Penerapan Algoritma Caesar Cipher Pada Aplikasi Pengaman Pesan Berbasis Website

Fajar Nurdiansyah[*1]
[1]Informatika, Universitas Siliwangi, Tasikmalaya, Indonesia
*Korespodensi author fajarnur27juni@gmail.com

Info Artikel

Diajukan: *24 April 2024*
Diterima: *6 Januari 2025*
Diterbitkan: *23 Januari 2025*

Keywords:
Caesar Cipher, Encryption, Cryptography, Message, Website

Kata Kunci:
Caesar Cipher; Enkripsi; Kriptograf; Pesan; Website

Abstract

Information technology is developing so fast that it does not rule out the possibility of various communications to get information. Many platforms provide communication features with message intermediaries. These messages need to be pro-tected from threats from unauthorized parties because they can contain sensitive data. Cryptography is the main method in securing messages, including the Caesar Cipher algorithm which is capable of encrypting and decrypting secret messages. The purpose of this research is to test the ability of the Caesar Cipher algorithm to secure messages. This research uses a literature study and implementation method which consists of three stages, including the literature study stage, program implementation stage, and application testing stage. The results of this research show that the application of the Caesar Cipher algorithm in message security applications can perform the encryption process well. With the help of ASCII code, the message can be scrambled and encoded easily without thinking about the number limit for the encryption key. There-fore, it can be concluded that Caesar Cipher is effective in simply performing encryption, this algorithm still has some weaknesses such as only being able to encrypt alphabetic characters and is vulnerable to being cracked using the brute force method.

Abstrak

Teknologi informasi berkembang sangat cepat sehingga tidak menutup kemungkinan terjadinya berbagai komunikasi untuk mendapatkan informasi. Banyak platform yang menyediakan fitur berkomunikasi dengan perantara pesan. Pesan inilah yang perlu dilindungi dari ancaman pihak tidak berkepentingan karena dapat memuat data sensitif. Kriptografi menjadi metode utama dalam mengamankan pesan, termasuk algoritma Caesar Cipher yang mampu melakukan proses enkripsi dan dekripsi pesan rahasia. Tujuan dari penelitian ini adalah untuk menguji kemampuan algoritma Caesar Ci-pher dalam mengamankan pesan. Penelitian ini menggunakan metode studi literatur dan implementasi yang terdiri dari tiga tahapan, diantaranya tahap studi literatur, tahap implementasi program, dan tahap pengujian aplikasi. Hasil penelitian ini menunjukkan bahwa penerapan algoritma Caesar Cipher pada aplikasi pengaman pesan dapat melakukan proses enkripsi dengan baik. Dengan bantuan kode ASCII, pesan dapat diacak dan disandikan dengan mudah tanpa memikirkan batasan angka untuk kunci enkripsi. Oleh karena itu, dapat diambil kesimpulan bahwa Caesar Cipher efektif dalam melakukan enkripsi secara sederhana, algoritma ini masih mempunyai beberapa kelemahan seperti hanya mampu mengenkripsi karakter alfabet saja dan rentan dipecahkan menggunakan metode brute force.

Cara mensitasi artikel:
F. Nurdiansyah. "Penerapan Algoritma Caesar Cipher Pada Aplikasi Pengaman Pesan Berbasis Website Sejahtera." *Jurnal Teknologi Informasi: Teori, Konsep, dan Implementasi (JTI-TKI), vol. 15, no. 2, pp. 91-98, Oktober 2024, https://doi.org/10.36382/jti-tki.v15i2.530*

PENDAHULUAN

Di zaman teknologi saat ini, penggunaan *internet* telah menjadi suatu keharusan dan tanggung jawab bagi banyak orang untuk mendapatkan serta menyebarkan informasi tanpa perlu berinteraksi secara langsung sehingga mengakibatkan munculnya berbagai platform jejaring sosial [1]. Ilmu pengetahuan dan teknologi berkembang dengan sangat cepat, terutama dalam bidang komunikasi, salah satu cara yang paling umum untuk berkomunikasi adalah melalui tulisan atau pesan [2]. Dalam platform jejaring sosial tidak menutup kemungkinan terjadi proses komunikasi atau pertukaran informasi berupa pesan antar pengguna. Pesan inilah yang sering kali tidak terlindungi dengan baik bahkan mengalami kebocoran kepada pihak tidak berwenang. Untuk meningkatkan keamanan pesan, kriptografi digunakan agar pesan terenkripsi tanpa menimbulkan kecurigaan, sehingga orang akan tetap percaya bahwa pesan tersebut tidak mengandung informasi rahasia [3].

Terdapat beragam metode untuk mengamankan data atau pesan, salah satunya adalah dengan memanfaatkan teknik kriptografi untuk menyamarkan data dan teknik *steganografi* untuk menyembunyikan informasi [4]. Kriptografi merupakan studi tentang teknik penyandian di mana teks asli diubah menggunakan suatu kunci enkripsi menjadi teks teracak yang sulit dibaca (*ciphertext*) oleh pihak yang tidak memiliki kunci dekripsi [5]. Sementara, dekripsi adalah proses mengembalikan pesan yang telah diubah ke keadaan aslinya, sehingga bisa dipahami seperti semula [6]. Dalam lingkungan komunikasi yang rawan terhadap ancaman seperti jaringan internet, penggunaan enkripsi data membantu menjaga kerahasiaan informasi pribadi, transaksi keuangan, data bisnis, dan informasi sensitif lainnya dari akses oleh pihak yang tidak berhak [7].

