Jurnal Qua Teknika Vol.15 No.1 Maret 2025 ISSN
ISSN 2088-2424 (Cetak) : ISSN 2527-3892 (Elektronik)
Fakultas Teknik Universitas Islam Balitar, Blitar
Https://ejournal.unisbablitar.ac.id/index.php/qua;
E-mail: quateknikajurnal@gmail.com

Muhammad Najibulloh Muzaki , Rina Firliana , Rini Indriati , Anita Sari Wardani , Erna Daniati.
MULTILEVEL ENKRIPSI MENGGUNAKAN KOMBINASI ALGORITMA KRIPTOGRAFI BASE64 DAN PEPPER
Jurnal Qua Teknika, (2025), 15(1): 14-24

MULTILEVEL ENKRIPSI MENGGUNAKAN KOMBINASI ALGORITMA KRIPTOGRAFI BASE64 DAN PEPPER

Muhammad Najibulloh Muzaki [1)], Rina Firliana[2)] , Rini Indriati[3)] , Anita Sari Wardani [4)] , Erna Daniati [5)]
Fakultas Teknik dan Ilmu Komputer, Universitas Nusantara PGRI Kediri
Jln. Ahmad Dahlan No.76, Mojoroto, Kec. Mojoroto, Kota Kediri, Jawa Timur
email: [1]m.n.muzaki@unpkdr.ac.id
[2]rina@unpkediri.ac.id
[3] rini.indriati@unpkediri.ac.id
[4]anita@unpkediri.ac.id
[5]ernadaniati@unpkediri.ac.id

ABSTRAK
Pertukaran data melalui internet rentan terhadap masalah keamanan. Upaya pengamanan terhadap sistem dilakukan dengan berbagai metode. Salah satu metode yang digunakan adalah dengan melakukan enkripsi terhadap data. Pada penelitian ini menggunakan metode enkripsi kombinasi Base64 dan Pepper. Algoritma Base64 merupakan algoritma hashing dua arah dimana digunakan untuk mengenkripsi dan dekripsi data. Algoritma Pepper digunakan untuk menyisipkan beberapa karakter terhadap data. Kombinasi kedua algoritma tersebut menggunakan teknik multilevel. Dari hasil pengujian dapat diketahui bahwa penerapan kombinasi algoritma Base64 dan Pepper memberikan hasil enkripsi yang lebih kompleks jika dibandingkan dengan hasil dari algoritma Base64 murni. karena menghasilkan karakter yang tidak dapat dikenali.

Kata kunci— Keamanan, Enkripsi, Kriptografi, Base64, Pepper

ABSTRACT
Data exchange over the internet is susceptible to security issues. Efforts to secure the system are carried out using various methods. One method used is to encrypt data. This research uses a combination of Base64 and Pepper encryption methods. The Base64 algorithm is a two-way hashing algorithm which is used to encrypt and decrypt data. The Pepper algorithm is used to insert several characters into the data. The combination of the two algorithms uses multilevel techniques. From the test results it can be seen that the application of the combination of the Base64 and Pepper algorithms provides more complex encryption results when compared to the results from the pure Base64 algorithm. because it produces unrecognizable characters.

Keywords— Security, Encryption, Cryptography, Base64, Pepper

PENDAHULUAN

Teknologi internet telah menyentuh berbagai sisi aktivitas manusia, membawa dampak besar dalam kehidupan sehari-hari. Dengan adanya internet, pengguna dapat mengakses berbagai informasi, layanan, dan hiburan hanya dengan beberapa klik. Keunggulannya terletak pada kemudahan akses yang ditawarkannya, di mana internet tidak lagi terbatas hanya pada perangkat tertentu maupun lokasi tertentu. Pengguna kini bisa mengakses dunia maya melalui berbagai jenis perangkat seperti ponsel, laptop, atau bahkan perangkat pintar lainnya, yang memungkinkan mereka tetap terhubung kapan saja dan di mana saja. Hal ini memungkinkan kolaborasi, pembelajaran jarak jauh, dan transaksi bisnis yang lebih efisien, sekaligus memfasilitasi pertukaran informasi dalam waktu yang lebih singkat dan tanpa batas. Internet

memberikan keluasan akses bagi user karena dapat diakses dimanapun dengan menggunakan gadget yang dimiliki [1].

Pertukaran data melalui internet memiliki salah satu permasalahan klasik, yaitu mengenai jaminan keamanan terhadap data. Transaksi lalu lintas data melalui internet sangat rawan untuk diretas, disabotase serta dapat disalahgunakan. Potensi terhadap pencurian data akan terus selalu ada [2]. Data yang tidak diamankan berpotensi dapat diakses dan dimanfaatkan oleh pihak yang tidak bertanggung jawab dan masalah keamanan selalu menjadi perhatian penting pada setiap perusahaan. Keamanan data merupakan bentuk pengamanan aset penting bagi perusahaan [3].

Salah satu cara atau metode yang efektif untuk meningkatkan keamanan data adalah dengan melakukan enkripsi data. Enkripsi data merupakan proses mengubah informasi menjadi format yang tidak dapat dibaca oleh pihak yang tidak berwenang, sehingga hanya pihak yang memiliki kunci enkripsi yang tepat yang dapat mengakses atau mengembalikan data ke bentuk semula.. Algoritma enkripsi adalah metode atau formula yang bertujuan untuk melindungi atau mengamankan data [4].

