Implementasi Kriptografi Dalam Pengamanan Database E-Voting Menggunakan Algoritma Rsa Dan Base64 Berbasis Progresive Web Apps

(Studi Kasus: Pemilihan Presiden Mahasiswa STMIK Tasikmalaya)

Yuda Pratama Putra[1], Fitri Nuraeni[2], Rijal Ajji Jatnika[3]
[1,2,3]Program Studi Teknik Informatika, STMIK Tasikmalaya, Tasikmalaya
Email : [1]yudaestilo@gmail.com, [2]nufi3@stmik-tasikmalaya.ac.id, [3]jatnika10@gmail.com

Abstrak

Evoting adalah suatu system pemilihan dimana data dicatat, disimpan, dan di proses pelaksanaannya dimulai dari pendaftaran pemilih, pelaksanaan pemilih, dan pemungutan hasil suara [1]. Pemilihan umum saat ini sudah banyak yang memanfaatkan teknologi sebagai medianya. Salah satunya dengan menggunakan sistem evoting, dimana semua proses dari pendaftaran peserta hingga pemungutan suara dilakukan secara digital. Namun beberapa orang belum sepenuhnya percaya dengan pemilihan suara secara digital karena kemungkinan hasil suara yang diperoleh dapat dimanipulasi oleh orang yang tidak bertanggung jawab. Maka dari itu perlu dibuat sebuah sistem yang dapat menjamin keamanan akurasi hasil dari evoting, integritas data, dan validasi pemilih. Pada penelitian ini penulis menerapkan metode keamanan menggunakan kiptografi dengan algoritma RSA dan Base64 untuk mengenkripsi data hasil pemilihan menjadi teks yang tidak dapat dibaca. Media Penerapan evoting-nya menggunakan progresive web apps dan untuk menguji tingkat keamanan dilakukan pengujian korelasi, entropy, dan waktu enkripsi dan hasilnya cukup mengesankan dengan rata-rata korelasi 0,0503, rata-rata entrophy plainteks adalah 2,60360198 dan rata-rata entrophy cipherteks adalah 3,089234653, serta rata-rata waktu proses enkripsi adalah 0,9544 milidetik. Sehingga dapat disimpulkan bahwa pengamanan database dengan enkripsi RSA dan base64 sangat bagus.

Kata kunci: *Evoting*, Super Enkripisi, kriptografi, RSA, base64

Abstrack

Evoting is an electoral system where data is recorded, stored, and in the process of its implementation starting from voter registration, voter implementation, and voting results [1]. Today's general election has used technology as a medium. One way is to use an evoting system, where all processes from participant registration to voting are done digitally. However, some people do not fully believe in digital voting because the possibility of the votes obtained can be manipulated by irresponsible people. Therefore, it is necessary to create a system that can guarantee the security of the accuracy of the results from voting, data integrity and voter validation. In this study the authors applied a security method using a kiptography with RSA and Base64 algorithms to encrypt the results of the election data into unreadable text. Media Application of evoting using progressive web apps and to test the level of security testing the correlation, entropy, and encryption time and the results are quite impressive with an average correlation of 0.0503, the average plaintext entrophy is 2.60360198 and the average entrophy ciphertext is 3.089234653, and the average encryption process time is 0.9544 milliseconds. Finaly concluded that securing the database with RSA and base64 encryption is very good.

Keyword: *Evoting, cryptography, encryption, RSA, base64*

1. Pendahuluan

Penyelenggaraan pemilihan umum yang bebas sudah menjadi sarana bagi orang-orang dalam menentukan pemimpin dengan bebas, jujur, dan adil[1]. Pemilihan umum ini dilakukan pada setiap pemilihan pemimpin baik di sekolah, desa, kecamatan, kabupaten, provinsi, hingga presiden. Pemilihan umum dilakukan dengan menggunakan kertas suara sebagai media untuk pemilih memberikan suaranya. Setelah itu pemilih memasukan surat suara ke dalam kotak suara yang nanti akan dihitung sebagai suara, terlepas dari surat suara itu sah dan tidak sah. Pemilihan suara seperti ini cukup beresiko, dimulai dengan manipulasi suara, surat suara tidak sah, kerusakan surat suara, dan penggunaan biaya kertas yang tidak

efektif. Sehingga seiring perkembangan zaman, untuk mengatasi kekurangan dari pemilihan umum secara manual maka dikembangkanlah sebuah sistem yang disebut Electronic Voting yang biasa disebut dengan E-Voting.

