BAB II
TINJAUAN PUSTAKA

2.1 Keamanan Data Komputer

Keamanan komputer adalah pencegahan kejahatan yang dimediasi komputer. Keamanan yang dibutuhkan meliputi keamanan fisik (ruang server dan infrastruktur pendukung), keamanan akses (manusia sebagai pengguna), keamanan data (virus dan pencurian data), dan keamanan sistem operasi komputer ((Siregar, 2019)).saat mengembangkan keamanan komputer, aspek kerahasiaan, integritas, otentikasi, non-penolakan, dan ketersediaan harus dipertimbangkan. Aspek kerahasiaan bertujuan untuk mencegah agar data yang ada di dalam komputer tidak jatuh ke tangan yang tidak berhak sehingga dapat disalahgunakan. Aspek integritas menyangkut konsistensi informasi data sehingga tidak dapat diubah atau dirusak oleh pihak lain. Dalam hal ini, metode enkripsi sering digunakan untuk penyandian ((Pratiwi & WP, 2016)).

Aspek otentikasi melibatkan identifikasi keaslian pengguna dan keaslian sumber data. Pada saat yang sama, dalam hal non-repudiation, mencegah pihak yang seharusnya bertanggung jawab atas data menolak akses data. Aspek Ketersediaan menekankan pada ketersediaan informasi jika pengguna tidak dapat mengakses data di komputer karena kejahatan komputer. Untuk meningkatkan keamanan komputer, tindakan pencegahan diambil untuk mencegah pengguna yang tidak berwenang mengakses data (kepercayaan) dan kemungkinan manipulasi/korupsi data (integritas) melalui penggunaan Pihak yang menyediakan data akses otentikasi.((Gunawan et al., 2016)).

2.2 Kriptografi

Kriptografiberasal dari bahasa Yunani, *crypto* dan *graphia*. *Crypto* berarti *secret* (rahasia) dan *graphia* berarti *writing* (tulisan). Kriptografi adalah sebuah teknik penyandian pesan yang dilakukan agar pesan dapat dikirim dan diterima dengan aman. Kriptografi bertujuan untuk menjaga kerahasian data dan informasi agar tidak disalah gunakan oleh pihak yang tidak sah ((Yusfrizal, 2019)). Agar kriptografi dapat berjalan dengan baik haruslah terdapat empat elemen utama didalamnya, yang paling berkait satu sama lain ((Setyaningsih et al., 2015)), yaitu:

1. *Plainpdf*

   Merupakan sebagai pesan gambar awal atau pesan asli yang dikirim pada proses komunikasi. *Plainpdf* inilah yang kemudian dienkripsi dan dideskripsi.

2. *Cipherpdf*

   Merupakan pesan gambar yang tersembunyi, yaitu pesan pdf asli (*plainpdf*) yang telah dienkripsi pada proses kriptografi. *Cipherpdf* ini dapat diubah kembali kebentuk aslinya (*plainpdf*) memanfaatkan *key* yang telah disediakan.

3. *Cryptography Key*

   Merupakan kunci yang digunakan untuk melakukan enkripsi dan deskripsi pada proses kriptografi. Tanpa adanya kunci (*key*) yang sama maka proses enkripsi dan deskripsi tidaka dapat dilakukan dengan baik. Kunci (*key*) merupakan informasi yang padat menjadi kendali terhadap proses terjadinya kriptografi.

4. *Encryption Decryption Algorithm*

   Komponen terakhir yang juga sama pentingnya dalam proses kriptografi adalah algoritma yang di gunakan untuk enkripsi dan dekripsi.

2.2.1 Aspek Keamanan Kriptografi

Keamanan telah menjadi aspek penting dari sebuah sistem informasi yang biasanya hanya diperlihatkan kepada kelompok tertentu karena penting untuk melindungi sebuah sistem informasi agar tidak jatuh ke tangan orang lain yang tidak berkepentingan. Salah satu upaya perlindungan sistem informasi yang dapat dilakukan adalah kriptografi yang memiliki beberapa aspek keamanan informasi ((Agustina & Kurniati, 2015)), yaitu :

1. Kerahasiaan (*confidentiality*)

   adalah layanan yang dirancang untuk mencegah pesan dibaca oleh pihak yang tidak berkepentingan..

2. Integritas Data (*integrity*)

   adalah layanan yang menjamin bahwa pesan lengkap/asli atau tidak dimanipulasi selama pengiriman.