Seni kriptografi memiliki berbagai macam metode yang digunakan untuk melakukan proses penyandian data. *Cipher Caesar* adalah salah satu metode kriptografi tertua

yang termasuk dalam kategori substitusi, di mana setiap karakter dalam teks asli digeser secara seragam dengan jumlah tertentu untuk membentuk teks terenkripsi [8].

Beberapa penelitian telah dilakukan terkait dengan penerapan algoritma *Caesar Cipher* dalam mengamankan pesan atau data. Pertama, penelitian mengenai "Implementasi Kriptografi *Caesar Chiper* Pada Aplikasi Enkripsi Dan Dekripsi" yang dilakukan oleh Ardiansyah dan kawan-kawan pada tahun 2023 [2], mengatakan bahwa *Caesar Cipher* merupakan salah satu teknik kriptografi yang sederhana yang digunakan untuk melakukan enkripsi dan dekripsi teks. Penelitian tersebut melakukan proses enkripsi dan dekripsi dengan HTML yang bertujuan untuk menguji efektivitas dan efisiensi algoritma *Caesar Cipher* dalam mengamankan informasi. Penelitian dilakukan dengan cara studi literatur melalui tiga tahapan termasuk, tahap perancangan, tahap implementasi, dan tahap pengujian. Hasil dari penelitian ini Menunjukkan bahwa penerapan *Caesar Cipher* dalam pembuatan aplikasi sederhana untuk enkripsi dan dekripsi menggunakan bahasa pemrograman HTML dapat dilakukan dengan lancar.

Kedua, penelitian mengenai "Implementasi Kriptografi Dengan Algoritma *Caesar Cipher* Untuk Keamanan Data Microsoft Office Word Dan Excel" yang dilakukan oleh Alasi pada tahun 2019 [9]. Pada penelitian ini dikatakan bahwa enkripsi *caesar cipher*, karakter-karakter dalam pesan diubah menjadi nilai ASCII, lalu digeser sebanyak n karakter sesuai dengan nilai kunci enkripsi yang telah ditentukan. Metode penelitian ini dilakukan berdasarkan studi literatur terkait kriptografi dan algoritma *Caesar Cipher*. Hasil penelitian ini mengatakan bahwa teknik keamanan data dengan menggunakan metode pergeseran kunci berdasarkan keputusan pengguna dapat dilakukan dengan memanfaatkan kode ASCII.

Ketiga, penelitian mengenai "Penerapan Algoritma *Caesar Cipher* Dan Algoritma *Vigenere Cipher* Dalam Pengamanan Pesan Teks" yang dilakukan oleh Priyono pada tahun 2016 [3]. Pada penelitian ini paparkan tahapan atau cara kerja dari algoritma *Caesar Cipher* meliputi, menentukan nilai kunci (bilangan bulat positif), mengkonversi setiap karakter *plainteks* ke desimal atau nilai ASCII, proses enkripsi dilakukan dengan rumus Ci = (Pi+K) Mod 256, terakhir konversikan kembali setiap nilai Ci ke karakter. Pada penelitian ini disimpulkan bahwa dalam proses enkripsi, digunakan rumus *modulo* 256 dalam mode ASCII agar tidak hanya 26 karakter yang dapat dienkripsi, melainkan semua karakter yang ada pada tabel ASCII dapat digunakan.

Penelitian ini hadir bertujuan untuk membangun pemahaman terkait kriptografi menggunakan algoritma *Caesar Cipher*, sehingga dapat mengetahui mekanisme dan proses perhitungan algoritma *Caesar Cipher* dalam mengenkripsi pesan dan menerapkannya ke dalam aplikasi. Melalui algoritma sederhana dari caesar cipher akan dilakukan penelitian terkait proses enkripsi-dekripsi pesan rahasia melalui pendekatan nilai ASCII yang sudah terintegrasi teknologi *website* seperti HTML, PHP, dan *Framework* Tailwind CSS. Dengan harapan dapat membantu proses pengamanan pesan serta cenderung menjadi media pembelajaran bagi civitas akademik.

METODE

A. Kajian Teori

1) Kriptografi

Kriptografi memiliki asal-usul dari bahasa Yunani, yaitu "krupto" yang berarti tersembunyi, dan "grapho" yang berarti tulisan, yang merujuk pada tulisan yang disembunyikan. Kriptografi merupakan ilmu yang mempelajari teknik matematika yang berkaitan dengan keamanan informasi, seperti kerahasiaan data, keabsahan data, integritas data, serta autentikasi data. Secara umum, kriptografi terbagi menjadi dua bagian utama, yaitu enkripsi dan dekripsi [10].

Kriptografi bertujuan untuk memberikan layanan keamanan, yang terdiri dari beberapa aspek keamanan, kerahasiaan adalah layanan yang bertujuan untuk memastikan bahwa pesan tidak dapat dibaca oleh pihak yang tidak berhak, integritas data adalah layanan yang memastikan bahwa pesan tetap asli dan belum pernah dimanipulasi selama proses pengiriman, otentikasi adalah layanan yang berkaitan dengan identifikasi, baik untuk mengidentifikasi keaslian pihak yang berkomunikasi maupun untuk memastikan kebenaran identitas mereka, non-repudiation adalah layanan yang menjaga agar entitas yang berkomunikasi tidak dapat menyangkal tindakan atau pesan yang telah mereka kirimkan [11].

