Terbit *online* pada laman : http://teknosi.fti.unand.ac.id/

Jurnal Nasional Teknologi dan Sistem Informasi

| ISSN (Print) 2460-3465 | ISSN (Online) 2476-8812 |

Artikel Penelitian

Sistem Keamanan Basis Data Klien P.T. Infokes Menggunakan Kriptografi Kombinasi RC4 Dan Base64

Nurhikmah Taliasih [a], *Irawan Afrianto* [b*]

[a] *Teknik Informatika Universitas Komputer Indonesia, Jl. Dipati Ukur No. 112-116, Bandung 40132, Indonesia*
[b*] *Teknik Informatika Universitas Komputer Indonesia, Jl. Dipati Ukur No. 112-116, Bandung 40132, Indonesia*

INFORMASI ARTIKEL

Sejarah Artikel:
Diterima Redaksi: 16 Desember 2020
Revisi Akhir: 29 Mei 2020
Diterbitkan *Online*: 31 Mei 2020

KATA KUNCI

Keamanan Basis Data,

Kriptografi,

RC4,

Base64

KORESPONDENSI

E-mail: irawan.afrianto@email.unikom.ac.id*

A B S T R A C T

PT Infokes Indonesia merupakan sebuah perusahaan yang bergerak dalam pembangunan perangkat lunak yang berfokus pada layanan kesehatan. Saat ini pada PT Infokes transaksi basis data dengan klien dilakukan secara konvensional yaitu dengan mengirimkan basis data yang disimpan dalam media penyimpanan sekunder melalui kurir ekspedisi. Hal tersebut menyebabkan basis data dapat dengan mudah dibaca dan dikhawatirkan menjadi celah kebocoran informasi yang bersifat privasi kepada publik.Dalam penelitian ini menggunakan algoritma kriptografi RC4 yang memiliki kelebihan dalam kecepatan maupun tingkat efesiensi dalam penyimpanan data dari hasil enkripsinya. Kriptografi RC4 yang digunakan, dikombinasikan dengan Base64 untuk menambah penyandian setelah dilakukan enkripsi. Hasil pengujian black box dan white box pada penelitian ini menunjukan bahwa implementasi algoritma kriptografi RC4 dan algoritma Base64 berjalan dengan baik. Selain itu, hasil pengujian dengan menggunakan Wireshark menunjukkan bahwa jalur transaksi basis data telah diamankan dengan protokol HTTPS dan basis data dalam keadaan terenkripsi. Hasil pengujian dengan melakukan cryptanalysis menggunakan CrypTool pada kombinasi algoritma RC4 dan Base64 menunjukkan bahwa pengkombinasian algoritma RC4 dan Base64 memiliki tingkat kemanan yang lebih tinggi untuk mengamankan basis data dibandingkan ketika hanya menggunakan algoritma RC4 saja.

1. PENDAHULUAN

P.T. Infokes Indonesia yang berfokus pada pengembangan produk dan solusi teknologi informasi kesehatan online dan terpadu di Indonesia telah menghasilkan beberapa produk di antaranya yaitu ePuskesmas, eHospital, eClinic, dan lain-lain. Pada PT Infokes arsip basis data klien disimpan dalam media penyimpanan sekunder. Selain itu transaksi basis data dengan klien dilakukan secara konvensional, yaitu dengan mengirimkan basis data yang sudah tersimpan di dalam media penyimpanan sekunder melalui kurir ekspedisi. Hal tersebut menyebabkan basis data dapat dibaca dengan mudah oleh orang lain dan dikhawatirkan menjadi celah kebocoran informasi yang bersifat privasi dan rahasia kepada publik. Hal itulah menyebabkan munculnya kebutuhan dari P.T. Infokes Indonesia untuk dapat mengamankan basis data klien yang dimilikinya, karena hal tersebut merupakan rahasia perusahaan yang harus dilindungi.

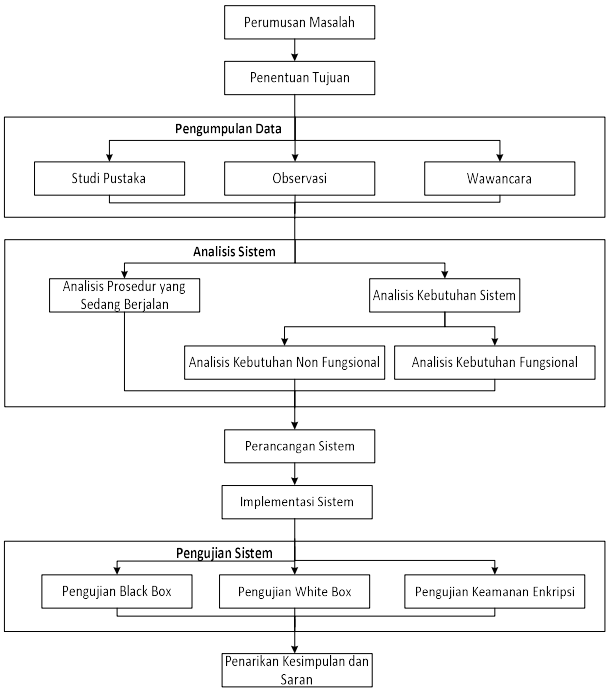
Kebutuhan pengamanan terhadap informasi menjadi suatu kebutuhan yang penting bagi perusahaan, khususnya guna mengamankan aset yang menjadi tanggung jawab perusahaan tersebut [1]. Teknik kriptografi merupakan salah satu alternatif solusi yang dapat digunakan dalam pengamanan informasi, beberapa penelitian [2][3][4][5], menunjukkan peran dan pentingnya mengamankan data menggunakan kriptografi. Algoritma RC4 dipilih karena memiliki kelebihan dalam kecepatan pemrosesan data [6] bahkan mencapai 10 kali lebih cepat dari DES [7]. Selain itu, RC4 memiliki tingkat efisiensi yang baik dalam penyimpanan data pada basis data, karena hasil enkripsi yang dihasilkan sama jumlahnya dengan karakter aslinya [8] dan pada RC4, jika masukkan yang akan dienkripsi mengandung kata

