KOMUNIKASI DATA DENGAN METODE CAESAR CIPHER DAN ALGORITMA PENGACAKAN FISHER-YATES BERBASIS MOBILE

Marwondo
Fakultas Teknologi dan Informatika, UNIBI

Abstrak

Saat ini, banyak orang menggunakan internet untuk komunikasi data, tetapi sedikit orang yang menyadari bahwa pentingnya keamanan data pada komunikasi tersebut. *Kriptografi* adalah sebuah teknik untuk memproteksi data atau pesan dengan mengubah pesan menjadi kode unik, kemudian hanya pengirim dan penerima yang dapat mengerti pesan tersebut. *Caesar cipher* adalah salah satu metode *kriptografi* dengan teknik *subtisusi*, dimana setiap karakter pada pesan ditukar dengan karakter pada tabel *hash,* dengan syarat pengirim dan penerima memiliki kunci dan tabel tabel *hash* yang sama.

Pada dasarnya metode *Caesar cipher* menggunakan tabel *hash* dengan karakter A-Z, dimana tabel *hash* ini diacak dari kanan kekiri berdasarkan kunci yang sudah disepakati untuk proses *enkripsi* dan *dekripsi*. Sehingga jika menggunakan kunci dengan kelipatan 26 maka orang lain mudah mendapatkan pesan rahasianya.

Penelitian ini mengkombinasi antara metode *Caesar cipher* dan algoritma pengacakan *Fisher-Yates,* sehingga proses pengacakan karakter pada tabel *hash* sulit ditemukan, karena algoritma *Fisher-Yates* tidak membatasi angka atau kunci untuk proses pengacakan pada tabel *hash*, maka pesan diproteksi dengan lebih kuat.

Kata Kunci : *Kriptografi, Caesar cipher, Subtisusi, Hash, Fisher-Yates.*

Abstract

Currently, many people are using internet for data communication, but the people do not care about data security on the communication. Criptography is a technic for protecting of data or message by changing the message into unique code, then only sender and receiver can be open the original message. Caesar cipher is a method of criptography by subtitution technic, furthermore each of messsage character change to character in the hash table, where a sender and a receiver use the equal key and the hash table.

Basically, Caesar cipher method is using hash table when A until Z as characters, futhermore randomize each alphabet character of hash table from right to left based on the key for encription and decription process. If the key is multiple of 26, then the alphabet character order of hash table back to the beginning, the impact is other people can be obtain the secrete message.

This research is combine between Caesar method and Fisher-Yates randomized algorithm, then randomized process of hash table does not back to the beginning, since Fisher-Yates algorithm does not limit the number or key for the randomization process in the hash table, the message is strongly protected.

Keywords: *Criptography, Caesar cipher, Subtitution, Hash, Fisher-Yates*

1. PENDAHULUAN

1.1 Latar Belakang

Saat ini pertumbuhan komunikasi data sangat cepat, apalagi adanya internet sehingga mempermudah dan mempercepat proses komunikasi antara dua orang atau lebih. Namun orang yang tidak berkepentingan dalam komunikasi tersebut juga bebas mengakses informasi dan komunikasi di internet, sehingga komunikasi yang bersifat rahasia sangatlah berbahaya jika tidak ada keamanan didalam komunikasi data tersebut.

Kriptografi dan steganografi adalah teknik untuk memanipulasi informasi atau pesan dengan mengubah atau menyembunyikan pesan tersebut [1]. Kriptografi adalah teknik untuk melindungi pesan dengan cara megubah atau mengacak pesan sehingga hanya pengirim dan penerima yang dapat membaca pesan tersebut [1]. Steganografi adalah teknik untuk melindungi pesan dengan menyembunyikan pesan ke dalam sebuah media, media dapat berupa text, suara, gambar, video dan media sosial [5].

Ada beberapa teknik dasar yang dapat dilakukan di kriptografi yaitu subtitusi, blocking, permutasi, ekspansi atau pemampatan (compression). Caesar cipher adalah salah satu teknik kriptografi dengan metode subtitusi, dimana informasi asli diganti dengan informasi lainnya [3].

Pengirim dan penerima menentukan kunci yang digunakan untuk mengacak tabel hash, dimana tabel hash berisi informasi alfabet dari A sampai Z, kunci tersebut berupa angka acak. Sehingga hanya pengirim dan penerima yang dapat mengetahui informasi asli dari kunci tersebut. Pengacakan karakter alfabet dilakukan dari kanan kekiri sesuai kunci yang telah ditentukan. Maka dari itu Caesar cipher aman jika menggunakan kunci $mod\ 26$, artinya jika menggunakan kunci dengan kelipatan 26 maka pengacakan karakter pada tabel hash akan kembali ke awal dengan urutan alfabet A sampai Z, sehingga mempermudah orang yang tidak berkepentingan mendapatkan pesan rahasianya.

