CESS (Journal of Computer Engineering System and Science) **p-ISSN :2502-7131**
Vol. 5 No. 2 Juli 2020 **e-ISSN :2502-714x**

BASE64 SEBAGAI KUNCI KEAMANAN PADA ONE TIME PAD (OTP)

Oris Krianto Sulaiman*, Khairuddin Nasution, Satria Yudha Prayogi

Universitas Islam Sumatera Utara
Jl. Sisingamangaraja No.Kelurahan, Teladan Barat, Kec. Medan Kota, Kota Medan, Sumatera Utara 20217
**oris.ks@ft.uisu.ac.id*

Abstrak— **Keamanan pesan dalam komunikasi menjadi hal yang sangat penting untuk menjaga kerahasiaan dan keutuhan pesan. Pesan yang dikirimkan harus tersampaikan secara utuh dan hanya tersampaikan sesuai dengan tujuan. *One Time Pad* atau OTP merupakan sebuah algoritma yang dapat melakukan enkripsi pesan menjadi *ciphertext* sehingga keamanan pesan dapat terjamin. OTP merupakan algoritma kriptografi simetri, yaitu kunci untuk enkripsi dan dekripsi adalah kunci yang sama. Untuk melakukan enkripsi maka panjang *plaintext* OTP harus sama dengan panjang kunci. Hal ini akan menyulitkan ketika terdapat panjang *plaintext* yang sangat panjang sehingga kunci juga harus menyesuaikan hal tersebut. Kunci OTP juga terdapat kelemahan jika sebuah kunci sudah digunakan lebih dari sekali. Oleh sebab itu dibutuhkan pembangkin kunci. Pada penelitian ini pembangkit kunci yang digunakan adalah *base64*. *Base64* digunakan untuk merubah bit dari *plaintext* yang akan digunakan oleh OTP. Pada penelitian ini *base64* dapat *encode plaintext* untuk kemudian digunakan sebagai kunci OTP sehingga berhasil menjadikan pembangkit kunci dari *encode base64*.**

Kata Kunci— ***Base64, One Time Pad, Kunci Keamanan, Keamanan Pesan*.**

Abstract— **Message security in communication is very important to maintain the confidentiality and integrity of messages. The message that is sent must be conveyed in its entirety and only delivered according to its purpose. One Time Pad or OTP is an algorithm that can encrypt messages into ciphertext so that message security can be guaranteed. OTP is a symmetric cryptographic algorithm, that is, the key for encryption and decryption is the same. To perform encryption, the OTP plaintext length must be the same as the key length. This will make it difficult when there is a very long length of plaintext, so the key must also adjust it. OTP keys also have a weakness if a key has been used more than once. Therefore a key builder is needed. In this research, the key generator used is base64. Base64 is used to change the bits of plaintext that will be used by OTP. In this study, base64 can encode plaintext to be used as an OTP key so that it succeeded in making the key generator from base64 encoding.**

Keywords— ***Base64, One Time Pad, Security Key, Message Security.***

I. PENDAHULUAN

Dalam berkomunikasi sangat penting menjaga kerahasiaan dan keutuhan dari pesan yang disampaikan. Banyak algoritma yang dapat menjadikan pesan tersebut menjadi teks acak atau sandi yang dikenal juga dengan sebutan *ciphertext*. Salah satu algoritma tersebut adalah *One Time Pad* atau OTP. Dalam proses enkripsi, algoritma OTP mengharuskan jumlah karakter teks asli atau *plaintext* sama dengan jumlah karakter kunci [1], [2].

Ada banyak teknik yang dapat digunakan dalam membuat kunci otomatis untuk OTP agar keamanan dan efisiensi pembuatan kunci jadi lebih baik. *Base64* merupakan sebuah algoritma yang merubah biner dari 8 bit menjadi 6 bit untuk setiap karakternya [3]. Pada penelitian terdahulu Rahim dkk. [4] dalam artikelnya yang berjudul "*Combination Base64 Algorithm and EOF Technique for Steganography*" membahas permasalahan keamanan pada *steganography* dan meningkatkan keamanan menggunakan *steganography* dengan kombinasi algoritma *Empirical Orthogonal Functions* (EOF), pada penelitian ini menghasilkan proses EOF dari encoding *base64*. Minarni [3] dalam artikelnya yang berjudul *"Implementasi Algoritma Base64 untuk Mengamakan SMS pada Smartphone"* membahas peningkatan keamanan SMS dan menghasilkan keamanan text SMS dengan encode *base64*. Rahim dkk. [5] pada artinya yang berjudul "*Base64, End of File and One Time Pad for Improvement Steganography Securit"* membahas peningkatan teknik *steganography* dengan menggunakan *base64* dan OTP, pada penelitian ini kunci yang digunakan dibuat tersendiri. Pada penelitian ini *base64* akan digunakan sebagai kunci yang ada pada OTP untuk memberikan kunci otomatis dan meningkatkan kunci keamanan pada OTP pada saat enkripsi pesan.