https://doi.org/10.25077/ TEKNOSI.v6i1.2020.9-18

berulang maka akan menghasilkan ciphertext yang acak [9][10]. Meskipun demikian, algoritma RC4 memiliki kerentanan terhadap Bit Flipping Attack atau BFA [11] dan cryptanalysis yang dapat mengetahui pesan asli dari analisis terhadap kunci yang mungkin digunakan [12]. Penelitian pengembangan algoritma RC4 dilakukan oleh [13] dimana algoritma RC4 dikombinasikan dengan kompresi Lzw guna mengamankan basis data perusahaan. Penelitian [14] mengkombinasikan algoritma RC4 dan mekanisme *rail fence* guna mengamankan basis data mahasiswa. Penelitian serupa yaitu kombinasi RC4 dan Base64 dilakukan oleh [15] yang diperuntukkan untuk melakukan kriptografi pada gambar. Semetara penelitian [16] menggunakan skema hybrid (AES,RC4 dan Elgamal) pada pengamanan data. Dengan merujuk pada penelitian-penelitian sebelumnya dan untuk membedakan peruntukannya, penelitian ini memiliki tujuan meningkatkan kinerja algoritma RC4 dengan menambahkan kombinasi menggunakan algoritma Base64 untuk mengamankan transaksi basis data klien terhadap eksploitasi *cryptanalysis bit flipping attack*.

2. METODE

Metode penelitian merupakan proses yang digunakan untuk memecahkan masalah secara logis. Gambar 1 menunjukkan bagan alir metode penelitian yang digunakan.



Gambar 1. Bagan Alir Penelitian

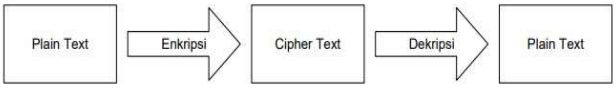
3. HASIL

3.1. Basis Data

Secara umum, database berarti koleksi data yang saling terkait. Secara praktis, database dapat dianggap sebagai suatu penyusun data yang terstruktur yang disimpan dalam media pengingat (hard disk) yang tujuannya adalah agar data tersebut dapat diakses dengan mudah dan cepat [17].

3.2. Kriptografi

Teknik kriptografi adalah ilmu yang bersandarkan pada teknik matematika untuk berurusan dengan keamanan informasi seperti kerahasiaan, keutuhan data dan otentikasi entitas [18]. Pesan atau data asli sebelum dienkripsi disebut plaintext. Sedangkan pesan yang sudah diacak disebut ciphertext. Proses pengubahan plaintext menjadi ciphertext disebut dengan enkripsi, sedangkan proses pengubahan ciphertext kembali menjadi plaintext disebut dengan dekripsi [19].



Gambar 2. Konsep kriptografi

3.3. Kriptografi RC4

Sistem sandi RC4 dikembangkan oleh Ronald Rivest pada tahun 1987 merupakan algoritma stream cipher yang paling banyak digunakan. Misalnya pada protokol SSL/TLS. RC4 merupakan stream cipher yang berorientasi byte. Masukan algoritma enkripsi RC4 merupakan sebuah byte, kemudian dilakukan operasi XOR dengan sebuah byte kunci, dan menghasilkan sebuah byte sandi [6] [12].

Persamaan proses enkripsi:

$$ci = pi \oplus ki \quad (1)$$

Persamaan proses dekripsi:

$$pi = ci \oplus ki \quad (2)$$

3.4. Base64

Algoritma Base64 merupakan salah satu algoritma untuk encoding dan decoding. Tujuan dari encoding adalah untuk mengubah bentuk atau format data. Algoritma Base64 mengubah suatu data ke dalam format ASCII yang didasarkan pada bilangan dasar 64 atau bisa dikatakan sebagai salah satu metode yang digunakan untuk melakukan encoding (penyandian) terhadap data binary. Karakter yang dihasilkan pada transformasi Base64 ini terdiri dari A..Z, a..z dan 0..9, serta ditambah dengan dua karakter terakhir yang bersimbol yaitu + dan / serta satu buah karakter sama dengan (=) yang digunakan untuk penyesuaian dan menggenapkan data binary atau istilahnya disebut sebagai padding [4].

 https://doi.org/10.25077/TEKNOSI.v6i1.2020.9-18

Index (6 bit data)	Char Encoding Base64	Index (6 bit data)	Char Encoding Base64	Index (6 bit data)	Char Encoding Base64	Index (6 bit data)	Char Encoding Base64
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	41	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	/
						pad	=

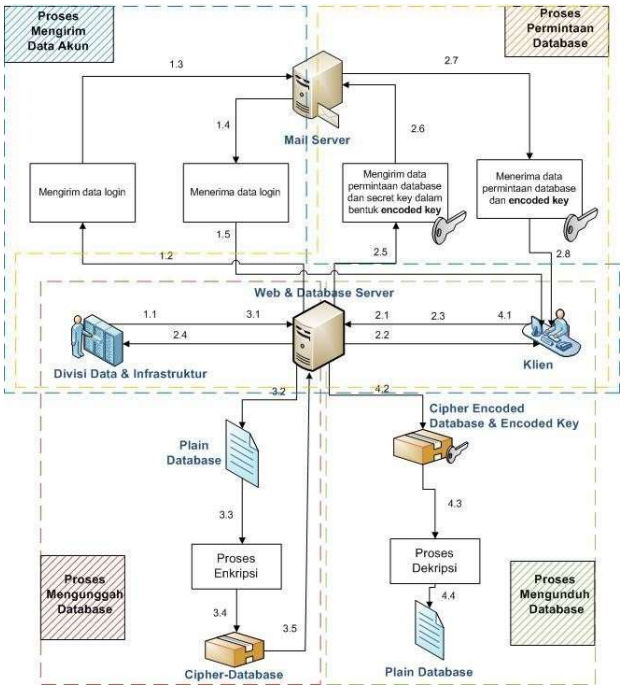
Gambar 3. Indeks Base64 [4]

3.5. HTTPS (HyperText Transfer Protocol)-Secure

HTTPS merupakan HTTP yang menggunakan SSL (Secure Socket Layer). SSL adalah protokol enkripsi yang dipanggil melalui web server dengan menggunakan HTTPS. Penggunaan protokol SSL pada jaringan sangat penting untuk mengamankan data di dalam sambungan jaringan.. SSL merupakan jenis sockets communications yang berada di antara transmision control protocol/internet protocol (TCP/IP) dan application layer [20].

