Jurnal Teknik Elektro dan Komputer vol. 19 no. 3 July-September 2024, pp. 193-200
p-ISSN : 2301-8402, e-ISSN : 2685-368X , available at : https://ejournal.unsrat.ac.id/index.php/elekdankom

Security Enhancements Of Authentication Data On E-Voting By Using Luc Algorithm

Peningkatan Keamanan Autentikasi Data Pada E-Voting Dengan Menggunakan Algoritma Luc

Bernad Jumadi Dehotman Sitompul, Nancy Jeane Tuturoong, AdeYusupa
Dept. of Electrical Engineering, Sam Ratulangi University Manado, Kampus Bahu St., 95115, Indonesia
e-mails: bernadjdsitompul@unsrat.ac.id, rendysyahputra@unsrat.ac.id, nancy.tuturoong@unsrat.ac.id

Received: 03 March 2024; revised: 29 March 2024; accepted: 31 March 2024

Abstract* — *Electronic Voting* or what is abbreviated as E-voting is a voting system in an online election process. An e-voting system is very necessary to make it easier to carry out online elections such as the election of student council chairman, sub-district head election, student president election, etc. The problems that will be studied in the e-voting system are how to create a secure web-based e-voting, as well as how to control e-voting system to prevent double voting and fraud by administrators or unauthorized parties. Usually the person voting must enter a password first so that other people cannot vote using that person's rights. By applying Luc algorithm to e-voting, it is hoped that the authentication data stored in the encrypted database cannot be read directly by system administrators or unauthorized parties. Based on testing with Avalanche Effect, text lengths ranging from 8 to 16 have an Avalanche Effect value above 50%. Meanwhile, text lengths from 17 to 35 have an Avalanche Effect below 25%. So it can be concluded that the use of the Luc algorithm can be applied to text lengths of around 8 to 16 with a better level of security based on the Avalanche Effect value obtained.

***Key words* — algorithm, authentication *e-voting*, cryptografi, luc.**

Abstrak* — *Electronic Voting* atau yang disingkat dengan *E-voting* adalah sebuah sistem pemungutan suara dalam proses sebuah pemilihan yang dilakukan secara *online*. Sistem e-*voting* sangat diperlukan untuk mempermudah dalam melakukan pemilihan secara *online* seperti pemilihan ketua osis, pemilihan camat, pemilihan presiden mahasiswa dan lain sebagainya. Permasalahan yang akan diteliti dalam sistem *e-voting* adalah bagaimana cara membuat sistem *e-voting* berbasis web yang aman, serta bagaimana cara mengontrol sistem *e-voting* agar tidak terjadi pemberian suara ganda dan kecurangan yang dilakukan oleh administrator atau pihak yang tidak berwenang. Biasanya orang yang memberikan suara harus memasukan password terlebih dahulu agar orang lain tidak bisa memberikan suara memakai hak orang tersebut. Dengan menerapkan algoritma Luc pada *e-voting*, diharapkan data autentikasi yang tersimpan ke dalam basis data yang sudah terenkripsi tidak dapat dibaca langsung oleh administrator sistem maupun pihak yang tidak berwenang. Berdasarkan pengujian dengan *Avalanche Effect*, panjang teks dengan rentang 8 sampai dengan 16 memiliki nilai *Avalanche Effect* diatas 50%. Sementara untuk panjang teks 17 sampai dengan 35 memiliki *Avalanche Effect* dibawah 25 %. Sehingga dapat disimpulkan bahwa penggunaan algoritma Luc dapat diterpakan untuk panjang teks sekitar 8 sampai dengan 16 dengan Tingkat keamanan yang lebih baik berdasarkan nilai *Avalanche Effect* yang diperoleh.

***Kata kunci* — algoritma, autentikasi, *e-voting*, kriptografi, luc.**

I. PENDAHULUAN

Electronic voting atau yang disingkat *e-voting* merupakan sebuah sistem pemungutan suara pada suatu pemilihan yang dilakukan secara *online*. *E-voting* adalah proses pemilihan umum yang memanfaatkan penggunaaan teknologi informasi dimana seluruh atau sebagian proses kegiatannya, dimulai dari pendaftaran pemilih, pemungutan suara, sampai penghitungan suara, dilaksanakan secara elektronik [1]. Di Indonesia, penggunaan *e-voting* sebagai sarana pemungutan suara secara elektronik sudah dilakukan pada tahun 2009. Daerah pertama di Indonesia yang sudah menerapkan sistem *e-voting* adalah Kabupaten Jembrana, Provinsi Bali pada pemilihan kepala dusun [2]. Dengan adanya sistem *e-voting* dalam pemilihan umum tentunya dapat mengurangi biaya yang dikeluarkan dan waktu yang selama ini menjadi kendala atau masalah yang terjadi pada kegiatan pemilihan konvensional. Namun sebenarnya tujuan penggunaan *e-voting* tidak hanya untuk mempercepat proses kegiatan pemilihan umum, namun juga untuk menjaga keabsahan dari suara pemilih, kerahasiaan pemilih, dan keakuratan dari penghitungan suara[3]. Aspek yang paling diutamakan dalam *e-voting* adalah keamanan dan integritas data dari setiap orang peserta pemungutan suara. Keamanan tidak hanya untuk menjaga kerahasiaan hak pilih yang dilakukan pemilih namun juga memperhatikan data yang lainnya [4]. Setiap pemilih yang menggunakan hak pilih tentunya tidak ingin hak pilihnya diketahui oleh orang lain. Oleh karena itu kerahasiaan dari *e-voting* ini tentunya menjadi hal yang sangat penting dan harus tetap terjaga dari pihak yang tidak berwenang [5].