A. *One Time Pad* (OTP)

OTP merupakan algoritma kriptografi yang dapat mengenkripsi pesa dengan sangat baik. OTP

menggunakan kunci simetri, dimana kunci enkripsi dan dekripsi pesan menggunakan kunci yang sama. Panjang kunci dari OTP harus sama dengan panjang pesan yang akan di enkripsi atau dekripsi [1], [3], [6]–[8]. OTP menggunakan formula

$c = (p + k) \bmod n$

$p = (c – k) \bmod n$

jika *plaintext* oris dan kunci srie maka setelah di enkripsi menggunakan OTP akan menjadi giqw. Dalam hal ini OTP juga berpengaruh pada *n* yaitu berapa banyak jumlah karakter yang digunakan pada algoritma OTP.

B. Base64

Merupakan algoritma *encode* dan *decode* bit pada teks [3], [4], [9], [10]. Tiap-tiap karakter yang ada pada teks terdiri dari 8 bit dalam bilangan binner yang sesuai dengan ASCII. Pada *base64* akan digabung bit-bit tersebut dan disusun menjadi 6 bit dari tabel *base64.* Sehingga menhasilkan krakter yang berbeda dari teks sebelumnya.

Plaintext: oris

Encode base64: b3Jpcw==

II. METODE PENELITIAN

Metode yang digunakan pada penelitian ini menggunakan algoritma *One Time Pad* (OTP) dan *Base 64. Plaintext* atau teks asli akan di enkripsi menggunakan algoritma *One Time Pad* (OTP) dengan kunci yang panjangnya sesuai dengan teks yang akan di enkripsi.

$c = (p + k) \bmod n$

$p = (c – k) \bmod n$

dimana:

p = plaintext

c = ciphertext

k = key

n = number of letters used

Pada penelitian ini akan digunakan karakter yang berjumlah 26 yang diambil dari tabel ASCII. Tabel I memperlihatkan karakter yang digunakan pada penelitian ini berjumlah 26 karakter, sehingga modulo yang digunakan untuk enkripsi dari OTP adalah modulo 26.

TABEL I
ASCII TABLE

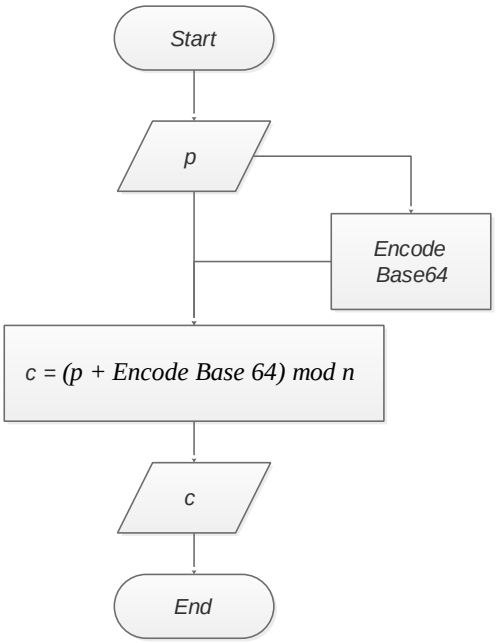
Idx	Binary	Char	Idx	Binary	Char
0	01100001	a	13	01101110	n
1	01100010	b	14	01101111	o
2	01100011	c	15	01110000	p
3	01100100	d	16	01110001	q
4	01100101	e	17	01110010	r
5	01100110	f	18	01110011	s
6	01100111	g	19	01110100	t
7	01101000	h	20	01110101	u
8	01101001	i	21	01110110	v
9	01101010	j	22	01110111	w
10	01101011	k	23	01111000	x
11	01101100	l	24	01111001	y
12	01101101	m	25	01111010	z

Kunci OTP yang ditentukan akan di *encode* oleh *base64* dan akan di *decode* kembali ketika proses dekripsi akan dilakukan. Tabel II memperlihatkan indeks base64.

