IMPLEMENTASI DOUBLE CAESAR CIPHER MENGGUNAKAN ASCII

Ahmad Tantoni[1], Mohammad Taufan Asri Zaen[2]

[1]Teknik Informatika, STMIK Lombok
[2]Sistem Informasi, STMIK Lombok
Jln. Basuki Rahmat No.105 Praya Lombok Tengah 83511
[1] ahmad.tantoni@students.amikom.ac.id, [2]opanzain@gmail.com

Abstract

This study will discuss about the merger between caesar cipher with caesar cipher or called double caesar cipher, and will show the design of the double caesar cipher algorithm in order to secure the database given the database login user form to make it safer. Double caesar cipher is the development of symmetrical caesar cipher algorithms. The encryption and description process has the same key, each of which has a key. Not only that, the research will also show the double caesar cipher algorithm script using the ASCII (American Standard Code for Information Interchange) table and also in the .php programming language (hypertext preprocessor), then how to run a double caesar cipher, and trying to enter plaintext into the double caesar cipher program then convert it to ciphertext and vice versa from ciphertext to plaintext again.

Keywords : caesar cipher, double caesar cipher, cryptography, ASCII

Abstrak

Penelitian ini akan membahas tentang pemgabungan antara caesar cipher dengan caesar cipher atau disebut dengan double caesar cipher, dan akan menunjukan rancangan algoritma double caesar cipher untuk pengamanan database yang diberikan form user login database agar lebih aman. Double caesar cipher merupakan pengembangan dari algoritma caesar cipher yang simetris. Proses enkripsi dan deskripsinya memiliki kunci yang sama, masing-masing memiliki kunci. Tidak hanya itu, penelitian juga akan memperlihatkan script algoritma double caesar cipher dengan penggunaan tabel ASCII (American Standard Code for Information Interchange) dan juga dalam bahasa pemrograman .php (hypertext preprocessor), cara menjalankannya double caesar cipher, dan mencoba memasukan plaintext kedalam program double caesar cipher kemudian mengubahnya ke chipertext dan sebaliknya dari chipertext ke plaintext lagi.

Kata kunci : caesar cipher, double caesar cipher, kriptografi, ASCII

1. Pendahuluan

Penggunaan kriptografi di masa ini sudah sangat banyak. Penggunaan kriptografi adalah untuk meningkatkan keamanan penyampaian pesan dari satu instansi ke instansi lain. Kriptografi mengikuti perkembangan zaman. (Rahmatullah dan Shahih, 2015)

Pada masa ini, kriptografi memasuki era barunya yang disebut era kriptografi modern, dimana algoritma-algoritma yang dikembangkan memainkan dan mengolah bit dari pesan yang hendak dienkripsi. Semakin banyaknya penggunaan komputer digital merupakan salah satu faktor yang mendorong terjadinya perkembangan kriptografi untuk menjaga kerahasiaan informasi digital. (Silvanus dan Putri, 2015)

Caesar Cipher merupakan salah satu algoritma cipher tertua dan paling diketahui

dalam perkembangan ilmu kriptografi. Caesar Cipher merupakan salah satu jenis cipher substitusi yang membentuk cipher dengan cara melakukan penukaran karakter pada plainteks menjadi tepat satu karakter pada chiperteks. Teknik seperti ini disebut juga sebagai chiper abjad tunggal. (Anonymous, 2011)

Dalam penelitian ini akan menunjukan rancangan algoritma double caesar cipher, guna untuk pengamanan database yang diberikan form user login database agar lebih aman. Double caesar cipher merupakan pengembangan dari algoritma caesar cipher yang simetris. Proses enkripsi dan deskripsinya memiliki kunci yang sama, masing-masing memiliki kunci.

Tidak hanya itu, penelitian juga akan memperlihatkan script algoritma double caesar cipher dengan penggunaan tabel ASCII (American Standard Code for Information Interchange) dan juga dalam bahasa pemrograman .php (hypertext preprocessor), lalu bagaimana cara menjalankannya double caesar cipher, yang terakhir adalah mencoba memasukan plaintext kedalam program double caesar cipher kemudian mengubahnya ke chipertext dan sebaliknya dari chipertext ke plaintext lagi.

2. Tinjuan Pustaka

Nassirudin dan Ramandhani (2015) dalam penelitiannya mengenai Algorima Block Cipher Baru diberi nama FC (Friendzone Cipher) mengemukakan fitur utama dari rancangan algoritma yang diajukan ini adalah penggunaan sejumlah relatif prima dari sebuah bilangan untuk menyamarkan pola. Analisis keamanan dilakukan dengan membandingkan frekuensi kemunculan huruf alphabet antara plainteks dengan cipherteks. Frekuensi kemunculan kata ditampilkan dalam representasi grafik batang untuk mempermudah pembandingan. Perbandingan frekuensi kemunculan huruf dari pesan asli dan cipherteks dengan modus ECB. Hasil eksperimen dan analisis menunjukkan algoritma yang diajukan mampu melakukan enkripsi dan dekripsi pesan dengan benar serta memiliki tingkat keamanan yang cukup baik.