Kriptografi terdiri dari empat elemen utama, yakni [12]:

- Pesan Asli (*Plaintext*), yang merupakan pesan yang dapat dibaca.
- Pesan Terenkripsi (*Ciphertext*), yang merupakan pesan sandi atau pesan acak yang tidak dapat dibaca.
- Kunci (*Key*), yang digunakan untuk menerapkan teknik kriptografi.
- Algoritma, yang merupakan metode untuk melakukan enkripsi dan dekripsi.

Proses dasar dalam kriptografi dapat dikelompokkan menjadi dua bagian, yakni Enkripsi dan Dekripsi. Contoh teknik kriptografi klasik termasuk [12]:

- Substitusi, di mana teknik ini mengganti satu atau beberapa bit pada blok plainteks tanpa mengubah urutannya.
- Transposisi, yaitu teknik yang memindahkan posisi bit pada blok plainteks sesuai dengan aturan tertentu.

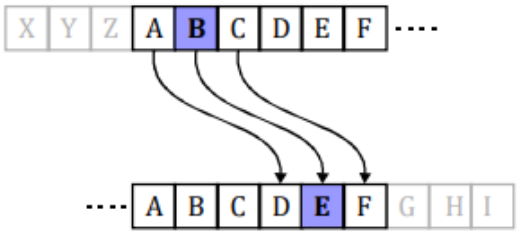
2) Caesar Cipher

Algoritma *Caesar Cipher* merupakan algoritma penyandian data yang paling sederhana, di mana data dienkripsi dan didekripsi dengan cara melakukan pergeseran sebanyak n [13]. *Caesar Cipher* adalah sebuah

metode substitusi *cipher* yang menggunakan pergeseran karakter dengan *modulo* 26 [1]. Metode ini adalah jenis sandi pengganti di mana setiap huruf dalam teks asli digantikan oleh huruf beberapa posisi di bawah alfabet yang tetap [9]. Menurut Alasi dalam [1] pada enkripsi *Caesar cipher*, karakter-karakter dalam pesan diubah menjadi nilai ASCII, lalu digeser sebanyak n karakter sesuai dengan nilai kunci enkripsi yang telah ditentukan.

Caesar Cipher memiliki tahapan sebagai berikut [14]:

- Menentukan jumlah pergeseran karakter yang akan digunakan untuk mengkonversi *plaintext* menjadi *ciphertext*.
- Mengubah setiap karakter pada *plaintext* menjadi *ciphertext* dengan mengikuti pergeseran yang telah ditentukan sebelumnya. Sebagai contoh, jika pergeserannya adalah 3, maka huruf A akan digantikan oleh huruf D, huruf B akan menjadi huruf E, dan seterusnya.



Gambar 1. Pergeseran Huruf Pada Algoritma Caesar Cipher

Pada **Gambar 1** terlihat adanya pergeseran sebanyak 3 (tiga) karakter, di mana A menjadi D, B menjadi E, dan C menjadi F, serta seterusnya. Sehingga proses enkripsi dari *Caesar Cipher* dapat dituliskan dengan notasi matematika sebagai berikut:

$$C = P + K \bmod 26$$

Dengan melakukan operasi yang berkebalikan, penerima pesan dapat melakukan proses dekripsi untuk menerima *plaintext*, yang dapat dijelaskan secara matematis sebagai berikut:

$$P = C - K \bmod 26$$

Di mana C = *Ciphertext*, P = *Plaintext*, K = Kunci, dan *mod* 26 merupakan jumlah alphabet keseluruhan.

Kekurangan dari *Caesar Cipher* adalah bahwa algoritma ini rentan terhadap serangan brute force, di mana penyerang mencoba semua kemungkinan kunci secara berturut-turut untuk mencari kunci yang benar [15].

3) Website

Web atau situs *web* adalah sekelompok halaman *web* yang terkait satu sama lain, sering kali disebut juga sebagai situs, situs *web*, atau portal [16]. Menurut Elgamar dalam [17] *website* adalah platform yang terdiri dari berbagai halaman yang saling terhubung melalui *hyperlink*, dimana fungsi utama *website* adalah untuk menyediakan informasi dalam berbagai bentuk seperti teks, gambar, video, audio, dan animasi, atau kombinasi dari semua elemen tersebut.

4) PHP

Menurut Peranginangin dalam [18] PHP adalah kependekan dari *Personal Home Page*, merupakan bahasa standar yang digunakan di dunia *website*. PHP merupakan bahasa pemrograman berbentuk *script* yang dijalankan di dalam *server web*. PHP dapat diartikan sebagai *Hypertext Preprocessor*. Ini adalah bahasa yang berjalan di sisi *server*, di mana hasilnya dapat ditampilkan pada klien. *Interpreter* PHP dalam mengeksekusi kode PHP di sisi *server* disebut sebagai *server-side*.