3. Otentikasi (*authentication*)

   Ini adalah layanan yang terkait dengan otentikasi, yang dapat mengidentifikasi keaslian kedua pihak yang berkomunikasi (autentikasi pengguna) dan keaslian sumber pesan (otentikasi asal data)

4. Nir penyangkalan (*non repudiation*)

   Ini adalah layanan yang mencegah entitas yang berkomunikasi untuk menyangkal bahwa pengirim pesan telah menolak untuk mengirimkannya atau bahwa penerima pesan telah menolak untuk menerima pesan tersebut..

2.2.2 Serangan Terhadap Kriptografi

Setelah data dienkripsi, bukan berarti tidak ada pihak yang mau mengetahui dan menggunakan hasil password yang dienkripsi, pengetahuan seperti ini disebut dengan kriptoanalisis. Jenis serangan terhadap pesan terenkripsi adalah sebagai berikut ((Hasugian, 2017)) :

1. *Ciphertext only attack* adalah serangan yang hanya mendapatkan pesan yang disandikan..
2. *Known plaintext attack* serangan yang password dan pesan aslinya sudah diketahui.
3. *Choosen plaintext attack*adalah merupakan model serangan kriptanalisis yang mengasumsikan bahwa penyerang dapat memperoleh ciphertext dari sembarang plaintext. Tujuan serangan ini adalah untuk mendapatkan informasi yang mengurangi keamanan skema enkripsi.

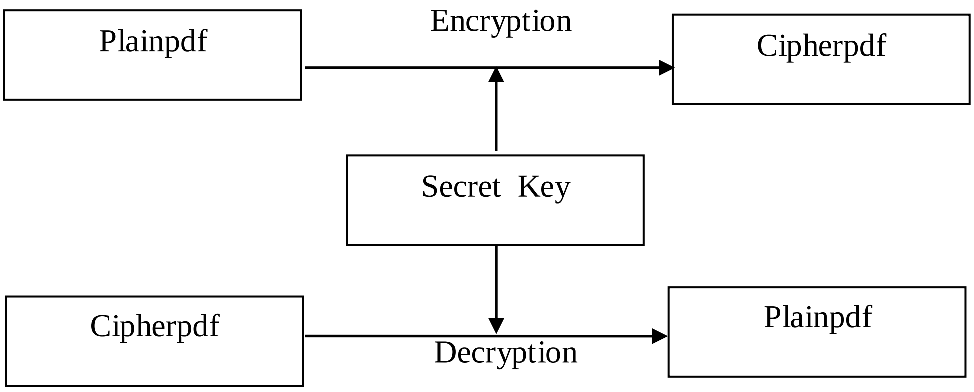
2.2.3 Pengelompokkan Algoritma Kriptografi

Menurut jenis kunci yang digunakan dalam proses enkripsi dan dekripsi, algoritma kriptografi diklasifikasikan menjadi dua jenis, yaitu cipher simetris dan cipher asimetris. Perbedaan utama antara keduanya adalah apakah kunci yang digunakan dalam proses enkripsi sama dengan kunci yang digunakan dalam proses dekripsi (Aleisa, 2015)).

1. Algoritma Kriptografi Simetris

   Enkripsi simetris atau kunci tunggal adalah salah satu metode enkripsi tertua dan seringkali sesederhana menggeser huruf teks dengan angka tertentu. Algoritma kunci simetris hanya menyediakan satu kunci untuk proses enkripsi dan dekripsi, yang dapat berupa angka acak, kata atau huruf, siapa pun yang memiliki kunci memiliki kemampuan untuk mendekripsi ciphertext. Bagian tersulit dari proses ini adalah meneruskan kunci dari pengirim ke penerima dan memastikan penerima telah menerima kuncinya**.** Jika kuncinya hilang ataudiperoleh oleh pihak ketiga, maka data yang dienkripsi menjadi tidak tersedia. Salah satu yang hebatmanfaat dari sistem kunci rahasia adalah kemampuan untuk menerjemahkan kata sandi dengan mudah dengan kuncidan kecepatan enkripsi. Bila dibandingkan dengan sistem kunci asimetris,sistem kunci simetrik adalah metode yang menarik karena aplikasinya tidak memerlukanketerlibatan pengguna eksternal ((Aleisa, 2015)). Contoh dari algoritma ini adalah *Data Encryption Standard* (DES), *Triple Data Encryption Standard* (3DES), *International Data Encryption Algorithm* (IDEA), *Advanced Encryption Standard* (AES), dan sebagainya. Adapun diagram enkripsi kunci simetri dapat dilihat pada gambar 2.1((Wibowo et al., 2015)) berikut