Silvanus dan Putri (2015) dalam penelitiannya mengenai Calogerus Cipher Blok membahas mengenai rancangan algoritma cipher blok. Blok yang diambil berupa matriks berukuran 4x4 dengan tiap sel matriks berisi 8 bit dari plainteks yang sudah diubah ke bentuk biner. Matriks ini akan dibalik urutan baris dan kolomnya. Kunci yang dimasukkan harus sebesar 128 bit. Kunci ini akan dibagi per 16 bagian, yang bagian-bagian tersebut akan disubstitusi dengan berpatokan pada sebuah substitution box. Kunci baru tersebut akan dilakukan operasi XOR dengan kunci lama dan hasilnya akan disimpan sebagai baris pertama dari sebuah matriks substitusi. Baris-baris selanjutnya akan dibangun berdasarkan pengulangan langkah di atas. Hal ini dilakukan berulang-ulang sebanyak 16 kali. Selanjutnya blok awal yang berisi plainteks akan disubsitusi dengan mengacu pada matriks yang telah dibangun dan digeser per baris dan kolomnya. Setelah itu dilakukan operasi XOR terhadap blok tersebut dengan baris pertama dan baris terakhir dari matriks substitusi yang telah dibangun. Terdapat 3 mode operasi untuk algoritma ini, yaitu mode Electronic Code Book, Chain Block Cipher, dan Cipher Feedback. Jika dibandingkan dengan algoritma Rijndael, perbedaan waktu yang dihasilkan jauh berbeda. Namun hal ini dikarenakan adanya pembangkitan matriks substitusi berkalikali. Sementara dalam algoritma Rijndael, matriks substitusi yang digunakan bersifat konstan. Karena panjang kunci dalam algoritma kami sepanjang 128 bit, maka akan terdapat sebanyak 2128 kemungkinan kunci. Bila saat ini komputer sudah dapat mencoba 1 juta kunci tiap detiknya, maka dibutuhkan waktu sekitar 5.4 x 1024 tahun untuk mencoba semua kemungkinan kunci. kesimpulan jika dekripsi yang dilakukan dengan mode CFB dapat 10 hingga 25 kali lebih cepat dibandingkan dekripsi yang dilakukan dengan mode ECB dan CBC. Sementara enkripsi yang dilakukan dengan mode CFB lebih cepat dibanding mode ECB dan CBC meskipun tidak menunjukkan perbedaan yang signifikan.

Nugroho dan Erwin (2015) dalam penelitiannya ICBC “Inverse Circular Block Cipher” mengusulkan sebuah algoritma enkripsi dan dekripsi berbasis block cipher baru yang bernama <NAMA ALGO>. Block cipher ini menggunakan blok sepanjang 64-bit dan kunci yang panjangnya 128-bit. Salah satu cara yang sering digunakan untuk meningkatkan kekuatan enkripsi pada block cipher adalah metode confusion dan diffusion dari Shannon. Metode ini tidak dapat diserang dengan metode konvensional karena perubahan 1 bit pada plaintext, ciphertext, ataupun key akan menyebabkan perubahan drastis secara keseluruhan. Untuk menambah kekuatan enkripsi, digunakan jaringan Feistel. Jaringan Feistel menggunakan prinsip XOR dan membagi plaintext ke dalam 2 blok yang seimbang. Kedua blok tersebut saling mempengaruhi hasil yang satu dengan yang lain dalam setiap bit yang

dioperasikan yang menyebabkan jaringan ini mempunyai tingkat kerumitan yang tinggi.

Ophie dan Rikysamuel (2015) dalam penelitiannya RICHIE — A New Block Cipher Algorithm membahas algoritma enkripsi RICHIE, suatu algoritma enkripsi block cipher yang memanfaatkan Feistel Network, S-Box dan dapat digunakan dalam mode ECB, CBC, dan CFB. Pengenkripsian pada teknik ini dilakukan perblock, sehingga plain text dan key harus dipecah terlebih dahulu ke block-block yang sudah ditentukan. Pada mode CBC, pengenkripsian pada satu block plain text tidak akan menghasilkan cipher text yang sama pada block plain text yang sama, sehingga akan menambah kesulitan untuk seorang cryptanalyst untuk dapat memecahkan cipher text yang didapatkan. Algoritma yang diajukan memiliki kekuatan dari serangan known plain attack. Karena algorita ini melakukan proses pengulangan berkali-kali sehingga jika ketika menggunakan mode CBC, hasil enkripsi plaintext suatu blok tidak akan menghasilkan ciphertext yang sama. Selain itu proses iterasi yang dilakukanpun sangat banyak sehingga "pola" semakin samar dan akan menjadi sangat sulit untuk mencari kesamaan dari setiap plaintext dan cipher text. Bahkan jika menggunakan ECB, hasil enkripsi tidak akan menghasilkan ciphertext yang sama pula kecuali pada kasus khusus yaitu dimana jumlah karakter pada plaintext berkelipatan delapan, dimana pola pada plaintext sendiri berada pada kelipatan delapan. Kesimpulan penelitian ini algoritma ini memang tidak menggunakan algoritma yang tidak terlalu kompleks, namun memberikan keamanan yang cukup, juga tidak membutuhkan waktu komputasi yang cukup lama. Untuk memecahkannya dengan Brute Force, membutuhkan waktu yang tidak mungkin.

