Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK) DOI: 10.25126/jtiik.20241117448
Vol. 11, No. 1, Februari 2024, hlm. 85-94 p-ISSN: 2355-7699
Akreditasi KEMENRISTEKDIKTI, No. 36/E/KPT/2019 e-ISSN: 2528-6579

PENGAMANAN TEKS DENGAN KOMBINASI METODE *ELECTRONIC CODE BOOK* (ECB) DAN KODE *SEVEN SEGMENT DISPLAY*

Kiswara Agung Santoso[*1], Erick Delenia[2], Agustina Pradjaningsih[3],

[1,2,3]Universitas Jember
Email: kiswara.fmipa@unej.ac.id, [2]erickdelenia08@gmail.com, [3]agustina.fmipa@unej.ac.id
*Penulis Korespondensi

(Naskah masuk: 6 Juli 2023, diterima untuk diterbitkan: 3 Februari 2024)

Abstrak

Layanan media sosial merupakan salah satu contoh perkembangan teknologi di bidang informasi. Salah satu cara meningkatkan sistem keamanan yaitu digunakannya ilmu kriptografi. Sudah banyak penelitian yang menghasilkan algoritma kriptografi, mulai dari algoritma yang baru, modifikasi algoritma bahkan kombinasi dari beberapa algoritma. Berdasarkan kebiasaan algoritma tersebut hacker tentu akan mencari celah untuk melakukan dekripsi dengan cara mencari algoritma dasar dari proses enkripsi untuk kemudian melakukan hack. Untuk mengantisipasi hal tersebut peneliti ingin melakukan modifikasi bukan pada algoritma pembentuknya melainkan modifikasi dari konversi system bilangan basis 2 berdasarkan *Seven Segment Display.* Salah satu metode yang sering digunakan untuk proses enkripsi yaitu metode *Electronic Code Book* (ECB). *Seven Segment Display* merupakan sebuah tampilan yang terbentuk dari tujuh kelompok segmen LED (*Light Emitting Diode*) yang dirangkai sedemikian sehingga membentuk angka–angka dari 0 hingga 9. Segmen LED dinotasikan dengan 1 jika menyala dan 0 jika mati, pola tersebut dapat digunakan untuk memanipulasi bit biner 7 bit khususnya karakter angka 0 hingga 9. Modifikasi terletak pada proses pembentukan bit kunci yang dibangkitkan berdasarkan aturan *Seven Segment Display.* Aturan ini digunakan untuk mengganti nilai bit, bila pada umumnya nilai bit didapat dari sistem bilangan basis 2 maka disini nilai bit didapat dari aturan *seven segment* display dan inilah yang merupakan *state of the art* dari penelitian ini karena belum pernah digunakan sebelumnya. Hasil penerapannya menunjukkan bahwa data yang dienkripsi menghasilkan chiperteks acak berupa karakter *printable* pada ASCII dan chiperteks dapat dikembalikan secara utuh tanpa ada informasi yang hilang.

Kata kunci: *kriptografi, pengamanan teks, ECB, electronic code book, seven segment display*

TEXT SECURITY WITH A COMBINATION OF ELECTRONIC CODE BOOK(ECB) METHOD AND SEVENT SEGMENT DISPLAY CODE

Abstract

Social media services are an example of technological developments in the information sector. One way to improve the security system is to use cryptography. A lot of research has produced cryptographic algorithms, starting from new algorithms, algorithm modifications, and even combinations of several algorithms. Based on these algorithm habits, hackers will look for loopholes to carry out decryption by looking for the basic algorithm of the encryption process and then hacking. To anticipate this, the researchers want to make modifications not to the forming algorithm but to modify the conversion of the base 2 number system based on the Seven Segment Display. One method that is often used for the encryption process is the Electronic Code Book (ECB) method. Seven Segment Display is a display formed from seven groups of LED (Light Emitting Diode) segments which are arranged in such a way as to form numbers from 0 to 9. LED segments are denoted by 1 if they are on and 0 if they are off, this pattern can be used to manipulate bits of 7-bit binary, especially the character numbers 0 to 9. Modification lies in the process of forming key bits which are generated based on the Seven Segment Display rules. This rule is used to replace bit values. If in general the bit value is obtained from the base 2 number system then here the bit value is obtained from the seven-segment display rule and this is the state of the art of this research because it has never been used before. The results of its application show that the encrypted data produces random ciphertext in the form of printable characters in ASCII and the ciphertext can be returned intact without any information being lost.

Keywords: *cryptography, text security, ECB, electronic code book, seven segment display, data security, social media service*

1. PENDAHULUAN

Perkembangan teknologi yang semakin pesat berdampak pada berbagai aspek kehidupan manusia, salah satunya dalam bidang informasi. Perkembangan teknologi informasi ini membawa paradigma baru bagi masyarakat luas dalam menjalankan kehidupan sehari-hari. Teknologi informasi memberikan cara baru dalam berkomunikasi, dimana manusia memungkinkan untuk saling berkomunikasi atau bahkan bertatap muka meski berada dalam dua belahan dunia yang berbeda (Kaunang, dkk., 2021). Layanan media sosial merupakan salah satu contoh perkembangan teknologi di bidang informasi. Melalui layanan media sosial seperti Facebook, Instagram, Twitter, Whatsapp dan lain sebagainya, informasi beredar dengan sangat mudah dan cepat. Layanan tersebut dapat diakses di manapun dan kapan saja hanya dengan menggunakan sebuah smartphone yang hampir semua punya. Layanan media sosial umunya berisikan informasi publik dan pribadi, termasuk alamat email, nama, username, jumlah follower, tanggal pembuatan akun, nomor telepon, password akun dan lain sebagainya. Data-data tersebut memang bisa diakses secara publik oleh semua pengguna, tetapi tidak untuk nomor telepon, alamat email ataupun password akun yang merupakan informasi bersifat pribadi. Layanan media sosial tentu harus mampu menjamin keamanan dan kerahasiaan informasi pribadi agar tidak diakses atau diretas pihak yang tidak memiliki hak akses, sehingga informasi tersebut tidak disalahgunakan pihak lain.