3.6. Gambaran Umum Sistem

Sistem yang akan dibangun pada penelitian ini adalah sistem keamanan dengan kombinasi algoritma kriptografi dan encoding berbasis web. Di mana divisi Data dan Infrastruktur akan membuatkan akun untuk klien sesuai surat pesanan dan data akun tersebut akan dikirim kan melalui email. Selanjutnya klien dapat melakukan permintaan database melalui sistem. Divisi Data dan Infrastruktur akan mengunggah database sesuai permintaan dan melakukan enkripsi melalui sistem yang sudah dibangun. Proses enkripsi dilakukan dengan algoritma kriptografi RC4 dan encoding Base64. Kemudian database yang sudah terenkripsi atau berbentuk cipher encoded database akan tersimpan dalam database server. Cipher encoded database ini selanjutnya dapat diunduh klien untuk didekripsi menjadi file database asli dengan memasukkan secret key yang dikirim melalui email. Secret key yang dikirim dalam bentuk encoded key karena sudah dilakukan pengubahan sebelumnya dengan menggunakan algoritma encoding Base64.



Gambar 4. Arsitektur Sistem Yang Dikembangkan

3.7. Analisis Algoritma

Pada penelitian ini algoritma yang digunakan adalah algoritma kriptografi RC4 dan algoritma Base64. Analisis algoritma digunakan untuk mengetahui rangkaian proses dari algoritma yang digunakan sehingga dapat diterapkan ke dalam sistem yang dibangun.

3.7.1. Analisis Pembangkitan Kunci RC4

Proses pembangkitan kunci dilakukan terlebih dahulu sebelum melakukan enkripsi maupun dekripsi pada algoritma RC4. RC4 yang merupakan bagian dari tipe stream cipher memiliki cara yang sama dengan konsep umum membangkitkan keystream pada stream cipher. Keystream dibangkitkan dengan keystream generator lalu dilakukan XOR antara key tersebut dengan plaintext. Algoritma RC4 memakai S-box (Substitution-box) dengan larik 256 byte yang berukuran 16 x 16 sebagai keystream generator. Proses pembangkitan kunci pada RC4 adalah sebagai berikut:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

Gambar 5. Inisiasi S-Box RC4

1) Inisialiasasi S-Box

Pertama, inisialisasi S-Box dengan panjang 256 byte, dengan S[0]=0, S[1]=1, S[2]=2,…, S[255]=255 sehingga array S menjadi seperti Gambar 5.

2) Melakukan padding kunci K sehingga panjang kunci K = 256

Inisialisasi 256 byte kunci array K. Misalkan kunci (*secret key*) adalah: ifk ulang kunci sampai memenuhi seluruh array k dan ubah format ke dalam bentuk ASCII. k[0]=i=105, K[1]=f=102, K[2]=k=107,…,[255]=i=105. Sehingga *array* K menjadi seperti terlihat pada Gambar 6.

105	102	107	105	102	107	105	102	107	105	102	107	105	102	107	105
102	107	105	102	107	105	102	107	105	102	107	105	102	107	105	102
107	105	102	107	105	102	107	105	102	107	105	102	107	105	102	107
105	102	107	105	102	107	105	102	107	105	102	107	105	102	107	105
102	107	105	102	107	105	102	107	105	102	107	105	102	107	105	102
107	105	102	107	105	102	107	105	102	107	105	102	107	105	102	107
105	102	107	105	102	107	105	102	107	105	102	107	105	102	107	105
102	107	105	102	107	105	102	107	105	102	107	105	102	107	105	102
107	105	102	107	105	102	107	105	102	107	105	102	107	105	102	107
105	102	107	105	102	107	105	102	107	105	102	107	105	102	107	105
102	107	105	102	107	105	102	107	105	102	107	105	102	107	105	102
107	105	102	107	105	102	107	105	102	107	105	102	107	105	102	107
105	102	107	105	102	107	105	102	107	105	102	107	105	102	107	105
102	107	105	102	107	105	102	107	105	102	107	105	102	107	105	102
107	105	102	107	105	102	107	105	102	107	105	102	107	105	102	107
105	102	107	105	102	107	105	102	107	105	102	107	105	102	107	105

Gambar 6. Proses Padding Kunci Pada RC4

3) Permutasi untuk S-Box

Operasi permutasi S-Box akan menghasilkan array di dalam S-Box yang berbeda dari urutan sebelumnya. Dalam operasi ini digunakan variabel i dan j ke indeks array S[i] dan K[i]. Pada permulaan akan dilakukan inisialisasi i dan j dengan 0. Operasi ini dilakukan sebanyak 256 kali dengan pengulangan rumusan (j + S[i]+K[i]) mod 256 yang diikuti dengan penukaran S[i] dengan S[j].

for i = 0 to 255
j = (j + S[i] + K[i]) mod 256
swap S[i] dan S[j]

Sehingga Swap S[0] dan S[105] akan menghasilkan array pada Gambar 7.

105	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	0	106	107	108	109	110	111
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

Gambar 7. Proses Permutasi S-Box S[0] dan S[105] Pada RC4

Iterasi dilanjutkan sampai iterasi ke-256. Akhir dari iterasi dan *swap* yang dilakukan akan menghasilkan *array* sesuai Gambar 8.

105	97	61	102	92	108	242	125	210	165	194	42	0	18	139	176
121	44	226	161	17	8	65	1	215	23	55	135	201	227	162	239
122	112	140	85	12	46	250	83	225	177	231	134	218	211	59	4
53	51	155	131	101	164	116	156	74	224	133	93	86	149	228	5
117	178	234	22	157	75	254	127	90	16	172	111	77	196	64	118
179	203	57	56	251	123	202	252	186	39	245	174	28	50	151	113
34	98	37	153	79	24	119	130	146	76	144	173	212	190	114	10
63	147	115	246	30	168	132	15	232	189	126	214	197	160	206	120
171	222	198	19	248	62	185	217	88	229	240	100	71	60	243	237
29	110	163	109	205	89	26	191	148	27	183	2	45	166	142	167
95	182	106	124	20	58	3	6	223	187	209	181	43	145	141	66
169	67	47	87	72	244	170	193	82	200	73	188	103	68	216	255
192	221	54	154	159	184	249	220	199	235	40	99	152	21	204	38
136	70	158	36	49	78	208	104	81	207	230	96	32	137	52	238
236	94	107	241	219	253	129	233	9	69	143	91	31	128	80	7
41	180	35	175	150	84	33	213	14	25	195	13	48	11	247	138