Penelitian ini mengkombinasi Caesar cipher dan algoritma pengacakan Fisher-Yates. Algoritma Fisher-Yates mengacak karakter di tabel hash sesuai kunci yang ditentukan, tetapi tidak secara berurutn atau dari kanan kiri. Sehingga kunci yang menjadi kesepakatan antara pengirim dan penerima tidak terbatas. Kemudian dengan algoritma ini kunci sulit ditebak oleh orang yang tidak berkepentingan, sehingga membuat pesan rahasia sulit untuk didapatkan.

1.2 Identifikasi Masalah

Metode Caesar cipher aman digunakan jika menggunakan kunci $mod\ 26$, sebaliknya jika menggunakan kunci dengan kelipatan 26 maka riskan untuk diketahui pesan rahasianya, karena teknik yang utama dari metode Caesar cipher adalah pengacakan pada tabel hash.

1.3 Rumusan Masalah

Perumusan masalah yang mendasari penelitian ini adalah bagaimana memodifikasi metode original Caesar cipher pada proses pengacakan tabel hash.

1.4 Batasan Masalah

Tabel hash yang digunakan pada penelitian ini adalah karakter alfabet dari A sampai Z secara berurutan dengan jumlah 26.

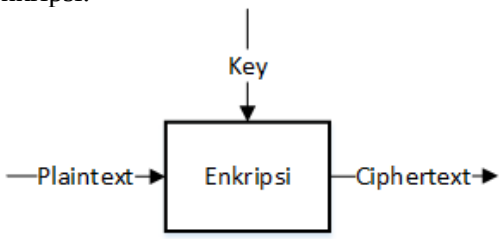
2. LANDASAN TEORI

2.1 Kriptografi

Kriptografi adalah teknik mengubah pesan ke bentuk tertentu dengan menggunakan ilmu matematika untuk proses enkripsi dan dekripsi didalamnya [2]. Kriptografi digunakan untuk melindungi informasi atau pesan yang sensitif, sehingga orang tertentu saja yang boleh mengakses

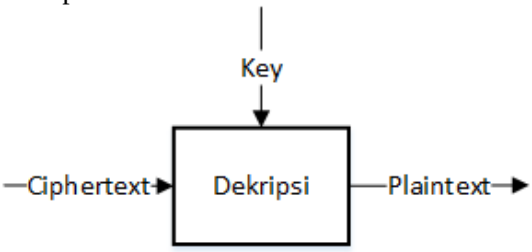
informasi tersebut. Jika kriptografi adalah ilmu untuk melindungi data, maka Kriptanalis adalah ilmu untuk menganalisa data atau informasi yang sudah dilindungi dengan teknik kriptografi. Ada beberapa istilah yang sering digunakan pada ilmu kriptogrfi yaitu, plaintext, ciphertext, key, enkripsi, dan dekripsi. Plaintext adalah informasi atau pesan asli sebelum proses enkripsi [4]. Ciphertext adalah informasi atau pesan setelah proses enkripsi, dimana pesan sudah diubah kedalam bentuk tertentu [4]. Key adalah sebuah variabel yang berbentuk angka atau huruf yang berfungsi untuk mengubah atau mendapakan pesan asli pada proses enkripsi atau dekripsi. Enkripsi adalah proses mengubah plaintext menjadi ciphertext. Dekripsi adalah kebalikan dari enkripsi yaitu proses mengubah ciphertext ke plaintext.

Pada gambar 1.1 menjelaskan proses enkripsi.



Gambar 1.1
Proses Enkripsi

Pada gambar 1.2 menjelaskan proses dekripsi.



Gambar 1.2
Proses Dekripsi

Ada beberapa teknik dasar kriptografi seperti subtitusi, transposisi atau permutasi, block cipher, stream cipher, expansi, dan pemampatan. Subtitusi adalah teknik kriptografi dengan menggantikan informasi satu dengan lainnya pada proses enkripsi [3], Caesar cipher adalah salah satu metoda kriptografi dengan menggunakan teknik subtitusi.

2.2 Caesar Cipher

Caesar cipher adalah metode konvensional dari kriptografi dengan menggunakan teknik subtitusi, dimana informasi asli diganti dengan informasi yang lain di tabel hash [3]. Salah satu teknik lama ini pertama kali di temukan oleh Julius Caesar.