5) ASCII

Menurut Wijaya dan kawan-kawan dalam [19] ASCII (*American Standard Code for Information Interchange*) adalah sebuah standar global dalam representasi huruf dan simbol, seperti *Hex* dan *Unicode*, namun ASCII lebih penggunaannya lebih *universal*. ASCII digunakan oleh komputer dan perangkat komunikasi lainnya untuk merepresentasikan teks. Kode ASCII terdiri dari urutan *biner* 8-bit, dari 00000000 hingga 11111111. Terdapat total 256 kombinasi yang dihasilkan, mulai dari kode 0 hingga 255 dalam sistem bilangan desimal. Kode ASCII tertera pada **Gambar 2**.

Dec	Hex	Name	Char	Ctrl-char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	0	Null	NUL	CTRL-@	32	20	Space	64	40	@	96	60	`
1	1	Start of heading	SOH	CTRL-A	33	21	!	65	41	A	97	61	a
2	2	Start of text	STX	CTRL-B	34	22	"	66	42	B	98	62	b
3	3	End of text	ETX	CTRL-C	35	23	#	67	43	C	99	63	c
4	4	End of xmit	EOT	CTRL-D	36	24	$	68	44	D	100	64	d
5	5	Enquiry	ENQ	CTRL-E	37	25	%	69	45	E	101	65	e
6	6	Acknowledge	ACK	CTRL-F	38	26	&	70	46	F	102	66	f
7	7	Bell	BEL	CTRL-G	39	27	'	71	47	G	103	67	g
8	8	Backspace	BS	CTRL-H	40	28	(	72	48	H	104	68	h
9	9	Horizontal tab	HT	CTRL-I	41	29	)	73	49	I	105	69	i
10	0A	Line feed	LF	CTRL-J	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	VT	CTRL-K	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	FF	CTRL-L	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage feed	CR	CTRL-M	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	SO	CTRL-N	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	SI	CTRL-O	47	2F	/	79	4F	O	111	6F	o
16	10	Data line escape	DLE	CTRL-P	48	30	0	80	50	P	112	70	p
17	11	Device control 1	DC1	CTRL-Q	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	DC2	CTRL-R	50	32	2	82	52	R	114	72	r
19	13	Device control 3	DC3	CTRL-S	51	33	3	83	53	S	115	73	s
20	14	Device control 4	DC4	CTRL-T	52	34	4	84	54	T	116	74	t
21	15	Neg acknowledge	NAK	CTRL-U	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	SYN	CTRL-V	54	36	6	86	56	V	118	76	v
23	17	End of xmit block	ETB	CTRL-W	55	37	7	87	57	W	119	77	w
24	18	Cancel	CAN	CTRL-X	56	38	8	88	58	X	120	78	x
25	19	End of medium	EM	CTRL-Y	57	39	9	89	59	Y	121	79	y
26	1A	Substitute	SUB	CTRL-Z	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	ESC	CTRL-[	59	3B	;	91	5B	[	123	7B	{
28	1C	File separator	FS	CTRL-\	60	3C	<	92	5C	\	124	7C	|
29	1D	Group separator	GS	CTRL-]	61	3D	=	93	5D	]	125	7D	}
30	1E	Record separator	RS	CTRL-^	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	US	CTRL-_	63	3F	?	95	5F	_	127	7F	DEL

Gambar 2. Kode ASCII (Sumber: commfront.com)

B. Persiapan Sistem

Pada tahap persiapan sistem dilakukan analisis kebutuhan yang akan dilakukan dalam proses penelitian dan implementasi aplikasi pengamanan pesan diantaranya yaitu kebutuhan akan perangkat lunak dan kebutuhan perangkat keras.

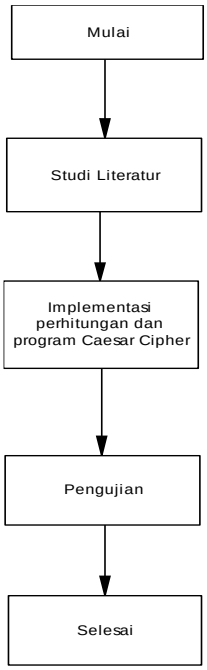
Perangkat lunak yang digunakan untuk pembuatan sistem enkripsi pesan berbasis *website* ini yaitu: Windows 10, XAMPP, Visual Studio Code, PHP, HTML, Tailwind CSS.

Perangkat keras yang digunakan untuk pembuatan sistem enkripsi pesan berbasis *website* ini adalah laptop

dengan spesifikasi sebagai berikut: Lenovo Ideapad Slim 1, *Processor* Intel Celeron N4020, *Memory* SSD 128 GB, *Memory* RAM 4 GB.

C. Analisis

Metode penelitian yang digunakan pada proses penelitian ini yaitu studi literatur dengan tujuan untuk mengumpulkan bahan-bahan sebagai referensi dan landasan meliputi kajian penelitian terdahulu, artikel, serta sumber referensi dari internet lainnya terkait penerapan algoritma *Caesar Cipher* pada pemrograman yang kemudian diikuti dengan tahap implementasi dan pengujian. Adapun alur penelitian diantaranya sebagai berikut:



Gambar 3. Alur Penelitian

Berdasarkan alur penelitian pada **Gambar 3** dapat dilakukan penguraian sebagai berikut:

a) Studi Literatur

Pada tahap ini peneliti melakukan proses kajian dan *review* terhadap beberapa jurnal penelitian mengenai implementasi algoritma *Caesar Cipher* dengan tujuan untuk memperoleh pemahaman yang akan digunakan selama penelitian berlangsung.

b) Implementasi

Pada tahap implementasi, peneliti mulai melakukan menerapkan materi mengenai algoritma *Caesar Cipher* hasil studi literatur ke dalam perhitungan manual dengan formula yang telah dijelaskan pada penelitian terdahulu serta melakukan implementasi pemrograman dalam bahasa PHP.

c) Pengujian

Pada tahap pengujian, peneliti mulai menguji program enkripsi dan dekripsi *Caesar Cipher* berbasis *website* dengan harapan program dapat bekerja sebagaimana mestinya.