E-Voting atau pemilihan secara elektronik merupakan sistem yang memungkinkan pemilih untuk mencatat pilihannya yang bersifat rahasia secara elektronik yang teramankan. Pengertian lainnya adalah sebuah proses yang dibuat untuk membuat surat suara, memberikan, menghitung, menayangkan perolehan suara, serta menghasilkan dan memelihara jejak audit secara digital[2]. E-Voting ini sudah dilakukan oleh beberapa instansi pendidikan salah satunyaditerapkan pada pemilihan ketua OSIS berbasis website di MA Nurul Ihsan NW Tilawah. E-voting ini berlangsung dengan mengharuskan setiap siswa masuk dengan NIK masing-masing, kemudian memilih calon ketua osisnya, setelah itu suara akan tersimpan kedalam database dan siap untuk dihitung. Penelitian yang telah dilakukan ini membuktikan bahwa dengan evoting, pemilihan ketua osis lebih mudah dilaksanakan, penggunaan kertas lebih sedikit dan irit biaya, proses perhitungan menjadi lebih cepat, dan dapat mengurangi siswa yang tidak memilih[3]. Namun pada sistem tersebut masih ada beberapa kekurangan, salah satunya karena sistem ini akan meyimpan seluruh data ke dalam database dimana data itu rentan untuk dimanipulasi oleh pihak yang tidak bertanggung jawab membuat suara yang diperoleh diragukan keasliannya. Maka untuk mengatasi masalah keamanan database evoting ini diperlukan sebuah metode pengamanan yang bisa menjaga data tersebut aman dari manipulasi, salah satu caranya adalah dengan menerapkan metode kriptografi.

Kriptografi adalah seni atau ilmu untuk mengamankan data yang didalamnya terdapat algoritma tertentu yang bertujuan sebagai pembingung, dengan cara mengubah teks asli (plaintext) menjadi teks yang tidak bisa dibaca (ciphertext). Dalam kriptografi ada dua jenis algoritma, yaitu simetris dan asimetris[4]. Simetris adalah metode kriptografi yang menggunakan satu kunci untuk enkripsi dan dekripsi dari sebuah teks. Sedangkan asimetris adalah metode kriptografi dengan menggunakan dua kunci yang berbeda baik enkripsi maupun dekripsi. contoh dari kriptografi simetris adalah Base64 dan asimetris adalah RSA (Rivest, Shamir, Adleman). Kedua algoritma kriptografi ini mempunyai kelebihan masing-masing. diantaranya adalah RSA yang mempunyai tingkat keamanan yang paling baik, karena memiliki dua kunci untuk enkripsi dan dekripsinya. Sedangkan algoritma Base64 terbilang cepat karena hasil dari enkripsinya berupa string.

Melihat karakteristik sistem E-voting yang digunakan oleh banyak orang dan berada ditempat umum, akan sangat beresiko apabila pengamanan database hanya menggunakan satu buah kunci. Maka alangkah baiknya bila menggunakan sistem dua kunci untuk pengamanan data yang lebih baik pada database E-voting ini. Kriptografi asimetris yang menggunakan dua kunci, yaitu algoritma RSA (Rivest, Shamir, Adleman). Untuk menambah keamanan dan performa dari E-voting ini digunakanlah algoritma Base64 sebagai bantuan untuk mempercepat proses pengiriman data.

Algoritma Kriptografi RSA merupakan algoritma asimetris, dimana kunci yang digunakan untuk enkripsi disebut kunci public berbeda dengan kunci yang digunakan untuk mendekripsi disebut kunci private. Penggunaan algoritma RSA ini terbilang aman karena sulitnya dalam memfaktorkan bilangan besar menjadi faktor-faktor bilangan primanya, sehingga semakin besar bilangan prima, makan semakin aman dan baik kualitas keamanan nya [5]. Proses dari Algoritma RSA terbagi menjadi tiga bagian utama yaitu proses pembangunan kunci, proses enkripsi, dan proses dekripsi [6].

Algoritma Kriptografi Base64 merupakan algoritma encoding dan decoding data untuk format ASCII, bisa berdasarkan bilangan 64 atau dikatakan metode yang digunakan untuk melakukan encoding (penyedian) untuk binary[4]. Algoritma ini termasuk kedalam jenis kriptografi simetris, karena mempunyai satu kunci untuk proses enkripsi dan dekripsinya.

Dalam sistem E-voting keamanan merupakan faktor penting, maka untuk menjaga keamanan data dalam database dibuatlah sebuah metode untuk mengamankannya salah satunya dengan menggunakan kriptografi, karena dengan kriptografi nilai yang ada dalam database berbeda dengan nilai yang ditampilkan pada program, sehingga hasil suara akan sulit dimanipulasi. Penggunaan algoritma RSA dan Base64 ini sangat cocok karena mempunyai tingkat keamanan dan kecepatan yang baik.