Teknik sederhana ini sering digunakan pada metode kriptografi, sehingga skema dari Caesar cipher menjadi bagian dari beberapa metode lainnya seperti Vigenere cipher dan ROT13. Keutamaan dari Caesar cipher adalah pengacakan karakter pada tabel hash, dimana karakter pada pesan asli digantikan oleh karakter pada hash tabel yang sudah diacak tergantung dari kunci yang sudah disepakati antara pengirim dan penerima. Dibawah ini dijelaskan langkah-langkah proses enkripsi dan dari Caesar cipher:

1. Diketahui pengirim menulis plaintext “BANDUNG” yang dikirim ke penerima.
2. Pengirim dan penerima menggunakan angka 3 sebagai kunci yang sudah disepakati dan pada tabel hash berisi 26 karakter alfabet.

Tabel 1.1
Tabel Hash

A	B	C	D	E	F	..	X	Y	Z

3. Langkah selanjutnya adalah proses enkripsi. Angka 3 sebagai kunci yang sudah disepakati, maka karakter pada tabel hash digeser sebanyak 3 kali dari kanan kekiri. Maka menghasilkan urutan seperti tabel 1.2.

Tabel 1.2

Tabel Hash Setelah di Geser

D E F G H I .. A B C

Dimana sekarang menjadi D = A, E = B, F = C dan seterusnya. Proses ini dapat digambarkan ke dalam modular aritmatic yaitu:

$$E_n(x) = (x + n)\ mod\ 26 \quad (1.1)$$

Diamana E_n adalah enkripsi, x adalah kunci dan n adalah jumlah index dari karakter.

4. Dengan menggunakan skema pada langkah ke 3 maka plaintext "BANDUNG" menjadi "EDQGXQJ" sebagai ciphertext. Maka data yang dikirim oleh pengirim ke penerima adalah "EDQGXQJ".

Dibawah ini dijelaskan langkah-langkah proses dekripsi dan dari Caesar cipher:

1. Diketahui penerima mendapatkan ciphertext adalah "EDQGXQJ".
2. Angka 3 sebagai kunci yang sudah disepakati dan pada tabel hash berisi 26 karakter alfabet.

Tabel 1.3
Tabel Hash

A B C D E F .. X Y Z

Langkah selanjutnya adalah proses dekripsi. Angka 3 adalah kunci yang sudah disepakati, maka karakter pada tabel hash digeser sebanyak 3 kali dari kanan kekiri. Maka menghasilkan urutan seperti tabel 1.4. Proses ini dapat digambarkan ke dalam modular aritmatic yaitu:

$$D_n(x) = (x - n)\ mod\ 26 \quad (1.2)$$

Diamana D_n adalah dekripsi, x adalah kunci dan n adalah jumlah index dari karakter.

Tabel 1.4

Tabel Hash Setelah di Geser

D E F G H I . A B C .

Dimana sekarang menjadi D = A, E = B, F = C dan seterusnya.

3. Dari tabel diatas maka ciphertext "EDQGXQJ" diubah menjadi "BANDUNG" sebagai plaintext.

2.3 Algoritma Fisher-Yates

Fisher-Yates adalah sebuah algoritma untuk menghasilkan permutasi yang acak dari sebuah urutan angka [6]. Algoritma ini di temukan oleh Ronald Fisher dan Frank Yates, ada dua algoritma yang dikenal, pertama adalah original Fisher-Yates dan Modern Fisher-Yates yaitu yang dimodifikasi oleh Durstenfeld. Jika dilihat dari komplesitas antara algoritma original dan modern Fisher-Yates, maka kompleksitas modern Fisher-Yates adalah $O(n)$ dan original Fisher-Yates adalah $O(n^2)$, modern lebih efesien daripada original, karena modern Fisher-Yates tidak perlu menghitung proses jumlah angka yang tersisa, tetapi memindahkan urutan angka pada index ke i ke index yang dipilih [5]. Pada tabel berikut dijelaskan proses modern Fisher-Yates:

1. Langkah pertama diketahui nilai permutasi A sampai C.

Tabel 1.5
Langkah 1 Modern Fisher-Yates

Jarak	Angka Acak	Alfabet	Hasil
1-3	2	A B C	

2. Pilih angka secara random dan tukar index alfabet sesuai angka random ke index terakhir.

Tabel 1.6
Langkah 2 Modern Fisher-Yates

Jarak	Angka Acak	Alfabet	Hasil
1-3	2	A C	B

3. Lakukan seperti langkah ke dua, dimana angka random adalah 2.

Tabel 1.7
Langkah 3 Modern Fisher-Yates

Jarak	Angka Acak	Alfabet	Hasil
1-2	2	A	B C

4. Lakukan seperti langkah ke dua, dimana angka random adalah 1.

Tabel 1.8
Langkah 4 Modern Fisher-Yates

Jarak	Angka Acak	Alfabet	Hasil
1-1	1	A	B C A

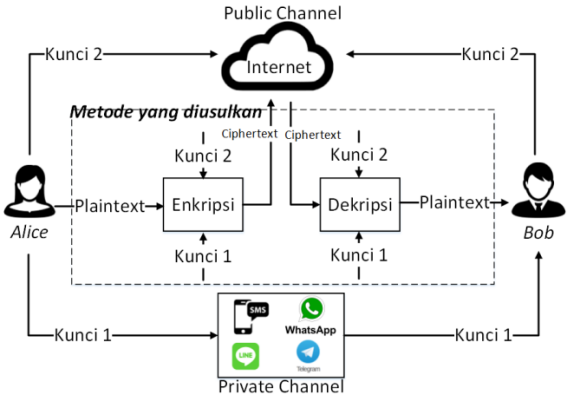
Maka alfabet yang berurutan "A B C" setelah melalui proses pengacakan menjadi "B C A". Dibawah ini adalah algorima modern Fisher-Yates.

Algoritma 2.1 Modern Fisher-Yates

```
1: procedure FISHER-YATES-NEWER
2:     Input: {a_i}_{i=1}^n, j ← 0, tmp ← 0
3:     for i = n − 1 to 1 do
4:         j ← RandomIntegerSuchThat 0 ≤ j ≤ i
5:         tmp ← a_i
6:         a_i ← a_j
7:         a_j ← tmp
8:     end for
9:     return {a_i}_{i=1}^n
10: end procedure
```

3. METODE PENELITIAN

Penelitian yang dilakukan adalah dengan mengkombinasi metode Caesar cipher dan algoritma modern Fisher-Yates. Algoritma Fisher-Yates digunakan untuk mengacak table hash berdasarkan kunci yang telah disepakati antara pengirim dan penerima.



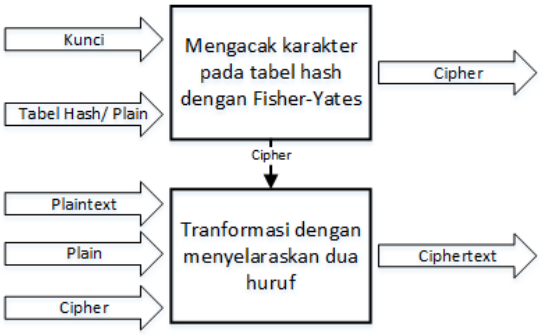
Gambar 3.1
Desain Proses

Pada gambar 3.1 adalah skenario dari data komunikasi, dimana Alice sebagai pengirim dan Bob sebagai penerima. Pada desain proses menggunakan dua channel untuk mengirimkan informasi, yang pertama private channel digunakan untuk menentukan kunci yang disepakati. Private channel pada kenyataannya dapat menggunakan komunikasi dengan sms, whatsapp, line, telegram dan lainnya. Kedua adalah public channel, digunakan untuk mengirim plaintext dengan metode yang diusulkan. Skenarionya adalah Alice ingin mengirim plaintext "CEGAH" ke pada Bob, sebelumnya Alice mengirimkan *kunci_1* yang bernilai 5 melalui private channel kepada Bob. Kemudian mereka sudah sepakat menggunakan *kunci_2* yang bernilai 7 yang didapat dari internet (public channel) dimana pada pertandingan UEFA Champions Legue semalam, Barcelona FC menang 7-0 atas Chealsea FC. Kemudian Alice melakukan proses enkripsi menggunakan kedua kunci tersebut dan mengirimkan ciphertext melalui status Facebook (public channel) yang dia publish pukul 10.00 AM.

Pada sisi penerima, Bob sebelumnya sudah mendapatkan *kunci_1* yang dikirim oleh Alice. Kemudian Bob juga sudah mendapatkan *kunci_2* dari informasi diinternet, dan ciphertext didapat dari status Alice di Facebook yang dibaca oleh Bob. Dari ketiga informasi tersebut Bob mendapatkan pesan asli dari Alice.

3.1 Enkripsi

Pada bagian ini dijelaskan proses enkripsi yang dilakukan pada sisi pengirim, dimana plaintext diubah dalam bentuk tertentu menggunakan Caesar cipher dan algoritma modern Fisher-Yates.



Gambar 3.2
Desain Proses Enkripsi

Berikut penjelasan dari langkah-langkah enkripsi yang terddapat pada gambar 3.2:

1. Langkah pertama, Alice dan Bob menentukan angka 5 sebagai *kunci_1* dan 7 sebagai *kunci_2*.
2. Kemudian langkah kedua Alice memiliki plaintext yaitu "CEGAH".
3. Langkah selanjutnya mengacak karakter pada tabel hash dengan algoritma modern Fisher-Yates. Sebelum melakukan pengacakan, nilai kunci dimasukkan kedalam persamaan untuk proses pembangkit angka acak sebagai index yang ditukar dengan index ke i. Dimana X adalah variabel index, K adalah nilai *kunci_1*, L adalah nilai *kunci_2*, dan i adalah index pada data.