Kriptografi merupakan cabang ilmu yang terus berkembang sangat pesat [5]. Berbagai algoritma enkripsi telah dikembangkan, dirumuskan, dan digunakan dalam berbagai bidang, terutama dalam dunia keamanan siber dan perlindungan data. Enkripsi data berfungsi sebagai mekanisme perlindungan informasi dengan mengubah teks asli (plain text) menjadi teks yang telah disandikan (ciphertext). Teks yang telah dienkripsi ini memiliki bentuk yang acak dan sulit dipahami tanpa adanya kunci dekripsi yang sesuai. Dengan demikian, enkripsi memastikan bahwa data tetap aman dan hanya dapat diakses oleh pihak yang memiliki otoritas atau izin yang sah. Penerapan enkripsi sangat penting dalam berbagai aspek kehidupan digital, seperti komunikasi daring, transaksi perbankan, serta penyimpanan data sensitif, guna mencegah akses yang tidak sah dan menjaga kerahasiaan informasi.

Terdapat dua karakteristik algoritma kriptografi menggunakan fungsi hash, yaitu fungsi satu arah (one-way function) dan ada fungsi dua arah (two-way function) [6]. Karakteristik utama dari one-way function adalah sifatnya yang tidak dapat dibalik, di mana teks asli (plain text) yang telah melalui proses enkripsi menjadi teks tersandi (ciphertext) tidak memungkinkan untuk didekripsi kembali ke bentuk semula. Fungsi ini banyak digunakan dalam berbagai bidang keamanan data, seperti hashing password, tanda tangan digital, dan autentikasi data, di mana integritas informasi lebih diutamakan dibandingkan kemampuan untuk mengembalikan data ke bentuk awalnya.

Sebaliknya, two-way function memiliki sifat yang memungkinkan proses enkripsi dan dekripsi dilakukan secara bolak-balik. Dalam mekanisme ini, teks asli (plain text) yang telah dienkripsi menjadi teks tersandi (ciphertext) dapat didekripsi kembali ke bentuk semula menggunakan kunci atau algoritma yang sesuai. Fungsi ini banyak diterapkan dalam sistem keamanan komunikasi, enkripsi file, serta perlindungan data dalam transaksi digital, di mana data yang dikodekan masih perlu diakses oleh pihak yang memiliki otorisasi. Dalam enkripsi tidak hanya menjamin keamanan data, tetapi juga menjamin keutuhan data [7].

Salah satu algoritma enkripsi yang memiliki karakteristik two-way function adalah Base64. Algoritma tersebut mengubah data ke dalam format ASCII menggunakan basis 64 karakter [8]. Jumlah basis 64 karakter yang digunakan tersusun dari karakter uppercase antara A – Z dengan total 26 karakter, karakter lowercase antara a sampai z dengan total 26 karakter, karakter angka 0 sampai 9 dengan total 10 karakter serta 2 karakter tambahan yaitu simbol (+) dan (/). Pada kasus tertentu dalam proses encoding jika memerlukan bit penggenap (padding), maka dapat ditambahkan dengan karakter (=) sebagaimana ditunjukkan pada tabel 1.

Penerapan algoritma Base64 juga dapat dimanfaatkan untuk pengkodean data [9]. Kelebihan lainnya adalah hasil pengkodean data dapat dengan mudah ditransfer menggunakan media apapun [10]. Seberapapun rumitnya ciphertext hasil enkripsi menggunakan algoritma Base64, tetap dapat di kembalikan ke bentuk semula [11]. Kemudahan tersebut menjadikan ciphertext yang dihasilkan sangat rentan karena mudah untuk dipecahkan sehingga tidak terjamin keamanan datanya. Algoritma Base64 memiliki kelemahan terhadap jaminan keamanan data [12].

TABEL 1. KARAKTER PADA ALGORITMA BASE64

Indeks	Karakter	Indeks	Karakter	Indeks	Karakter	Indeks	Karakter
0	A	16	Q	32	g	48	w
1	B	17	R	33	h	49	x
2	C	18	S	34	i	50	y
3	D	19	T	35	j	51	z
4	E	20	U	36	k	52	0
5	F	21	V	37	l	53	1
6	G	22	W	38	m	54	2
7	H	23	X	39	n	55	3
8	I	24	Y	40	o	56	4
9	J	25	Z	42	p	57	5
10	K	26	a	42	q	58	6
11	L	27	b	43	r	59	7
12	M	28	c	44	s	60	8
13	N	29	d	45	t	61	9
14	O	30	e	46	u	62	+
15	P	31	f	47	v	63	/
Karakter penggenap (*padding*) menggunakan simbol =							

Solusi yang dapat dilakukan untuk menutup kelemahan yang terdapat pada algoritma *Base64* telah banyak dilakukan, salah satunya adalah dengan mengkombinasikan algoritma tersebut dengan algoritma lainnya. Kombinasi algoritma dapat menghasilkan alur proses maupun hasil enkripsi yang berbeda dari algoritma alaminya. Dengan beragamnya algoritma kriptografi, dapat memudahkan dalam menerapkan teknik kombinasi antar agoritma [13].