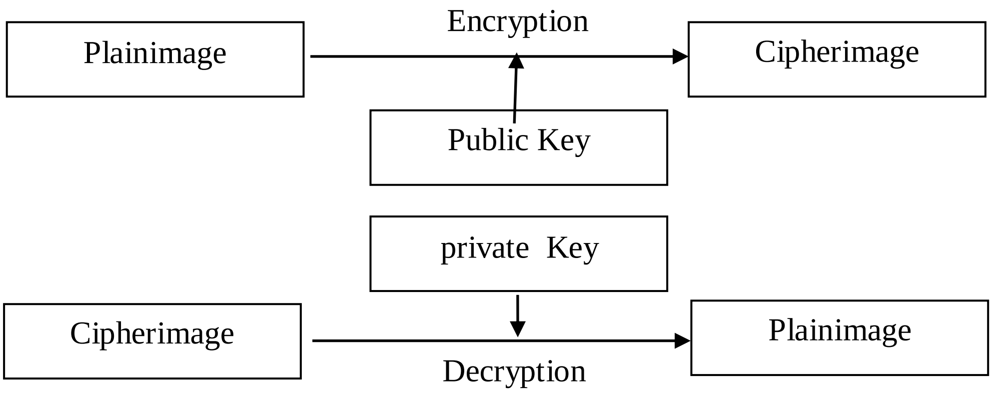
   :



Gambar 2.1 Diagram enkripsi dan dekripsi kunci simetris
Sumber : Thakur dan Khumar, 2011

2. Algoritma Kriptografi Asimetris

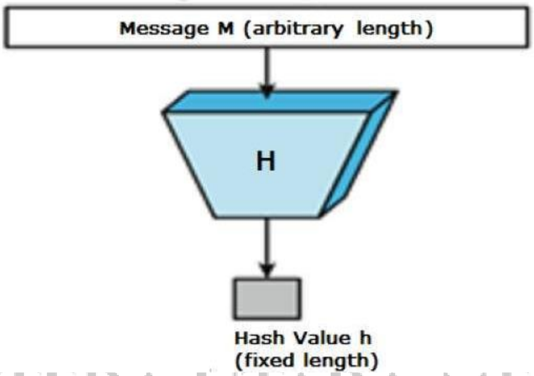
Sistem algoritme kunci asimetris, juga dikenal sebagai kriptografi kunci publik, lebih aman karena memerlukan dua kunci, publik dan pribadi, untuk mengenkripsi dan mendekripsi teks. Kunci publik diketahui semua orang, dan penerima memberikan kunci publik mereka sendiri kepada pengirim, yang menggunakannya untuk mengenkripsi teks yang akan dikirim. Penerima akan menerima ciphertext dan mendekripsi menggunakan kunci pribadinya. Kunci privat tidak pernah didistribusikan, oleh karena itu ancaman dari pihak ketiga sangat berkurang, karena tanpa kunci privat, teks tidak dapat didekripsi. Meskipun kunci publik dan kunci privat dalam kriptografi asimetris memecahkan masalah keamanan pengiriman kunci, masih ada kelemahan di bidang keamanan. Pertama, enkripsi kunci publik jauh lebih lambat daripada enkripsi kunci tunggal. Kedua, ini hanya efektif untuk sejumlah kecil data seperti email, tetapi tidak untuk enkripsi massal (Aleisa, 2015). Algoritma yang menggunakan kunci umum dan public ini antara lain *Digital Signature Algoritma* (DSA) , *Rivest Shamit-Adleman* (RSA), *Diffie-hellman* (DH), dan sebagainya. Proses enkripsi dekripsi kunci asimetri dapat dilihat pada gambar 2.2((Rohmanu, 2017)) berikut :



Gambar 2.2 Diagram enkripsi dan dekripsi kunci simetri
Sumber : Thakur dan Khumar, 2011