Dalam sistem *e-voting*, setiap pemilih memiliki data autentikasi masing-masing yang berfungsi sebagai kunci untuk masuk kedalam sistem *e-voting* [6]. Pada sistem komputer, autentikasi biasanya terjadi pada saat *login* atau permintaan akses. Autentikasi adalah suatu metode untuk menentukan atau memastikan bahwa seseorang atau entitas tersebut adalah asli atau benar [7]. Adapun proses validasi pengguna pada saat memasuki sistem yaitu nama dan kata sandi (*password*) dari pengguna melalui proses pengecekan user pada suatu *database*

yang diregistrasi sebelumnya oleh user itu sendiri. Kata sandi merupakan sebuah metode autentikasi yang paling umum, yaitu berupa rangkaian huruf, angka, atau karakter khusus [8].

Untuk melindungi akun, maka pengguna perlu membuat kata sandi yang kuat yang menyertakan kombinasi dari semua opsi yang memungkinkan. Namun sayangnya, kata sandi ini rentan terhadap serangan *phishing* dan serangan lainnya yang dapat melemahkan keefektifannya. Sebagian besar pengguna memiliki banyak akun online yang berbeda, namun hanya segelintir saja yang menggunakan kata sandi berbeda di seluruh akun mereka. Akibatnya, banyak pengguna yang memilih kenyamanan daripada keamanan. Karena hanya perlu mengingat satu kata sandi untuk semua akun daripada mengingat banyak kata sandi. Kebanyakan pengguna sistem juga menggunakan kata sandi yang sederhana daripada membuat kata sandi yang kuat karena lebih mudah diingat. Dari beberapa hal tersebut dapat dikatakan kata sandi sebenarnya memiliki banyak kelemahan dan tidak cukup untuk melindungi informasi terutama dalam jaringan. Peretas dapat dengan mudah menebak kredensial pengguna dengan menelusuri semua kemungkinan kombinasi sampai mereka menemukan kecocokan. Untuk itu, diperlukan teknik atau pendekatan untuk mengamankan kata sandi yang digunakan oleh pengguna akun [9].

Ada beberapa teknik yang digunakan untuk mengamankan data salah satu diantaranya adalah kriptografi. Kriptografi merupakan sebuah teknik, pengetahuan ataupun seni yang mempelajari sebuah cara mengamankan pesan atau kata oleh pemberi pesan kepada penerima pesan secara rahasia dengan berbagai cara agar orang yang tidak berkepentingan tidak dapat melihat atau membaca isi dari pesan tersebut [10]., pesan yang sudah disandi (*ciphertext*) adalah pesan yang Kriptografi terbagi menjadi beberapa bagian yaitu pesan awal (*plaintext*) adalah pesan yang semua orang dapat membacanya, kunci adalah sebuah infromasi atau tanda untuk mengubah sebuah pesan menjadi bentuk lain agar tidak mudah dibaca oleh seseorang yang tidak berhak membukanyatelah diubah mengunakan kunci dalam mengubah atau menyamarkan *plaintext* ke *ciphertext* disebut enkripsi dan mengembalikan ke pesan semula disebut dekripsi. Ada banyak teknik kriptografi yang digunakan untuk mengamankan suatu pesan. Teknik tersebut dibagi menjadi 3 golongan besar, yaitu teknik enkripsi kunci simetri, teknik enkripsi kunci asimetri dan fungsi hash [11].

Algoritma Luc merupakan varian dari algoritma kriptografi asimetris. Algoritma asimetris menggunakan kunci yang berbeda untuk proses enkripsi dan dekripsi, sehingga menambah tingkat keamanannya. Algoritma Luc menggunakan dua bilangan prima untuk membangkitkan kunci publik dan kunci rahasia. Algoritma Luc ini juga memiliki banyak kesamaan dengan algoritma Rivest Shamir Adleman (RSA) namun pada algoritma Luc fungsi pangkat pada algoritma RSA digantikan dengan fungsi Lucas [12]. Beberapa hasil penelitian terdahulu sudah pernah dilakukan dan dijadikan referensi pada penelitian ini. Pada penelitian yang dilakukan yaitu *hybrid* algoritma RC4 dan algoritma Luc. Dari hasil pengujian yang dilakukan diperoleh oleh [13] waktu rata-rata proses enkripsi kedua algoritma 287,06 ms dan 74,86. Sementara waktu rata-rata proses dekripsi keduanya 53,43 ms dan 94,26 ms. Kemudian penelitian yang dilakukan oleh [14] yaitu studi perbandingan antara tiga algoritma kriptografi yaitu ElGamal, RSA, dan Luc. Dari hasil pengujian. algoritma RSA menjadi yang tercepat dalam proses enkripsi dengan waktu 0,19 ms. Sementara algoritma Luc menjadi yang tercepat dalam proses dekripsi dengan waktu 0,31 ms. Penelitian berikutnya, dilakukan oleh [15]yaitu dengan mengkombinasikan algoritma Advanced Encryption Standard (AES) dan Luc. Dimana hasil yang diperoleh menunjukan waktu rata-rata proses enkripsi 12,91 ms dan waktu rata-rata proses dekripsi 0,066 ms. Berikutnya penelitian yang dilakukan oleh [16], yaitu melakukan perbandingan ElGamal dengan algoritma Luc. Berdasarkan hasil pengujian System Usability Scale (SUS), dengan nilai rata-rata interpretasi 83.75%. Dari hasil pengujian diketahui bahwa algoritma Luc sangat unggul dibandingkan algoritma ElGamal dalam kecepatan proses enkripsi. Sementara untuk proses dekripsi ElGamal lebih baik dalam proses dekripsi

II. METODE

A. Pengembangan Sistem

Rapid Application Development (RAD) adalah strategi siklus hidup yang ditujukan untuk menyediakan pengembangan yang jauh lebih cepat dan mendapatkan hasil dengan kualitas yang lebih baik dibandingkan dengan hasil yang dicapai melalui siklus tradisional [17]. Pada Gambar 1, dapat dilihat siklus dari model *Rapid Application Development* (RAD). Tahapan *Rapid Application Development* (RAD) adalah sebagai berikut:

1. Fase Persyaratan Proyek

Pada fase ini analisis kebutuhan untuk pengidentifikasian tujuan aplikasi atau sistem serta untuk mengidetifikasi syarat-syarat informasi yang ditimbulkan dari tujuan-tujuan tersebut.