Ide utama pada penelitian ini mengkombinasikan algoritma kriptografi *Base64* dengan teknik kombinasi yang belum pernah dilakukan pada penelitian sebelumnya. Algoritma kriptografi *Base64* akan dipadukan dengan algoritma *Pepper* untuk menghasilkan *ciphertext* yang tidak mudah untuk dipecahkan dengan algoritma dekripsi *Base64* murni. Algoritma *Pepper* jamak digunakan dalam *image processing*. Istilah *Salt* dan *Pepper* dalam *image processing* dapat berhubungan dengan *noise* pada citra atau enkripsi citra dengan merubah piksel asli sehingga citra asli dapat tersamarkan. Tampilan *Salt* dan *Pepper* berbentuk derau bintik piksel putih dan hitam pada citra [14]. Penambahan *Salt* dan *Pepper* dapat digunakan untuk mengenkripsi citra [15].

Dalam enkripsi teks, implementasi *Salt* dan *Pepper* dilakukan dengan cara menyisipkan teks tambahan ke dalam *plain text*, hal tersebut bertujuan untuk menyamarkan isi teks aslinya. *Salt* dan *Pepper* pada dasarnya adalah sama-sama berbentuk teks acak yang ditambahkan dalam *plain text*. Perbedan antara *Salt* dan *Pepper* adalah letak penyimpanannya, *Salt* disimpan dalam *database* bersama dengan *plain text* sedangkan *Pepper* tidak tersimpan di dalam *database* dan dapat tersembunyi di dalam kode program. Pada penelitian ini memilih *Pepper* karena memiliki kelebihan dalam keamanan, karena tidak dapat ditemukan dalam penyimpanan *database* yang pada dasarnya juga rentan untuk diretas.

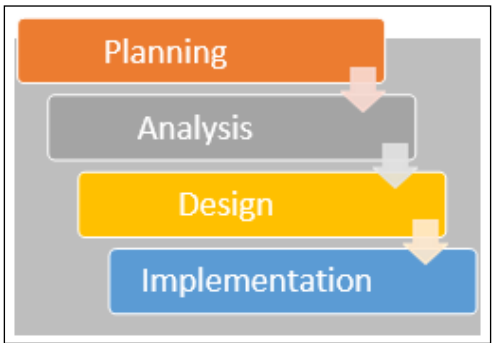
Perpaduan antara algoritma kriptografi Base64 dan Pepper diharapkan mampu menciptakan metode enkripsi yang unik dengan pendekatan algoritma yang berbeda. Kombinasi ini menawarkan fleksibilitas tinggi dalam perancangannya, memungkinkan berbagai strategi pengamanan data yang dapat disesuaikan dengan kebutuhan spesifik. Dengan menerapkan Base64 sebagai metode encoding yang dapat mengubah data menjadi format yang lebih aman untuk penyimpanan dan transmisi, serta mengombinasikannya dengan

Pepper—sebuah teknik tambahan dalam keamanan kata sandi yang menyisipkan nilai rahasia secara eksternal—keamanan data dapat ditingkatkan secara signifikan. Pendekatan ini tidak hanya membantu menjaga kerahasiaan informasi, tetapi juga memastikan keutuhan data dengan mempersulit upaya eksploitasi oleh pihak yang tidak berwenang.

METODE PENELITIAN

A. Metode Pengembangan Sistem

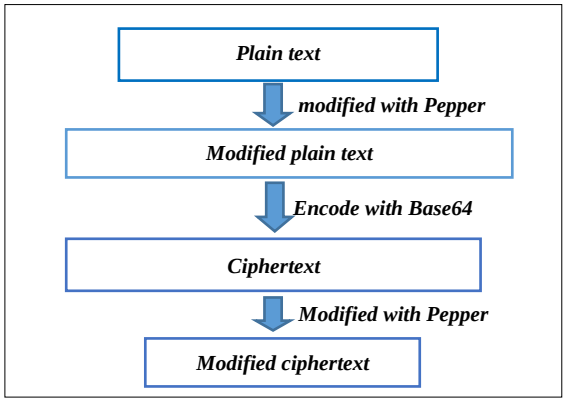
Pengembangan sistem multilevel enkripsi menggunakan kombinasi algoritma kriptografi base64 dan pepper menggunakan konsep *System Development Life Cycle* (SDLC) dengan metode yang digunakan adalah metode *Waterfall*, dengan tahapan yang ditunjukkan pada gambar 1.



Gambar 1. METODE WATERFALL

1. Tahap perencanaan (*Planning*) dengan merancang pengembangan sistem multilevel enkripsi menggunakan kombinasi algoritma kriptografi base64 dan pepper.
2. Tahap analisis (*Analysis*) dilakukan dengan menganalisis rancangan alur algoritma multilevel enkripsi menggunakan kombinasi algoritma kriptografi base64 dan pepper.
3. Tahapan desain (*Design*) dilakukan dengan mendesain rancangan kombinasi algoritma kriptografi Base64 dan Pepper.
4. Tahapan implementasi (*Implementation*) dilakukan dengan mengaplikasikan desain ke dalam bahasa pemrograman. Bahasa pemrograman yang digunakan, yaitu: PHP. Langkah selanjutnya adalah melakukan pengujian sistem.