Dalam penelitian ini subjek yang digunakan adalah Pemilihan Presiden Badan Eksekutif Mahasiswa di STMIK Tasikmalaya, dimana perguruan tinggi tersebut sudah mulai menerapkan sistem E-voting dalam pemilihan Presiden mahasiswa pada tahun 2019, sehingga dengan pengembangan berupa pengamanan menggunakan kriptografi dalam databasenya akan sangat membantu dalam menjaga keaslian dari setiap suara. Maka keamanan dari pemilihan presiden mahasiswa menjadi lebih baik dan tehindar dari manipulasi data berbagai macam resiko keamananan lainnya.

Berdasarkan dari uraian diatas penulis akan mengangkat penelitian yang berkenaan dengan pengamanan database dalam sistem e-voting menggunakan kriptografi. Dalam penelitian skripsi ini

penulis mengangkat judul "Implementasi Kriptografi Dalam Pengamanan Database e-voting Menggunakan Algoritma RSA dan Base64 Berbasis Progresive Web Apps".

2. Metode Penelitian

Metode penelitian yang digunakan adalah observasi, studi literatur, dan wawancara sehingga kita dapat mengetahui darimana kita dapat menemukan data, sehingga data yang diterima dapat dipastikan keasliannya. Algoritma enkripsi yang digunakan adalah RSA dan base64.

1. Algoritma Rivest Shamir Adleman (RSA)

Rivest Shamir Adleman atau RSA adalah salah satu kriptografi kunci asimetrik yang cukup terkenal dengan kekuatan enkripsinya. Algoritma enkripsi dan dekripsi sistem kriptografi RSA bersandar ada asumsi satu arah yang dibagun oleh fungsi eksponsial modular pada group perkalian (Z^* n,×) dan group perkalian (〖Z^*〗 _(Ø(n)),×) dengan proses n = p × q,p,q adalah bilangan prima dan Ø(n) = (p -1)(q-1). Algoritma ini ada tiga proses yaitu pembentukan kunci, proses enkripsi, dan dekripsi.

Berikut ini adalah proses pembentukan kunci dalam kriptografi RSA:

1. Memilih dua bilangan prima yang diberi simbol sebagai p dan q nilai $(p \neq q)$.
2. Menghitung nilai n = . q $(p \neq q$, karena jika $p = q$, maka nilai $n = p^2$ sehingga nilai p dapat diperoleh dengan menarik akar pangkat dua dari n).
3. Hitung $\varphi(n) = (p - 1)(q - 1)$.
4. Memilih kunci publik e yang relatif prima terhadap (n)
5. Bangkitkan kunci privat dengan persamaan $e.d \equiv 1 \ (mod\ \varphi(n))$ dimana $1 < d < \varphi(n)$. Perhatikan bahwa persamaan $e.d \equiv 1\ (mod\ \varphi(n))$ ekivalen dengan e . d = 1 + k (n), sehingga untuk mencari nilai d dapat dihitung dengan $d = \frac{1 + k\varphi(n)}{e}$

Hasil dari pembentukan pasangan kunci diatas adalah (e, n) adalah kunci publik, dan (d, n) adalah kunci rahasia. Nilai n tidak bersifat rahasia karena diperlukan pada saat perhitungan proses enkripsi dan dekripsi.

Contoh pembangkitan kunci RSA sebagai berikut:

1. Pilih bilangan dua bilangan prima, misalnya **p = 2711** dan **q = 3169**
2. Hitung nilai N dengan rumus
   $n = p * q$, $2711 * 3169 = 8591159$
   $n = 8591159$
3. Kemudian hitung nilai φ(N) dengan persamaan:
   $\varphi(n) = (p - 1)(q - 1) = 4292640$
4. Kemudian, misalnya yang dipilih sebagai e adalah **2089**
5. Sehingga kunci publik nya adalah **(2089, 8591159)**
6. Untuk mencari kunci privat atau d dapat menggunakan parameter $d = \frac{1+4292640 \times 106(8591159)}{2089}$ hasilnya adalah kunci privatnya adalah **437689**

Setelah membangkitkan kunci kemudian dilanjutkan pada tahap enkripsi RSA:

1. Ambil nilai plaintext nya.
2. Ambil kunci publik penerima pesan e dan modulus n atau (e, n).
3. Lakukan perhitungan dengan menggunakan rumus $C = P^e \ mod\ n$.
4. Ciphertext sudah ditemukan.