2. Fase *Prototype*

Pada tahap ini digambarkan bagaimana rancangan antar muka dan proses alur pencarian data dan penyimpanan data serta semua yang terkait dengan gambaran sistem yang akan dirancang.

3. Fase *Construction* dan *Feedback*

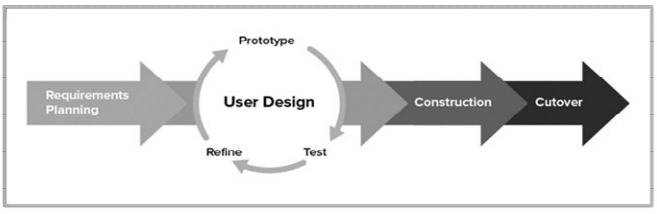
Pada tahapan ini dilakukan pengkodean terhadap rancangan-rancangan yang telah didefinisikan ke dalam kode program.

4. Fase *Cutover*

Fase finalisasi yang mencakup aspek *interface*, fungsi, estetika dan segala sesuatu yang terkait dengan software atau aplikasi yang dibuat [18].

B. Algoritma Luc

Algoritma Luc merupakan salah satu algoritma kriptografi dengan kunci publik yang dibentuk dari fungsi matematika pada persamaan (1) yaitu Deret Lucas. Untuk melakukan



Gambar. 1 Siklus *Rapid Application Development* [19]

proses enkripsi, teks ataupun karakter harus dikonversikan dalam bentuk kode *American Standard Code for Information Interchange* (ASCII). Hasil enkripsi adalah teks ataupun karakter yang telah disamarkan dari pihak yang memiliki akses atas informasi didalamnya [20]. Selanjutnya untuk membuka teks ataupun karakter yang telah terenkripsi dibutuhkan kunci privat (Private Key). Hasil dekripsi akan menghasilkan teks ataupun karakter yang sama dengan sebelum dienkripsi (teks asli).

$$f_{luc}(P) = Vn\,(P, 1)\ mod\ n \quad (1)$$

Dalam penerapan algoritma Luc terdapat 3 tahap utama yaitu pembangkitan kunci, proses enkripsi dan proses dekripsi.

C. Pembangkitan Kunci

Pilih secara acak dua bilangan prima p dan q, dimana p ≠ q, dengan syarat Faktor Persekutuan Terbesar (FPB) dari p dan q = 1.

Kemudian cari nilai N yaitu dengan mengalikan kedua bilangan prima tersebut, $N = p * q$

Selanjutnya hitung nilai m dengan persamaan *Euler* berikut:

$$m = (p + 1) \times (p - 1) \times (q + 1) \times (q - 1) \quad (3)$$

Pilih bilangan acak yang disimbolkan dengan e, dimana e lebih besar dari 1, dan lebih kecil dari
m (1 <e <m). e merupakan bilangan relatif prima dengan GCD (e, m) = 1

Setelah nilai e diperoleh, maka nilai e menjadi kunci publik.

Hitung nilai R (n):

$$R(n) = LCM\ (p + 1, p - 1, q + 1, q - 1) \quad (4)$$

Kemudian hitung nilai d:

$$e \times d\ Mod\ R(n) = 1 \quad (5)$$

Setelah didapatkan nilai d, maka nilai d tersebut menjadi kunci privat

D. Enkripsi

Proses dekripsi adalah tahap mengembalikan *chiper text* ke bentuk *plain text* (teks asli). Untuk mengenkripsi teks ataupun karakter, persamaan yang digunakan adalah sebagai berikut:

$$V[i] = (M * V[i-1] - V[i-2])\ mod\ n \quad (6)$$

E. Dekripsi

Untuk mengubah *cipher tex*t ke *plain text* (dekripsi), persamaan yang digunakan adalah sebagai berikut:

$$V[i] = (C * V[i-1] - V[i-2])\ mod\ n \quad (7)$$

Dimana C adalah *Cipher text*, V[i] adalah V [e] adalah urutan barisan dari rantai lucas [21].

F. Avalanche Effect (AE)

Pengujian *Avalanche Effect* dilakukan untuk mengetahui seberapa besar perubahan bit ketika karakter *plaintext* dirubah. Salah satu fungsi dari *Avalanche Effect* adalah untuk melihat tingkat keamanan suatu algoritma kriptografi [22]. Pengujian *Avalanche Effect* dianggap baik apabila terjadi perubahan bit yang menunjukkan antara 45-60% (50% adalah hasil yang dianggap baik dalam pengujian). Perubahaan sebesar 50% akan mengakibatkan masalah yang cukup sulit untuk pihak yang tidak berwenang melakukan serangan terhadap data atau informasi pengguna. Semakin besar persentase yang dihasilkan maka semakin bagus juga enkripsi yang dihasilkan dan begitu juga sebaliknya. Perhitungan nilai *Avalanche Effect* dilakukan dengan menggunakan persamaan berikut ini.

$$AE = \frac{Dx}{total}\ x\ 100\% \quad (8)$$

Dimana:

AE : merupakan nilai *Avalanche Effect*,
Dx : merupakan jumlah bit yang berbeda,
total : merupakan total dari keseluruhan bit dalam teks atau karakter yang telah terenkripsi [23].

