 16 digit.

Berdasarkan Tabel 4, 5 dan 6 terkait parameter panjang data *ciphertext* setelah dilakukan pengujian, maka jumlah total panjang data tersebut masih termasuk kategori yang layak untuk dikomunikasikan melalui aplikasi berbasi *mobile* atau media internet lainnya. Dari beberapa literatur disebutkan bahwa satu pesan yang disampaikan itu berkisar 160 *bytes* dapat dikirimkan ke jaringan *mobile*.

D. *Avalenche Effect (AE)*

a) Perubahan satu *bit* pada *plaintext*

Pada Tabel 9 terlihat perbedaan *bit* yang dihasilkan yaitu satu *bit* dari total 128 *bit* yang dihasilkan. Dengan kata lain, nilai *avalenche effect* yang diperoleh dari fungsi *jumlahbitberubah* dibagi dengan *jumlahbittotal* pada kasus pertama sebesar 78,74%. Nilai AE pada kasus ini sangat jauh dari kriteria AE yang baik yang berkisar dalam rentang nilai 45-60% [9].

Tabel 9. *Ciphertext* Pengujian AE

Data	Hasil	Hasil (dalam *binary*)
Ciphertext 1	"e91ec7e1a21a b546 559e831590073 fd8"	11101001 0001110 11000111 11100001 10100010 00011010 10110101 01000110 10101011 00111101 00000110 00101011 00100000 00001110 01111111 10110000
Ciphertext 2	"e91ec7e1a21a b546 559e831590073 fd9"	11101001 0001110 11000111 11100001 10100010 00011010 10110101 01000110 10101011 00111101 00000110 00101011 00100000 00001110 01111111 10110001

b) Perubahan satu *bit* pada *key*

Pada Tabel 10 terlihat data *ciphertext* 1 dan 2 menunjukkan perbedaan perubahan *bit* yang dihasilkan adalah 71 *bit* dari total 128 *bit* yang dihasilkan. Hal ini diartikan nilai AE yang diperoleh dari uji coba pada kasus kedua sebesar 55,9055 yang termasuk kategori kriteria AE dengan performansi yang sangat baik.

Tabel 10. *Ciphertext* Hasil Pengujian

Data	Hasil	Hasil (dalam *binary*)
Ciphertext 1	"d2cad19c1b99 89fb 76ea588fee55e3 f1"	11101001 0001110 11000111 11100001 10100010 00011010 10110101 01000110 10101011 00111101 00000110 00101011 00100000 00001110 01111111 10110000
Ciphertext 2	"737ebc5cee56 b94 6a7885b408f2e 026"	11101001 0001110 11000111 11100001 10100010 00011010 10110101 01000110 10101011 00111101 00000110 00101011 00100000 00001110 01111111 10110001

Dari pengujian pada kasus pertama dan kedua didapat indikasi bahwa *key* memiliki peran yang sangat penting. Oleh karena itu, di dalam pendistribusian *key* terutama pada kriptografi kunci *private* (simetrik), *key* yang akan disebarkan mesti dilakukan penambahan proses pendukung untuk pengamanannya [10][11].

V. PENUTUP

A. *Kesimpulan*

Adapun kesimpulan yang diperoleh setelah tahapan implementasi, pengujian dan analisis pada hasil adalah sebagai berikut. Waktu yang dibutuhkan selama proses transaksi dari pertama kali SMS terkirim sampai SMS respon, diperoleh rata-rata waktu komputasi sebesar 4,52 detik. Waktu rata-rata proses komputasi pada sistem yang diusulkan lebih cepat dibandingkan dengan sistem pada SMSSec yang berkisar antara 20 detik < waktu proses (detik) < 60 detik. Rata-rata konsumsi *memory* yang dibutuhkan untuk pemrosesan komputasi pada sistem yang diusulkan sebesar 3.294.211 *bytes*. Nilai parameter *avalenche effect* (AE) yang dilakukan pada kriptografi simetrik AES dengan inisialisasi pada *key generating*

sebesar 78,74% dan jauh termasuk nilai AE dengan kriteria baik.

B. Saran

Sedangkan saran yang dapat dilakukan pada penelitian selanjutnya, yakni pendekatan protokol SMSSec sebaiknya juga diimplementasikan pada algoritma kriptografi asimetrik dan simetrik lainnya. Semua *log* aktivitas yang telah diproses dilakukan pengamanan data dengan metode berbeda namun memberikan waktu yang lebih cepat lagi sebelum disimpan di database. Dimungkinkan dilakukan *watermarking* atau steganografi pada media lain misal audio atau image sebagai media untuk penyisipan data akun *client*.

DAFTAR PUSTAKA

[1] Li-Chang Lo, Johnny. Bishop, Judith. Eloff, J.H.P. "*SMSSec: an end-to-end protocol for secure SMS*". Computer Science Department. University of Pretoria. South Africa. 2010.

[2] Shah, Jolly. Saxena, Vikas. "*Performance Study on Image Encryption Schemes*". IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 4, No. 1. July 2011.

[3] Soleymani, Ali. Md Ali, Zulkarnain. Nordin, Md Jan. "*A Survey on Principal Aspects of Secure Image Transmission*". World academy of Science, Engineering and Technology 66 2012.

[4] Niels Ferguson and Bruce Schneier 2003. "*Practical Cryptography*". Indianapolis, Indiana: Wiley Publishing, 75-82, 89-91, 233, 350 – 352.

[5] Meyer, Carl H., Matyas Stephen M. 1982. "*Cryptography: A New Dimension in Computer Data Security*". New York: John Wiley & Sons.

[6] Denatama, M. I., Perdana, D., Negara, R. M. " Analisis Perbandingan Kinerja Protokol Routing DSDV dan OLSR Untuk Perubahan Kecepatan Mobilitas pada Standar IEEE 802.11ah". Jurnal Infotel Vol. 8 No.2 November 2016. ISSN : 2085-3688; e-ISSN : 2460-0997.

[7] Wahyuningrum, T. "Implementasi XML Encryption (XML Enc) Menggunakan Java". Jurnal Infotel Volume 4 Nomor 1 Mei 2012. ISSN : 2085-3688; e-ISSN : 2460-0997.

[8] Barja Sanjaya, Muhammad. Adolf Telnoni, Patrick. "Implementasi Blum-Blum-Shub dan Chaotic Function Untuk Modifikasi Key Generating pada AES". Jurnal Elektro Telekomunikasi Terapan, Vol. 2, No. 2. ISSN (p): 2407-1320. ISSN (e): 2442-4400. Desember 2015.

[9] Arya, I Putu Dharmadi. Ari M, Barmawi. Gandeva BS. "Enkripsi Gambar Parsial Dengan Kombinasi Metode Stream Cipher RC4 dan Chaotic Function". Fakultas Informartika, Institut Teknologi Telkom, Bandung. 2013.

[10] Jawanjal, Amol. Dagade, Rahul. "*A Secure Protocol for End to End Security to SMS Banking*". International Research Journal of Engineering and Technology (IRJET). Vol. 03 Issue 01, e-ISSN: 2395-0056, p-ISSN: 2395-0072. Jan 2016.

[11] Scheneir, Bruce., dkk. 1998. "*Related Key Cryptanalysis of 3-WAY, Biham-DES, CAST, DES-X, NewDes, RC2, and TEA*". Paper.