Contoh enkripsi RSA :
Misalkan *plaintext* yang akan dienkripsi adalah **x = 148**

1. Ambil nilai dari plaintext yaitu **x = 148**
2. Ubah plainteks tersebut kedalam bentuk ASCII Code sehingga menjadi **495256,** dimana 1 = **49**, 4 = **52**, 8 = **56**
3. Kemudian ambil kunci kunci publik yaitu **(2089, 8591159),** berarti **e = 2089** dan **n = 8591159**
4. Lakukan perhitungan dengan rumus $C = P^e \ mod\ n$ seperti dibawah ini:
   495256^{2089} mod 8591159 = 7967745

   Jadi *ciphertext* yang dihasilkan adalah
   Y = **7967745**

Untuk proses dekripsi dari cipherteks menjadi plainteks dari algoritma RSA adalah sebagai berikut:

1. Ambil pesan (cipherteks) yang telah diterima.
2. Kemudian ambil *private key* yaitu **437689**.
3. Lakukan perhitungan dengan rumus $P = C^d \, mod \, n$
4. Maka akan diperoleh hasilnya.

Contoh proses dekripsi adalah sebagai berikut:

1. Ambil pesan (cipherteks) yang diterima yaitu **7967745**
2. Ambil dan dekripsi menggunakan kunci privat d = **437689**
3. Lakukan perhitungan dengan menggunakan rumus $P = C^d \, mod \, n$ seperti dibawah ini: **7967745^{437689}** mod **8591159 = 495256**
4. Maka hasil enkripsi tersebut diubah kedalam karakter ASCII, dimana 49 = 1, 52 = 4, 56 = 8.
5. Maka hasil dekripsinya adalah **148,** sesuai dengan plaintext pertama

2. Algoritma base64

Transformasi Base64 merupakan salah satu algoritma untuk encoding dan decoding suatu data ke dalam format ASCII, yang didasarkan pada bilangan dasar 64 atau bisa dikatakan sebagai salah satu metoda yang digunakan untuk melakukan encoding (penyandian) terhadap data binary. Karakter yang dihasilkan pada transformasi Base64 ini terdiri dari A..Z, a..z dan 0..9, serta ditambah simbol "+" dan "/" serta satu buah karakter sama dengan (=) di dua karakter terakhir yang dipakai untuk pengisian pad atau dengan kata lain penyesuaian dan menggenapkan data binary. Karakter simbol yang akan dihasilkan akan tergantung dari proses algoritma yang berjalan[4]. Kriptografi transformasi Base64 banyak digunakan di dunia Internet sebagai media data format untuk mengirimkan data, penggunaan tersebut dikarenakan hasil dari encode Base64 berupa plaintext, maka data ini akan jauh lebih mudah dikirim, dibandingkan dengan format data yang berupa binary. Algoritma Base64 menggunakan kode ASCII dan kode index Base64 dalam melakukan proses enkripsi ataupun dekripsinya.

Langkah-langkah enkripsi dari *Base64*, jika sebuah *string (bytes)* yang akan disandikan ke algoritma *base64* maka tahapannya yaitu:

1. Hitung panjang karakter yang akan kita enkripsi. kemudian jumlah karakter dikali 8.
2. Hasil dari tahap pertama kemudian dibagi 6.
3. Ambil nilai *binary* dari setiap huruf yang akan di enkripsi;
4. Kemudian sejajarkan angka binary dari setiap huruf tersebut.
5. Kemudian ubah menjadi binary yang awalnya 8 bit menjadi binary berukuran 6 bit dalam blok ukuran 6 bit.
6. Hitung angka biner 6 bit tersebut dan ubah ke dalam bentuk *decimal*.
7. Ubah bentuk decimal ke dalam bentuk karakter berdasarkan tabel enkripsi base64.

Catatan apabila panjang karakter bukan merupakan kelipatan 3 atau dalam pembagian hasil bilangan menjadi 6 bit ada sisa pembagi, maka ditambahkan karakter *pad* (=) sebagai penggenap. Oleh karena itu, terkadang pada *Base64* akan muncul satu atau dua karakter(=)[8].

Contoh enkripsi menggunakan algoritma base64 adalah sebagai berikut:

1. Contoh pesannya adalah "evoting". Kata "evoting" memiliki 7 karakter kemudian dikali kan 8, hasilnya adalah 56. Kemudian dibagi 6 untuk merubah kedalam bentuk 6 bit. Sehingga 56 dibagi 6 hasilnya adalah 9 dengan sisa 2.