III. HASIL DAN PEMBAHASAN

A. Penyandian Dengan Algoritma Luc

Penyandian teks atau karakter dengan menggunakan algoritma Luc ada tiga tahapan yaitu pembangkitan kunci publik, proses enkripsi, dan proses dekripsi. Sebagai uji coba penyandian teks atau karakter pada penelitian ini adalah kata "VOTE2024".

B. Hasil Pembangkitan Kunci Publik

Sebelum proses enkripsi, maka dilakukan proses pembangkitan kunci yang nantinya akan digunakan untuk proses enkripsi dan dekripsi. Tahapan dan proses pembangkitan kunci publik dan kunci privat algoritma luc adalah sebagai berikut:

1. Pilih dua bilangan prima *p* dan *q* secara acak dimana $p \neq q$ dan Faktor Persekutuan Terbesar (FPB) bilangan prima *p* dan *q* = 1. Untuk contoh perhitungan pada penelitian ini, dipilih bilangan prima *p* = 43 and *q* = 257. Setelah dipilih bilangan prima *p* dan *q*, maka selanjutnya mencari nilai *n* dengan mengalikan bilangan *p* dan *q*, *n* = *p* * *q*, *n* = 43 * 257, *n* = 11051.
2. Hitung semua faktor prima terhadap N:

$$\Phi(N) = (p + 1) * (p - 1) * (q + 1) * (q - 1)$$
$$= (43 + 1) * (43 - 1) * (257 + 1) * (257 - 1)$$
$$= (44) * (42) * (258) * (256)$$
$$= 122056704$$

3. Berikutnya mencari nilai *e*:

Sebelum mencari nilai *e*, terlebih dahulu dicari bilangan relatif prima dari $(p + 1), (p - 1), (q + 1)$, dan $(q - 1)$.

Relatif prima $(p + 1)$ = Relatif Prima $(43 + 1) = 44\ \{3, 5, 7, 13, \ldots, \ldots, \ldots, 43\}$

Relatif prima $(p - 1)$ = Relatif Prima $(43 - 1) = 42\ \{5, 11, 13, \ldots, \ldots, \ldots, 42\}$

Relatif prima $(q + 1)$ = Relatif Prima $(257 + 1) = 258\ \{5, 7, 11, 13, \ldots, \ldots, \ldots, 257\}$

Relatif prima $(q - 1)$ = Relatif Prima $(257 - 1) = 256\ \{3, 5, 7, 11, 13, \ldots, \ldots, \ldots, 255\}$

Dari hasil pencarian, maka diperoleh bilangan 5 dan 13. Kemudian dipilih salah satu dari kedua bilangan tersebut untuk $FPB\ (e, \Phi(N)) = 1$ dimana $1 < e < \Phi(N)$.

$FPB\ (5, 122056704\ )) = 1$

$FPB\ (13, 122056704\ )) = 1$

Dari hasil tersebut, pada penelitian ini, dipilih bilangan 13 sebagai kunci publik, sehingga e = 13.

C. Hasil Enkripsi

Kemudian berikutnya adalah proses enkripsi yaitu proses pengubahan *plain text* menjadi *chiper text*. Sebelum proses enkripsi, terlebih dahulu *plain text* yaitu karakter "VO" dikonversi kedalam kode ASCII sehingga hasilnya adalah 8679. Proses enkripsi *plain text* = 8679 dengan kunci publik 13 dapat dilihat pada Tabel 1. Dari hasil enkripsi pada Tabel 2., nilai Vn_baru pada iterasi 5 diperoleh 6669 dan jika dikonversi ke dalam karakter ASCII menjadi "BE".

TABEL 1.
TAHAPAN DAN HASIL ENKRIPSI KARAKTER "VO"

x	k[x]	Vn_lama	Vj_lama	Vn_baru	Vj_baru
1	0	8679	2	1423	8679
2	1	1423	8679	8622	1423
3	0	8622	1423	9856	4868
4	0	9856	4868	2444	8989
5	1	2444	8989	**6669**	2444

Kemudian berikutnya adalah proses enkripsi dari *plain text* karakter "TE" menjadi *chiper text*. Sebelum proses enkripsi, terlebih dahulu *plain text* dikonversi kedalam kode ASCII sehingga hasilnya adalah 8469. Proses enkripsi *plain text* = 8469 dengan kunci publik 13 dapat dilihat pada Tabel 3. Dari hasil enkripsi pada Tabel 2., nilai Vn_baru pada iterasi 5 diperoleh 9931 jika dikonversi ke dalam karakter ASCII menjadi "cUS".

TABEL 2.
TAHAPAN DAN HASIL ENKRIPSI KARAKTER "TE"

x	k[x]	Vn_lama	Vj_lama	Vn_baru	Vj_baru
1	0	8469	2	2969	8469
2	1	2969	8469	6018	2969
3	0	6018	2969	2195	557
4	0	2195	557	10838	9587
5	1	10838	9587	**9931**	10838

Kemudian selanjutnya proses enkripsi dari *plain text* karakter "20" menjadi *chiper text*. Sebelum proses enkripsi, terlebih dahulu *plain text* dikonversi kedalam kode ASCII sehingga hasilnya adalah 5048. Proses enkripsi *plain text* = 5048 dengan kunci publik 13 dapat dilihat pada Tabel 3. Dari hasil enkripsi pada Tabel 3., nilai Vn_baru pada iterasi 5 atau iterasi yang terakhir diperoleh 7894 dan jika dikonversi kedalam kode ASCII menjadi karakter "N^".