Sistem keamanan diperlukan untuk menjaga informasi yang bersifat pribadi. Salah satu sistem keamanan yang sering digunakan adalah kriptografi. Kriptografi adalah ilmu dan seni dalam mengamankan atau merahasiakan sebuah pesan dengan menyandikannya ke bentuk yang tidak dapat dipahami maknanya. Algoritma kriptografi terdiri dari 3 fungsi dasar, yaitu enkripsi, dekripsi, dan kunci. Enkripsi ialah proses perubahan pesan asli (plainteks) menjadi kode-kode yang tidak dimengerti. Dekripsi merupakan kebalikan dari enkripsi yaitu pengembalian pesan yang telah dienkripsi ke bentuk pesan asli (plainteks). Kunci ialah kunci yang akan digunakan pada proses enkripsi dan dekripsi (Ariyus, 2008).

Metode *Electronic Code Book* (ECB) merupakan salah satu jenis algoritma kriptografi. Metode ini membagi plainteks menjadi blok-blok yang independen, sehingga kerusakan pada satu blok tidak akan mempengaruhi blok lainnya. Sifat dasar metode ECB yaitu blok plainteks yang sama selalu dienkripsi menjadi chiperteks yang sama pula (Simarmata, dkk., 2008). Penelitian oleh Mufid (2010), menjelaskan bahwa metode ECB memiliki kekurangan ketika karakter pada plainteks sering berulang akan menghasilkan enkripsi cipherteks yang sama. Ariandi et al (2020) dalam penelitiannya berkesimpulan bahwa metode ECB dapat membantu mengamankan data pegawai pada PDAM Tirta Sanita Sumber dengan baik. Arnawa dkk (2020) melakukan penelitian dengan membandingkan waktu enkripsi antara metode ECB dan *Chipher Block Chaining* (CBC) dalam algoritma *Blowfish* dan menghasilkan kesimpulan bahwa metode ECB membutuhkan waktu yang lebih cepat dalam melakukan enkripsi teks dibanding metode CBC. Penelitian oleh Syahputra (2022) yang mengombinasikan metode *Vigenere Chiper* dan ECB, berkesimpulan bahwa teknik pengamanan data dengan gabungan dua metode tersebut dapat dapat meningkatkan keamanan, karena kompleksitasnya yang jauh lebih rumit dibandingkan dengan menggunakan satu metode. Widarma dkk. (2019) dalam penelitiannya mengkombinasikan algoritma *Vigenere Chiper* dengan *Electronic Code Book* (ECB), cara terebut dapat meningkatkan keamanan karena kompleksitas dua algoritma menghasilkan chiperteks yang jauh lebih rumit namun membutuhkan waktu yang lebih lama. Putra dan Yuliani (2019) juga melakukan perancangan dan pengujian perangkat lunak dengan menggabung dua algoritma yaitu *Playfair Chiper* dengan *Electronic Code Book* (ECB), tujuan penggabungan tersebut yaitu untuk memperkuat penyandian klasik dengan adanya metode modern *block.* Karo (2020) dalam penelitiannya juga menggabungkan dua algoritma *Affine Chiper* dan algoritma *Electronic Code Book* (ECB) dalam pengamanan pesan teks, hasilnya gabungan dua algoritma tersebut dapat diterapkan pada pengamanan pesan. Masalah baru akan muncul, seiring kemajuan teknologi dan semakin banyak cara untuk memecahkan metode ini pada penerapannya dalam pengamanan data. Untuk itu salah satu cara untuk membuat data atau informasi menjadi lebih aman adalah dengan membuat pola atau metode baru dengan melakukan modifikasi pada metode ECB.

Seven Segment Display merupakan sebuah tampilan yang terbentuk dari tujuh kelompok segmen LED (*Light Emitting Diode*) yang dirangkai sedemikian sehingga membentuk angka–angka dari 0 hingga 9. Ketika segmen dalam tampilan dihidupkan dengan kombinasi yang tepat, salah satu dari angka 0 hingga 9 dapat ditampilkan (Misra, 2022). Penelitian yang dilakukan oleh Esnawan dan Antarnusa (2019), yaitu tentang sistem penskoran olahraga menggunakan *Seven Segment Display* dapat menggantikan papan skor manual sehingga dapat mempermudah proses pencatatan skor pada pertandingan olahraga. Penelitian serupa juga dilakukan oleh Surahmat dan Fu'ady (2020), menghasilkan kesimpulan bahwa *Seven Segment Display* dapat menampilkan sistem angka desimal dan dapat menjadi alternatif lain dari layar *dot-matrix* yang terdiri dari LED. Pola pada *Seven Segment Display* dapat diterapkan untuk melakukan modifikasi pada metode ECB. *Seven Segment Display* yang terdiri atas 7 segmen LED dinotasikan dengan 1 jika menyala dan 0 jika mati, pola tersebut dapat

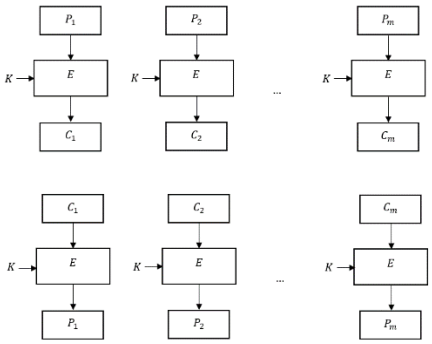
menggantikan bilangan biner 7 bit khususnya karakter angka 0 hingga 9. Modifikasi terletak pada proses pembentukan bit yang dibangkitkan berdasarkan aturan *Seven Segment Display* atau dengan kata lain *Seven Segment Display* digunakan untuk memanipulasi bit biner.

Berdasarkan latar belakang masalah tersebut, maka dilakukan penelitian tentang pengamanan teks menggunakan metode ECB yang dikombinasikan dengan kode *Seven Segment Display*. Perancangan metode dengan mengombinasikan metode ECB dan *Seven Segment Display* bertujuan untuk menciptakan suatu metode baru hasil modifikasi yang dapat membantu memperbaiki dan memperbarui metode yang ada agar lebih bervariatif.