Gambar 8. Hasil Akhir Permutasi S-Box RC4

4) Membangkitkan random key

Berikutnya adalah proses pembangkitan kunci yang diperoleh secara random dengan melakukan operasi perhitungan sebanyak jumlah byte dari plaintext. Perhitungannya adalah sebagai berikut: Pembangkitan random key ke-1. Inisialiasasi terlebih dahulu nilai i = 0; j = 0 i = (i + 1) mod 256

= (0 + 1) mod 256
= 1 j = (j + S[i]) mod 256
= (0 + S[1]) mod 256
= (0 + 97) mod 256
= 97

Sehingga hasilnya: S[i]=S[1] dan S[j]=S[97]. Kemudian lakukan swap terhadap array hasil iterasi terakhir :

S[i]=S[97]=98 dan S[j]=S[1]=97

t = (S[i] + S[j]) mod 256
= (98 + 97) mod 256
= 195

K = S[t] = S[195] = 154 (nilai yang akan di-XOR-kan dengan byte plaintext ke-1) .

Operasi pembangkitan random key dilanjutkan sampai sebanyak byte plaintext yang akan dienkripsi dan akan menghasilkan random key seperti pada Gambar 9.

154	99	54	33	86	200	242	36	0	29	165	238	174	108	144	91
241	6	248	77	197	179	69	196	243	137	109	16	173	8	179	230
238	150	4	202	58	201	209	141	48	221	244	112	157	35	178	39
195	203	237	172	15	231	51	85	37	20	190	11	195	188	9	215
118	46	87	13	46	253	133	170	67	181	209	85	…			

Gambar 9. Random Key RC4

3.7.2. *Analisis Algoritma Enkripsi RC4*

Algoritma enkripsi RC4 beroperasi dalam byte, berarti XOR dilakukan setiap satu byte *plaintext* dengan satu byte key dari keystream. *Flowchart* dari algoritma enkripsi RC4 dapat dilihat pada gambar 10.

 https://doi.org/10.25077/TEKNOSI.v6i1.2020.9-18

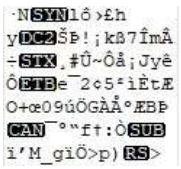
Gambar 10. Diagram Alir Algoritma Enkripsi RC4

Dari proses pembangkitan kunci sebelumnya telah diperoleh random key, selanjutnya dilakukan operasi XOR antara byte dari random key dengan plaintext yang ditampilkan pada gambar 11.

RANDOM KEY			PLAINTEXT			XOR		
Char	Des	Biner	Char	Des	Biner	Char	Des	Biner
š	154	10011010	-	45	00101101	·	183	10110111
c	99	01100011	-	45	00101101	N	78	01001110
6	54	00110110		32	00100000	[SYN]	22	00010110
!	33	00100001	M	77	01001101	1	108	01101100
V	86	01010110	y	121	01111001	ô	244	11110100
È	200	11001000	S	83	01010011	›	155	10011011
ò	242	11110010	Q	81	01010001	£	163	10100011
$	36	00100100	L	76	01001100	h	104	01101000
0	0	00000000		32	00100000		32	00100000
	29	00011101	d	100	01100100	y	121	01111001
¥	165	10100101	u	117	01110101	[DO2]	18	00010010
î	238	11101110	m	109	01101101	Š	138	10001010
®	174	10101110	p	112	01110000	Þ	222	11011110
1	108	01101100		32	00100000	!	33	00100001
□	144	10010000	1	49	00110001	¡	161	10100001
[	91	01011011	0	48	00110000	k	107	01101011
...								

Gambar 11. Operasi XOR Pada Enkripsi RC4

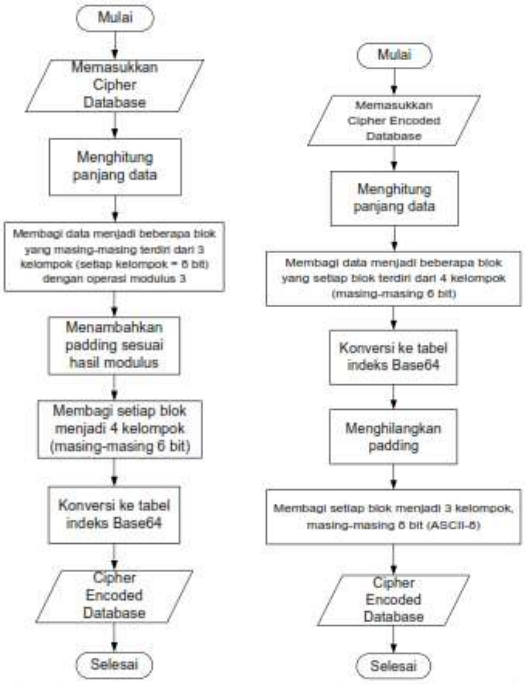
Dari proses XOR antara *plaintext* dengan *random key*, maka hasil enkripsi RC4 selengkapnya seperti yang ditunjukan pada gambar 12.



Gambar 12. Hasil Enkripsi RC4

3.7.3. Analisis Algoritma Encoding dan Decoding Base64

Berdasarkan hasil enkripsi RC4, selanjutnya dilakukan encoding menggunakan algoritma Base64 dengan tahapan proses seperti pada gambar 13(a) dan proses decoding Base64 pada gambar 13(b).



(a) (b)

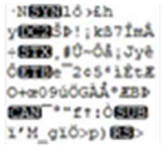
Gambar 13. Proses Encoding Base64 (a) - Proses Decoding Base64 (b)

Dalam penelitian ini, algoritma Base64 dioperasikan setelah algoritma RC4 sehingga masukkannya berupa hasil enkripsi RC4 (cipher database) yang akan menghasilkan *cipher encoded database* seperti pada gambar 14.

```
t04WbPSbo2ggeR
KK3iGha983zm3C
9wK4I9t+lOWhSn
nqlBdlrzKiNbLs
yHTGTyucMDn61k
fAxbrGQt4Yr7CT
ZoY60hrvkklfZ+
```

Gambar 14. Hasil Encoding Base64 Terhadap Enkripsi RC4

Sementara untuk proses decoding Base64, dilakukan terhadap data encoding Base64, yang akan menghasilkan data *cipher* enkripsi RC 4, seperti pada gambar 15.