TABEL 3.
TAHAPAN DAN HASIL ENKRIPSI KARAKTER "20"

x	k[x]	Vn_lama	Vj_lama	Vn_baru	Vj_baru
1	0	5048	2	9747	5048
2	1	9747	5048	9807	9747
3	0	9807	9747	394	3682
4	0	394	3682	520	9030
5	1	520	9030	**7894**	520

Selanjutnya adalah proses enkripsi dari *plain text* karakter "24" menjadi *chiper text*. Sebelum proses enkripsi, terlebih dahulu *plain text* dikonversi kedalam kode ASCII sehingga hasilnya adalah 5052. Proses enkripsi *plain text* = 5052 dengan kunci publik 13 dapat dilihat pada Tabel 4.

TABEL 4.
TAHAPAN DAN HASIL ENKRIPSI KARAKTER "24"

x	k[x]	Vn_lama	Vj_lama	Vn_baru	Vj_baru
1	0	5052	2	5943	5052
2	1	5943	5052	4468	5943
3	0	4468	5943	4916	3770
4	0	4916	3770	9568	6792
5	1	9568	6792	**4721**	9568

Dari hasil enkripsi pada Tabel 4., nilai Vn_baru pada iterasi 5 atau iterasi yang terakhir diperoleh 4721 dan jika dikonversi kedalam kode ASCII menjadi karakter "/NAK".
Sehingga jika digabungkan hasil keseluruhan enkripsi dari *plain text* "VOTE2024" adalah 6669993178944721. Dan jika dikonversi ke dalam karakter ASCII menjadi BEcUSN^/NAK.

D. Hasil Dekripsi

Setelah semua *plain text* selesai di enkripsi, maka proses selanjutnya adalah proses dekripsi. *Chiper text* dari proses enkripsi adalah "BEcUSN^/NAK". *Chiper text* tersebut kembali dikonversi ke dalam kode ASCII, sehingga menjadi "6669993178944721".
Sebelum dilakukan proses dekripsi, terlebih dahulu ditentukan kunci privat dengan menggunakan persamaan (7).

1. Hasil Dekripsi Karakter "BE"

Blok pertama yang akan dicari kunci privatnya adalah karakter "BE" yang terlebih dahulu di konversi ke karakter ASCII menjadi "6669". Kemudian dilakukan proses pembangkitan kunci privat dengan perhitungannya sebagai berikut.

$$\begin{aligned} D &= (C^2) - 4 \\ &= 6669^2 - 4 \\ &= 44.475.561 - 4 \\ &= 44.475.557 \end{aligned}$$

Legendre:

$$\frac{D}{p} = \frac{44.475.557}{43} = -1$$

$$\frac{D}{q} = \frac{44.475.557}{257} = 1$$

r = LCM (43 – (-1)), (257 – 1)
r = LCM (44), (256)
r = 2816

$$e * d \equiv 1 \bmod R(N)$$
$$13 * d \equiv 1 \bmod 2816$$
$$d = 1733$$

Dari perhitungan yang sudah dilakukan, maka diperoleh kunci privat (d) = 1733. Kemudian untuk dapat dilakukan proses dekripsi, maka tahap berikutnya adalah pembangkitan rantai *lucas* k[x] dari kunci privat atau *d* = 1733. Untuk lebih jelasnya hasil pembangkitan rantai lucas kunci privat karakter "BE" dapat dilihat pada Tabel 3.

TABEL 3.
RANTAI LUCAS KUNCI PRIVAT KARAKTER "BE"

x	k[x]	d = 1733
1	1	d-1 = 1732
2	0	d/2 = 866
3	0	d/2 = 433
4	1	d-1 = 432
5	0	d/2 = 216
6	0	d/2 = 108
7	0	d/2 = 54
8	0	d/2 = 27
9	1	d-1 = 26
10	0	d/2 = 13
11	1	d-1 = 12
12	0	d/2 = 6
13	0	d/2 = 3
14	1	d-1 = 2
15	0	d/2 = 1

Didapatkan k[*x*] = {1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0} dimana k[*x*] adalah rantai lucas. Kemudian untuk dapat digunakan urutan rantai lucas harus di urutkan ulang dimulai urutan dari yang terbawah, maka k[*x*] = {0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1}. Setelah diperoleh rantai lucas dari kunci privat, maka dapat dilakukan proses dekripsi dengan menggunakan (7). *Chiper text* yang akan didekripsi adalah karakter "BE". Hasil perhitungan proses dekripsi dapat dilihat pada Tabel 4.

TABEL 4.
HASIL DEKRIPSI KARAKTER "BE"

x	k[x]	Vn_lama	Vj_lama	Vn_baru	Vj_baru
1	0	6669	2	6335	6669
2	1	6335	6669	4524	6335
3	0	4524	6335	122	8679
4	0	122	8679	3831	2324
5	1	3831	2324	7754	3831
6	0	7754	3831	7074	4868
7	1	7074	4868	5970	7074
8	0	5970	7074	1423	10291
9	0	1423	10291	2594	5900
10	0	2594	5900	9826	3347
11	0	9826	3347	8738	4228
12	1	8738	4228	8622	8738
13	0	8622	8738	9856	8751
14	0	9856	8751	2444	1183
15	1	2444	1183	**8679**	2444

Berdasarkan Tabel 4, diperoleh hasil dekripsi dari karakter "BE" yaitu pada iterasi ke-15 dengan Vn_baru = 8679. Kemudian jika kode ASCII 8679 dikonversi ke dalam karakter maka diperoleh karakter atau *plain text* "VO". Sehingga hasil dekripsi dari karakter "BE"adalah karakter "VO".