3. Fungsi *Hash*

Fungsi *hash* adalah fungsi matematika yang mengubah nilai *input* numerik menjadi nilai numerik terkompresi lainnya. *Input* ke fungsi *hash* adalah panjang yang tidak ditentukan tetapi *output* selalu memiliki panjang tetap.Nilai yang dikembalikan oleh fungsi *hash* disebut pesan digest atau hanya nilai *hash* ((Sulastri & Putri, 2018)). Gambar berikut ini menggambarkan fungsi *hash* :



Gambar 2.3 Fungsi *Hash*
Sumber : Sulastri dan Putri, 2018

2.3Algoritma Data Encryption Standard(DES)

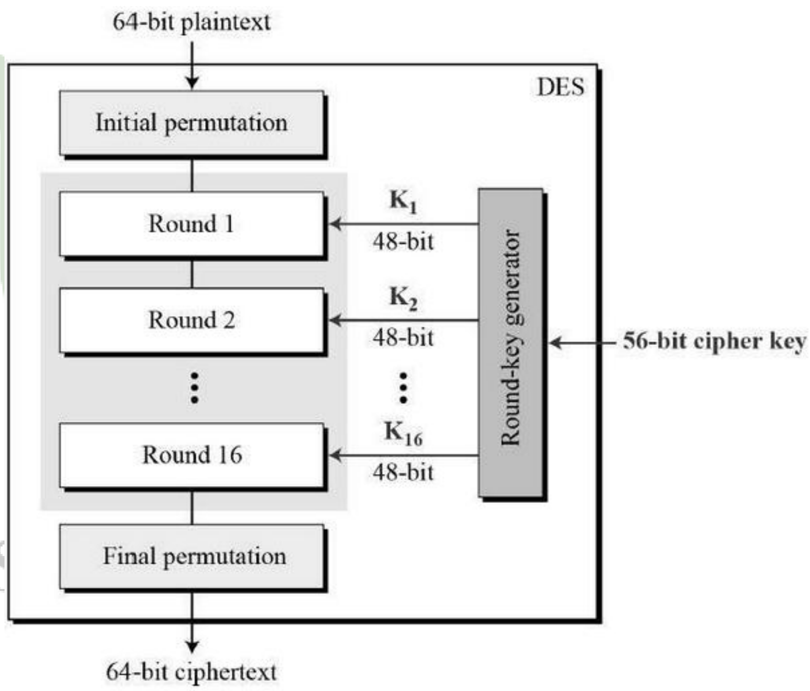
Algoritma *Data Encryption Standard* (DES) adalah *cipher* blok simetris-kunci yang diterbitkan oleh *National Institute of Standard and Technology*(NIST).DES adalah implementasi dari *Cipher Feistel*. Menggunakan 16 ronde struktur *Feistel*. Ukuran blok adalah 64-bit. Meskipun, panjang kunci adalah 64-bit, DES memiliki panjang kunci efektif 56 bit, karena 8 dari 64 bit kunci tidak digunakan oleh algoritma enkripsi ((Winafil et al., 2018)).

DES mentransformasikan input 64 bit dalam beberapa tahap enkripsi ke dalam *output* 64 bit. DES termasuk *block cipher*, dengan tahapan dan kunci yang sama, DES digunakan untuk membalik enkripsi. Kunci internal pada algoritma DES dibangkitkan dari kunci eksternal (*external key*) 64 bit ((Primartha, 2011)).

Tiga tahapan besar dalam DES yaitu:

1. *Plaintext* yang berukuran 64 bit dipermutasi dengan matriks permutasi awal.
2. Hasil permutasi awal kemudian di-enciphering sebanyak 16 kali (16 putaran). Setiap putaran menggunakan kunci internal yang berbeda.
3. Hasil *enciphering* kemudian dipermutasi dengan matriks permutasi balikan (invers initial permutation atau IP-1) menjadi blok *ciphertext*.

Struktur Umum DES digambarkan dalam ilustrasi berikut :



Gambar 2.4 Struktur DES
Sumber : Mohtashim, 2014

Karena DES didasarkan pada *Feistel Cipher* yang diperlukan untuk menentukan DES (Mohtashim,2014)adalah :