TABEL II
BASE64 INDEX

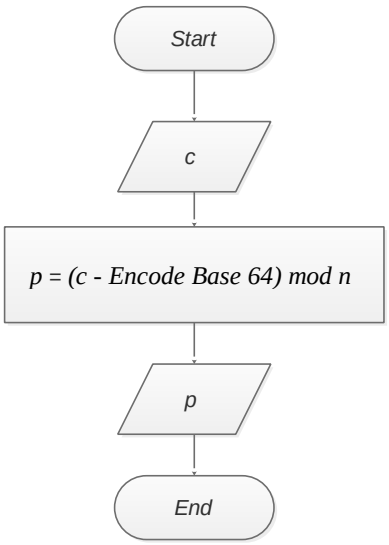
Index	Binary	Char	Index	Binary	Char
0	000000	A	33	100001	h
1	000001	B	34	100010	i
2	000010	C	35	100011	j
3	000011	D	36	100100	k
4	000100	E	37	100101	l
5	000101	F	38	100110	m
6	000110	G	39	100111	n
7	000111	H	40	101000	o
8	001000	I	41	101001	p
9	001001	J	42	101010	q
10	001010	K	43	101011	r
11	001011	L	44	101100	s
12	001100	M	45	101101	t
13	001101	N	46	101110	u
14	001110	O	47	101111	v
15	001111	P	48	110000	w
16	010000	Q	49	110001	x
17	010001	R	50	110010	y
18	010010	S	51	110011	z
19	010011	T	52	110100	0
20	010100	U	53	110101	1
21	010101	V	54	110110	2
22	010110	W	55	110111	3
23	010111	X	56	111000	4
24	011000	Y	57	111001	5
25	011001	Z	58	111010	6
26	011010	a	59	111011	7
27	011011	b	60	111100	8
28	011100	c	61	111101	9
29	011101	d	62	111110	+
30	011110	e	63	111111	/
31	011111	f	Padding		=
32	100000	g			

Berikut *flowchart* enkripsi dan dekripsi algoritma OTP dan Base 64 untuk keamanan kunci dari OTP.



Gbr. 1 Proses enkripsi

Pada gambar terlihat bahwa kunci yang digunakan untuk melakukan enkripsi oleh algoritma OTP berasal dari *encode* menggunakan *base64.* Setelah mendapatkan *encode base64* maka OTP akan diproses sehingga menghasilkan ciphertext.



Gbr. 1 Proses dekripsi

Untuk proses dekripsi maka *ciphertext* akan di proses dengan OTP dengan kunci yang sama yaitu *encode base* 64 dari *plaintext.* Sehingga akan menghasilkan *plaintext.*

III. HASIL DAN PEMBAHASAN

Percobaan yang dilakukan adalah dengan menggunakan *plaintext* "srie djoewita lisady" dan kunci akan dihasilkan dari *plaintext* tersebut berdasarkan proses yang dilakukan oleh *base64.*

Plaintext: srie djoewita lisady

Base64: c3JpZSBkam9ld2l0YSBsaXNhZHk=

Jumlah karakter *encode base64* disesuaikan dengan jumlah karakter yang ada pada *plaintext.* Karakter *plaintext* berjumlah 18 karakter maka *encode base64* yang diambil adalah 18 karakter juga. Sehingga menghasilkan *plaintext* dan *encode base64* berikut:

Plaintext: srie djoewita lisady

Base64: c3Jp ZSBkam9l d2l0YS

TABEL III
PROSES ENKRIPSI OTP DENGAN BASE64

Plaintext	***Base64***	***Ciphertext***
s	c	u
r	3	a
i	J	x
e	p	d
d	Z	v
j	S	k
o	B	y
e	k	e
w	a	i
i	m	t
t	9	w
a	l	l
l	d	j
i	2	a
s	l	s
a	0	a
d	Y	d
y	S	y

Dari proses perhitungan OTP dengan kunci yang dihasilkan dari *base64* maka didapatkan hasil dari *ciphertext:* uaxd vkyeitwl jasady. Untuk dekripsi maka digunakan kunci yang sama dari *base64* kemudian dilakukan proses OTP sehingga menghasilkan Kembali *plaintext:* srie djoewita lisady

TABEL IV
PROSES DEKRIPSI OTP DENGAN BASE64

Ciphertext	***Base64***	***Plaintext***
u	c	s
a	3	r
x	J	i
d	p	e
v	Z	d
k	S	j
y	B	o
e	k	e
i	a	w
t	m	i
w	9	t
l	l	a
j	d	l

a	2	i
s	l	s
a	0	a
d	Y	d
y	S	y

Dalam percobaan tersebut 18 karakter dari *plaintext* mendapatkan kunci yang dihasilkan dari *encode base64 plaintext.* Hal ini membantu dalam pembentukan kunci keamanan untuk *One Time Pad* (OTP), karena OTP membutuhkan kunci yang panjangnya sama dengan panjang dari karakter *plaintext.* Setelah mendapatkan kunci maka operasi OTP dapat dilakukan.