2. Hasil Dekripsi Karakter "cUS"
Dengan menggunakan persamaan (7) maka diperoleh kunci privat pada karakter "cUS" yaitu d = 1733. Maka selanjutnya tidak perlu dilakukan pembangkitan rantai lucasuntuk proses dekripsi karakter "cUS". Hal ini dikarenakan hasilnya akan sama dengan rantai lucas dari karakter sebelumnya. Hasil perhitungan proses dekripsi karakter "cUS" dapat dilihat pada Tabel 5.

TABEL 5.
HASIL DEKRIPSI KARAKTER "cUS"

x	k[x]	Vn_lama	Vj_lama	Vn_baru	Vj_baru
1	0	9931	2	5635	9931
2	1	5635	9931	41	5635
3	0	41	5635	1679	84
4	0	1679	84	1034	9544
5	1	1034	9544	3782	1034
6	0	3782	1034	3528	10705
7	1	3528	10705	5244	3528
8	0	5244	3528	4646	2578
9	0	4646	2578	2711	10275
10	0	2711	10275	604	8125
11	0	604	8125	131	1976
12	1	131	1976	6018	131
13	0	6018	131	2195	4857
14	0	2195	4857	10838	9071
15	1	10838	9071	**8469**	10838

Pada Tabel 5, diperoleh hasil dekripsi dari karakter "cUS" yaitu pada iterasi ke-15 dengan Vn_baru = 8469. Kemudian jika kode ASCII 8679 dikonversi ke dalam karakter maka diperoleh karakter atau *plain text* "TE". Sehingga hasil dekripsi dari karakter "BE"adalah karakter "TE".

3. Hasil Dekripsi Karakter "N^"
Seperti tahap sebelumnya, proses dekripsi memerlukan kunci privat. Sehingga untuk melakukan proses dekripsi karakter "cUS" harus dicari terlebih dahulu kunci privat. Kunci privat untuk karakter dihitung dengan menggunakan (7). Maka kunci privat yang diperoleh yaitu d = 1733. Maka selanjutnya tidak perlu dilakukan pembangkitan rantai lucasuntuk proses dekripsi karakter "N^". Hal ini dikarenakan hasilnya akan

sama dengan rantai lucas dari karakter sebelumnya. Untuk lebih detail, hasil perhitungan proses dekripsi karakter "N^" dapat dilihat pada Tabel 6.

TABEL 6.
HASIL DEKRIPSI KARAKTER "N^"

x	k[x]	Vn_lama	Vj_lama	Vn_baru	Vj_baru
1	0	7894	2	9696	7894
2	1	9696	7894	41	5635
3	0	41	5635	1679	84
4	0	1679	84	1034	9544
5	1	1034	9544	3782	1034
6	0	3782	1034	3528	10705
7	1	3528	10705	5244	3528
8	0	5244	3528	4646	2578
9	0	4646	2578	2711	10275
10	0	2711	10275	7061	5659
11	0	7061	5659	6658	940
12	1	6658	940	9807	6658
13	0	9807	6658	394	8855
14	0	394	8855	520	10962
15	1	520	10962	**5048**	520

Berdasarkan Tabel 6, diperoleh hasil dekripsi dari karakter "cUS" yaitu pada iterasi ke-15 dengan Vn_baru = 5048. Kemudian jika kode ASCII 5048 dikonversi ke dalam karakter maka diperoleh karakter atau *plain text* "20". Sehingga hasil dekripsi dari karakter "N^" adalah karakter "20".

4. Hasil Dekripsi Karakter "/NAK"

Dari hasil perhitungan kunci privat untuk karakter "/NAK", diperoleh hasil yang sama dengan karakter sebelumnya yaitu d = 1733. Sehingga tidak perlu pembangkitan rantai lucas karena hasilnya juga sama dengan karakter sebelumnya. Hasil perhitungan proses dekripsi karakter "/NAK" dapat dilihat pada Tabel 7.

TABEL 7.
HASIL DEKRIPSI KARAKTER "/NAK"

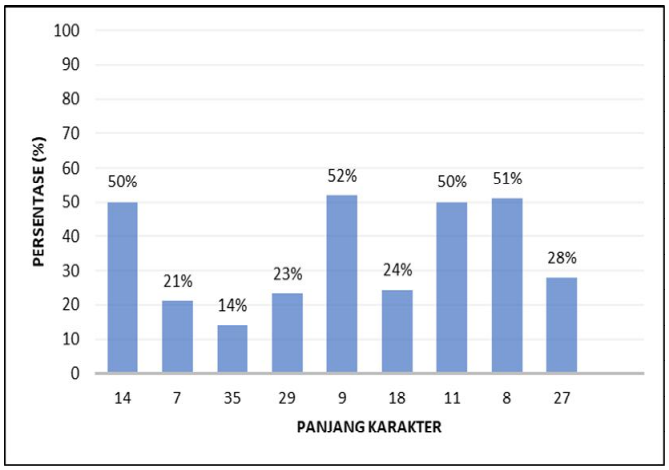
x	k[x]	Vn_lama	Vj_lama	Vn_baru	Vj_baru
1	0	4721	2	9023	4721
2	1	9023	4721	2308	9023
3	0	2308	9023	280	279
4	0	280	279	1041	7093
5	1	1041	7093	824	1041
6	0	824	1041	4863	2136
7	1	4863	2136	3160	4863
8	0	3160	4863	6545	1469
9	0	6545	1469	3347	6565
10	0	3347	6565	7744	9997
11	0	7744	9997	6808	10843
12	1	6808	10843	4468	6808
13	0	4468	6808	4916	1071
14	0	4916	1071	9568	39
15	1	9568	39	**5052**	9568

Dari hasil yang ada pada Tabel 7, dapat diketahui hasil dekripsi dari karakter "/NAK" yaitu pada iterasi ke-15 dengan Vn_baru = 5052. Kemudian jika kode ASCII 5052 dikonversi ke dalam karakter maka diperoleh karakter atau *plain text* "24". Sehingga hasil dekripsi dari karakter "N^" adalah karakter "24".