2. PERSAMAAN MATEMATIKA

2.1. Metode Electronic Code Book

Munir (2019) menyatakan bahwa pada metode ini, plainteks P dibagi menjadi blok-blok, dimana setiap blok P_i dienkripsi secara individual dan independen menjadi blok chiperteks C_i. Gambar 1 merupakan skema enkripsi m buah blok plainteks $P_1 \dots P_m$ dan dekripsi m buah blok chiperteks $C_1 \dots C_m$. Fungsi E dan D masing masing menyatakan fungsi enkripsi dan dekripsi sedangkan *K* merupakan kunci yang digunakan pada proses enkripsi dan dekripsi.



Gambar 1. Skema Enkripsi dan Dekripsi Metode ECB (Munir,2019)

Metode ECB yang akan dilakukan adalah dengan meng-*XOR*-kan blok plainteks P_i dengan K, kemudian hasilnya di geser secara *wrapping* satu bit ke kiri. *Wrapping* merupakan pergeseran bit secara sirkular artinya bit digeser ke arah kiri dengan bit paling kiri akan terlempar ke ujung kanan lainnya. Proses enkripsi pada plainteks $E_k(P)$ sederhana tersebut dinotasikan sebagai berikut:

$$E_k(P) = (P \oplus K) \ll 1 \qquad (1)$$

dengan:
$\oplus$: operasi logika XOR
$\ll$: pergeseran bit ke kiri
P : plainteks
K : kunci
E_k: fungsi enkripsi

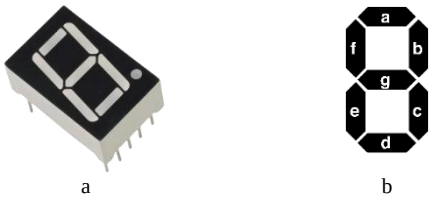
Proses dekripsi menggunakan fungsi D atau kebalikan dari fungsi E. Proses tersebut dimulai dengan menggeser bit tiap blok secara *wrapping* 1 bit ke kanan, kemudian dilakukan operasi XOR blok P_i dengan kunci K. Proses dekripsi $D_k(C)$ dinotasikan sebagai berikut:

$$D_k(C) = (C \gg 1) \oplus K \qquad (2)$$

dengan:
$\oplus$: operasi logika XOR
$\gg$: pergeseran bit ke kanan
C : chiperteks
K : kunci
D_k: fungsi dekripsi

2.2. Seven Segment Display

Menurut Misra (2022), *Seven Segment Display* merupakan sebuah perangkat digital yang biasanya digunakan untuk menampilkan angka dari 0 sampai 9. Dalam beberapa kasus, *Seven Segment Display* juga dapat menampilkan beberapa huruf alfabet, namun fungsi utamanya biasanya untuk menampilkan angka. Sebuah seven segment display ditunjukkan pada Gambar 2(a). *Seven Segment Display* memilililiki tujuh buah LED (light emitting diode) yang merepresentasikan segmen-segmennya. Segmen pada *Seven Segment Display* ditandai dengan notasi a hingga g yang ditunjukkan pada Gambar 2(b).

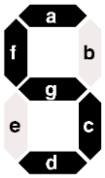


a (a) Tampilan *seven segment display*

b (b) Tampilan segmen

Gambar 2. Tampilan *Seven Segment Display*

Ketika segmen-segmen dihidupkan dengan kombinasi yang tepat, salah satu dari angka 0 sampai 9 dapat ditampilkan. Contoh dalam kasus untuk menampilkan angka 5 seperti pada Gambar 3, maka segment a, c, d, f, dan g menyala dan segmen b dan e tidak menyala.



Gambar 3. Tampilan Angka 5 pada *Seven Segment Display*

Seven Segment Display tersedia di pasar dalam bentuk dua tipe yaitu Common Cathode (CC) dan juga Common Anode (CA). Untuk menghidupkan

segmen pada tipe-CA (Common Anode), kita harus mengirimkan 5 V (atau biner 1) ke segmen tersebut, dan untuk mematikannya, kita harus mengirim 0 V (atau biner 0) ke segmen tersebut. Contoh kode untuk menampilkan angka 5 ditunjukkan pada Tabel 1.

Tabel 1. Kode untuk menampilkan angka 5 pada *seven segment display*

	Tampilan Segmen							Angka yang Tampil
	a	b	c	d	e	f	g	
Tingkat Tegangan (V)	5	0	5	5	0	5	5	5
Nilai Biner	1	0	1	1	0	1	1	

Adapun kode-kode untuk menampilkan angka lain dari 0 hingga 9 pada *seven segment display* ditunjukkan pada Tabel 2.

Tabel 2. Kode untuk Menampilkan Angka pada *Seven Segment Display*

Tampilan Segmen							Angka yang Tampil
a	b	c	d	e	f	g	
1	1	1	1	1	1	0	0
0	1	1	0	0	0	0	1
1	1	0	1	1	0	1	2
1	1	1	1	0	0	1	3
0	1	1	0	0	1	1	4
1	0	1	1	0	1	1	5
1	0	1	1	1	1	1	6
1	1	1	0	0	0	0	7
1	1	1	1	1	1	1	8
1	1	1	0	0	1	1	9

3. METODE PENELITIAN

Data yang digunakan pada penelitian ini terbagai menjadi dua yaitu data yang digunakan sebagai plainteks dan data yang digunakan sebagai kunci, dimana keduanya berupa teks. Teks tersebut terdiri kumpulan karakter dari mulai huruf, angka, tanda baca dan lain sebagainya yang tidak dibatasi jenisnya. Data yang digunakan sebagai plainteks merepresentasikan sebuah informasi pribadi (email, nomor hp, password akun dan lain sebagainya) yang akan diamankan menggunakan algoritma kriptografi yang diusulkan penulis. Data yang digunakan sebagai kunci akan dibangkitkan secara acak tidak ada kaitannya dengan data yang digunakan sebagai plainteks, namun keduanya akan saling berkaitan pada proses pengamanan data plainteks. Adapun tahapan penelitian yang akan dilakukan pada penelitian ini adalah sebagai berikut.

3.1. Tahap Enkripsi

Data (plainteks dan kunci) diolah menggunakan metode ECB yang dikombinasikan dengan kode *Seven Segment Display*, sehingga akan menghasilkan chiperteks (data yang tidak dapat dikenali).