2. Plainteks nya diubah ke dalam kode ASCII dan binary

Teks	ASCII	Binary
e	101	01100101
v	118	01110110
o	111	01101111
t	116	01110100
i	105	01101001
n	110	01101110
g	103	01100111

3. Kemudian kemudian ubah dari binary 8 bit ke binary 6 bit dengan menderetkan angka binary 8 bit. Dimana hasil 6 bit nya mewakili setiap 1 karakter pada base64

c	v	o	t	i	n	g			
01100101	01110110	01101111	01110100	01101001	01101110	01100111			
011001	010111	011001	101111	011101	000110	100101	101110	011001	11

Karena panjang kata bukan kelipatan 3 maka akan ada padding "=" sepanjang 2 karakter.

4. Kemudian ubah biner 6 bit kedalam desimal dan karakter base64

011001	010111	011001	101111	011101	000110	100101	101110	011001	11
25	23	25	47	29	6	37	46	25	48
Z	X	Z	v	d	G	l	u	Z	w

5. Sehingga hasil enkripsi menggunakan algoritma base64 adalah **ZXZvdGluZw==**

Untuk melakukan dekripsi algoritma *Base64*, dengan cara membalikan proses enkripsi. Tahapan dekripsi *base64* adalah sebagai berikut:

1. Untuk proses dekripsi pada base64 ini terbilang mudah contohnya adalah ciphertext "ZXZvdGluZw==". Pertama kita ubah dulu teks nya kedalam bentuk decimal dan biner

Z	X	Z	v	d	G	l	u	Z	w	=	=
25		23	25	47	29	6	37	46	25	48	
011001	010111	011001	101111	011101	000110	100101	101110	011001	11	00	00

Catatan: apabila terdapat padding maka setiap satu "=" bernilai 00.

2. Kemudian ubah menjadi kode biner itu menjadi 8 bit dengan mensejajarkan semua angka biner.

Z	X	Z	v	d	G	l	u	Z	w	=	=
25		23	25	47	29	6	37	46	25	48	
011001	010111	011001	101111	011101	000110	100101	101110	011001	11	00	00
01100101	01110110	01101111	01110100	01101001	01101110	01100111	0000				

3. Ubah angka biner 8 bit menjadi decimal

01100101	01110110	01101111	01110100	01101001	01101110	01100111	0000
101	118	111	116	105	110	103	0

4. Setelah menemukan bilangan decimal. Ubah bilangan decimalnya kedalam bentuk karakter dengan membandingkan tabel ASCII

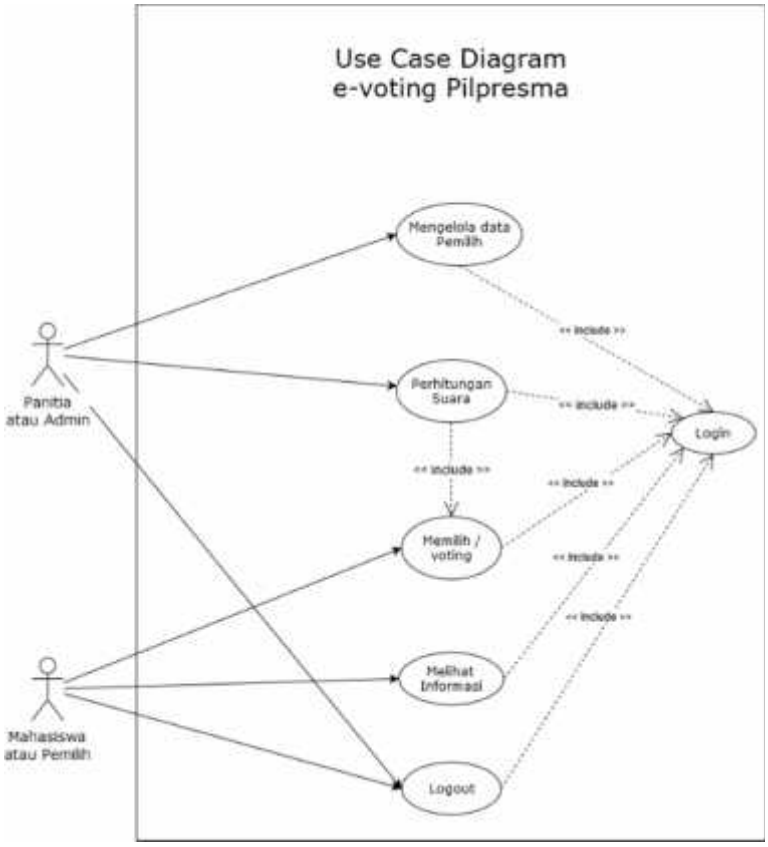
01100101	01110110	01101111	01110100	01101001	01101110	01100111	0000
101	118	111	116	105	110	103	0
e	v	o	t	i	n	g	

5. Dan hasil dari dekripsinya adalah "**evoting**"

Kedua algoritam tersebut digabungkan menjadi sebuah super enkripsi agar kekuatan enkripsinya lebih kuat, maka dari itu keamanan database pemilihan bisa menjadi lebih baik dan terhindar dari hal-hal yang tidak diinginkan. Keaslian data yang ada dalam database dapat dipertanggung jawabkan.