2.1 Konsep Dasar Kriptografi

Kriptografi adalah suatu teknik matematika yang berhubungan dengan aspek-aspek pengamanan informasi seperti data confidentiality, data integrity dan data authentication. Cryptographic algorithm adalah fungsi matematika yang digunakan untuk enkripsi dan dekripsi. Terdapat dua fungsi yang saling berhubungan yaitu satu untuk enkripsi dan satu lagi untuk dekripsi. Cryptanalysis adalah ilmu untuk mendapatkan plaintext pesan tanpa harus mengetahui kunci secara wajar. (Setiadi, 2004)

Enkripsi merupakan proses pengkodean sebuah pesan sehingga isi dari pesan tersebut tidak diketahui. Dekripsi adalah proses kebalikan dari enkripsi yaitu mentransformasi pesan yang dienkripsi kembali menjadi bentuk semula. Sebuah sistem enkripsi dan dekripsi disebut cryptosystem. Bentuk asli dari sebuah pesan disebut plaintext dan bentuk asli yang dienkripsi disebut chipertext. (Setiadi, 2004)

2.2 Ancaman Keamanan

Terjadi banyak petukaran informasi setiap detiknya di internet. Juga banyak terjadi pencurian atas informasi oleh pihak ketiga. Ancaman keamanan yang terjadi terhadap informasi adalah : (Ariyus. 2008)

a. Interruption, yakni suatu ancaman terhadap ketersediaan suatu informasi, dari yang asalnya ada menjadi tidak ada atau rusak (ancaman terhadap aspek keamanan availability).
b. Interception, yakni suatu ancaman keamanan komputer terhadap kerahasian informasi, sehingga informasi tersebut menjadi diketahui atau diakses oleh orang lain yang tidak berhak.
c. Modification, yakni ancaman terhadap keaslian suatu informasi, yang mengakibatkan informasi yang diperoleh menjadi tidak asli karena telah mengalami perubahan/modifikasi oleh orang lain.
d. Fabrication, yakni suatu ancaman keamanan komputer pemalsuan informasi yang kita peroleh, sehingga kita menyangka bahwa informasi yang kita peroleh adalah asli padahal merupakan hasil tiruan informasi atau informasi palsu.

2.3 Algoritma kriptografi

Terjadi banyak petukaran informasi setiap detiknya di internet. Juga banyak terjadi pencurian atas informasi oleh pihak ketiga. Ancaman keamanan yang terjadi terhadap informasi adalah : (Ariyus. 2008)

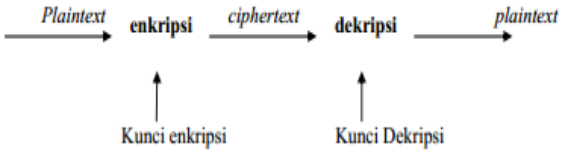
a. Enkripsi merupakan hal yang sangat penting dalam kriptografi, merupakan pengamanan data yang dikirimkan agar terjaga kerahasiannya. Pesan asli disebut plaintext, yang diubah menjadi kode-kode yang tidak dimengerti. Enkripsi bisa diartikan dengan cipher atau kode. Untuk mengubah teks asli ke bentuk teks kode digunakan algoritma yang dapat mengkodekan data.
b. Mengkodekan data.Dekripsi merupakan kebalikan dari enkripsi. Pesan yang telah dienkripsi dikembalikan ke bentuk asalnya (plaintext) disebut dengan dekripsi.

c. Kunci yang dipakai untuk melakukan enkripsi dan dekripsi. Kunci terbagi menjadi dua bagian yaitu kunci rahasia (private key) dan kunci umum (public key).

Biasanya algoritma kriptografi dapat dinotasikan sebagai berikut :

- Plaintext(M),
- Ciphertext(C),
- Enkripsi (fungsi E),
- Dekripsi (fungsi D).

Kriptografi itu sendiri terdiri dari dua proses utama yakni proses enkripsi dan proses dekripsi. Seperti yang telah dijelaskan di atas, proses enkripsi mengubah plaintext menjadi ciphertext (dengan menggunakan kunci tertentu) sehingga isi informasi pada pesan tersebut sukar dimengerti. Adapun alur dari proses enkripsi dan dekripsi pada kriptografi dapat dilihat pada gambar 2.1



Gambar 2.1 Diagram Proses Enkripsi dan Deskripsi

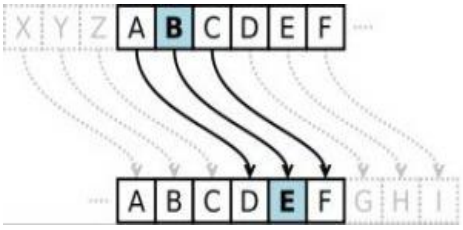
2.4 Algoritma Caesar Cipher

Substitusi kode yang pertama dalam dunia penyandian dikenal dengan Kode Kaisar, karena penyandian terjadi pada saat pemerintahan Yulius Caesar dengan algoritma ROT3. (Anonymous, 2015)

Inti dari algoritma kriptografi Caesar Cipher adalah melakukan pergeseran terhadap semua karakter pada plainteks dengan nilai pergeseran yang sama. Langkah-langkah untuk membentuk chiperteks dengan Caesar Cipher adalah : (Anonymous, 2015)

a. Menentukan besarnya pergeseran karakter yang digunakan dalam membentuk cipherteks ke plainteks.
b. Menukarkan karakter pada plainteks menjadi cipherteks dengan berdasarkan pada pergeseran yang telah ditentukan sebelumnya.