$$X_i = ((K * (i-1)) + (K * L)) \; mod \; 26 \qquad (3.1)$$

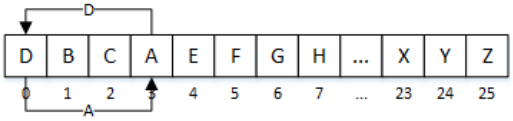
Diketahui tabel hash adalah:

A	B	C	D	E	F	G	H	...	X	Y	Z
0	1	2	3	4	5	6	7	...	23	24	25

Gambar 3.3
Hash

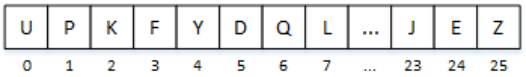
Kemudian nilai kunci dimasukkan pada persamaan 3.1, maka index pertama yang didapat adalah $X_1 = ((5 * (i-1)) + (5 * 7)) \; mod \; 26 = 4$. Jika hasil persamaan adalah 1, maka $K * n$ dan $L * n$ dimana n adalah jumlah pesan.

4. Langkah keempat nilai pada index ke 4 ditukar dengan nilai pada index ke $i = 1$, dapat dilihat pada gambar 3.4.



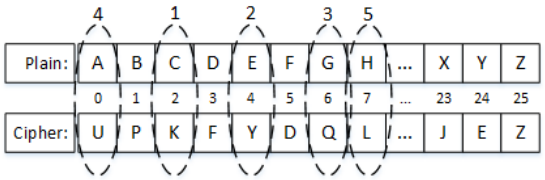
Gambar 3.4
Menukar Nilai Pada Index

5. Langkah Kelima adalah mengulangi langkah 3 dan 4 sebanyak karakter yang ada ditabel hash yaitu 26. Sehingga hasil pengacakan dapat dilihat pada gambar 3.5.



Gambar 3.5
Cipher

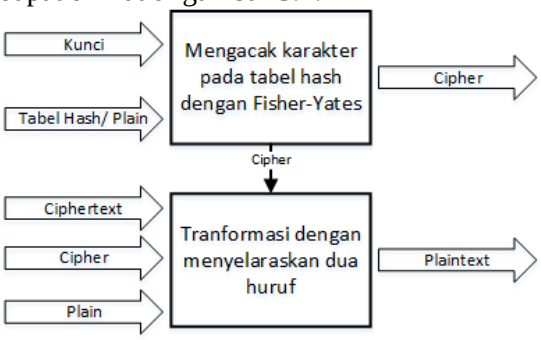
6. Langkah keenam adalah mentranformasikan plaintex "CEGAH" dengan cipher atau tabel hash yang sudah diacak, maka C=K, E=Y, G=Q, A=U, dan H=L, sehingga ciphertext yang dikirim ke Bob adalah "KYGQUL", hasilnya dapat dilihat pada gambar 3.6.



Gambar 3.6 Ciphertext

3.2 Dekripsi

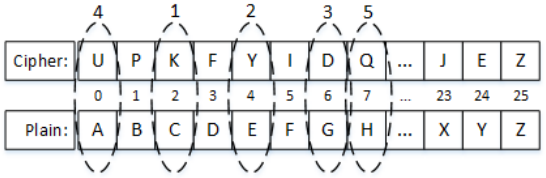
Dekripsi adalah proses untuk mendapatkan plaintext atau pesan asli pada sisi penerima. Sebelumnya Bob sudah menyepakati bersama Alice *kunci_1* dan *kunci_2* yang digunakan, maka proses dekripsi sebenarnya kebalikan dari enkripsi dapat dilihat di gambar 3.7.



Gambar 3.7
Desain Proses Dekripsi

Dibawah ini dijelaskan langkah-langkah dekripsi:

1. Langkah pertama mengacak karakter pada tabel hash "A, B, C, D, E, F, G, H, .., X, Y, Z" dengan algoritma modern Fisher-Yates. Sebelum melakukan pengacakan, nilai kunci dimasukkan kedalam persamaan 3.1 untuk mendapatkan index yang ditukar dengan index terakhir. Maka index pertama yang didapat adalah $X_1 = ((5 * (i - 1)) + (5 * 7))\ mod\ 26 = 4$ dan nilai pada index ke 4 akan ditukar dengan nilai pada index ke $i = 1$. Jika hasil persamaan adalah 1, maka $K * n$ dan $L * n$ dimana n adalah jumlah pesan. Sehingga tabel hash sekarang menjadi "D, B, C, A, E, F, G, H, .., X, Y, Z".
2. Langkah kedua mengulangi langkah pertama sebanyak 26, sehingga cipher atau hasil pengacakan dari karakter pada tabel hash adalah "U, P, K, F, Y, D, Q, L, .., J, Y, Z".
3. Langkah ketiga adalah mentranformasikan ciphertext "KYGQUL" dengan cipher atau tabel hash yang sudah diacak, sehingga plaintext yang didapat oleh Bob adalah "CEGAH", hasilnya dapat dilihat pada gambar 3.8.