Uraian mengenai tahapan enkripsi adalah sebagai berikut:

1. Konversi Kunci (K) ke Bentuk Kunci Desimal (KD) berdasarkan Kode ASCII
   Tiap karakter pada kunci (K) dikonversi ke bentuk desimal berdasarkan aturan kode ASCII. Hasil konversi menghasilkan kunci desimal (KD)
2. Konversi Kunci Desimal (KD) ke Bentuk Kunci Biner (KB) berdasarksan Kode *Seven Segment Display*.
   Ubah tiap digit desimal kunci desimal (KD) ke bentuk kunci biner (KB) berdasarkan aturan kode *Seven Segment Display*. Aturan tersebut dapat dilihat pada Tabel 3.

Tabel 3. Tabel konversi desimal-biner pada aturan seven ssegment display

Bentuk Biner	Bentuk Desimal
1111110	0
0110000	1
1101101	2
1111001	3
0110011	4
1011011	5
1011111	6
1110000	7
1111111	8
1110011	9

3. Konversi Plainteks (P) ke Bentuk Plainteks Desimal (PD) berdasarkan Kode ASCII
   Tiap karakter pada plainteks (P) dikonversi ke bentuk desimal berdasarkan aturan kode ASCII. Hasil konversi menghasilkan plainteks desimal (PD)
4. Konversi Plainteks Desimal (PD) ke Bentuk Plainteks Biner (PB)
   Tiap desimal pada plainteks desimal (PD) dikonversi ke bentuk biner 7 bit, sehingga menghasilkan plainteks biner (PB).
5. Proses Enkripsi Metode ECB
   Kunci biner (KB) dan plainteks biner (PB) diproses pada tahap enkripsi menggunakan metode seperti pada persamaan (5).
   Pada persamaan tersebut terdapat dua proses yaitu
   a. Proses XOR
      Proses XOR tiap blok biner PB dengan KB menghasilkan chiperteks biner (CB).
   b. Pergeseran Bit
      Pergeseran bit dilakukan di tiap blok biner pada chiperteks biner (CB) dengan menggeser bit sejauh 1 bit ke kiri, bit paling kiri akan terlempar ke posisi bit paling kanan.
6. Konversi Chiperteks Biner (CB) ke Bentuk Chiperteks Desimal (CD)
   Konversi tiap blok biner pada chiperteks biner (CB) ke bentuk desimal berdasarkan persamaan (2). Hasil konversi menghasilkan chiperteks desimal (CD)
7. Konversi Chiperteks Desimal (CD) ke bentuk Karakter Chiperteks (C)

Ubah tiap desimal pada chiperteks desimal (CD) ke bentuk karakter, sehingga menghasilkan chiperteks (C).

3.2. Tahap Dekripsi

Data (chiperteks dan kunci) diolah pada tahap dekripsi untuk mengembalikan data (chiperteks) ke bentuk semula (plainteks) menggunakan metode ECB yang dikombinasikan dengan kode.
Uraian mengenai tahapan enkripsi adalah sebagai berikut:

1. Konversi Kunci (K) ke Bentuk Kunci Desimal (KD) berdasarkan Kode ASCII
   Tiap karakter pada kunci (K) dikonversi ke bentuk desimal berdasarkan aturan kode ASCII. Hasil konversi menghasilkan kunci desimal (KD)
2. Konversi Kunci Desimal (KD) ke Bentuk Kunci Biner (KB) berdasarksan Kode *Seven Segment Display*.
   Ubah tiap digit desimal kunci desimal (KD) ke bentuk kunci biner (KB) berdasarkan aturan kode *Seven Segment Display*. Aturan tersebut dapat dilihat pada Tabel 3.
3. Konversi Plainteks (P) ke Bentuk Plainteks Desimal (PD) berdasarkan Kode ASCII
   Tiap karakter pada chiperteks (C) kemudian dikonversi ke bentuk desimal berdasarkan kode ASCII, sehingga hasil konversi menghasilkan chiperteks desimal (CD)
4. Konversi Chiperteks Desimal (CD) ke Bentuk Chiperteks Biner (CB)
   Tiap desimal pada chiperteks desimal (CD) dikonversi ke bentuk biner 7 bit, sehingga menghasilkan chiperteks biner (CB).
5. Proses Enkripsi Metode ECB
   Kunci biner (KB) dan chiperteks biner (CB) diproses pada tahap dekripsi menggunakan algoritma seperti pada persamaan (6). Pada persamaan tersebut terdapat dua proses yaitu
   a. Pergeseran bit
      Pergeseran bit dilakukan di tiap blok biner pada chiperteks biner (CB) dengan menggeser sejauh 1 bit ke kanan, bit yang paling kanan akan terlempar ke posisi paling kiri.
   b. Proses XOR
      Proses XOR tiap blok biner CB dengan KB menghasilkan plainteks biner (PB).
6. Konversi Plainteks Biner (PB) ke Bentuk Plainteks Desimal (PD)
   Konversi tiap blok biner pada plainteks biner (PB) ke bentuk. Hasil konversi menghasilkan plainteks desimal (PD)
7. Konversi Plainteks Desimal (PD) ke bentuk Karakter Plainteks (P)
   Ubah tiap desimal pada plainteks desimal (PD) ke bentuk karakter, sehingga menghasilkan plainteks (P).

3.3. Pembuatan Program

Program dibuat dengan menggunakan framework Flutter. Flutter sendiri merupakan teknologi yang dibuat oleh google, bersifat open source yang bertujuan untuk membuat aplikasi pada perangkat mobile (Android atau iOS). Flutter dikembangkan dengan menggunakan bahasa Dart, dan saat ini telah dikembangkan untuk aplikasi multiplatform baik mobile, website maupun dekstop (Ridwan dan Bustami, 2021). Pembuatan program dimulai dari menyusun code/skrip bagian *back-end* berdasarkan metode yang diajukan penulis yaitu kombinasi metode ECB dan kode *Seven Segement Display*, kemudian dilanjutkan dengan membuat tampilan antarmuka (*user interface*) mulai dari background, tombol, label (*Text*), input text (*TextField*) dan lain sebagainya. Program dirancang dengan menggunakan Visual Studio Code sebagai IDE (*Integrated Development Environment*).