Dari hasil keseluruhan proses dekripsi dari setiap blok karakter, diperoleh nilai ASCII dari gabungan keseluruhan blok yang didekripsi yaitu "8679846950485052". Kemudian jika dikonversi ke dalam karakter ASCII, hasilnya adalah VOTE2024. VOTE2024 merupakan *plain text* yang diperoleh dan sesuai dengan teks yang digunakan pada tahap awal proses enkripsi.

E. Hasil Pengujian Avalanche Effect

Untuk mengetahui tingkat keamanan algoritma Luc, maka harus dilakukan pengujian. Dalam pengujian *Avalanche Effect* terdapat dua buah *plaintext* yang berbeda satu bit yang dienkripsi dengan satu kunci yang sama. Pengujian ini dilakukan sebanyak 10 kali, sehingga terdapat total 20 buah *plaintext* dengan panjang karakter yang berbeda, serta 10 buah kunci. Grafik hasil pengujian algoritma Luc dengan menggunakan *Avalanche Effect* dapat dilihat pada Gambar 5.



Gambar 5. Grafik Hasil Pengujian dengan *Avalanche Effect*

Hasil pengujian pada Gambar 5 menunjukkan bahwa algortima Luc menghasilkan persentase sekitar diatas 50%. Tetapi dapat dilihat bahwa semakin panjang suatu teks, maka hasil persentase dari *Avalanche Effect* akan semakin menurun.

IV. KESIMPULAN DAN SARAN

Berdasarkan hasil pengujian dan analisis penerapan algoritma *luc* pada *e-voting* yang telah dilakukan dapat disimpulkan bahwa: Algoritma Luc dapat diimplementasikan pada sistem *e-voting* untuk untuk keamanan autentikasi data serta mencegah penyalahgunaan informasi dan hak pilih dari

pemilih atau pengguna (*user*) Hasil pengujian dengan *Avalanche Effect*, menunjukkan bahwa algoritma luc dinilai cocok untuk menjaga data dari serangan pasif dengan tingkat keamanan yang menengah ke tinggi dengan hasil persentase sekitar 50% untuk data yang memiliki panjang teks rentang 15 karakter. Dan dapat disimpulkan bahwa kekuatan dari algoritma luc tidak bergantung pada panjang teks atau karakter. Karena justru panjang teks atau karakter = 8 memiliki nilai Avalanche Effect diatas 50% yang dapat diartikan tingkat keamanannya lebih tinggi. Untuk penelitian berikutnya *e-voting* ini dapat dikembangkan dengan menambahkan algoritma kunci publik lainnya, seperti algoritma Rabin dan ElGamal. Dan untuk penelitian berikutnya sebaiknya dapat memproses enkripsi dan dekripsi pada semua data yang ada di basis data sistem *e-voting*. Dan sistem e-voting ini dapat dikembangkan menjadi aplikasi berbasis *mobile*. Sehingga nantinya aplikasi *e-voting* ini dapat diakses dimana saja dan kapan saja.

ACKNOWLEDGEMENT

Terimakasih banyak disampaikan kepada Lembaga Penelitian dan Pengabdian Masyarakat Universitas Sam Ratulangi yang telah memberikan kesempatan dan juga pendanaan sehingga penelitian ini dapat terselenggara dengan baik.

V. KUTIPAN

[1] N. Keerthi, A. Raghuram, and R. Jayaraman, "Interfacing of Online and Offline Voting System with an E-Voting Website," in 2022 6th International Conference on Devices, Circuits and Systems (ICDCS), IEEE, Apr. 2022, pp. 223–228. doi: 10.1109/ICDCS54290.2022.9780681.

[2] D. I. Sensuse, P. B. Pratama, and Riswanto, "Conceptual Model of E-Voting in Indonesia," in 2020 International Conference on Information Management and Technology (ICIMTech), IEEE, Aug. 2020, pp. 387–392. doi: 10.1109/ICIMTech50083.2020.9211156.

[3] M. Mehta, "Online Voting System," Int J Res Appl Sci Eng Technol, vol. 10, no. 5, pp. 1471–1476, May 2022, doi: 10.22214/ijraset.2022.42552.

[4] A. Olumide S., B. Olutayo K., and S. E. Adekunle, "A Review of Electronic Voting Systems: Strategy for a Novel," International Journal of Information Engineering and Electronic Business, vol. 12, no. 1, pp. 19–29, Feb. 2020, doi: 10.5815/ijieeb.2020.01.03.

[5] C. Denis González, D. Frias Mena, A. Massó Muñoz, O. Rojas, and G. Sosa-Gómez, "Electronic Voting System Using an Enterprise Blockchain," Applied Sciences, vol. 12, no. 2, p. 531, Jan. 2022, doi: 10.3390/app12020531.

[6] Y.-X. Kho, S.-H. Heng, and J.-J. Chin, "A Review of Cryptographic Electronic Voting," Symmetry (Basel), vol. 14, no. 5, p. 858, Apr. 2022, doi: 10.3390/sym14050858.

[7] A. Ezugwu et al., "Password-based authentication and the experiences of end users," Sci Afr, vol. 21, p. e01743, Sep. 2023, doi: 10.1016/j.sciaf.2023.e01743.

[8] A. Aditya, N. Singh, M. Singh, A. Begum, and Ambika, "Password encryption-decryption based online voting system using OTP," 2024, p. 020055. doi: 10.1063/5.0184820.