Gambar 3.8
Plaintext

3.3 Algoritma Pengacakan

Pada algoritman 3.1 menunjukkan algoritma pengacakan pada metode yang diusulkan.

Algoritma 3.1 Pengacakan Tabel Hash

```
procedure PENGACAKAN TABEL HASH
    Input: {P_i}_{i=1}^n, j ← 0, tmp ← 0, K, L
    for i = n − 1 to 1 do
        j ← (K * (i − 1) + (K * L)) mod 26
        tmp ← P_i
        P_i ← P_j
        P_j ← tmp
    end for
    return {P_i}_{i=1}^n
end procedure
```

4. HASIL DAN PEMBAHASAN

Pada bagian ini menjelaskan hasil dari metode yang diusulkan dengan metode asli dari Caesar Cipher dengan menganalisa keamanan data. Kemudian mencoba 10 pesan yang berbeda. Setelah itu ditampilkan tampilan antarmuka dalam bentuk mobile dari motode yang diusulkan.

4.1 Evaluasi Keamanan Data

Pada bagian ini dilakukan analisa keamanan data pada metode yang diusulkan dan sebelumnya. Evaluasi dilakukan dengan melihat berapa jumlah kombinasi pengacakan karakter tabel hash dengan menggunakan kunci dari 1 sampai 30. Berikut hasilnya d tabel 4.1.

Tabel 4.1.
Kombinasi tabel hash dengan 30 kunci

Kunci	Sebelum	Diusulkan
1	BCDEFGHIJKL MNOPQRSTU VWXYZA	QPMHOLKDIN GFCJEBAZYX WVUTSR
2	CDEFGHIJKL MNOPQRSTU VWXYZAB	NCHGPKBORS JWTAFIVQLY XMDEZU
3	DEFGHIJKLM NOPQRSTUV WXYZABC	ABCJKFSLUD EHMNOTWXG PIVQRYZ
4	EFGHIJKLMN OPQRSTUVW XYZABCD	NELMJUVIPK DSBAXYRWT GHCZQFO
5	FGHIJKLMNO PQRSTUVWX YZABCDE	AVQLGZERM TYDINSXCHO JWBUPKF
6	GHIJKLMNOP QRSTUVWXY ZABCDEF	NGVSHYLQX MTOPARUCEF WZIJBKD
7	HIJKLMNOPQ RSTUVWXYZ ABCDEFG	APETIBMXGL CZKNURODW FSHYJQV
8	IJKLMNOPQR STUVWXYZA BCDEFGH	NIXYJORGFU PQVABCDWZ ETSHKLM
9	JKLMNOPQRS TUVWXYZAB CDEFGHI	ADGBEPCHIJ MVKNQFORS TYLWZUX
10	KLMNOPQRST UVWXYZABC DEFGHIJ	NKVEJYHSTM DIBARCZWPU FGXQLO
11	LMNOPQRSTU VWXYZABCD EFGHIJK	ATMFGBIZCP ORQNYDEHUJ WXKLSV
12	MNOPQRSTU VWXYZABCD EFGHIJKL	NMPKRITUFW DYBAZCXEV GHSJQLO
13	NOPQRSTUV WXYZABCDE FGHIJKLM	YNABCDEFG HIJKZMLOPQ RSTUVWX
14	OPQRSTUVW XYZABCDEFG HIJKLMN	NOPQRSTUV WXYZABCDE FGHIJKLM
15	PQRSTUVWX YZABCDEFGH IJKLMNO	AHOBIJCFQLS DYNKRMVWZ ETGPUX
16	QRSTUVWXY ZABCDEFGHIJ KLMNOP	NQCWXGPILO ZURAJSDMTF VHBKYE
17	RSTUVWXYZ ABCDEFGHIJ KLMNOPQ	AJSBKLCPYD MTENQVWZG FIHORUX
18	STUVWXYZA BCDEFGHIJKL MNOPQR	NSLKTMRWZ UPQIAVYXEO GHBJCDF
19	TUVWXYZAB CDEFGHIJKL MNOPQRS	ALIHQROJGV WDUNEZCTY XSBMPKF
20	UVWXYZABC DEFGHIJKLM NOPQRST	NURIVWZKG YCMHAPBTE XOSDJFQL
21	VWXYZABCD EFGHIJKLMN OPQRSTU	AFKPUBWJOH CXSNIDYTMR EZGLQV
22	WXYZABCDE FGHIJKLMNO PQRSTUV	NWTOZGPYIQ RUXAJCDMSF VHBKLE
23	XYZABCDEFG HIJKLMNOPQ RSTUVW	ADGJMHSPUB ETKNWFOXC VYLQZIR
24	YZABCDEFGH IJKLMNOPQR STUVWX	NYZUGQXMP KVSFATCRED BWOJILH
25	ZABCDEFGHIJ KLMNOPQRST UVWXY	CBALQVWXK PIDYTOJEZUN SFGHMR
26	ABCDEFGHIJ KLMNOPQRST UVWXYZ	ZABCDEFGHIJ KLMNOPQRST UVWXY
27	BCDEFGHIJKL MNOPQRSTU VWXYZA	QPMHOLKDIN GFCJEBAZYX WVUTSR
28	CDEFGHIJKL MNOPQRSTU VWXYZAB	NCHGPKBORS JWTAFIVQLY XMDEZU
29	DEFGHIJKLM NOPQRSTUV WXYZABC	ABCJKFSLUD EHMNOTWXG PIVQRYZ
30	EFGHIJKLMN OPQRSTUVW XYZABCD	NELMJUVIPK DSBAXYRWT GHCZQFO