Analisis Hasil dan Kesimpulan

Analisis hasil dilakukan dengan tujuan untuk memvalidasi hasil yang diperoleh apakah sudah sesuai dengan konsep yang dirancang. Kesimpulan dari penelitian diperoleh dari hasil analisis proses pengamanan teks menggunakan metode yang telah diusulkan baik pada tahap enkripsi maupun tahap dekripsi. Penelitian dikatakan berhasil jika data (plainteks) dapat diubah menjadi bentuk yang tidak dikenali (chiperteks) melalui tahapan enkripsi dan dapat dikembalikan kembali ke bentuk semula (plainteks) melalui tapahan dekripsi.

4. HASIL DAN PEMBAHASAN

Berikut data yang digunakan:
Plainteks : erick@gmail.com
Kunci : 79Jbr

Data tersebut disimulasikan untuk dilakukan pengamanan dimulai dari tahap enkripsi dimana pada tahap tersebut plainteks dan kunci akan diproses sehingga menghasilkan chiperteks kemudian chiperteks tersebut akan digunakan sebagai input bersamaan dengan kunci ketika proses dekripsi untuk dikembalikan ke bentuk semula (plainteks). Tahapan tersebut akan dipaparkan sebagai berikut:

4.1. Tahap Enkripsi

Langkah-langkah pada tahap enkripsi adalah sebagai berikut:

1. Konversi Kunci (K) Ke Bentuk Kunci Desimal (KD) berdasarkan kode ASCII
   Karakter 1 sampai dengan karakter 5 masing-masing akan dikonversi ke bentuk desimal, dimulai dari karakter ke-1 atau yang dinotasikan

K_1 yaitu "7", berdasarkan aturan kode ASCII, karakter "7" memiliki desimal 55, berlaku juga untuk K_2 hingga K_5 , sehingga keseluruhan kunci desimal (KD) adalah sebagai berikut:

KD : {55,57,74,98,114}

2. Konversi Kunci Desimal (KD) ke Bentuk Kunci Biner (KB) berdasarkan Kode *Seven Segment Display*

Kunci desimal (KD) kemudian dipecah tiap digitnya, dari mulai $KD_1 = 55$, dipecah menjadi {5,5}, begitupun untuk KD_2, hingga KD_5, sehingga kunci desimal (KD) yang baru adalah sebagai berikut:

KD : {5,5,5,7,7,4,9,8,1,1,4}

Kemudian kunci desimal (KD) dikonversi ke bentuk kunci biner (KB), dimulai dari $KD_1 = 5$, jika dilihat pada Tabel 3 maka konversi binernya adalah 1011011, berlaku juga untuk KD_2 hingga KD_{11} , sehingga keseluruhan kunci biner (KB) adalah sebagai berikut:

(KB) : { 1011011, 1011011, 1011011, 1110000, 1110000, 0110011, 1110011, 1111111, 0110000, 0110000, 0110011 }

3. Konversi Plainteks (P) ke Bentuk Plainteks Desimal (PD) berdasarkan Kode ASCII

Tiap karakter pada plainteks (P) dikonversi ke bentuk desimal dimulai dari karakter ke-1 atau yang dinotasikan P_1 yaitu "e", berdasarkan aturan kode ASCII, karakter "e" memiliki desimal 101, berlaku juga untuk P_2 hingga P_{15} , sehingga keseluruhan plainteks desimal (PD) adalah sebagai berikut:

PD : {101, 114, 105, 99, 107, 64, 103, 109, 97, 105, 108, 46, 99, 111, 109 }

4. Konversi Plainteks Desimal (PD) ke bentuk Plainteks Biner (PB)

Plainteks desimal (PD) kemudian di konversi ke bentuk plainteks biner (PB) dengan panjang 7 bit, dimulai dari $PD_1 = 101$, bentuk binernya adalah 1100101, sehingga $PB_1 = 1100101$. Begitupun untuk PD_2 hingga PD_{15} , sehingga keseluruhan plainteks biner (PB) adalah sebagai berikut:

PB: {1100101, 1110010, 1101001, 1100011, 1101011, 1000000, 1100111, 1101101, 1100001, 1101001, 1101100, 0101110, 1100011, 1101111, 1101101}

5. Proses Enkripsi Metode ECB

Plainteks biner (PB) dan kunci biner (KB) kemudian di proses pada inti metode ECB.

a. Proses XOR

Proses XOR antara plainteks biner (PB) dengan kunci biner (KB) menghasilkan chiperteks biner (CB). Plainteks biner (PB) dan kunci biner (KB), masing masing memilki panjang 15 dan 11. Panjang yang dimaksud adalah banyak blok biner. Proses XOR dirumuskan seperti pada Persamaan (3).

$$CB_i = PB_i \oplus KB_{(i-1 \bmod m)+1} \quad (3)$$

dengan i menyatakan indeks pada PB yang bergerak dari 1 hingga sepanjang PB, dan m menyatakan panjang kunci (K).

Berikut perhitungan untuk mencari chiperteks biner (CB), mulai dari CB_1, berdasarkan Persamaan (3) maka CB_1 dihasilkan dari proses XOR antara PB_1 dengan $KB_{(1-1 \bmod 11)+1}$.

$CB_1 = PB_1 \oplus KB_1$

PB_1 : 1100101

KB_1 : 1011011

0111110 ⊕

$CB_1 = 0111110$.

Perhitungan di atas berlaku juga untuk menghitung CB_2 hingga CB_{15}, sehingga keseluruhan dihasilkan chiperteks biner (CB) sebagai berikut:

CB: {0111110, 0101001, 0110010, 0010011, 0011011, 1110011, 0010100, 0010010, 1010001, 1011001, 1011111, 1110101, 0111000, 0110100, 0011101}

b. Pergeseran Bit

Pergeseran bit dilakukan dengan menggeser sejauh 1 bit ke kiri, bit yang paling kiri akan terlempar ke posisi paling kanan, dilakukan dari mulai blok chiperteks biner pertama (CB_1) hingga terakhir (CB_{15}).