3. Hasil dan Pembahasan

Untuk mengubah menjadi sebuah super enkripsi maka dibuatlah suatu fungsi enkripsi dan dekripsi yang ditambahkan pada system evoting untuk pemilihan presiden mahasiswa di STMIK Tasikmalaya. Pada gambar 3.1 dibawah ini merupakan alur yang terdapat pada pemilihan umum presiden mahasiswa. Dimana setiap siswa diharuskan login untuk mengambil hak pilihnya sehingga dengan begitu mahasiswa dapat memilih calon presiden sesuai dangan haknya. Sedangkan untuk dekripsinya kunci sudah ditentukan pertama oleh ketua KPUM dimana semua data akan terlihat dan muncul setelah dilakukan dekripsi oleh ketua pada saat pembukaan hasil suara.

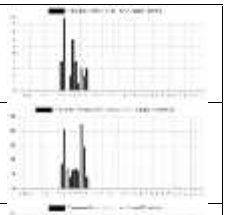
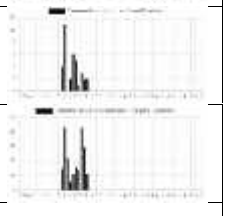
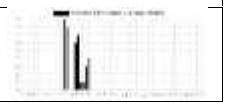


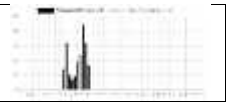
Gambar 3.1 usecase diagram sistem pemilihan evoting dengan super enkripsi

1. Pengujian enkripsi

Untuk mengetahui kualitas dari super enkripsi ini, dilakukan pengujian pada proses enkripsi dengan 100 sampel data hasil pemilihan presiden mahasiswa. Dimana pengujian nya difokuskan pada ukuran file, entropi, korelasi. Dengan asumsi bahwa semakin rendah korelasi dan semakin tinggi nilai entrophy antar variable maka kualitas dari enkripsi akan semakin bagus.

Tabel 3.1 Pengujian

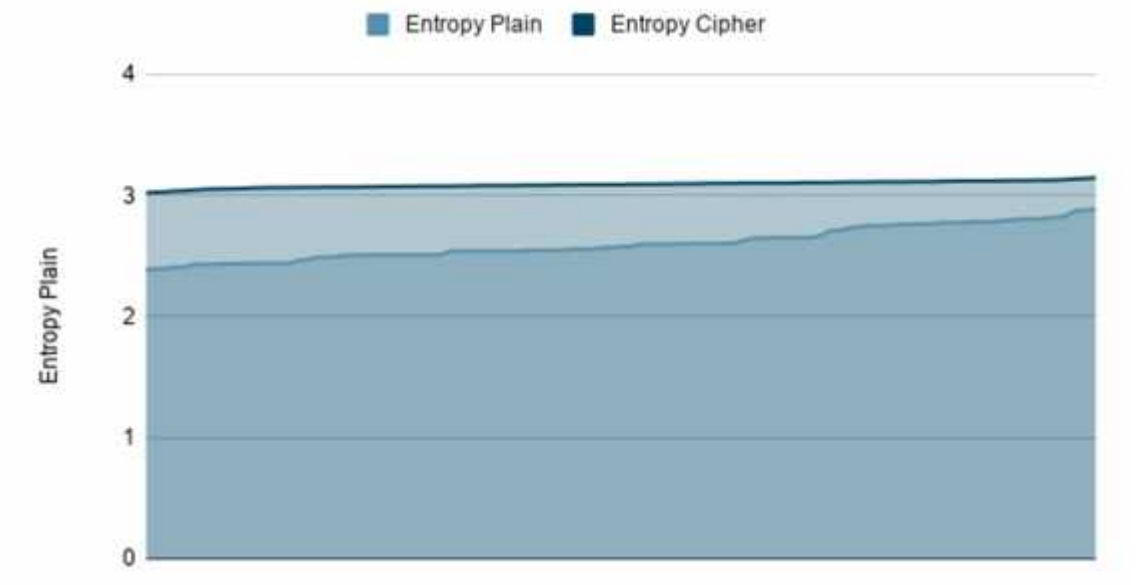
p1.txt	40 Btyes	2.7302217752085	0.014570378207348 korelasi sangat lemah	1,4499	
c1.txt	203 Btyes	3.0435466823054 Enkripsi lebih bagus			
p2.txt	38 Btyes	2.7630878867903	0.046742167822391 korelasi sangat lemah	1,2349	
c2.txt	213 Btyes	3.0635077419141 Enkripsi lebih bagus			
…	…	…	…	…	…
p100.txt	41 Btyes	2.6463184440544	0.019070529377561 korelasi sangat lemah	0,7749	

c100.txt	211 Btyes	3.0883255483692 Enkripsi lebih bagus			

Hasil pengujian diatas diubah kedalam bentuk grafik supaya mudah dibaca. Dapat dilihat pada gambar dibawah ini hasil dari korelasi, entropy, dan waktu enkripsinya:

Gambar 3.2 Grafik dari korelasi

Gambar 3.3 Perhitungan waktu enkripsi



Gambar 3.3 Perbandingan dari entropy dari setiap data.

2. Pengujian Sistem

Selain pengujian terhadap enkripsi, dilakukan pula pengujian terhadap sistem pemilihan umumnya, dimana ini merupakan alat untuk menjalankan pemilihan umum dan menjalankan enkripsinya. Pengujian system ini menggunakan metode Black Box.

Tabel 3.2 Pengujian Pendaftaran Mahasiswa

No	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil uji	Kesimpulan
1	Mengosongkan text field nama dan NPM, kemudian	Nama: - Npm: -	Sistem akan menolak akses daftar dan menampilkan	✕	Tidak Berhasil

	menekan tombol “Daftar”		pesan “Harap masukan data dengan benar”.		
2	Mengisi nama pada isian nama dan mengosongkan npm, kemudian menekan tombol “Daftar”	Nama: Galih Npm: -	Sistem akan menolak akses login dan menampilkan pesan “Data belum lengkap, harap lengkapi data anda”	✕	Tidak Berhasil
3	Mengisi nama pada field nama dan field npm, kemudian menekan tombol “Daftar”	Nama: Gani NPM : 06160732	Sistem akan merespon dan menyimpan data ke dalam database, kemudian data tampil pada tabel peserta.	✔	Berhasil

Tabel 3.3 Pengujian halaman perhitungan suara

No	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil uji	Kesimpulan
1	Mengosongkan field kunci kemudian menekan tombol “Buka Kunci”	Kunci: -	Sistem akan menolak kunci dan menampilkan pesan “Harap masukan kunci dengan benar”.	✔	Berhasil
2	Mengisi field kunci dengan kunci yang salah, kemudian menekan tombol “Buka Kunci”	Kunci: 251247 (salah)	Sistem akan menolak akses dan menampilkan pesan kalau kunci yang dimasukan salah.	✔	Berhasil
3	Mengisi field kunci dengan kunci yang benar, kemudian menekan tombol “Buka Kunci”	Kunci: 138725205973429859411 (benar)	Sistem akan merespon kunci kemudian menampilkan halaman perhitungan suara.	✔	Berhasil

3. Tampilan Porgram

Tampilan program untuk peserta dapat memilih sesuai dengan hak pilihnya. Program ini dibuat dengan menggunakan pwa, dimana dapat digunakan dalam berbagai platform dengan menggunakan web browser.

Gambar 3.4 Perbandingan dari entropy dari setiap data.

Gambar 3.5 Perbandingan dari entropy dari setiap data.

Gambar 3.6 Perbandingan dari entropy dari setiap data.

Gambar 3.7 Tampilan untuk membuka hasil pemilihan

Gambar 3.8 Halaman pemilihan kandidat presiden

4. Kesimpulan

Berdasarkan penelitian tentang Implementasi Kriptografi Dalam Pengamanan Database e-voting menggunakan Algoritma RSA dan Base64 Berbasis Progresive Web Apps, dapat ditarik kesimpulan

1. Dari penelitian ini telah dihasilkan aplikasi e-voting dengan berbasis *progresive web apps* sehingga dapat digunakan pada *website* dan *mobile*.

2. Solusi untuk menggunakan enkripsi pada database dalam meningkatkan keamanan dan kerahasiaan data dengan menggunakan algoritma RSA dan base64, sehingga keamanan lebih terjamin.
3. Berdasarkan hasil pengujian dan validasi algoritma telah didapat bahwa kualitas keamanan dari penggabungan algoritma RSA dan base64 sangat baik karena rata-rata waktu proses adalah 0,9544 milidetik, sehingga dikatakan sangat baik karena waktu yang didapat kurang dari 1 detik.
4. Dengan hasil penelitian yang telah dilakukan dapat diketahui bahwa tingkat keamanan pada penyimpanan data e-voting terbilang aman, dengan melihat rata-rata korelasi adalah 0,0503 dimana angka korelasi ini terbilang lemah (bagus) dengan dasar teori korelasi product moment yaitu 0,00 – 0,19 merupakan hasil korelasi yang sangat lemah. Sehingga dapat disimpulkan bahwa hubungan antara plainteks dan cipherteks sangat berbeda. Rata-rata entrophy dari plainteks adalah 2,60360198 dan rata-rata entrophy cipherteks adalah 3,089234653 sehingga dapat disimpulkan bahwa terjadi perubahan dari plainteks ke chiperteks yang cukup signifikan.