Dari tabel 4.1 dilakukan pengujian untuk mencari berapa banyak kombinasi pengacakan tabel hash dari kedua metode, dimana kunci yang digunakan adalah bernilai 1 sampai 30.

Hasilnya adalah metode sebelumnya menghasilkan 25 kombinasi pengacakan, jika menggunakan kunci yang bernilai 26 maka urutan karakter kembali keawal, jika menggunakan kunci bernilai 27, 28, 29, dan 30 maka urutan pengacakan sama dengan kunci 2, 3, 4, dan 5, ini dikarenakan pengacakan pola pengacakan yang beraturan dengan menggeser setiap karakter dari kanan ke kiri, sehingga dapat dipastikan jumlah pengacakan sebanyak -1, dimana x adalah jumlah karakter pada tabel hash

Sementara itu metode yang diusulkan menghasilkan 26 kombinasi pengacakan yang berbeda dimana pada tabel 4.1 menggunakan *kunci_1* dari 1 sampai 30 dan *kunci_2* adalah 1, hal ini disebabkan oleh penggunaan persamaan dari pembangkit bilangan acak dan algoritma permutasi dari Fisher-Yates. Jika dihitung berapa banyak jumlah pengacakan yang bisa dilakukan oleh metode yang diusulkan maka *kunci_1* * *kunci_2* yaitu 26 * 26 = 676.

4.2 Evaluasi Pengujian Waktu Eksekusi

Pada bagian ini adalah pengujian dengan skenario 10 pesan berbeda dijalankan di metode sebelumnya dan metode yang diusulkan, kemudian dihitung waktu eksekusi enkripsi pada setiap pesan untuk masing-masing metode, satuan waktu yang digunakan adalah mikro detik.

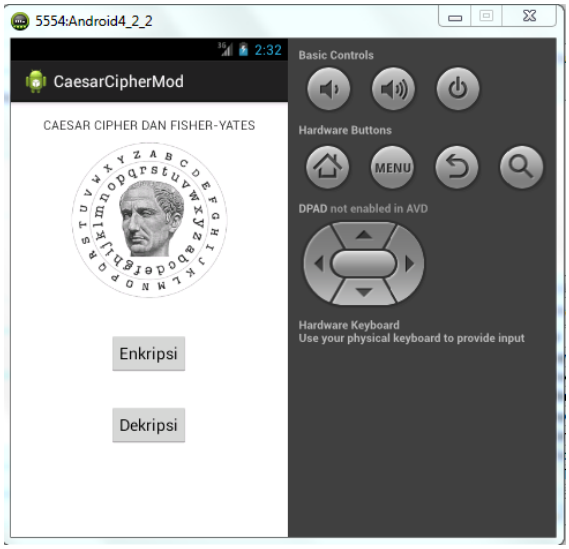
Tabel 4.2.
Waktu Eksekusi

Kunci	Pesan	Mikro detik	
		Sebelum	Diusulkan
1	CEGAH PERTEMUAN ANTARA TUAN X DAN TUAN Z YANG AKAN TERJADI	968	663
2	CEGAH PERTEMUAN ANTARA TUAN X DAN TUAN Z YANG AKAN TERJADI HARI SENIN DIHOTEL	997	894
3	CEGAH PERTEMUAN ANTARA TUAN X DAN TUAN Z YANG AKAN TERJADI HARI SENIN DIHOTEL BLUE PUKUL	1049	1001
4	CEGAH PERTEMUAN ANTARA TUAN X DAN TUAN Z YANG AKAN TERJADI HARI SENIN DIHOTEL BLUE PUKUL SEMBILAN	1534	1113
5	CEGAH PERTEMUAN ANTARA TUAN X DAN TUAN Z YANG AKAN TERJADI HARI SENIN DIHOTEL BLUE PUKUL SEMBILAN MALAM	2922	1121

Dapat dilihat dari tabel 4.2 dengan menggunakan 5 pesan dan 5 kunci yang berbeda, menyatakan waktu eksekusi metode yang diusulkan lebih efesien dibandingkan metode sebelumnya. Hasil ini dipengaruhi algoritma pengacakan yang diterapkan metode diusulkan, jika dinotasikan adalah $O(n)$, sedangkan metode sebelumnya adalah $O(n^2)$.