B. Rancangan Algoritma Enkripsi



Gambar 2. RANCANGAN ALGORITMA MULTILEVEL ENKRIPSI KOMBINASI BASE64 DAN PEPPER

Rancangan algoritma dari kombinasi algoritma *Base64* dan *Pepper* dapat dilihat pada gambar 2. Secara garis besar terdiri dari 3 level enkripsi. Level enkripsi yang pertama adalah melakukan enkripsi menggunakan *pepper* terhadap plain text untuk menyamarkan *plain text*. Level ekripsi yang kedua adalah melakukan enkripsi menggunakan base64 terhadap hasil modifikasi *plain text* dengan *pepper* sehingga menjadi *ciphertext*. Level enkripsi yang ketiga adalah memodifikasi karakter pada *ciphertext* dengan menggunakan *pepper* untuk menyamarkan *ciphertext* agar tidak dapat di dekripsi dengan mudah.

Secara rinci tahapan tersebut terdiri dari beberapa urutan langkah.

1. Modifikasi *plain text* diawali dengan membangkitkan sejumlah karakter acak atau *pepper* untuk ditempatkan pada posisi tertentu pada *plain text*. Penempatan *pepper* dapat sisipkan di depan, tengah atau dibelakang karakter *plain text*, sehingga memiliki kemungkinan posisi sebanyak n+1, dimana n adalah jumlah karakter *plain text.* Panjang dari karakter *pepper* dapat bervariasi sehingga memiliki tingkat kerumitan karakter yang kompleks. Selain itu pemilihan karakter *pepper* dapat dipilih secara random sehingga dapat menyamarkan *plain text.* Modifikasi menggunakan *pepper* menghasilkan *plain text* termodifikasi dengan susunan teks yang sama sekali berbeda dari teks aslinya

   Contoh :
   - ***Plain text*** **:** X125
   - ***Proses modifikasi*** **:** **“**Tf”(*pepper*)+”X”(*plain text*)+”7r”(*pepper*)+”125”(*plain text*)+”7k”(*pepper*)
   - ***Modified Plain text*** **:** TfX7r1257k
2. *Plaintext* yang telah mengalami modifikasi dengan penambahan karakter *Pepper* selanjutnya diproses lebih lanjut melalui tahap encoding menggunakan algoritma enkripsi Base64. Proses ini bertujuan untuk mengubah *plaintext* yang telah diperkuat dengan karakter *Pepper* menjadi ciphertext yang lebih aman dan sulit untuk dikembalikan ke bentuk aslinya tanpa mengetahui karakter Pepper yang digunakan. Dengan demikian, kombinasi antara modifikasi plaintext dan encoding Base64 menghasilkan ciphertext yang lebih kompleks

   Contoh :
   - ***Modified Plain text*** **:** TfX7r1257k
   - **Proses encoding menggunakan Base64 :**

     Tahapan pertama dari *encoding Base64* adalah konversi masing-masing karakter ke dalam kode ASCII. Setiap nilai Kode ASCII yang diperoleh kemudian diterjemahkan ke nilai biner seperti ditunjukkan pada tabel 2.

KONVERSI KARAKTER KE ASCII DAN BINER

Karakter	ASCII	Biner
T	84	1010100
f	102	1100110
X	88	1011000
7	55	110111
r	114	1110010
1	49	110001
2	50	110010
5	53	110101
7	55	110111
k	107	1101011

- Kode biner yang dihasilkan dari proses konversi setiap karakter kemudian digabungkan secara berurutan hingga membentuk sebuah rangkaian biner yang panjang dan berkesinambungan. Rangkaian biner ini merepresentasikan keseluruhan data dalam format biner, yang selanjutnya dapat digunakan dalam tahap pemrosesan berikutnya.
  Kode biner :
  010101000110011001011000001101110111001000110001001100100011010100110111011011011
- Rangkaian kode biner yang telah terbentuk kemudian dipecah kembali menjadi kelompok-kelompok kecil, masing-masing kelompok dipecah dalam rentang blok dengan panjang 6 bit. Pembagian ini dilakukan untuk menyesuaikan format encoding yang digunakan dalam algoritma Base64, di mana setiap blok 6 bit akan dikonversi ke dalam karakter yang sesuai dalam tabel karakter Base64. Proses pemecahan ini bertujuan untuk memastikan bahwa data nantinya dapat dikodekan dengan benar, sehingga menghasilkan output yang lebih terstruktur dan sesuai dengan standar enkripsi base64.s
  Kode biner :
  010101, 000110, 011001, 011000, 001101, 110111, 001000, 110001, 001100, 100011, 010100, 110111, 011010, 11
- Setiap kelompok 6 bit yang dihasilkan dari proses pemecahan kemudian dikonversi ke dalam bentuk bilangan desimal. Nilai desimal ini digunakan untuk menentukan indeks karakter yang sesuai dalam tabel karakter Base64, sehingga setiap segmen dapat dikodekan dengan benar. Namun, jika pada proses pemecahan terdapat segmen terakhir yang hanya berisi 2 bit, maka perlu dilakukan penyesuaian dengan menambahkan *padding* sebesar 4 bit tambahan. Penambahan *padding* ini bertujuan agar segmen terakhir tetap memiliki panjang 6 bit, sehingga dapat diproses dengan cara yang sama seperti segmen lainnya dalam proses encoding Base64. Hasil konversi ke dalam karakter Base64 ditunjukkan pada tabel 3.