Untuk CB_1

Bentuk biner = 0111110

Hasil Pergeseran Bit = 1111100

sehingga keseluruhan chiperteks biner (CB) yang baru yaitu:

CB: {1111100, 1010010, 1100100, 0100110, 0110110, 1100111, 0101000, 0100100, 0100011, 0110011, 0111111, 1101011, 1110000, 1101000, 0111010}

6. Konversi Chiperteks biner (CB) ke Bentuk Chiperteks Desimal (CD)

Chiperteks biner (CB) kemudian diubah ke bentuk desimal (CD), mulai dari CB_1 hingga CB_{15}.

Untuk $CB_1 = 1111100$

Bentuk desimal CB_1 yaitu 124, sehingga $CD_1 = 124$, berlaku juga untuk CD_2 hingga CD_{15} ,sehingga diperoleh chiperteks desimal (CD) sebagai berikut:

CD: {124, 82, 100, 38, 54, 103, 40, 36, 35, 51, 63, 107, 112, 104, 58}

7. Konversi Chiperteks Desimal (CD) ke bentuk Chiperteks Karakter (C)

Mula-mula tiap chiperteks desimal (CD) akan dicari nilai k, dan s dari Persamaan (4) yaitu:

$$CD_i = 95k + s \quad (4)$$

dengan i menyatakan indeks yang bergerak dari 1 hingga sebanyak desimal pada chiperteks desimal (CD). Nilai k dan s akan menentukan hasil konversi, berikut ketentuannya:

a. Untuk $k = 0$ terdapat dua kasus:
   1. Jika $s + 32 \in [48,57]$, maka hasil konversi dirumuskan seperti pada Persamaan (5)

C_i= k digabung dengan karakter ke- $(s + 32)$ pada ASCII (5)

2. Jika $s + 32 \in [48,57]$, maka hasil konversi dirumuskan seperti pada Persamaan (6)

C_i=karakter ke$-$ $(s + 32)$ (6)

b. Untuk $k \neq 0$, maka hasil konversi dirumuskan seperti pada Persamaan (7).

C_i= bilangan acak (1-9) digabung dengan karakter ke- $(s + 32)$ pada ASCII (7)

Untuk $CD_1 = 124$:
$CD_1 = 95k + s$

$124 = 95 * 1 + 29$

Maka $k = 1$ dan $s = 29$, $s + 32 = 61$,
Karena k bernilai 1, maka CD_1 dikonversi dengan Persamaan (7) yaitu sebarang bilangan acak (1-9) digabung dengan karakter ke-61 pada ASCII, dalam hal ini misalkan angka acak yang dipilih adalah 6 dan karakter ke-61 pada ASCII adalah " = ", konversinya menghasilkan " 6= "
Dilanjutkan untuk CD_2 hingga CD_{15}, sehingga hasil konversi keseluruhan adalah sebagai berikut:

Chiperteks (C) : 6=r7%FV3(HDCS_3,118)Z

$95 + \mathrm{CD_{i+1}} - 32$ (9)

4.2. Tahap Dekripsi

Langkah-langkah pada tahap enkripsi adalah sebagai berikut:

1. Konversi Kunci (K) Ke Bentuk Kunci Desimal (KD) berdasarkan kode ASCII

Karakter 1 sampai dengan karakter 5 masing-masing akan dikonversi ke bentuk desimal, dimulai dari karakter ke-1 atau yang dinotasikan K_1 yaitu "7", berdasarkan aturan kode ASCII, karakter "7" memiliki desimal 55, berlaku juga untuk K_2 hingga K_5, sehingga keseluruhan kunci desimal (KD) adalah sebagai berikut:
KD : {55,57,74,98,114}

2. Konversi Kunci Desimal (KD) ke Bentuk Kunci Biner (KB) berdasarkan Kode *Seven Segment Display*

Kunci desimal (KD) kemudian dipecah tiap digitnya, dari mulai $KD_1 = 55$, dipecah menjadi {5,5}, begitupun untuk KD_2, hingga KD_5, sehingga kunci desimal (KD) yang baru adalah sebagai berikut:
KD : {5,5,5,7,7,4,9,8,1,1,4}
Kemudian kunci desimal (KD) dikonversi ke bentuk kunci biner (KB), dimulai dari $KD_1 = 5$, jika dilihat pada Tabel 3, maka konversi binernya adalah 1011011, berlaku juga untuk KD_2 hingga KD_{11}, sehingga keseluruhan kunci biner (KB) adalah sebagai berikut:
(KB) : { 1011011, 1011011, 1011011, 1110000, 1110000, 0110011, 1110011, 1111111, 0110000, 0110000, 0110011 }

3. Konversi Chiperteks (C) ke Bentuk Chiperteks Desimal (CD) berdasarkan Kode ASCII

Berikut chiperteks (C) yang dihasilkan:
C: 6=r7%FV3(HDCS_3,118)Z
Chiperteks (C) tersebut terdiri dari 21 karakter, tiap karakter pada chiperteks (C) dikonversi ke bentuk desimal dimulai dari karakter ke-1 atau yang dinotasikan C_1 yaitu "6", berdasarkan aturan kode ASCII, karakter "6" memiliki desimal 54, sehingga CD_1 bernilai 54. Berlaku juga untuk C_2 hingga C_{21}, yang menghasilkan CD_2 hingga CD_{21} , sehingga keseluruhan chiperteks desimal (CD) adalah sebagai berikut:
CD : {54, 61, 114, 55, 37, 70, 86, 51, 40, 72, 68, 67, 83, 95, 51, 44, 49, 49, 56, 41, 90}
Chiperteks desimal (CD) kemudian ditransformasi dengan ketentuan sebagai berikut:

a. Apabila $CD_i \in [48,57]$, maka CD_i dan CD_{i+1} ditransformasi menjadi bilangan baru dengan rumus sebagai berikut:
   1) Jika $CD_i = 48$, maka transformasinya dirumuskan seperti pada Persamaan (8).

$\mathrm{CD_{i+1}} - 32$ (8)

   2) Jika $CD_i \neq 48$, maka transformasinya dirumuskan seperti pada Persamaan (9).

dan proses dilanjutkan dengan $i = i + 2$.

b. Apabila $CD_i \notin [48,57]$ maka CD_i ditransformasi menjadi bilangan baru dengan rumus seperti pada Persamaan (10).