Misalkan pergeseran = 3, maka huruf A menjadi huruf D, huruf B menjadi huruf E, dan seterusnya.



Gambar 2.2 Pergeseran Caesar Cipher

Jika pergeseran sebanyak tiga kali, maka kunci untuk dekripsinya adalah 3. Pergeseran kunci yang dilakukan tergantung keinginan pengiriman pesan contohnya a = 7, b = 9, dan seterusnya. Cara kerja sandi ini dapat diilustrasikan dengan membariskan dua set alfabet. Misalnya sandi Caesar dengan kunci 3 sebagai berikut :

Alfabet Biasa: ABCDEFGHIJKLMNOPQRSTUVWXYZ
Alfabet Sandi: DEFGHIJKLMNOPQRSTUVWXYZABC

Contoh penyandian sebuah pesan adalah sebagai berikut :

teks Biasa : KIRIM PASUKAN KE SAYAP KIRI
teks tersandi : NLULP SDVXNDQ NH VDBDS NLUL

Proses enkripsi dapat secara matematis menggunakan operasi modulus dengan mengubah huruf-huruf menjadi angka, A = 0, B = 1, sampai Z = 25. Proses enkripsi dari "huruf" x dengan pergeseran n dan proses pemecahan kode, hasil dekripsi secara matematis dituliskan dengan :

$$Enkripsi: c_i = E(p_i) = (p_i + k) \bmod 26 \quad (1)$$
$$Dekripsi: p_i = D(c_i) = (c_i - k) \bmod 26 \quad (2)$$
$$k = kunci\ rahasia$$

Kalau di caesar cipher menggunakan mod 26 sedangkan di ascii menggunakan mod 256 dan mempunyai rumus dan tabel ascii-nya seperti di bawah ini :

$$Enkripsi: c_i = E(p_i) = (p_i + k) \bmod 256 \quad (3)$$
$$Dekripsi: p_i = D(c_i) = (c_i - k) \bmod 256 \quad (4)$$
$$k = kunci\ rahasia$$

2.5 Alogaritma cipher blok

Algoritma cipher blok merupakan salah satu dari algoritma kriptografi modern. Algoritma ini beroperasi dalam mode bit. Kunci, plainteks, dan cipherteks diproses dalam rangkaian bit. Algoritma ini tetap menggunakan gagasan pada algoritma kriptografi klasik seperti subtitusi dan transposisi, tetapi lebih rumit dan sangat sulit untuk dipecahkan. Prinsipnya adalah sebagai berikut : (Juzar dkk, 2015)

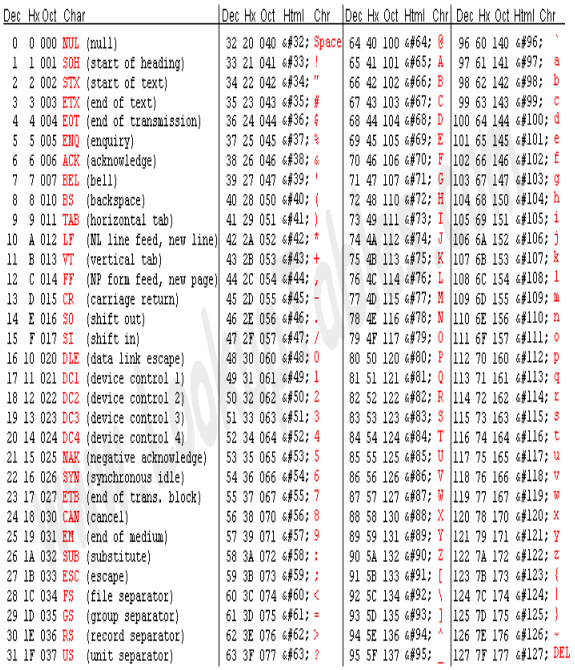
- Pesan dalam rangkaian bit dipecah menjadi beberapa blok.
- Padding bits : merupakan bit-bit tambahan jika ukuran blok terakhir tidak mencukupi panjang blok. Padding bits mengakibatkan ukuran plainteks hasil deskripsi sedikit lebih besar dari plainteks semula.
- Pesan juga dapat dinyatakan dalam kode heksadesimal.
- Bit-bit plainteks dibagi menjadi blok dengan panjang yang sama.
- Panjang kunci enkripsi = panjang blok
- Enkripsi dilakukan terhadap blok bit plainteks menggunakan bit-bit kunci.
- Algoritma enkripsi menghasilkan bok cipherteks yang panjangnya sama dengan panjang blok plainteks.