TABEL 3. KONVERSI 6 BIT KE DESIMAL DAN KARAKTER BASE64

Biner	Desimal	Karakter Base 64
010101	21	V
000110	6	G
011001	25	Z
011000	24	Y
001101	13	N
110111	55	3
01000	8	I
110001	49	x
001100	12	M
100011	35	j
010100	20	U
110111	55	3
011010	26	a
11+0000	48	w

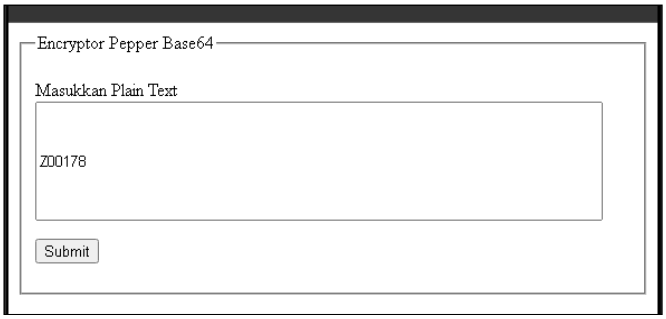
Biner	Desimal	Karakter Base 64
Padding		= =

- Hasil *encoding* dengan base64
  ***Modified Plain text* :** TfX7r1257k
  ***Ciphertext* :** VGZYN3IxMjU3aw==

3. Tahapan terakhir dalam proses ini adalah memodifikasi *ciphertext* dengan menambahkan karakter *Pepper*. Penambahan karakter *Pepper* bertujuan untuk meningkatkan keamanan data dengan membuat *ciphertext* yang dihasilkan menjadi lebih kompleks dan sulit untuk didekode menggunakan algoritma Base64 murni. Dengan adanya *Pepper*, hasil encoding tidak dapat langsung dikembalikan ke bentuk aslinya tanpa mengetahui karakter *Pepper* yang digunakan, sehingga menambah lapisan perlindungan terhadap potensi serangan atau upaya dekripsi yang tidak sah.
   Contoh :
   ***Ciphertext* :** VGZYN3IxMjU3aw==
   ***Modified Ciphertext* :** d3jKdVGZYN3IxMjU3aw==

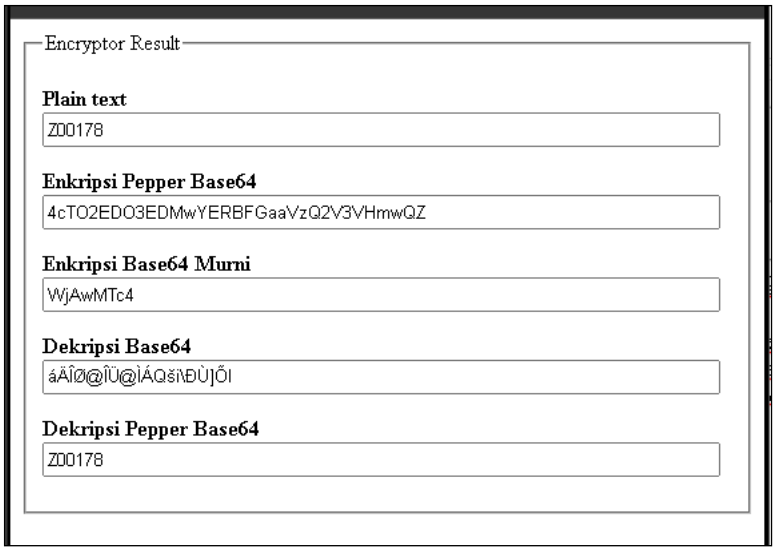
HASIL DAN PEMBAHASAN

Data uji untuk algoritma kombinasi Base64 dan Pepper menggunakan 10 buah plain text. Sebagai alat bantu pengujian, menggunakan hasil implementasi sistem enkripsi kombinasi Base64 dan Pepper menggunakan bahasa pemrigraman PHP. Tampilan halaman input plain text seperti ditunjukkan pada gambar 3.



Gambar 3. TAMPILAN INPUT PROGRAM ENKRIPSI BASE64 DAN PEPPER

Tampilan halaman hasil enkripsi baik yang menggunakan kombinasi *Base64* dan *Pepper*, maupun hasil enkripsi *Base64* murni akan ditampilkan beserta hasil dekripsi menggunakan base64 murni maupun hasil dekripsi algoritma *Base64* dan *Pepper*. secara detail tampilan tersebut ditunjukkan pada gambar 4.



Gambar 4. TAMPILAN INPUT PROGRAM ENKRIPSI BASE64 DAN PEPPER

Tabel 4 menyajikan rangkuman hasil percobaan yang dilakukan dengan menggunakan 10 *plaintext* sebagai data uji. Percobaan ini bertujuan untuk menganalisis hasil encoding menggunakan dua metode berbeda, yaitu algoritma *Base64* murni serta algoritma *Base64* yang dikombinasikan dengan *Pepper*. Dalam tabel tersebut, ditampilkan *plaintext* awal beserta *ciphertext* yang dihasilkan dari masing-masing metode *encoding*. Hasil percobaan ini memberikan gambaran mengenai perbedaan antara *ciphertext* yang dihasilkan oleh *Base64* tanpa tambahan *pepper* dan yang diperoleh dengan penambahan *pepper* untuk meningkatkan keamanan data.