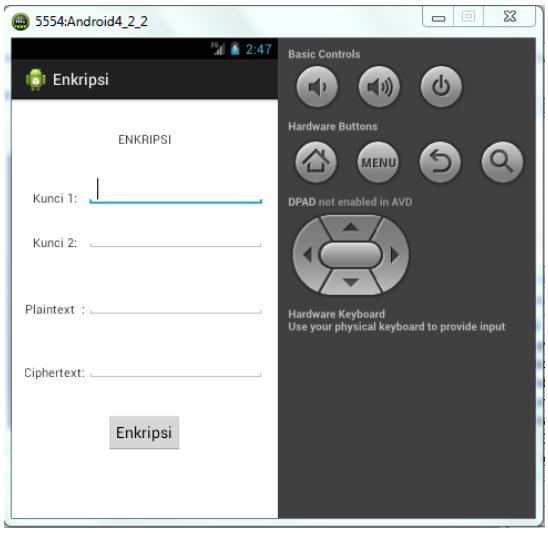
4.3 Tampilan Pengguna Mobile

Disini dijelaskan dan ditampilkan hasil implementasi dari metode yang diusulkan dalam bentuk mobile. Pada gambar 4.1 menjelaskan tampilan awal.



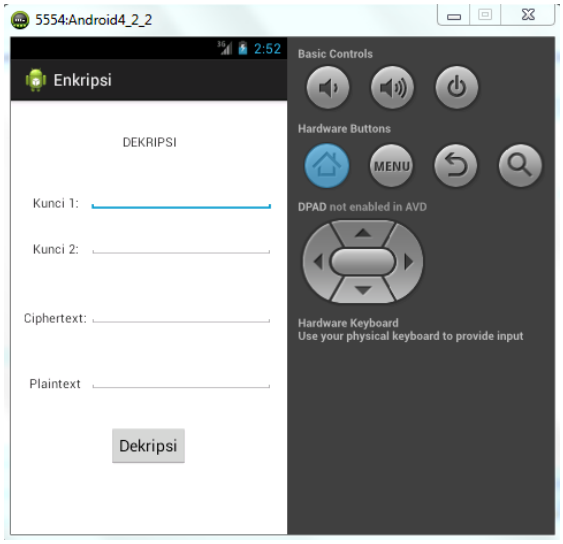
Gambar 4.1
Tampilan Awal Aplikasi

Kemudian pada gambar 4.2 memperlihatkan tampilan untuk proses enkripsi untuk sisi pengirim.



Gambar 4.2
Tampilan Enkripsi

Pada gambar 4.3 menunjukkan tampilan dekripsi untuk sisi penerima.



Gambar 4.3
Tampilan Dekripsi

5. KESIMPULAN

Berdasarkan masalah yang diangkat oleh penulis pada metode sebelumnya, bahwa metode Caesar cipher hanya memiliki 25 kombinasi pengacakan, sehingga memudahkan oranglain menebak kunci dan pesan. Sementara metode yang diusulkan menghasilkan 676 kombinasi pengacakan, dan hasil pengujian waktu eksekusi algoritma pengacakan dapat dinotasikan adalah $O(n)$.

Ada kelemahan yang ada di metode yang diusulkan yaitu jika menggunakan kunci bernilai 1 maka hasil persamaan adalah 0 sehingga pengacakan pada karakter tidak berubah.

6. REFERENSI

[1] A Joseph dan V Sundaram, *Cryptography and Steganography – A Survey*. IJCTA, 2(3):626-630, February 2011.

[2] Network Associates dan Affiliated Companies, *An Introduction to Cryptography*, Version 6.5.2, 1990-1999.

[3] Kuo Cheng-Jing, *Cryptography*, 2015

[4] PGP Corporation, *An Introduction to Cryptography*, Version 8.0, October 2002.

[5] C.A Henk van Tilborg, *Fundamentals of Cryptology A Profesional Reference and Interactive Tutorial*, Kluwer Academic Publisher, 1999.

[6] Jorg Arndt. *Generating random permutations*. PhD thesis, University of Australian National, 2010.