HASIL DAN PEMBAHASAN

Pada proses enkripsi dan dekripsi dilakukan perhitungan secara manual atau matematis terkait dengan perubahan *plaintext* (P) menjadi *ciphertext* (C) melalui pendekatan nilai ASCII. Peneliti sedikit melakukan modifikasi formula atau rumus *Caesar Cipher* agar pengguna dapat menginput nilai *key* (K) lebih besar dari rentang indeks 1–26 atau lebih besar dari 26. Dengan logika apabila angka *key* yang diinputkan lebih dari 26 atau melebihi huruf Z, maka pergeseran karakter akan dimulai dari huruf A kembali. Contoh *key* yang dimasukkan adalah 29, maka 29–26 = 3 indeks dari awal yaitu huruf C. Proses ini diperoleh dengan menggunakan rumus matematis sebagai berikut:

$$C = ((P - 65 + K) \bmod 26) + 65$$

$$P = ((C - 65 - K) \bmod 26) + 65$$

A. Implementasi

1) Enkripsi

Pada tahap enkripsi akan peneliti melakukan penyandian pesan berupa *plaintext* (P) "INFORMATIKA UNSIL" menjadi *ciphertext* (C) dengan *key* (K) = 10, menggunakan rumus yang telah didefinisikan sebelumnya.

Tabel 1. Perhitungan Enkripsi

Perhitungan	ASCII
I : ((73 – 65 + 10) mod 26) + 65 = 83	S
N : ((78 – 65 + 10) mod 26) + 65 = 88	X
F : ((70 – 65 + 10) mod 26) + 65 = 80	P
O : ((79 – 65 + 10) mod 26) + 65 = 89	Y
R : ((82 – 65 + 10) mod 26) + 65 = 66	B
M: ((77 – 65 + 10) mod 26) + 65 = 87	W
A : ((65 – 65 + 10) mod 26) + 65 = 75	K
T : ((84 – 65 + 10) mod 26) + 65 = 68	D
I : ((73 – 65 + 10) mod 26) + 65 = 83	S
K : ((75 – 65 + 10) mod 26) + 65 = 85	U
A : ((65 – 65 + 10) mod 26) + 65 = 75	K
U : ((85 – 65 + 10) mod 26) + 65 = 69	E
N : ((78 – 65 + 10) mod 26) + 65 = 88	X
S : ((83 – 65 + 10) mod 26) + 65 = 67	C
I : ((73 – 65 + 10) mod 26) + 65 = 83	S
L : ((76 – 65 + 10) mod 26) + 65 = 86	V

Dari hasil perhitungan pada **Tabel 1**, maka diperoleh *ciphertext* (C): “SXPYBWKDSUK EXCSV”.

2) Dekripsi

Proses dekripsi merupakan tahap kebalikan dari enkripsi, yaitu mengubah kembali *ciphertext* menjadi *plaintext* agar penerima pesan dapat mengetahui makna dibalik pesan yang diterima.

Ciphertext (C) yang diketahui adalah “SXPYBWKDSUK EXCSV”, maka akan didekripsi menjadi *plaintext* (P) dengan *key* (K) yang sama yaitu 10, menggunakan rumus yang telah didefinisikan sebelumnya.

Hasil perhitungan pada **Tabel 2**, *plaintext* yang didapat adalah “INFORMATIKA UNSIL”.

Tabel 2. Perhitungan Dekripsi

Perhitungan	ASCII
S : ((83 – 65 – 10) mod 26) + 65 = 73	I
X : ((88 – 65 – 10) mod 26) + 65 = 78	N
P : ((80 – 65 – 10) mod 26) + 65 = 70	F
Y : ((89 – 65 – 10) mod 26) + 65 = 79	O
B : ((66 – 65 – 10) mod 26) + 65 = 82	R
W: ((87 – 65 – 10) mod 26) + 65 = 77	M
K : ((75 – 65 – 10) mod 26) + 65 = 65	A
D : ((68 – 65 – 10) mod 26) + 65 = 84	T
S : ((83 – 65 – 10) mod 26) + 65 = 73	I
U : ((85 – 65 – 10) mod 26) + 65 = 75	K
K : ((75 – 65 – 10) mod 26) + 65 = 65	A
E : ((69 – 65 – 10) mod 26) + 65 = 85	U
X : ((88 – 65 – 10) mod 26) + 65 = 78	N
C : ((67 – 65 – 10) mod 26) + 65 = 83	S
S : ((83 – 65 – 10) mod 26) + 65 = 73	I
V : ((86 – 65 – 10) mod 26) + 65 = 76	L

3) Program

Perhitungan manual diimplementasikan ke dalam program menggunakan bahasa pemrograman PHP. Seperti pada **Gambar 4**, jika tombol “enkripsi” ditekan, maka pesan dan kunci yang diinput oleh pengguna akan diproses untuk menghasilkan teks terenkripsi. Sebaliknya, jika tombol “dekripsi” ditekan, maka teks akan didekripsi kembali menjadi pesan asli berdasarkan kunci yang sama.