$\mathrm{CD_i} - 32$ (10)

dan proses dilanjutkan dengan $i = i + 1$.
dengan i menyatakan indek pada CD yang bergerak dari mulai 1 hingga maksimal sebanyak desimal pada CD.

Proses transformasi dimulai dari CD_1, karena $CD_1 = 54$ maka CD_1 dan CD_2 menjadi desimal baru dengan rumus sesuai dengan Persamaan (9), dimana $CD_2 = 61$, maka transformasinya adalah sebagai berikut:

$$95 + CD_2 - 32$$

$$95 + 61 - 32 = 124$$

dilanjutkan untuk CD_3 hingga CD_{21} , sehinnga keseluruhan chiperteks desimal (CD) hasil transformasi adalah sebagai berikut:
CD: {124,82,100,38,54,103,40,36,35,51,63,107,112,104,58}

4. Konversi Chiperteks Desimal (CD) ke bentuk Chiperteks Biner (CB)

Chiperteks desimal (CD) kemudian diubah ke bentuk chiperteks biner (CB) dengan panjang 7 bit.
Untuk $CD_1 = 124$, bentuk binernya adalah 1111100, sehingga $CB_1 = 1111100$.
Begitupun untuk CD_2 hingga CD_{15} ,sehingga keseluruhan chiperteks biner (CB) adalah sebagai berikut:

CB: {1111100, 1010010, 1100100, 0100110, 0110110, 1100111, 0101000, 0100100, 0100011, 0110011, 0111111, 1101011, 1110000, 1101000, 0111010}

5. Proses Dekripsi Metode ECB

a. Pergeseran Bit

Pergeseran bit dengan menggeser sejauh 1 bit ke kanan, dan bit yang paling kanan akan terlempar ke posisi paling kiri, dilakukan dari mulai blok chiperteks biner pertama (CB_1) hinnga terakhir (CB_{15}).
Untuk CB_1

Bentuk biner = 1111100
Hasil Pergeseran Bit = 0111110

begitu seterusnya hingga CB_{15}, sehingga keseluruhan chiperteks biner (CB) yang baru yaitu:

CB: {0111110, 0101001, 0110010, 0010011, 0011011, 1110011, 0010100, 0010010, 1010001, 1011001, 1011111, 1110101, 0111000, 0110100, 0011101}

b. Proses XOR

Proses XOR antara chiperteks biner (CB) dengan kunci biner (KB) menghasilkan plainteks biner (PB). Chiperteks biner (CB) dan kunci biner (KB), masing masing memilki panjang 15 dan 11. Panjang yang dimaksud adalah jumlah blok biner. Proses XOR dirumuskan seperti pada Persamaan (11).

$$CB_i = PB_i \oplus KB_{(i-1 \bmod m)+1} \quad (11)$$

dengan i menyatakan indeks pada CB yang bergerak dari 1 hingga sepanjang CB, dan m menyatakan panjang kunci (K).
Berikut perhitungan untuk mencari plainteks biner (PB), mulai dari PB_1, berdasarkan Persamaan (11) maka PB_1 dihasilkan dari proses XOR antara CB_1 dengan $KB_{(1-1 \bmod 11)+1}$.
$PB_1 = CB_1 \oplus KB_1$

CB_1 : 0111110
KB_1 : 1011011
1100101 $\oplus$

$PB_1 = 1100101$.

Perhitungan di atas berlaku juga untuk menghitung PB_2 hingga PB_{15}, sehingga keseluruhan dihasilkan plainteks biner (PB) sebagai berikut:

PB: {1100101, 1110010, 1101001, 1100011, 1101011, 1000000, 1100111, 1101101, 1100001, 1101001, 1101100, 0101110, 1100011, 1101111, 1101101}

6. Konversi Plainteks biner (PB) ke Bentuk Plainteks Desimal (PD)

Plainteks biner (PB) kemudian dikonversi ke bentuk plainteks desimal (PD), mulai dari PB_1 hingga hingga PB_{15}
Untuk $PB_1 = 1100101$, bentuk desimal dari PB_1 adalah 101, sehingga $PD_1 = 101$,sehingga keseluruhan plainteks desimal (PD) adalah sebagai berikut:

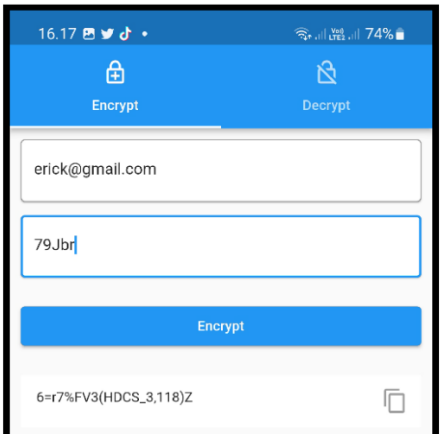
PD: {101, 114, 105, 99, 107, 64, 103, 109, 97, 105, 108, 46, 99, 111, 109}

7. Konversi Plainteks Desimal (PD) ke bentuk Plainteks (P)

Plainteks desimal (PD) menjadi ke bentuk karakter (C) dimulai dari PD_1 yang bernilai 101, berdasarkan aturan kode ASCII, 101 merupakan desimal dari karakter " e ", maka $P_1 = e$. Berlaku juga untuk PD_2 hingga PD_{15} yang menghasilkan P_2 hingga P_{15} , sehingga keseluruhan plainteks (P) adalah sebagai berikut:
Plainteks (P): erick@gmail.com

4.3. Pembuatan Program

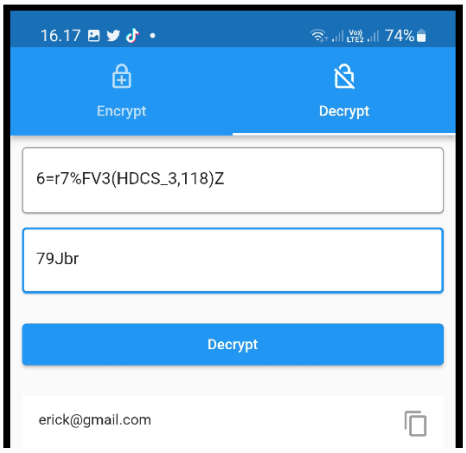
Pembuatan program berbasis aplikasi mobile bertujuan untuk mempermudah perhitungan dan mempercepat waktu apabila memiliki data yang ukurannya besar. Program kemudian disimulasikan. Simulasi program dimulai dengan proses enkripsi. Pada proses ini ouput dari program adalah chiperteks. Input yang diperlukan program yaitu plainteks yang diinput pada TextField pertama serta kunci yang diinput pada TextField kedua. Plainteks yang digunakan yaitu erick@gmail.com dan kunci yang digunakan yaitu 79Jbr. Terlihat pada Gambar 5, jika diproses program menghasilkan output chiperteks yaitu 6=7r%FV3(HDCS_3,118)Z.