Gambar 15. Hasil Decoding Base64 Terhadap Data Encoding Base64

3.7.4. Analisis Algoritma Dekripsi RC4

RC4 menggunakan fungsi dekripsi dan enkripsi yang sama karena operasi yang dilakukan hanyalah XOR antara random key yang dibangkitkan dengan plaintext. Sehingga pada dekripsinya, proses

XOR dilakukan antara random key dengan cipher database. Diagram alir proses dekripsi RC4 dapat dilihat pada gambar 16.

Gambar 16. Diagram Alir Algoritma Dekripsi RC4

Pada proses dekripsi menghasilkan pembangkitan kunci yang sama ketika melakukan enkripsi. Operasi XOR yang dilakukan pada tahap dekripsi seperti pada gambar 17.

CIPHER DATABASE			*RANDOM KEY*			HASIL XOR		
Char	**Des**	**Bin**	**Char**	**Des**	**Bin**	**Char**	**Des**	**Bin**
·	183	10110111	š	154	10011010	-	45	00101101
N	78	01001110	c	99	01100011	-	45	00101101
	22	00010110	6	54	00110110		32	00100000
l	108	01101100	!	33	00100001	M	77	01001101
ô	244	11110100	V	86	01010110	y	121	01111001
›	155	10011011	È	200	11001000	S	83	01010011
£	163	10100011	ò	242	11110010	Q	81	01010001
h	104	01101000	$	36	00100100	L	76	01001100
	32	00100000	0	0	00000000		32	00100000
y	121	01111001		29	00011101	d	100	01100100
	18	00010010	¥	165	10100101	u	117	01110101
Š	138	10001010	î	238	11101110	m	109	01101101
Þ	222	11011110	®	174	10101110	p	112	01110000
!	33	00100001	l	108	01101100		32	00100000
¡	161	10100001	□	144	10010000	1	49	00110001
k	107	01101011	[	91	01011011	0	48	00110000
.....								

Gambar 17. Operasi XOR Pada Dekripsi RC4

Dari proses XOR antara cipher database dengan random key, menghasilkan plain database pada gambar 18.

```
-- MySQL
dump
10.16
Distrib
10.3.10-Ma
riaDB,
```

Gambar 18. Hasil Dekripsi RC4

4. PEMBAHASAN

Pembahasan yang dilakukan dalam penelititian ini mencakup Spesifikasi Kebutuhan Perangkat Lunak (SKPL), pemodelan menggunakan UML, implementasi dan pengujian sistem yang dikembangkan.

4.1. Spesifikasi Kebutuhan Perangkat Lunak (SKPL)

Guna menunjang berjalannya sistem yang dibangun diperlukan analisis terhadap kebutuhan perangkat lunak yang digunakan. Adapun analisis yang dilakukan mencakup analisis SKPL fungsional (tabel 1) dan SKPL non fungsional.(tabel 2).

Tabel 1. SKPL Fungsional

No	SKPL-F-ID	Deskripsi
1	SKPL-F-001	Melakukan Login
2	SKPL-F-003	Melakukan Enkripsi
3	SKPL-F-004	Melakukan Dekripsi
4	SKPL-F-005	Mengunggah Data
5	SKPL-F-006	Mengunduh Data
6	SKPL-F-007	Mengelola Data *User*
7	SKPL-F-008	Sistem Logout

Tabel 2. SKPL Non-Fungsional

SKPL-NF-ID	Parameter	Deskripsi
SKPL-NF-01	*Availability*	24 jam
SKPL-NF-02	*Reability*	Tidak pernah gagal
SKPL-NF-03	*Ergonomy*	Tampilan antarmuka menggunakan Graphical *User* Interface (GUI)
SKPL-NF-04	*Portability*	Mudah diadopsi pada berbagai lingkungan sistem operasi, berbagai browser internet dan berbagai device asalkan memenuhi batasan kebutuhan perangkat lunak
SKPL-NF-05	*Memory*	Menyesuaikan dengan data yang ada di *database*

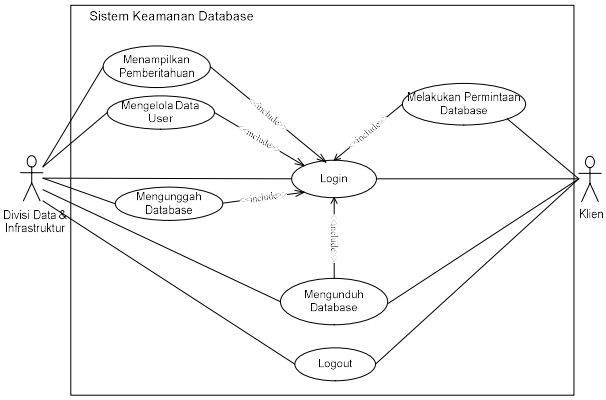
4.2. Pemodelan Sistem

Pemodelan sistem digunakan untuk menjelaskan kebutuhan yang diperlukan agar sistem dapat berjalan dengan baik serta sesuai dengan kebutuhan sistem keamanan database di PT Infokes. Adapun penggambaran model sistem menggunakan diagram UML.

4.2.1. Diagram Use Case

Diagram Use case merupakan diagram yang menggambarkan interaksi antara sistem yang dibangun dengan pengguna yang disebut aktor maupun dengan sistem lain atau eksternal. Diagram use case untuk sistem keamanan basis data P.T. Infokes dapat dilihat dapat gambar 19.