1. *Key Schedule*

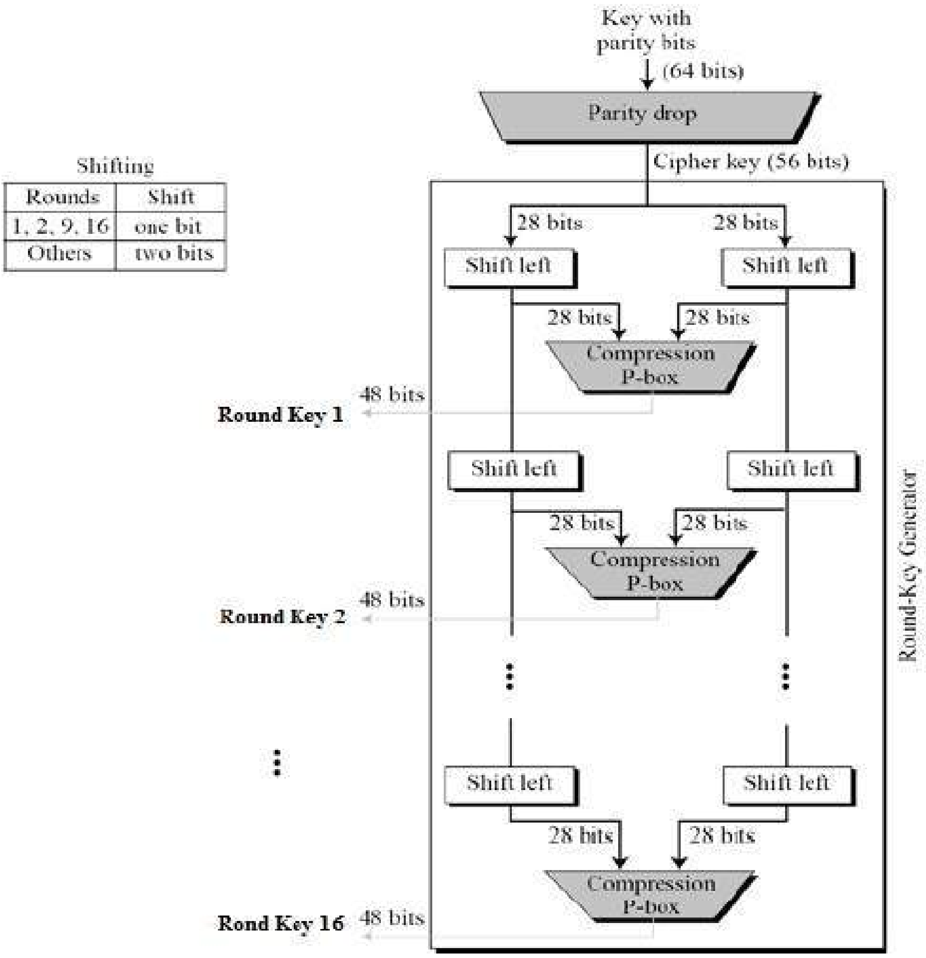
2. *Round Function*

3. *Initial* dan *final permutation*

Berikut tranformasi dari algoritma DES :

1. *Genrate Key*

Generator putaran kunci menciptakan enam belas kunci 48-bit dari kunci *cipher* 56-bit. Proses pembangkitan kunci digambarkan dalam ilustrasi berikut :



Gambar 2.5 Proses Pembangkitan Kunci DES
Sumber : Mohtashim, 2014

Adapun nilai tabel kompresi *permuted choice* 1 (PC^{-1}) dan tabel *permuted choice 2* (PC^{-2}) adalah sebagai berikut :

Tabel 2.1*Permuted Choice* 1 (PC^{-1})

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

Sumber : Jagbir Dhillon et al, 2011

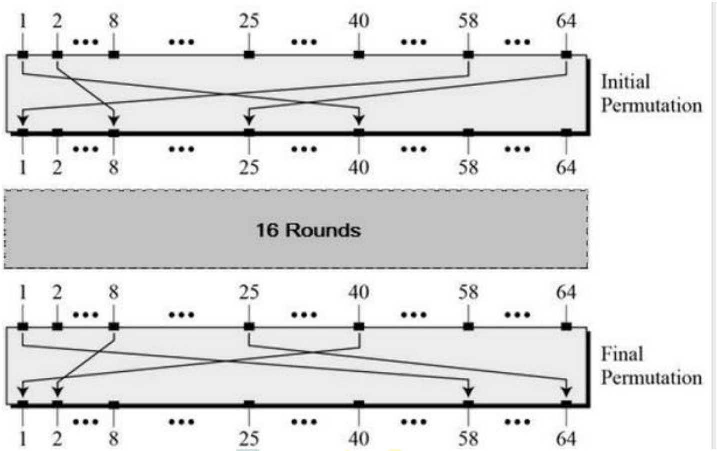
Tabel 2.2*Permuted Choice* 2 (PC^{-2})