Gambar 5. Tampilan simulasi enkripsi pada aplikasi

Simulasi program selanjutnya yaitu proses dekripsi, menu bar yang dipilih adalah dekripsi. Pada proses ini ouput dari program adalah plainteks. Input yang diperlukan program yaitu chiperteks yang diinput pada *TextField* pertama serta kunci yang

diinput pada *TextField* kedua. Chiperteks diambil dari hasil simulasi proses enkripsi yaitu 6=7r%FV3(HDCS_3,118)Z dan kunci yang digunakan yaitu 79Jbr. Terlihat pada Gambar 6, jika diproses program menghasilkan output plainteks yaitu erick@gmail.com.



Gambar 6. Tampilan simulasi dekripsi pada aplikasi

5. KESIMPULAN

Berdasarkan hasil dan pembahasan yang sudah dipaparkan, kombinasi metode Electronic Code Book (ECB) dan kode Seven Segment Display (SSD dapat digunakan untuk mengamankan data teks dengan baik. Data tersebut dapat diubah ke bentuk yang tidak dapat dikenali (chiperteks) melalui tahap enkripsi. Chiperteks yang dihasilkan berupa karakter printable pada ASCII. Chiperteks juga dapat dikembalikan secara utuh tanpa adanya informasi yang hilang melalui tahap dekripsi.

DAFTAR PUSTAKA

ARIANDI, W., S. WIDYASTUTI, dan L. HARIS. 2020. Implementasi *Block Cipher Electronic Code Book* (ECB) untuk Pengamanan Data Pegawai. *Jurnal Ilmiah Teknologi Informasi UMUS*. 2(02): 65-74.

ARIYUS, D. 2008. *Pengantar Ilmu Kriptografi: Teori Analisis & Implementasi*. Yogyakarta: CV. ANDI OFFSET.

ARNAWA, I. A. W., P. E. W. HARRY, dan A. A. G. B. PUTRA. 2020. Perbandingan Waktu Enkripsi Antara Metode *Electronic Code Book* (ECB) dan *Chipher Block Chaining* (CBC) dalam Algoritma Blowfish. *Jurnal Ilmu Komputer Indonesia (JIKI)*. 5(2):50-54.

ASTI, M., A. KAMSYAKAWUNI, dan K. A. SANTOSO. 2018. Pengamanan *Image* dengan Modifikasi *Algoritma Electronic Code Book* (ECB). *Jurnal Ilmiah Matematika dan Statistika*. 18(2): 91-104.

ESMAWA, A., dan G. ANTARNUSA. 2019. Perancangan Sistem Penskoran Olahraga dengan Tampilan *Seven Segment*. *Jurnal Ilmiah Penelitian dan Pembelajaran Fisika*. 5(1): 99-108.

KARO, E. S. Br. 2020. Penerapan Algoritma Affine Chiper dan Algoritma Electronic Code Book (ECB) dalam Pengamanan Pesan Teks. *Jurnal Teknik Informatika Unika St. Thomas*. 05(01):28-32.

KAUNANG, F. J., A. KARIM, J. SIMAMARTA, A. ISKANDAR, D. P. Y. ARDIANA, R. S. SEPTARINI, E. S. NEGARA, HAZRIANI, R. D. WIDYASTUTI. 2021. *Konsep TeknologiInformasi*. https://www.google.co.id/books/edition/Konsep_Teknologi_Informasi/cIUeEAAAQBAJ?hl=en&gbpv=0. [Diakses pada tanggal 7 januari 2023].

MISRA, Y. 2022. *Programming and Interfacing with Arduino*. https://www.google.co.id/books/edition/Programming_and_Interfacing_with_Arduino/DsA5EAAAQBAJ?hl=en&gbpv=1&dq=Programming++and+Interfacing+with+Arduino&printsec=frontcover. [Diakses pada tanggal 10 Desember 2022].

MUFID, A. 2010. Teknik Enkripsi Dan Deskripsi Menggunakan Algorithma Electronic Code Book (ECB). *Jurnal Tehnik – UNISFAT*. 6(1): 21-25.

MUNIR, R. 2019. *Kiptografi Jilid II*. Bandung: Informatika Bandung.

PUTRA, A. Y. dan I. D. A. E. Yuliani. 2019. Perancangan dan Pengujian Perangkat Lunak Kriptografi Gabungan Playfair Chiper dan Electronic Code Book (ECB). *Jurnal ENTER*. 2(2019):560-570.

RIDWAN dan BUSTAMI. 2021. Konsep dan Perancangan Aplikasi: Membangun Aplikasi Mobile Menggunakan Flutter. Aceh: Syiah Kualas University Press.

SIMARMATA, J., SRIADI, R. RAHIM. 2019. *KRIPTOGRAFI Tehnik Keamanan Data dan Informasi*. Yogyakarta: CV. ANDI OFFSET.

SURAHMAT, A. dan Tb. D. FU'ADY. 2020. Simulasi Rangkaian Seven Segment menggunakan Multisim pada Pembelajaran Rangkaian Elektronika Analog Dan Digital di SMKS Informatika Sukma Mandiri. *Jurnal Of Innovation and Future Technology*.2(1):15-28.

SYAHPUTRA, M. R. 2022. Kombinasi Metode Vigenere Chiper dan Electronic Code Book untuk Keamanan Data. *Jurnal Cyber Area*. 2(9):1-10

WIDARMA, A., H. F. SIREGAR. Dan M. D. IRAWAN. 2019. Teknik Keamanan Data MenggunakanVigenere Chiper dan Electronic Code Book (ECB). Jurnal Sains Komputer & Informatika. 3(2):393-400.