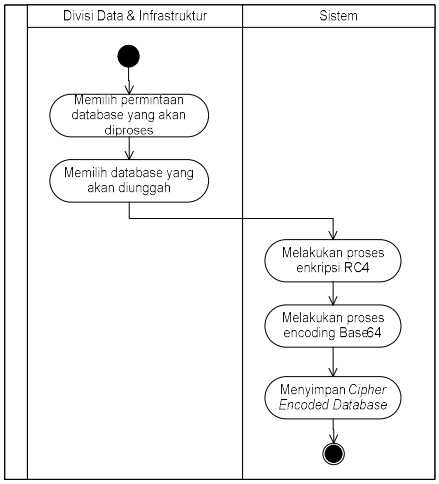
 https://doi.org/10.25077/TEKNOSI.v6i1.2020.9-18



Gambar 19. Diagram Use Case Keamanan Basis Data

4.2.2. Diagram Activity

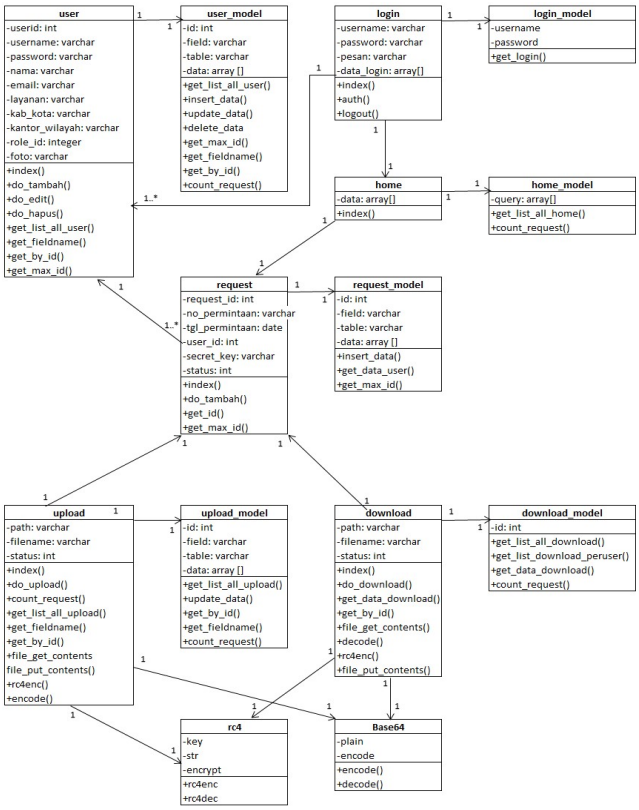
Diagram Activity menggambarkan berbagai alur aktivitas atau aliran kerja dari sistem yang sedang dirancang, bagaimana masing-masing alur berawal dan bagaimana alur tersebut berakhir. Gambar 20, menunjukkan diagram activity untuk aktivitas mengunggah basis data.



Gambar 20. Diagram Activity Unggah Basis Data

4.2.3. Diagram Class

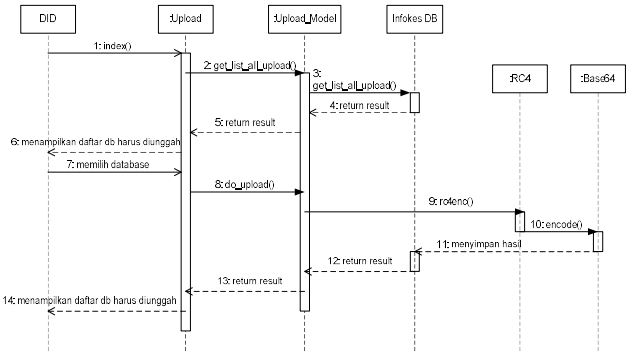
Diagram class dalam pembangunan sistem berfungsi untuk menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat, baik itu kelas yang terlibat maupun hubungan antar kelasnya. Gambar 21 menunjukkan diagram class untuk sistem keamanan basis data.



Gambar 21. Diagram Class Sistem Keamanan Basis Data

4.2.4. Diagram Sequence

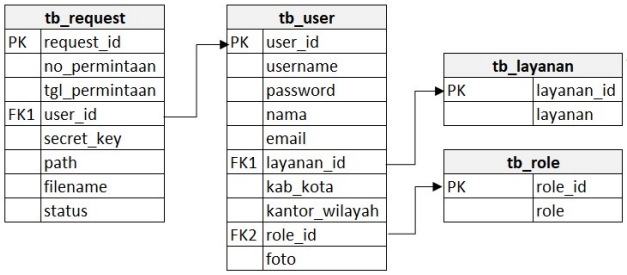
Diagram sequence berfungsi untuk memberikan gambaran mengenai urutan aktivitas yang terjadi di dalam sistem. Gambar 22 menunjukkan diagram sequence mengunggah basis data.



Gambar 22. Diagram Sequence Mengunggah Basis Data

4.3. Skema Relasi

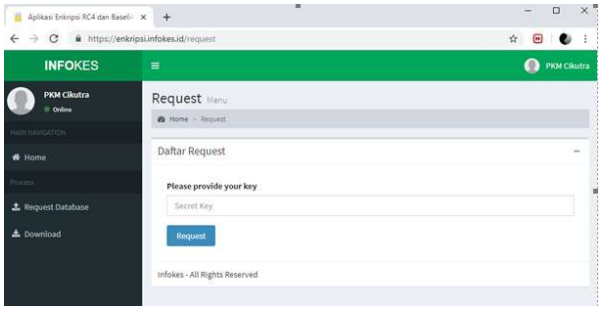
Skema relasi merupakan rangkaian hubungan antara dua tabel atau lebih pada sistem database. Gambar 23 adalah skema relasi untuk sistem keamanan database P.T. Infokes.



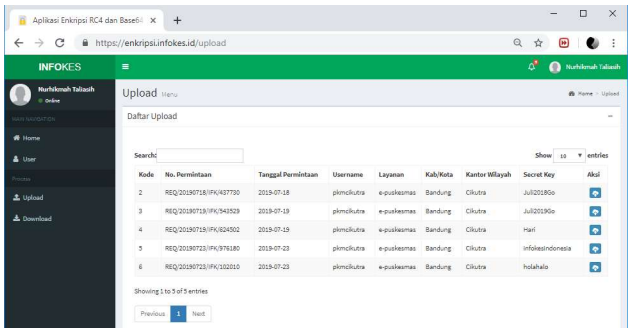
Gambar 23. Skema Relasi Sistem Keamanan Basis Data

4.4. *Antar Muka Sistem*

Beberapa implementasi antar muka untuk sistem keamanan database P.T. Infokes dapat dilihat pada gambar 24 dan 25.



Gambar 24. Antar Muka Permohonan Kunci



Gambar 25. Antar Muka Unggah Basis Data

4.5. *Pengujian Sistem*

Pengujian yang dilakukan ada beberapa proses yaitu pengujian *black box*, pengujian *white box* dan pengujian keamanan database dengan menggunakan CrypTool versi 1.4.41 dan Wireshark versi 3.0.1.