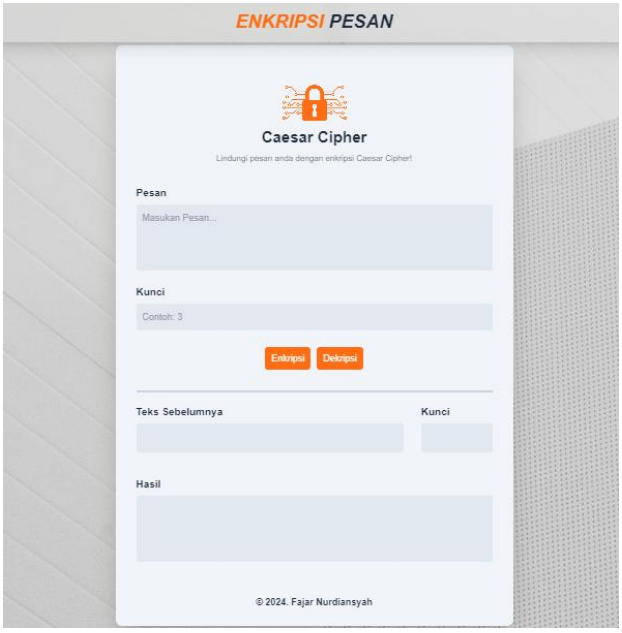
Program PHP ini dimulai dengan inisialisasi variabel $pesan, $key, dan $hasil. Terdapat dua blok kondisional utama yaitu enkripsi dan dekripsi. Dalam blok enkripsi, terdapat loop yang mengiterasi melalui setiap karakter pesan. Jika karakter adalah huruf, maka dilakukan pergeseran sesuai algoritma *Caesar Cipher*. Jika karakter bukan huruf, maka karakter tersebut tetap tidak berubah. Proses enkripsi dan dekripsi ini melibatkan tahap pengkonversian karakter menjadi kode ASCII dalam desimal dengan menggunakan fungsi *ord()*.

```php
<?php
$pesan = '';
$key = '';
$hasil = '';
if (isset($_POST['enkripsi'])) {
  $pesan = $_POST['pesan'];
  $key = $_POST['key'];
  // Proses enkripsi
  for ($i=0; $i < strlen($pesan); $i++) {
    $char = $pesan[$i];
    if (ctype_alpha($char)) {
      // Menentukan rentang ASCII yang sesuai berdasarkan huruf besar atau kecil
      $start = ord(ctype_upper($char) ? 'A' : 'a');
      $hasil .= chr(($start + ((ord($char) - $start + $key) % 26)));
    } else {
      // Tambahkan karakter yang tidak perlu dienkripsi
      $hasil .= $char;
    }
  }
} else if (isset($_POST['dekripsi'])) {
  $pesan = $_POST['pesan'];
  $key = $_POST['key'];
  // Proses dekripsi
  for ($i=0; $i < strlen($pesan); $i++) {
    $char = $pesan[$i];
    if (ctype_alpha($char)) {
      // Menentukan rentang ASCII yang sesuai berdasarkan huruf besar atau kecil
      $start = ord(ctype_upper($char) ? 'A' : 'a');
      $ascii = (ord($char) - $start - $key) % 26;
      // Pengecekan apakah ascii bernilai positif atau negatif
      if ($ascii <= -1 && $ascii >= -25) {
        $hasil .= chr($start + ($ascii + 26));
      } else {
        $hasil .= chr($start + $ascii);
      }
    } else {
      // Tambahkan karakter yang tidak perlu didekripsi
      $hasil .= $char;
    }
  }
}
?>
```

Gambar 4. Program PHP Caesar Cipher

B. Pengujian

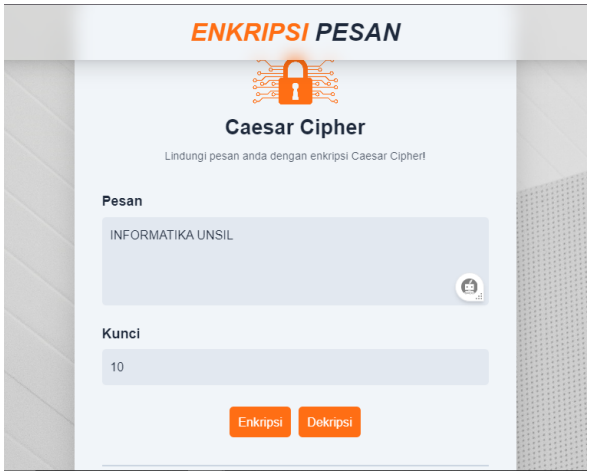
1) Landing Page



Gambar 5. Landing Page

Landing page pada **Gambar 5** merupakan tampilan utama dari *website* ketika pengguna mengunjungi portal *web* melalui url. Pada tampilan ini terdapat form untuk melakukan enkripsi dan dekripsi pesan.

2) Input Pesan Plaintext



Gambar 6. Input Plaintext

Setelah pengguna memasuki halaman utama dari *website*, pengguna dapat mulai melakukan enkripsi pesan. Sebagai contoh pada **Gambar 6**, pengguna menginputkan pesan "INFORMATIKA UNSIL" untuk dienkripsi dengan kunci (*key*) 10.