TABEL 4. PENGUJIAN ENCODING

No	*Plain text*	*Enkripsi Base64*	Kombinasi *Base64* dan *Pepper*
1	Z00178	WjAwMTc4	4EjN5ADO3EDMwQFaGFTZapkZaJTNsrn06
2	P12878	UDEyODc4	zUDN4EDO3gjMx8EO4BzcQhkNBR2T8QBW6
3	R98234	Ujk4MjM0	5MjM3EDNzIDO5UHUz4kZSZzMutmYDCe7h
4	UV4234	VVY0MjM0	5MDNwUDNzIDNWVGMpVDbV1GTCFjWq49IP
5	TY3272	VFkzMjcy	5cTNxgjM3IzMZBXb2cDTURzSIdTY64uBw
6	E90324	RTkwMzI0	1YjM0EDNyMDM5MmQ1cnWF9ER2o3Vaz0Dg
7	B43241	QjQzMjQx	1EzM4ITM0IzM0U3cTdEeCp3MCt2U9hJf3
8	H38493	SDM4NDkz	ykzN1QzM5QDOzIHTtBXZIlFSuhXR1c2fm
9	U66585	VTY2NTg1	2EjM5gTN4UjN20WdyAzTVhjavdGUY5ArF
10	K34688	SzM0Njg4	zQDOxkDO4YDNzInNy00MLhUOCNDMh4isS

Dari hasil uji coba dapat diketahui bahwa *ciphertext* hasil *encoding* algoritma *Base64* murni dan *ciphertext* hasil algoritma *Base64* dan *Pepper* sama-sama dapat menyamarkan *plain text* dengan baik karena memiliki hasil yang sama sekali berbeda dengan *plain text*. *Ciphertext* hasil algoritma *Base64* dan *Pepper* menghasilkan rentang karakter yang lebih panjang jika dibandingkan dengan hasil enkripsi pada algoritma

Base64 murni, karena pengaruh dari penambahan karakter *Pepper*. Hasil tersebut menunjukkan bahwa kompleksitas enkripsi algoritma kombinasi *Base64* dan *Pepper* lebih rumit untuk dipecahkan.

TABEL 5. PENGUJIAN DEKRIPSI DATA

No	Kombinasi *Base64* dan *Pepper*	Decode *Base64*	Decode *Base64* dan *Pepper*
1	4EjN5ADO3EDMwQFa GFTZapkZaJTNsrn06	àHÍä□ÎÜ@ÌÁZTÙj ™h”Í²¹ô	Z00178
2	zUDN4EDO3gjMx8EO 4BzcQhkNBR2T8QBW 6	Í@Íà@ÎÞÌÇÁàÜB	P12878
3	5MjM3EDNzIDO5UHU z4kZSZzMutmYDCe7h	äÈÌÜ@ÍÎ€ÎåAÔÏ‰Iœ Ì°Ù˜	R98234
4	5MDNwUDNzIDNWV GMpVDbV1GTCFjWq4 9IP	Ä€Íâ@ÍÎ€Í[Œ'•• U• Þ¥QŽ5ý#	UV4234
5	5cTNxgjM3IzMZBXb2c DTURzSIdTY64uBw	åÄÍÆÌÜŒÌdÛÙÀÓQ Ò!ÔØë‹•	TY3272
6	1YjM0EDNyMDM5M mQ1cnWF9ER2o3Vaz0 Dg	請绦@腿捞渖惕芍□ 趨諨=———	E90324
7	1EzM4ITM0IzM0U3cT dEeCp3MCt2U9hJf3	ÔLÌà„ÌÐŒÌÑMÜMÑ -	B43241
8	ykzN1QzM5QDOzIHTt BXZIlFSuhXR1c2fm	ÊLÍÕ	H38493
9	2EjM5gTN4UjN20Wdy AzTVhjavdGUY5ArF	ØHÌæ——— ÍáHÍÛE• È	U66585
10	zQDOxkDO4YDNzInN y00MLhUOCNDMh4is S	Í□ÎÆ@Îá€ÍÎ‰ÍËM	K34688

Tabel 5 menyajikan perbandingan hasil dekripsi ciphertext yang dilakukan menggunakan dua metode berbeda, yaitu algoritma Base64 murni serta kombinasi algoritma Base64 dan *Pepper*. Dalam percobaan ini, ciphertext yang digunakan sebagai pembanding merupakan hasil dari proses encoding menggunakan kombinasi algoritma Base64 dan *Pepper*.

Hasil dekripsi dengan algoritma Base64 murni menunjukkan bahwa metode ini tidak mampu mengembalikan *ciphertext* ke bentuk *plaintext* aslinya dengan benar. Akibatnya, hasil dekripsi berupa karakter acak yang tidak sesuai dengan teks asli, menandakan bahwa proses deskripsi tidak dapat dilakukan dengan baik.