4.5.1. *Pengujian Black Box*

Pengujian *black box* merupakan tahap uji secara fungsional yang dilakukan untuk mengetahui kesesuaian fungsi-fungsi di dalam sistem beserta masukan dan keluaran dengan spesifikasi yang dibutuhkan. Tabel 3, menunjukkan pengujian black box untuk fungsi permohonan basis data.

Tabel 3. Pengujian *Black Box* Sistem

Kelas Uji	**Butir Uji**	**Hasil Uji**
Login	Verifikasi *Username*	Diterima
	Verifikasi Password	Diterima
Melakukan Permintaan *Database*	Tambah Data	Diterima
	Kirim Email	Diterima
Menampilkan Pemberitahuan	Tampil Data	Diterima
Mengelola Data *User*	Tambah Data	Diterima
	Edit Data	Diterima
	Hapus Data	Diterima
	Kirim Email	Diterima
Mengunggah *Database*	Enkripsi & *Encoding* Data	Diterima
	Unggah Data	Diterima
Mengunduh *Database*	Dekripsi & *Decoding* Data	Diterima
	Unduh Data	Diterima
	Kirim Email	Diterima

4.5.2. *Pengujian White Box Sistem*

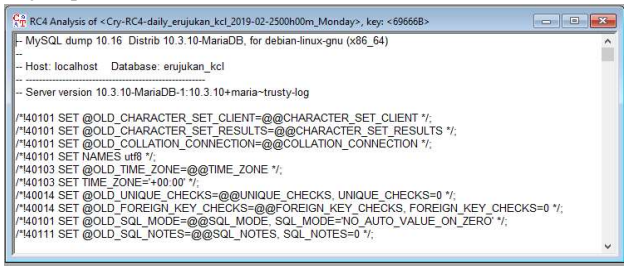
Pada tahap ini, pengujian *white box* yang akan digunakan merupakan bentuk *basis path testing*. Pengujian dilakukan pada proses pembangkitan kunci dan enkripsi RC4. Adapun langkah-langkah pengujian white box adalah sebagai berikut :

1. Memeriksa Source Code Pembangkitan Kunci dan Enkripsi RC4.
2. Membuat *Flowgraph* Pembangkitan Kunci dan Enkripsi RC4.
3. Menghitung *cyclomatic complexity*.
4. Menentukan *Independet Path.*
5. Membuat *Graph* Matriks Pembangkitan Kunci dan Enkripsi RC4.

Berdasarkan pengujian white box pada pembangkitan kunci dan enkripsi RC4 dihasilkan nilai *cyclomatic complexity* yang sama yaitu 5. Sehingga dapat disimpulkan bahwa sistem berjalan dengan baik, karena setiap pengujian menghasilkan nilai yang sama.

4.5.3. *Pengujian Menggunakan CrypTool*

Pengujian ini dilakukan untuk membuka database menggunakan *CrypTool*. Untuk pengujian keamanan enkripsi, dilakukan perbandingan pengujian hasil enkripsi RC4 dan pengujian terhadap hasil kombinasi enkripsi RC4 dan encoding Base64. Gambar 26 menunjukkan bahwa proses *cryptanalysis* yang dilakukan, mampu membuka basis data yang di enkripsi RC4 menjadi *plain text*.



Gambar 26. Implementasi *CrypTool* Pada Enkripsi RC4

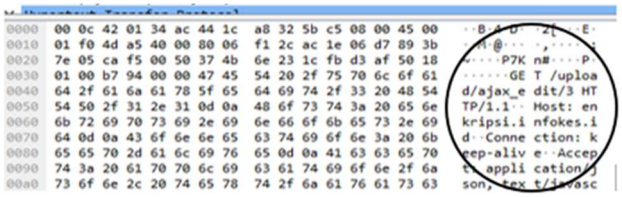
 https://doi.org/10.25077/TEKNOSI.v6i1.2020.9-18

Sementara, pengkombinasian algoritma RC4 dan Base64, menunjukkan bahwa aplikasi *CrypTool*, tidak mampu mengembalikan basis yang yang di enkripsi ke bentuk *plain text*, seperti yang terlihat pada gambar 27.

Gambar 27. Implementasi *CrypTool* Pada Kombinasi Enkripsi RC4 dan Base64

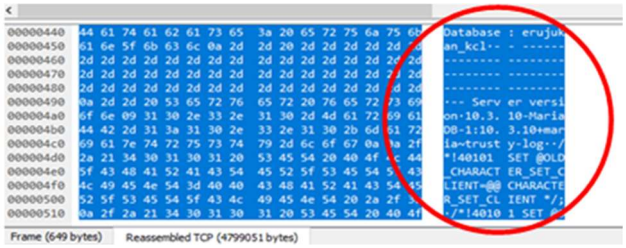
4.5.4. Pengujian Menggunakan Wireshark

Pengujian keamanan database menggunakan aplikasi Wireshark dilakukan untuk mengetahui apakah database yang dikirimkan dalam keadaan aman atau tidak. Pengujian dilakukan pada dua tahap yaitu pengujian pada saat sistem web yang dibangun masih menggunakan protokol HTTP dan pengujian setelah menggunakan protokol HTTPS. Gambar 28 menunjukkan bahwa lalu lintas basis data menggunakan protokol HTTP masih dapat dengan mudah dilihat dan diterjemahkan karena bersifat *plain text.*



Gambar 28. Monitoring Wireshark Pada Protokol HTTP

Sementara, setelah sistem pengamanan basis data menggunakan SSL dan protokol berubah menjadi TPPS, lalu lintas data data pada aplikasi Wireshark menjadi sulit dibaca, karena proses transmisinya telah di enkripsi oleh SSL (gambar 29).



Gambar 29. Monitoring Wireshark Pada Protokol HTTPS

5. KESIMPULAN

Berdasarkan pengujian sistem yang telah dilakukan, dapat disimpulkan bahwa sistem dapat mengamankan database dengan kriptografi. Berdasarkan pengujian black box dan white box menunjukan bahwa proses unggah dengan enkripsi RC4 serta encoding Base64 dan unduh dengan decoding Base64 serta dekripsi RC4 pada database berjalan dengan baik. Berdasarkan pengujian menggunakan aplikasi *CrypTool*, kombinasi algoritma dengan menambahkan Base64 pada algoritma kriptografi RC4 dapat meningkatkan kinerja RC4 dari serangan cryptanalysis yang dilakukan. Pengujian menggunakan aplikasi *Wireshark*, menujukkan bahwa hasil pemonitoran pada lalu lintas data tidak dapat membaca database asli sehingga sistem dapat mengamankan proses transaksi database dari Divisi Data dan Infrastruktur kepada klien.