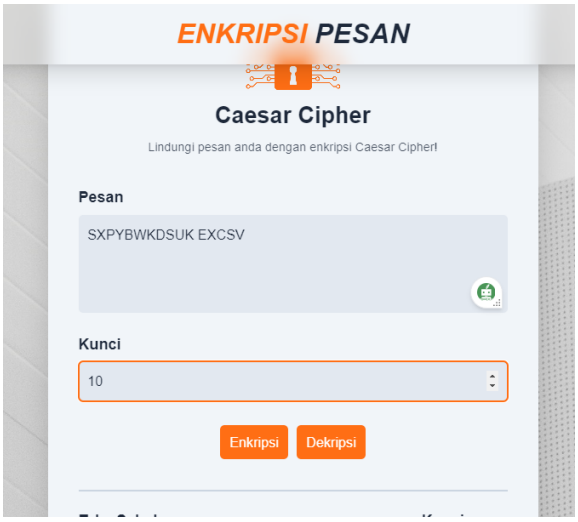
3) Enkripsi Pesan

Gambar 7. Hasil Enkripsi Plaintext

Setelah menginputkan pesan dan kunci, untuk melakukan proses enkripsi, pengguna dapat menekan tombol enkripsi sehingga seperti **Gambar 7** akan tampil *output* berupa enkripsi dari pesan (*ciphertext*), teks, dan kunci yang diinputkan sebelumnya.

4) Input Pesan Ciphertext

Untuk mengetahui makna sebenarnya dari pesan yang telah dienkripsi, seperti pada **Gambar 8** pengguna dapat melakukan proses dekripsi pesan dengan menginputkan kembali *ciphertext* hasil enkripsi yaitu "SXPYBWKDSUK EXCSV" dan kunci yang sama seperti sebelumnya.



Gambar 8. Input Ciphertext

5) Dekripsi Pesan

Gambar 9. Hasil Dekripsi Ciphertext

Setelah pengguna meng-*input*-kan *ciphertext*, untuk melakukan proses dekripsi, pengguna dapat menekan tombol dekripsi sehingga akan memperoleh *output* berupa makna sebenarnya dari pesan yang telah dienkripsi. Seperti pada **Gambar 9** hasil dekripsinya yaitu "INFORMATIKA UNSIL". Hal ini menunjukkan persamaan antara perhitungan manual dengan implementasi program.

KESIMPULAN DAN SARAN

Penyandian atau pengamanan pesan dapat dilakukan dengan beberapa metode kriptografi, salah satunya adalah algoritma *Caesar Cipher* yang mampu melakukan proses enkripsi pesan dengan cara menggeserkan posisi huruf asli berdasarkan kunci yang telah ditetapkan pada huruf alfabet.

Pesan yang dapat dienkripsi hanya dapat terdiri dari karakter-karakter alfabet karena algoritma *Caesar Cipher* hanya mampu melakukan pergeseran karakter untuk penyandian pesan yang termuat dalam 26 huruf alfabet.

Proses enkripsi dan dekripsi pesan menggunakan algoritma *Caesar Cipher* dapat dibantu dengan nilai ASCII untuk mempermudah pergeseran karakter berdasarkan nilai desimal yang dikonversi menjadi karakter pada tabel ASCII, serta penggunaan rumus modifikasi yang membuat angka kunci menjadi lebih dari 26.

Algoritma *Caesar Cipher* dapat bekerja dengan baik dalam melakukan enkripsi pesan, namun kurang tepat apabila digunakan untuk tujuan keamanan informasi tingkat tinggi karena algoritma ini tergolong algoritma sederhana yang mudah untuk dipecahkan oleh metode pemecah sandi seperti *brute force*.

Aplikasi pengaman pesan berbasis *website* dapat bekerja dengan baik dalam menampilkan hasil enkripsi dan dekripsi sehingga mampu dipergunakan untuk mengamankan pesan rahasia.

Saran yang diharapkan untuk penelitian selanjutnya adalah mampu mengembangkan aplikasi pengamanan pesan ini ke dalam tingkat yang lebih tinggi untuk ranah keamanan informasi dan komunikasi dengan cara menggunakan kunci asimetris, mengkombinasikan dua atau lebih algoritma kriptografi lain seperti *Vigenere Cipher*, RSA atau bahkan menggunakan penerapan algoritma kriptografi yang lebih modern seperti AES, DES, dan lain sebagainya dalam mengamankan pesan atau informasi.

REFERENSI

[1] M. H. Alfirdaus *et al.*, "Perancangan Aplikasi Enkripsi Deskripsi Mengunakan Metode Caesar Chiper Berbasis Web," *J. Tek. Mesin, Ind. Elektro Dan Inform.*, vol. 2, no. 2, pp. 64–76, 2023.

[2] Ardiansyah *et al.*, "Implementasi Kriptografi Caesar Chiper Pada *Aplikasi* Enkripsi Dan Dekripsi," *J. Ilm. Sist. Inf. dan Ilmu Komput.*, vol. 3, no. 1, pp. 105–112, 2023, doi: 10.55606/juisik.v3i1.438.

[3] P. Priyono, "Penerapan Algoritma Caesar Cipher Dan Algoritma Vigenere Cipher Dalam Pengamanan Pesan Teks," *J. Ris. Komput.*, vol. 3, no. 5, *pp*. 351–356, 2016.