IV. PENUTUP

Penelitian menggunakan *base64* sebagai pembangkin bilangan untuk kunci pada *One Time Pad* (OTP) telah berhasil dilakukan. Hasil yang diperoleh menunjukkan bahwa kunci secara otomatis akan membentuk *base64* dan tidak memerlukan pembuatan kunci secara manual. Panjang kunci yang dihasilkan dari *encode plaintext base64* lebih banyak daripada panjang kunci *plaintext*, hal ini disebabkan karena bit yang digunakan per-karakter sebanyak 6 bit. Dalam proses OTP jumlah karakter kunci *encode base64* disesuaikan dengan jumlah karakter *plaintext* yang akan di enkripsi. Kunci OTP yang dibangkitkan dengan pembangkit bilangan acak akan lebih meningkatkan keamanan pesan, namun kelemahan model ini terletak pada *base64*, sehingga perlu adanya modifikasi *base64* agar pembangkit bilangan sulit untuk di ketahui kriptanalis.

REFERENSI

[1] J. Clawdia, N. Khairina, and M. K. Harahap, "Implementasi Algoritma Kriptografi One Time Pad (Otp) Dengan Dynamic Key Linear Congruential Generator (Lcg)," *Konferensi Nasional Teknologi Informasi dan Komputer (KOMIK)*, vol. I, pp. 12–14, 2017.

[2] N. Khairina and M. K. Harahap, "Analisis Algoritma One Time Pad Dengan Algoritma Cipher Transposisi Sebagai Pengamanan Pesan Teks," *SinkrOn*, vol. 1, no. 2, p. 58, Jun. 2017, doi: 10.33395/sinkron.v1i2.42.

[3] R. Minarni, "Implementasi Algoritma Base64 untuk Mengamakan SMS p ada Smartphone," *Building Informatics, Technology and Science (BITS)*, vol. 1, no. 1, pp. 28–33, 2019.

[4] R. Rahim *et al.*, "Combination Base64 Algorithm and EOF Technique for Steganography," *Journal of Physics: Conference Series*, vol. 1007, no. 1, 2018, doi: 10.1088/1742-6596/1007/1/012003.

[5] R. Rahim, R. Ratnadewi, D. Prayama, E. Asri, and D. Satria, "Base64, End of File and One Time Pad for Improvement Steganography Security," *IOP Conference Series: Materials Science and Engineering*, vol. 407, no. 1, 2018, doi: 10.1088/1757-899X/407/1/012161.

[6] R. Aulia, A. Zakir, and M. Zulhafiz, "Penerapan Algoritma One Time Pad & Linear Congruential Generator Untuk Keamanan Pesan Teks," *InfoTekJar (Jurnal Nasional Informatika dan Teknologi Jaringan)*, vol. 4, no. 1, pp. 37–41, Sep. 2019, doi: 10.30743/infotekjar.v4i1.1590.

[7] L. E. Pratiwi, R. Marwati, and I. Yusnitha, "Program Aplikasi Kriptografi Penyandian One Time Pad Menggunakan Sandi Vigenere," *Jurnal EurekaMatika*, vol. 2, no. 1, pp. 43–53, 2014.

[8] S. M. Hardi, D. Hamonangan, and M. Zarlis, "IMPLEMENTASI KRIPTOGRAFI HIBRID DENGAN ALGORITMA ELGAMAL DAN ALGORITMA ONETIME PAD(OTP) DALAM PENGAMANAN FILE AUDIO BERBASIS DESKTOP," *TECHSI - Jurnal Teknik Informatika*, vol. 10, no. 2, p. 129, Oct. 2018, doi: 10.29103/techsi.v10i2.903.

[9] E. Gunadhi and A. P. Nugraha, "Penerapan Kriptografi Base64 Untuk Keamanan URL (Uniform Resource Locator) Website Dari Serangan SQL Injection," *Jurnal Algoritma*, vol. 13, no. 2, pp. 391–398, 2017, doi: 10.33364/algoritma/v.13-2.391.

[10] S. Siswanto, M. Anif, and W. Gata, "Penerapan Algoritma Kriptografi TEA Dan Base64 Untuk Mengamankan Email Data Policy Asuransi," *Jurnal ELTIKOM*, vol. 2, no. 1, pp. 34–41, 2018, doi: 10.31961/eltikom.v2i1.44.