5. Saran

Penulis menyadari masih banyak kekurangan dari penelitian ini, penulis memberikan beberapa saran untuk penelitian selanjutnya yang berkaitan dengan penelitian ini, yaitu:

1. Penelitian ini menggunakan sistem berbasis web yang dapat digunakan pada platform web dan mobile, namun belum bisa terinstall secara permanen, karena tidak berbentuk aplikasi android
2. Kombinasi algoritma RSA dan base64 sudah cukup baik, namun dapat dicoba menggunakan kombinasi algoritma yang lain untuk mendapat hasil yang lebih baik lagi.
3. Kesesuaian perhitungan korelasi dan entorphy secara manual dengan penerapan kode program yang dapat dibuktikan secara tertulis.
4. Perhitungan korelasi mempunyai banyak jenis dan rumus perhitungan, sehingga penggunaan rumus korelasi yang tepat dapat menghasilkan nilai korelasi paling kecil dan sesuai dengan permasalahan.

6. Daftar Pustaka

[1] “1. MODEL SISTEM ELECTRONIC VOTING (E-VOTING) BERBASIS WEB DENGAN MENERAPKAN QUICK RESPONSE CODE (QR-CODE) SEBAGAI SISTEM KEAMANAN DALAM PEMILIHAN LEGISLATIF.pdf.”.

[2] E. W. Rumaf, “IMPLEMENTASI ALGORITMA BLOWFISH UNTUK PRIVACY DATA E-VOTING,” p. 7.

[3] M. T. A. Zaen and R. Putra, “APLIKASI VOTING PEMILIHAN KETUA ORGANISASI SISWA INTRA SEKOLAH (OSIS) PADA MA NURUL IHSAN NW TILAWAH BERBASIS WEB,” J. Manaj. Inform. Dan Sist. Inf., vol. 1, no. 2, p. 43, Aug. 2018, doi: 10.36595/misi.v1i2.48.

[4] S. Suhandinata, R. A. Rizal, and D. OngkyWijaya, “ANALISIS PERFORMA KRIPTOGRAFI HYBRID ALGORITMA BLOWFISH DAN ALGORITMA RSA,” J. Teknol. Dan Sist. Inf., no. 1, p. 11, 2019.

[5] M. Ridwan and Z. Arifin, “RANCANG BANGUN E-VOTING DENGAN MENGGUNAKAN KEAMANAN ALGORITMA RIVEST SHAMIR ADLEMAN (RSA) BERBASIS WEB (STUDI KASUS: PEMILIHAN KETUA BEM FMIPA),” p. 7.

[6] J. J. S. Thehok, “MODEL MODIFIKASI KRIPTOGRAFI ALGORITMA RSA UNTUK KEAMANAN DATA PADA DATABASE E-VOTING,” vol. 11, no. 2, p. 15, 2017.

[7] Harma Oktafia Lingga Wijaya, Ed., “E-Voting Berbasis Website Pada Pemilihan Kades Di Rantau Jaya (Lake) Dengan Keamanan Data Menggunakan Enkripsi Base 64,” Jurasik J. Ris. Sist. Inf. Dan Tek. Inform., vol. 2, no. 1, p. 48, Jul. 2017, doi: 10.30645/jurasik.v2i1.18.

[8] E. Gunadhi, A. P. Nugraha, and Sekolah Tinggi Teknologi Garut, “Penerapan Kriptografi Base64 Untuk Keamanan URL (Uniform Resource Locator) Website Dari Serangan SQL Injection,” J. Algoritma, vol. 13, no. 2, pp. 391–398, Feb. 2017, doi: 10.33364/algoritma/v.13-2.391.

[9] “Implementasi progressive web app sebagai solusi untuk meningkatkan kinerja aplikasi berbasis website.pdf.”.

[10] Betha Sidik, Javascript. Informatika Bandung.

[11] Al-Bahra Bin Ladjamudin, Rekayasa Perangkat Lunak. Graha Ilmu.
[12] Rosa A.S M. Salahuddin, Rekayasa Perangkat Lunak Terstruktur dan Berorietasi Objek. Informatika Bandung.
[13] Iqbal Hasan, Pokok-Pokok Materi Statistik. Bumi Aksara, 1999.