2.6 Tabel ASCII

Kode Standar Amerika untuk Pertukaran Informasi atau ASCII (American Standard Code for Information Interchange) merupakan suatu standar internasional dalam kode huruf dan simbol seperti Hex dan Unicode tetapi ASCII lebih bersifat universal, contohnya 124 adalah untuk karakter "|". Ia selalu digunakan oleh komputer dan alat komunikasi lain untuk menunjukkan teks. Kode ASCII sebenarnya memiliki komposisi bilangan biner sebanyak 7 bit. Namun, ASCII disimpan sebagai sandi 8 bit dengan menambakan satu angka 0 sebagai bit significant paling tinggi. Bit tambahan ini sering digunakan untuk uji prioritas. Karakter control pada ASCII dibedakan menjadi 5 kelompok sesuai dengan penggunaan yaitu berturut-turut meliputi logical communication, Device control, Information separator, Code extention, dan physical communication. Code ASCII ini banyak dijumpai pada papan ketik (keyboard) komputer atau instrument-instrument digital. (Anonymous, 2011)

Jumlah kode ASCII adalah 255 kode. Kode ASCII 0..127 merupakan kode ASCII untuk manipulasi teks; sedangkan kode ASCII 128..255 merupakan kode ASCII untuk manipulasi grafik. Kode ASCII sendiri dapat dikelompokkan lagi kedalam beberapa bagian:

- Kode yang tidak terlihat simbolnya seperti Kode 10(Line Feed), 13(Carriage Return), 8(Tab), 32(Space).
- Kode yang terlihat simbolnya seperti abjad (A..Z), numerik (0..9), karakter khusus (~!@#$%^&*()_+?:"{}).
- Kode yang tidak ada dikeyboard namun dapat ditampilkan. Kode ini umumnya untuk kode-kode grafik.



Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Source: www.LookupTables.com

Gambar 2.3 Tabel ASCII

3. Metodologi Penelitian dan Rancangan Sistem

3.1 Metodologi

Melakukan studi pustaka penelitian terdahulu tentang kriptografi modern. Penelitian terdahulu diambil lima tahun terkhir yang memiliki nilai keterbaruan.

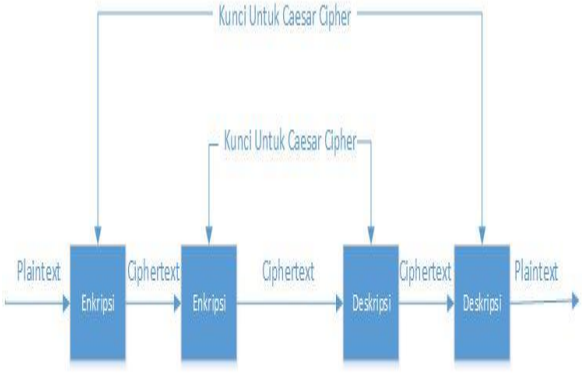
3.2 Analisis keamanan

Aspek keamanan merupakan aspek yang selalu diutamakan dalam pembuatan algoritma kriptografi. Keamanan mencakup aspek kerahasiaan, aspek integritas, serta aspek otentik. Cakupan ini merupakan tujuan utama dari ilmu kriptografi modern.

Menggunakan ascii yang mod 256 setidaknya lebih baik dripada menggunakan mod 26 karena mod 26 hanya menggunakan abjad A-Z sedangkan mod 256 sudah menggunakan 256 abjad, angka ataupun symbol sehingga lebih banyak kemungkinan ciphertext yang akan muncul. Meskipun begitu Double Caesar cipher masih memiliki bug yang harus ditutupi.

3.3 Analisis dan Rencangan Skema Double Caesar Cipher

Analisis dan rancangan memiliki skema double caesar cipher sebagai berikut :



Gambar 3.1 Skema Double Caesar Cipher

Dari plaintext akan dienkripsi ke bentuk ciphertext akan dienkripsi lagi. Lalu ciphertext akan dideskripsi ke bentuk ciphertext dan hadil ciphertext akan dideskripsi lagi ke dalam plaintext.

3.4 Script Rancangan Double Caesar Cipher

Script rancangan interface aplikasi double caesar cipher ditunjukkan dalam gambar 3.1 di bawah ini.

```
if(!empty($_POST)){ //if do action
    $plus = $_POST['n1']+1+$_POST['n2'];
    $string = $_POST['plaintext'];
    $newstring = $_POST['plaintext'];
    if(isset($_POST['btn_encrypt'])) {//jika melakukan encrypt
        for ($i=0;$i<strlen($string);$i++) {
            $ascii = ord($string[$i]);
            $ascii = $ascii + ($plus);
        if($ascii == 90) { //uppercase bound 90 'Z'
            $ascii = 65; //reset back to 'A' 65
        }
        else if($ascii == 122) { //lowercase bound 122 'z'
            $ascii = 97; //reset back to 'a' 97
        }
        else {
            $ascii++;
        }
        $newstring[$i] = chr($ascii);
        }
    }else
    if(isset($_POST['btn_decrypt'])) { //jika melakukan decrypt

        for ($i=0;$i<strlen($string);$i++) {
            $ascii = ord($string[$i]);
            $ascii = $ascii - ($plus+2);
        if($ascii == 90) { //uppercase bound 90 'Z'
            $ascii = 65; //reset back to 'A' 65
        }
        else if($ascii == 122) { //lowercase bound 122 'z'
            $ascii = 97; //reset back to 'a' 97
        }
        else {
            $ascii++;
        }
        $newstring[$i] = chr($ascii);
        }
    }
```