[9] I. Anastasaki, G. Drosatos, G. Pavlidis, and K. Rantos, "User Authentication Mechanisms Based on Immersive Technologies: A Systematic Review," Information, vol. 14, no. 10, p. 538, Oct. 2023, doi: 10.3390/info14100538.

[10] M. Sbai El Idrissi, Y. F. Ebobissé Djéné, P.-M. Tardif, and B. El-Bhiri, "Efficient Energy Consumption of IoT Network Security Based on Symmetric and Asymmetric Cryptography and Hash Function," 2024, pp. 73–80. doi: 10.1007/978-3-031-46849-0_8.

[11] Z. Arif and A. Nurokhman, "Analisis Perbandingan Algoritma Kriptografi Simetris Dan Asimetris Dalam Meningkatkan Keamanan Sistem Informasi," Jurnal Teknologi Sistem Informasi, vol. 4, no. 2, pp. 394–405, Sep. 2023, doi: 10.35957/jtsi.v4i2.6077.

[12] N. Permata Dewi, D. J. M. Sembiring, R. BR. Ginting, and M. BR. Ginting, "Pengamanan Data dengan Kriptografi Hibrida Algoritma Hill Cipher dan Algoritma Luc Serta Steganografi Chaotic Lsb," Jurnal Syntax Admiration, vol. 3, no. 2, pp. 341–361, Feb. 2022, doi: 10.46799/jsa.v3i2.389.

[13] D. Rachmawati, M. A. Budiman, and D. F. Perangin-angin, "A hybrid cryptosystem approach for information security by using RC4 algorithm and LUC algorithm," J Phys Conf Ser, vol. 1321, no. 3, p. 032013, Oct. 2019, doi: 10.1088/1742-6596/1321/3/032013.

[14] P. P. Sari, E. Budhiarti Nababan, and M. Zarlis, "Comparative Study of LUC, ElGamal and RSA Algorithms in Encoding Texts," in 2020 3rd International Conference on Mechanical, Electronics, Computer, and Industrial Technology (MECnIT), IEEE, Jun. 2020, pp. 148–151. doi: 10.1109/MECnIT48290.2020.9166586.

[15] W. Ady Putra, S. Suyanto, and M. Zarlis, "Performance Analysis Of The Combination Of Advanced Encryption Standard Cryptography Algorithms With Luc For Text Security," SinkrOn, vol. 8, no. 2, pp. 890–897, Apr. 2023, doi: 10.33395/sinkron.v8i2.12202.

[16] N. R. D. P. Astuti, D. P. Setiawan, and D. C. Hakika, "Comparative Study of Elgamal and Luc Algorithm in Cryptographic Key Generation," ASEAN Engineering Journal, vol. 13, no. 4, pp. 61–68, Oct. 2023, doi: 10.11113/aej.v13.19184.

[17] Pratiwi, M., Mayola, L., Kris Hiburan Laoli, V., Ilhami Arsyah, U., & Pratiwi, N. (2022). Medical Record Information System with Rapid Application Development (RAD) Method. Journal of Information Systems and Technology Research, 1(2), 124–130. https://doi.org/10.55537/jistr.v1i2.170

[18] Maulany, R., Hasan, B., Abdullah, A. G., & Rohendi, D. (2021). Design Of Learning Applications Using the Rapid Application Development Method. IOP Conference Series: Materials Science and Engineering, 1098(2), 022090. https://doi.org/10.1088/1757- 899X/1098/2/022090

[19] Faqih, H., Hikmah, A. B., & Azizah, W. (2022). Implementasi Metode Rapid Application Development Pada Pengembangan Aplikasi e-Fin Mosque Z. Indonesian Journal on Software Engineering (IJSE), 8(1), 83-91.

[20] Putra, W. A., Suyanto, S., & Zarlis, M. (2023). Performance Analysis Of The Combination Of Advanced Encryption Standard Cryptography Algorithms With Luc For Text Security. Sinkron: jurnal dan penelitian teknik informatika, 7(2), 890-897.

[21] Rachmawati, D., M. A. Budiman, and D. F. Perangin-angin. "A hybrid cryptosystem approach for information security by using RC4 algorithm and LUC algorithm." Journal of Physics: Conference Series. Vol. 1321. No. 3. IOP Publishing, 2019.

[22] Sanap, S. D., & More, V. (2021, May). Performance analysis of encryption techniques based on avalanche effect and strict avalanche criterion. In 2021 3rd International Conference on Signal Processing and Communication (ICPSC) (pp. 676-679). IEEE.

[23] K. Mohamed, M. N. M. Pauzi, F. H. H. M. Ali, and S. Ariffin, "Analyse On Avalanche Effect In Cryptography Algorithm," Oct. 2022, pp. 610–618. doi: 10.15405/epms.2022.10.57.

Bernad J. D. Sitompul, adalah salah satu dosen aktif yang bertugas di Fakultas Teknik Progaram Studi Teknik Informatika Universitas Sam Ratulangi Manado. Menempuh Pendidikan Sarjana di Universitas Budi Darma Medan pada tahun 2008. Kemudian menyelesaikan Pendidikan Magister di Universitas Sumatera Utara pada tahun 2018 dengan judul "Peningkatan Hasil Evaluasi Clustering Davies Bouldin Index Dengan Penentuan Tititk Pusat Cluster Algoritma K-Means". Dan sudah dipublikasikan dalam bentuk jurnal di Scopus pada tahun 2019. Riset yang ditekuni sampai saat ini adalah Kecerdasan Buatan. Beberapa topik penelitian yang sudah diselesaikan antara lain Sistem Pendukung Keputusan, String Matching, Data Mining, dan lain sebagainya.