Sebaliknya, dekripsi menggunakan kombinasi algoritma Base64 dan *Pepper* berhasil mengembalikan *ciphertext* ke *plaintext* yang sesuai dengan data aslinya. Hal ini membuktikan bahwa metode kombinasi ini tetap menjaga keutuhan data, memastikan bahwa informasi yang dienkripsi dapat dipulihkan secara akurat tanpa kehilangan atau perubahan dalam isi pesan.

Gambar 5. PLAIN TEXT PADA TAUTAN

Pada gambar 5 menunjukkan pengujian untuk enkripsi data *plain text* yang ada dalam URL (*Uniform Resource Locator*). Dari gambar 5 diketahui bahwa data tersebut sangat mudah dibaca oleh pengguna, sehingga apabila diketahui pihak yang tidak bertanggung jawab maka dapat dimanfaatkan untuk kepentingan tertentu.

Gambar 6. HASIL ENKRIPSI TAUTAN MENGGUNAKAN KOMBINASI BASE64 DAN PEPPER

Gambar 6 menunjukkan hasil enkripsi pada URL menggunakan algoritma kombinasi *Base64* dan *Pepper*. Data menjadi lebih sulit dibaca dan dipahami maupun dipecahkan. Hal ini membuktikan bahwa implementasi algoritma kombinasi *Base64* dan *Pepper* dapat dilakukan untuk mengamankan data yang muncul pada URL sehingga dapat menjamin keamanan data tersebut.

SIMPULAN

Dari hasil penelitian yang telah dilakukan dapat disimpulkan bahwa, penerapan multilevel enkripsi menggunakan kombinasi algoritma *Base64* dan *Pepper* memberikan hasil enkripsi yang lebih kompleks jika dibandingkan dengan hasil dari algoritma *Base64* yang dijalankan secara murni.

Uji coba dekripsi memberikan hasil bahwa hasil penerapan multilevel enkripsi kombinasi algoritma *Base64* dan *Pepper* tidak dapat didekripsi dengan algoritma *Base64* murni, karena menghasilkan karakter yang tidak dapat dikenali. Meskipun begitu hasil enkripsi kombinasi algoritma *Base64* dan *Pepper* tetap dapat didekripsi tanpa merubah keutuhan dari *plain text*.

Penerapan multilevel enkripsi kombinasi algoritma *Base64* dan *Pepper* dapat diimplementasikan untuk mengamankan data yang dapat tampil pada URL. Sehingga data tidak dapat dengan mudah terbaca secara langsung. Kombinasi karakter pepper serta tingkat multilevel secara fleksibel dapat dimodifikasi sesuai dengan rancangan yang diinginkan karena tidak memiliki bentuk baku, sehingga setiap orang dapat memodifikasi kombinasi algoritma *Base64* dan *Pepper* sesuai versinya masing-masing. Penelitian selanjutnya dapat difokuskan untuk mengkombinasikan karakter *pepper* dengan algoritma enkripsi yang lain untuk lebih meningkatkan tingkat kerumitan hasil enkripsi.

UCAPAN TERIMA KASIH

Ucapan Terima kasih ditujukan kepada Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) Universitas Nusantara PGRI Kediri atas pendanaan hibah Penelitian Dosen.

REFERENSI

[1] M. I. Suri and A. S. Puspaningrum, "SISTEM INFORMASI MANAJEMEN BERITA BERBASIS WEB," Jurnal Teknologi dan Sistem Informasi, vol. 1, no. 1, pp. 8–14, Jun. 2020, doi: 10.33365/jtsi.v1i1.128.

[2] E. Setyawati, C. E. Widjayanti, R. R. Siraiz, and H. Wijoyo, "Pengujian keamanan komputer kriptografi pada surat elektronik berbasis website dengan enkripsi metode MD5," *Jurnal Manajamen Informatika Jayakarta*, vol. 1, no. 1, p. 56, Feb. 2021, doi: 10.52362/jmijayakarta.v1i1.367.

[3] S. Nurul, S. Anggrainy, and S. Aprelyani, "FAKTOR-FAKTOR YANG MEMPENGARUHI KEAMANAN SISTEM INFORMASI: KEAMANAN INFORMASI, TEKNOLOGI INFORMASI DAN NETWORK (LITERATURE REVIEW SIM)," vol. 3, no. 5, 2022, doi: 10.31933/jemsi.v3i5.

[4] A. Adil Yazdeen, S. R. M. Zeebaree, M. Mohammed Sadeeq, S. F. Kak, O. M. Ahmed, and R. R. Zebari, "FPGA Implementations for Data Encryption and Decryption via Concurrent and Parallel Computation: A Review," *Qubahan Academic Journal*, vol. 1, no. 2 SE-Articles, pp. 8–16, Mar. 2021, doi: 10.48161/qaj.v1n2a38.

[5] M. H. Saracevic *et al.*, "Data Encryption for Internet of Things Applications Based on Catalan Objects and Two Combinatorial Structures," *IEEE Trans Reliab*, vol. 70, no. 2, pp. 819–830, 2021, doi: 10.1109/TR.2020.3010973.