Gambar 3.2 Script Rancangan Double Caesar Cipher

3.5 Script Interface Tempat hasil Double Caesar Cipher

Script interface tempat hasil dari ciphertext atau plaintext pada aplikasi double caesar cipher sebagai berikut :

```
echo '<div style="border:1px solid gray">';
echo '<p><center>Plaintext : </strong>',$_POST['plaintext'],'</p>';
echo '<p><strong>angka pertama : </strong>',$_POST['n1'],'</p>';
echo '<p><strong>angka kedua : </strong>',$_POST['n2'],'</p>';
```

Gambar 3.3 Script Interface Tempat hasil Double Caesar Cipher

3.6 Cara Menjalankan Aplikasi Double Caesar Cipher

Double caesar cipher dirancang berbasis pemrograman web yang semua bisa menjalankan disemua platform dengan metode client-server. Berikut langkah-langkah menjalankan aplikasi double caesar cipher sebagai berikut :

1) Jalankan Aplikasi XAMPP Control Panel.
2) Aktifkan Apache.
3) Setelah itu copy folder "aplikasi-double-caesar-chiper".
4) Buka difolder XAMPP --> kemudian buka folder htdocs kemudian paste disitu.
5) Akses dibrowser dengan "Localhost/aplikasi-double-caesar-chiper".
6) Selesai.

3.7 Interface Double Caesar Cipher

Rancangan interface aplikasi double caesar cipher sebagai berikut :

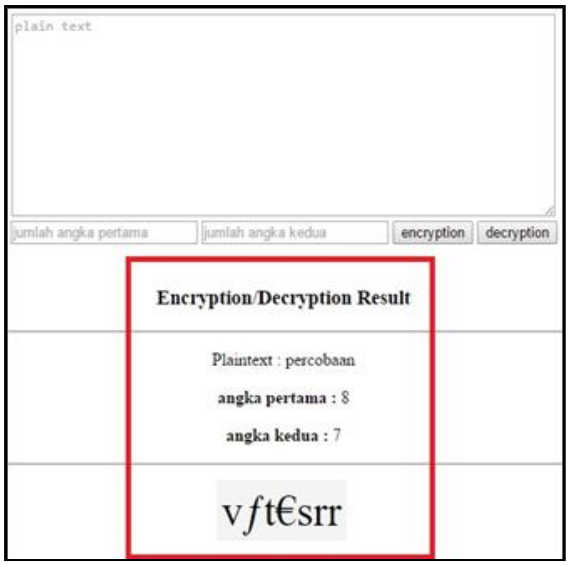
Gambar 3.4 Interface Double Caesar Cipher

Dari gambar 3.4 diatas dilihat bahwa terdapat kolom untuk memasukan plaintext yang akan diciphertext. Lalu dibawah terdapat kolom jumlah angkka petama dan jumlah angka kedua

yaitu berapa pergeseran caesar cipher yang akan ciphertext dan terdapat tombol encryption dan decryption yang berfungsi untuk menenkripsi atau sebaliknya mendeskripsi text.

3.8 Hasil Ciphertext/Plaintext

Hasil interface aplikasi double caesar cipher yang sudah di enkripsi atau dideskripsi sebagai berikut :



Gambar 3.5 Hasil Ciphertext/Plaintext

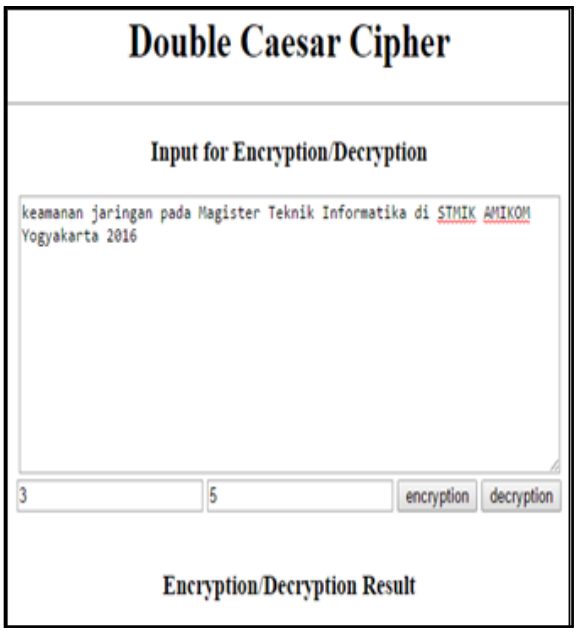
Dari gambar 3.5 merupakan hasil ciphertext atau plaintext yang plaintextnya : "percobaan", angka pertama : "8", angka kedua : "7" dan hasilnya adalah "vft€srr".