[4] V. M. Hidayah, D. I. Mulyana, and Y. Bachtiar, "Algoritma Caesar Cipher atau Vigenere Cipher pada Pengenkripsian Pesan Teks," *J. Educ.*, vol. 5, no. 3, pp. 8563–8573, 2023, doi: 10.31004/joe.v5i3.1647.

[5] A. Suprayogi, N. S. Guna, F. R. Darmawan, M. A. Maulana, and R. D. Nurhaq, "Implementasi aplikasi kriptografi caesar cipher dengan php," *Researchgate*, no. January, 2021.

[6] I. M. Yusup, C. Carudin, and I. Purnamasari, "Implementasi Algoritma Caesar Cipher Dan Steganografi Least Significant Bit Untuk File *Dokumen*," *J. Tek. Inform. dan Sist. Inf.*, vol. 6, no. 3, pp. 434–441, 2020, doi: 10.28932/jutisi.v6i3.2817.

[7] R. M. S. Awalsyah, P. S. Harahap, and M. Dono, "Implementasi Caesar Cipher Dalam Mengenkripsikan Pesan Pada Serangan Man in," *J. JOCOTIS - J. Sci. Inform. Robot.*, vol. 1, no. 1, pp. 64–72, 2023.

[8] I. Gunawan, "Kombinasi Algoritma Caesar Cipher dan Algoritma RSA untuk pengamanan File Dokumen dan Pesan Teks," *InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)*, vol. 2, no. 2, pp. 124–129, 2018, doi: 10.30743/infotekjar.v2i2.266.

[9] T. S. Alasi, "Implementasi Kriptografi Dengan Algoritma Ceasar Cipher Untuk Keamanan Data Microsoft Office Word Dan Excel," *J. Inf. Komput. Log.*, vol. 1, no. 2, pp. 1–4, 2019, [Online]. Available: http://ojs.logika.ac.id/index.php/jikl/article/download/26/26

[10] E. Dika Santosa, "Implementasi Algoritma Caesar cipher dan Hill *cipher* pada TB MITA Jepara," *J. Teknol. Inf.*, 2015.

[11] R. Silalahi, S. Iin Parlina, I. Gunawan, and W. Saputra, "*Implementasi* Algoritma Caesar Cipher dan Algoritma RSA untuk Keamanan Data Surat Wasiat pada Kantor Notaris/PPAT Robert Tampubolon, S.H.," *J. Sos. dan Teknol.*, vol. 1, no. April, pp. 282–293, 2021, [Online]. Available: http://sostech.greenvest.co.id

[12] M. D. Irawan, "Implementasi Kriptografi Vigenere Cipher Dengan Php," *J. Teknol. Inf.*, vol. 1, no. 1, pp. 11–21, 2017, doi: 10.36294/jurti.v1i1.21.

[13] A. B. Nasution, "Implementasi Pengamanan Data Dengan *Menggunakan* Algoritma Caesar Cipher Dan Transposisi Cipher," *J. Teknol. Inf.*, vol. 3, no. 1, pp. 1–6, 2019, doi: 10.36294/jurti.v3i1.680.

[14] A. Basuki, U. Paranita, and R. Hidayat, "Perancangan Aplikasi Kriptografi Berlapis menggunakan Algoritma Caesar, Transposisi, Vigenere, dan Blok Chiper Berbasis Mobile," *Semin. Nas. Teknol. Inf. Dan Multimed. 2016*, vol. 1, no. 4, pp. 31–35, 2016.

[15] Y. D. Putri, R. Rosihan, and S. Lutfi, "Penerapan Kriptografi Caesar Cipher Pada Fitur Chatting Sistem Informasi Freelance," *JIKO (Jurnal Inform. dan Komputer)*, vol. 2, no. 2, pp. 87–94, 2019, doi: 10.33387/jiko.v2i2.1319.

[16] D. I. G. Hutasuhut, F. Rozi Lubis, F. Aulia Pratama, H. Ikhsanul Hasan, and H. Aldi Farisi, "IMPLEMENTASI ALGORITMA KRIPTOGRAFI UNTUK KEAMANAN DATA SISWA PADA SMK TUNAS KARYA BATANG KUIS BERBASIS WEB IMPLEMENTATION OF CRYPTOGRAPHIC ALGORITHMS FOR STUDENT DATA SECURITY AT TUNAS KARYA VOCATIONAL SCHOOL BATANG KUIS," *UNES J. Inf. Syst.*, vol. 8, no. 1, pp. 20–27, 2023, [Online]. Available: https://fe.ekasakti.org/index.php/UJIS

[17] A. Y. A. Putra, T. Willay, and A. Risky, "Perancangan Aplikasi *Pengamanan* Data Berbasis Web Dengan Penerapan Algoritma Kriptografi 3Des Dan Twofish," *J. InTekSis*, vol. 10, no. 2, p. 53, 2023.

[18] Y. Trimarsiah and M. Arafat, "Analisis dan Perancangan Website *Sebagai* Sarana," *J. Ilm. MATRIK*, vol. 19, no. 1, pp. 1–10, 2017.

[19] Sutardi, "Implementasi dan analisis kinerja algoritma shannon- fano untuk *kompresi* file text," *Din. J. Ilm. Tek. Mesin*, vol. 6, no. 1, pp. 53–60, 2014.