DAFTAR PUSTAKA

[1] I. Afrianto, T. Suryana, and others, “Pengukuran dan Evaluasi Keamanan Informasi Menggunakan Indeks KAMI-SNI ISO/IEC 27001: 2009,” *Ultim. InfoSys J. Ilmu Sist. Inf.*, vol. 6, no. 1, pp. 43–49, 2015.

[2] A. D. Hidayat and I. Afrianto, “Sistem Kriptografi Citra Digital Pada Jaringan Intranet Menggunakan Metode Kombinasi Chaos Map dan Teknik Selektif,” *Ultim. J. Tek. Inform.*, vol. 9, no. 1, pp. 59–66, 2017.

[3] M. Septian, “Penerapan Enkripsi Rabbit Stream Cipher Algorithm Untuk Mengamankan File Bersifat Rahasia Di Polsek Mangkubumi Tasikmalaya,” Universitas Komputer Indonesia, 2018.

[4] R. Aulia, A. Zakir, and D. A. Purwanto, “Penerapan Kombinasi Algoritma Base64 Dan Rot47 Untuk Enkripsi Database Pasien Rumah Sakit Jiwa Prof. Dr. Muhammad Ildrem,” *InfoTekJar J. Nas. Inform. dan Teknol. Jar.*, vol. 2, no. 2, pp. 146–151, 2018.

[5] A. P. Rahangiar, F. de Fretes, and others, “Penerapan Algoritma Gabungan Rc4 Dan Base64 Pada Sistem Keamanan E-commerce,” *J. Fak. Huk. UII*.

[6] A. A. Okedola and Y. N. Asafe, “RSA and RC4 Cryptosystem Performance Evaluation Using Image and Text File,” *Int J Sci Eng Res*, vol. 6, pp. 289–294, 2015.

[7] W. Komputer, “Memahami model enkripsi dan security data,” *Penerbit Andi, Yogyakarta*, 2003.

[8] S. A. Agusta and others, “Implementasi Algorithma Stream Cipher RC4 dalam Aplikasi Pendataan Alumni STMIK Amik Riau,” *INOVTEK Polbeng-Seri Inform.*, vol. 1, no. 1, pp. 1–8, 2016.

[9] Y. Prayudi and I. Halik, “Studi dan Analisis Algoritma RIVEST CODE 6 (RC6) Dalam Enkripsi/Dekripsi Data,” *J. Fak. Huk. UII*, 2005.

[10] H. Pandiangan, “Perancangan Media Pengiriman Pesan Teks Dengan Penyandian Pesan Menggunakan Algoritma RC4 Berbasis WEB,” *J. Mantik Penusa*, vol. 19, no. 1, 2016.

[11] V. Arintamy, N. Cahyani, and A. Mulyana, “Analisis Algoritma Rc4 Sebagai Metode Enkripsi Wpa-Psk Pada Sistem Keamanan Jaringan Wireless Lan,” *eProceedings Eng.*, vol. 1, no. 1, 2014.

[12] P. Xue, T. Li, H. Dong, C. Liu, W. Ma, and S. Pei, “GB-RC4: Effective brute force attacks on RC4 algorithm using GPU,” in *2016 Seventh International Green and Sustainable Computing Conference (IGSC)*, 2016, pp. 1–6.

[13] A. R. Wahid and M. Syafrullah, “Implementasi Algoritma Rc4 Dan Kompresi Lzw Untuk Pengamanan Database Pada Pt. Mpp International Development Indonesia,” *SKANIKA*, vol. 1, no. 3, pp. 1045–1050, 2018.

[14] R. S. Siregar, M. S. Asih, and N. Wulan, “Penerapan Algoritma RC4 Dan Rail Fence Untuk Enkripsi Database Mahasiswa Pada Kampus Poltekkes Kemenkes Medan,” *JITEKH*, vol. 7, no. 02, pp. 51–56, 2019.

[15] M. D. Putra and others, "Enkripsi Dan Dekripsi Gambar Dengan Menggunakan Perpaduan Algoritma Base64 Dan Rc4," *SEMNASTEKNOMEDIA ONLINE*, vol. 6, no. 1, pp. 2–14, 2018.
[16] A. Widarma, "Kombinasi Algoritma AES, RC4 dan Elgamal Dalam Skema Hybrid Untuk Keamanan Data," *Comput. Eng. Sci. Syst. J.*, vol. 1, no. 1, pp. 1–8, 2016.
[17] A. Kadir, "Tuntunan Praktis Belajar Database Menggunakan MySQL," *Yogyakarta Andi*, 2008.
[18] R. Sadikin, "Kriptografi untuk keamanan jaringan," *Penerbit Andi, Yogyakarta*, 2012.
[19] D. Ariyus, *Pengantar ilmu kriptografi: teori analisis & implementasi*. Penerbit Andi, 2008.
[20] H. Pranata, L. A. Abdillah, and U. Ependi, "Analisis Keamanan Protokol Secure Socket Layer (SSL) Terhadap Proses Sniffing di Jaringan," *arXiv Prepr. arXiv1508.05457*, 2015.

BIODATA PENULIS

Nurhikmah Taliasih
Alumni program studi Teknik Informatika – Universitas Komputer Indonesia, saat ini aktif menjadi *freelancer* pada beberapa projek TIK di Kota Bandung.

Irawan Afrianto
Saat ini aktif sebagai dosen tetap di program studi Teknik Informatika Universitas Komputer Indonesia (UNIKOM) Bandung. Menyelesaikan studi S1 di Universitas Komputer Indonesia pada tahun 2002 dan menyelesaikan studi S2 di Institut Teknologi Bandung pada tahun 2010. Saat ini sedang menyelesaikan studi S3 Ilmu Komputer di *IPB University* Bogor.

 https://doi.org/10.25077/TEKNOSI.v6i1.2020.9-18