4. Hasil dan Pembahasan

Percobaan yang dilakukan terhadap plaintext terhadap aplikasi double caesar cipher sebagai berikut :

4.1 Interface Awal Double Caesar Cipher

Interface awal dari aplikasi double caesar cipher dengan menginputkan plaintext yang akan diubah menjadi ciphertext bisa dilihat dibawah ini sebagai berikut :



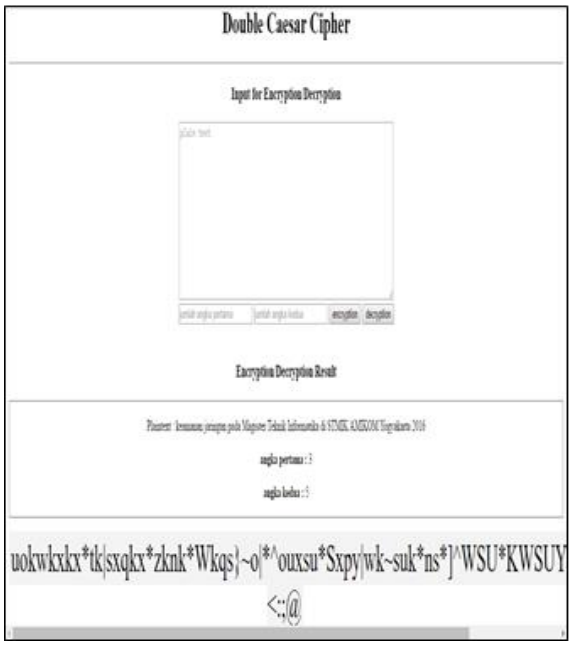
Gambar 4.1 Interface Awal Double Caesar Cipher

Pada gambar 4.1 akan dilakukan percobaan untuk menggubah plaintext ke ciphertext yang disebut enkripsi teks atau sebaliknya dari ciphertext ke plaintext yang disebut deskripsi teks. Dengan melakukan percobaan interface memasukan teks "keamanan jaringan pada Magister Teknik Informatika di STMIK AMIKOM Yogyakarta 2016" dengan angka pertama "3" dan angka kedua "5" lalu kan mengklik encryption supaya mendapatkan ciphertext.

4.2 Hasil Plaintext ke Ciphertext

Hasil plaintext ke ciphertext dari aplikasi double caesar cipher bisa dilihat dalam gambar 4.2. dibawah ini.

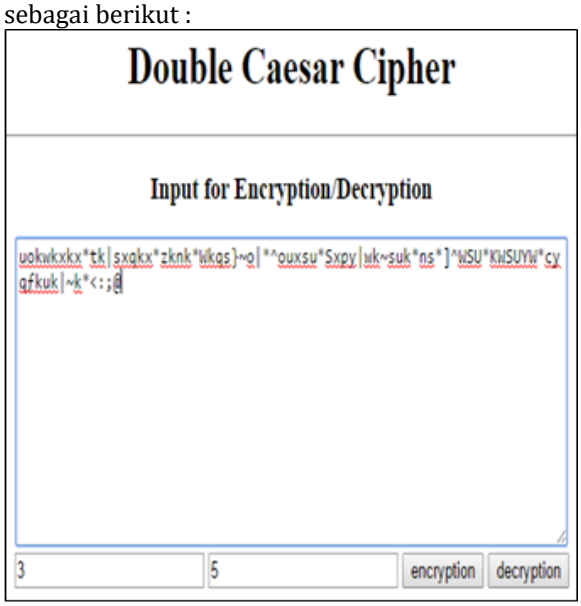
Pada gambar 4.2 menunjukan langkah berikutnya setelah mengklik enkripsi pada gambar 4.1 dengan plaintext sebelumnya dengan kalimat "keamanan jaringan pada Magister Teknik Informatika di STMIK AMIKOM Yogyakarta 2016" lalu hasil yang didapatkan sebuah ciphertext dengan kalimat "uokwkxkx*tk|sxqkx*zknk*Wkqs}~o|*^ouxsu*Sx py|wk~suk*ns*]^WSU*KWSUYW*cyqfkuk|~k*<:; @" dengan dilakukan dengan rumus angka pertama "3" dan angka kedua "5".



Gambar 4.2 Hasil Plaintext ke Ciphertext

4.3 Deskripsi Ciphertext ke Plaintext

Deskripsi dari aplikasi double caesar cipher dengan menginputkan ciphertext yang akan diubah menjadi plaintext bisa dilihat dibawah ini sebagai berikut :



Gambar 4.3 Deskripsi Ciphertext ke Plaintext

Pada gambar 4.3 akan dilakukan percobaan untuk menggubah ciphertext ke plaintext yang disebut deskripsi teks. Dengan melakukan inputan teks sesuai yang sudah dienkripsi lagi maka hasil enkripsi dari kalimat tersebut akan dideskripsikan lagi sebagai berikut kalimatnya "uokwkxkx*tk|sxqkx*zknk*Wkqs}~o|*^ouxsu*Sxpy|wk~suk*ns*]^WSU*KWSUYW*cyqfkuk|~k*<:;@" dengan angka pertama "3" dan angka kedua "5" lalu kan mengklik denskryption supaya mendapatkan plaintext yang diubah sebelumnya.

4.4 Hasil Ciphertext ke Plaintext

Hasil ciphertext ke plaintext dari aplikasi double caesar cipher bisa dilihat pada gambar 4.5 dibawah ini.