[6] S. Sulastri, R. Defi, and M. Putri, "Implementasi Enkripsi Data Secure Hash Algorithm (SHA-256) dan Message Digest Algorithm (MD5) pada Proses Pengamanan Kata Sandi Sistem Penjadwalan Karyawan," *Jurnal Teknik Elektro*, vol. 10, no. 2, pp. 70–74, Dec. 2018, doi: 10.15294/JTE.V10I2.18628.

[7] R. Andriyanto, K. Khairijal, and D. Satria, "Penerapan Kriptografi AES Class Untuk Pengamanan URL WEBSITE Dari Serangan SQL INJECTION," *JURNAL UNITEK*, vol. 13, no. 1, pp. 34–48, 2020, doi: 10.52072/unitek.v13i1.153.

[8] K. Algoritma Base, D. Caesar Cipher Pada Aplikasi Yudo Devianto, W. Gunawan, and B. Sukowo, "Kombinasi algoritma Base64 dan caesar cipher pada aplikasi," *Faktor Exacta*, vol. 17, no. 1, pp. 1979–276, May 2024, doi: 10.30998/FAKTOREXACTA.V17I1.20680.

[9] Y. Zhang *et al.*, "Information stored in nanoscale: Encoding data in a single DNA strand with Base64," *Nano Today*, vol. 33, 2020, doi: 10.1016/j.nantod.2020.100871.

[10] S. Supiyandi, H. Hermansyah, and K. A. P. Sembiring, "Implementasi dan Penggunaan Algoritma Base64 dalam Pengamanan File Video," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 4, no. 2, p. 340, Apr. 2020, doi: 10.30865/mib.v4i2.2042.

[11] A. Kodir and W. Pramusinto, "IMPLEMENTASI KRIPTOGRAFI DENGAN MENGGUNAKAN METODE RC4 DAN BASE64 UNTUK MENGAMANKAN DATABASE SEKOLAH PADA SDN GROGOL UTARA 10," *SKANIKA*, vol. 4, no. 1, pp. 7–14, Jan. 2021, doi: 10.36080/skanika.v4i1.884.

[12] 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, Nov. 2016, doi: 10.33364/ALGORITMA/V.13-2.391.

[13] N. F. Ginting and M. Ginting, "Perbandingan Kriptografi RSA dengan Base64," *Jurnal Teknik Informatika UNIKA Santo Thomas*, pp. 47–52, Dec. 2017, doi: 10.17605/JTI.V2I2.190.

[14] C. Wadisman, I. Nozomi, and S. Rahmawati, "PENGAMANAN DATABASE MENGGUNAKAN KOMBINASI ALGORITMA (CEST CRYPTOGRAPHY) DAN ALGORiTMA BASE64," *JURTEKSI (Jurnal Teknologi dan Sistem Informasi)*, vol. 7, no. 1, pp. 33–38, 2020, doi: 10.33330/jurteksi.v7i1.896.

[15] Soleman, D. Budiman, and S. Mubaroq, "Combination RC4 Algorithm and Base64 Encryption on The Least Significant Bit Method," *Jurnal CoreIT*, vol. 8, no. 2, 2022, doi: 10.24014/coreit.v8i2.20106.

[16] M. Firman Arif and M. Misdram, "IMPLEMENTASI ENKRIPSI URL PADA WEBSITE MENGGUNAKAN METODE BASE64 DAN ROTATION13," *SPIRIT*, vol. 12, no. 1, pp. 20–25, Jul. 2020, doi: 10.53567/SPIRIT.V12I1.166.

[17] A. Rifaâ€™i and L. C. Sumartini, "IMPLEMENTASI KRIPTOGRAFI MENGGUNAKAN METODE BLOWFISH DAN BASE64 UNTUK MENGAMANKAN DATABASE INFORMASI AKADEMIK PADA KAMPUS AKADEMI TELEKOMUNIKASI BOGOR BERBASIS WEB-BASED," *Jurnal E-Komtek (Elektro-Komputer-Teknik)*, vol. 3, no. 2, pp. 87–96, Nov. 2019, doi: 10.37339/E-KOMTEK.V3I2.133.

[18] S. W. Jeon, K. S. Kwack, J. S. Yun, S. M. Gho, and S. Park, "Salt-and-pepper noise sign on fat-fraction maps by chemical-shift–encoded MRI: A useful sign to differentiate bone islands from osteoblastic metastases—a preliminary study," *American Journal of Roentgenology*, vol. 214, no. 5, pp. 1139–1145, 2020, doi: 10.2214/AJR.19.22177.

[19] S. Y. Doo, S. Tena, and V. M. Ndolu, "IMPLEMENTASI PENGAMANAN DATA MENGGUNAKAN METODE KRIPTOGRAFI HILL CIPHER DAN STEGANOGRAFI LEAST SIGNIFICANT BIT (LSB) PADA MEDIA CITRA DIGITAL," *Jurnal Media Elektro*, vol. VIII, no. 2, pp. 90–96, Oct. 2019, doi: 10.35508/JME.V0I0.1778.

[20] M. Faisal, "RANCANG BANGUN SISTEM INFORMASI HOUSEKEEPING INVENTORY DENGAN METODE WATERFALL," *Jurnal Infortech*, vol. 1, no. 1, pp. 28–34, 2019, doi: 10.31294/infortech.v1i1.6999