Gambar 4.5 Hasil Ciphertext ke Plaintext

Pada gambar 4.5 menunjukan langkah berikutnya setelah mengklik enkripsi pada gambar 4.3 dengan ciphertext sebelumnya dengan kalimat "uokwkxkx*tk|sxqkx*zknk*Wkqs}~o|*^ouxsu*Sxpy|wk~suk*ns*]^WSU*KWSUYW*cyqfkuk|~k*<:;@"" lalu hasil yang didapatkan sebuah plaintext dengan kalimat "keamanan jaringan pada Magister Teknik Informatika di STMIK AMIKOM Yogyakarta 2016" dengan dilakukan dengan rumus angka pertama "3" dan angka kedua "5".

5. Kesimpulan dan saran

5.1. Kesimpulan

Dari penulisan jurnal yang dibuat maka dapat disimpulkan sebagai berikut :

1) Double caesar cipher merupakan penggabungan dari caesar cipher yang dienkripsikan dua kali dan didiskripsikan dua kali juga sehingga disebut dengan double caesar cipher.
2) Substitusi kode yang pertama dalam dunia pengenkripsian dikenal dengan kode caisar, karena deskripsi ini terjadi pada pemerintahan Yulius Caesar dengan

algoritma ROT3 dan juga algoritma tertua dalam perkembangan ilmu kriptografi.

3) Caesar cipher pada awal di temukan menggunakan mod 26 sedangkan di ascii menggunakan mod 256.

5.2. Saran

Untuk meningkatkan keamanan kriptografi khusus pada double caesar cipher maka saran-saran dari prnulisan jurnal ini sebagai berikut :

1) Kriptografi Double caesar cipher perlu mengembangkan lebih baik lagi sehingga bug yang masih ada bisa diatasi.
2) Semoga menjadi refrensi bagi penelitian senanjutnya.

Daftar Pustaka:

[1] Ariyus, Dony., 2008, Pengantar Ilmu Kriptografi Teori, Analisis dan Implementasi, Penerbit Andi, Yogyakarta
[2] Juzar, Mario Tressa dan dkk., 2015, Cipher Blok JAF, Sekolah Teknik Elekto dan Teknik Informatika, ITB Bandung
[3] Nassirudin, Muhammad dan Ramandhani, Mohamad Rivai., 2015, FC – Algorima Block Cipher Baru. Sekolah Teknik Elekto dan Teknik Informatika, ITB Bandung
[4] Nugroho, Andreas Dwi dan Erwin., (2015), ICBC “Inverse Circular Block Cipher”, Sekolah Teknik Elekto dan Teknik Informatika, ITB Bandung.
[5] Kalaena, L. S., & Bagye, W. (2018). Implementasi Network Attached Storage (NAS) Menggunakan Freenas Pada STMIK Lombok. Jurnal Manajemen Informatika dan Sistem Informasi, 1(1), 6-10.
[6] Sunardi, Sunardi, and Sofiansyah Fadli. "SISTEM INFORMASI PENGOLAHAN DATA KELAPA SAWIT BERBASIS CLIENT-SERVER." *Jurnal Manajemen Informatika dan Sistem Informasi* 1.2 (2018): 23-28.
[7] Rahmatullah, Yusuf dan Khaidzir Muhammad Shahih., 2015, Block Cipher Menggunakan Permutasi Diagonal dan Feistel Berbasiskan AES-128, Sekolah Teknik Elekto dan Teknik Informatika, ITB Bandung
[8] Ophie, Edmund dan Rikysamuel., (2015), RICHIE — A New Block Cipher Algorithm, Sekolah Teknik Elekto dan Teknik Informatika, ITB BandungSetiadi, Budi., 2004, Analisis Sistem Keamanan Data Dengan menggunakan Metode DES dan Metode Gost, Bidang Khusus Kendali & Sistem Cerdas Program Magister Teknik Elektro, ITB Bandung
[9] Silvanus, Andarias dan Cilvia Sianora Putri., 2015, Calogerus Cipher Blok-Pengembangan Algoritma Cipher Blok dengan Matriks Substitusi Dinamis, Sekolah Teknik Elekto dan Teknik Informatika, ITB Bandung
[10] Bakti, Wira, Khairul Imtihan, and Ahmad Susan Pardiansyah. "Proxy Server dan Management Bandwidth Jaringan Komputer Menggunakan Mikrotik RB952Ui5ac2nD (Studi Kasus MA Ishlahul Ikhwan Nahdlatul Wathan Mispalah Praya)." *Jurnal Informatika dan Rekayasa Elektronik* 1.1 (2018): 44-49.
[11] Kalaena, Lalu Supriadi, and Wire Bagye. "Implementasi Network Attached Storage (NAS) Menggunakan Freenas Pada STMIK Lombok." *Jurnal Manajemen Informatika dan Sistem Informasi* 1.1 (2018): 6-10.
[12] Anonymous, https://sholeh012.wordpress.com/2011/10/03/caesar-cipher-dan-cipher-key/ diakses pada tanggal 02 Oktober 2018
[13] Anonymous, http://jurnal-bebas-q.unkris.my.id/eng/2811-2697/Ascii_49078_unkris_jurnal-bebas-q-unkris.html diakses pada tanggal 02 Oktober 2018
[14] Anonymous, website update pada 11 Maret 2015, Sandi Caesar. wikipedia: https://id.wikipedia.org/wiki/Sandi_Caesar diakses pada tanggal 02 Oktober 2018