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

Sumber : Jagbir Dhillon et al, 2011

2. *Initial* dan *Final Permutation* (Permutasi Awal dan Akhir)

Permutasi awal dan akhir adalah kotak *Permutation* lurus (P-*boxes*) yang invers satu sama lain. Permutasi awal dan akhir ditampilkan sebagai gambar 2.6 berikut :



Gambar 2.6 Proses Permutasi DES
Sumber : Mohtashim, 2014

Adapun nilai tabel permutasi awal (IP) dan tabel permutasi akhir (IP^{-1}) adalah sebagai berikut :

Tabel 2.3*Initial Permutation* (IP)

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Sumber : Jagbir Dhillon et al, 2011

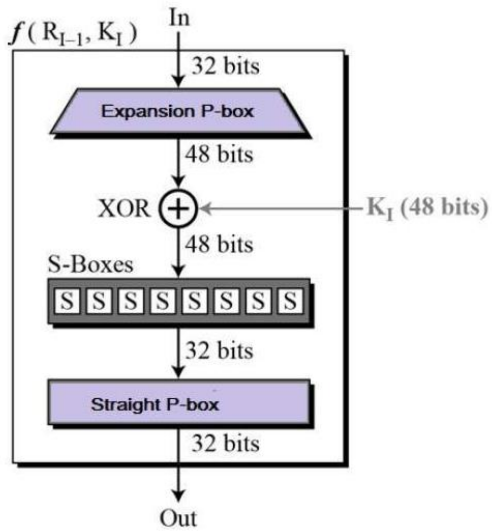
Tabel 2.4*InverseInitial Permutation* (IP^{-1})

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

Sumber : Jagbir Dhillon et al, 2011

3. *Round Function*

Inti dari *cipher* ini adalah fungsi *f*. Fungsi DES menerapkan kunci 48-bit ke 32 bit paling kanan untuk menghasilkan output 32-bit.

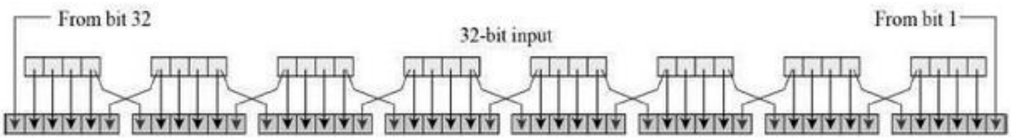


Gambar 2.7 Proses *Round Function* DES
Sumber : Mohtashim, 2014

Adapun keterangan pada gambar 2.7 adalah :

a. *Expansion Permutation Box*

Karena input yang benar adalah 32-bit dan kunci ronde adalah 48-bit, pertama-tama perlu memperluas masukan yang tepat ke 48 bit. Ilustrasi permutasi dapat di lihat pada gambar 2.8 di bawah ini :



Gambar 2.8 Proses Ekspansi DES
Sumber : Mohtashim, 2014

32	01	02	03	04	05
04	05	06	07	08	09
08	09	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	31	31	32	01

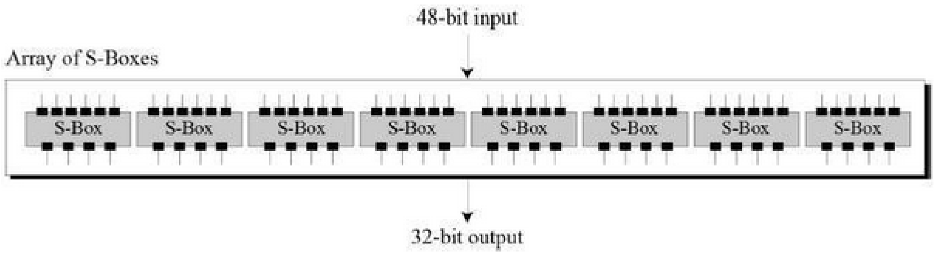
Gambar 2.9 Tabel Ekspansi DES
Sumber : Mohtashim, 2014

b. Proses XOR

Setelah proses permutasi ekspansi, DES melakukan operasi XOR pada bagian kanan yang diperluas dan kunci ronde. Kunci ronde hanya digunakan dalam operasi ini.

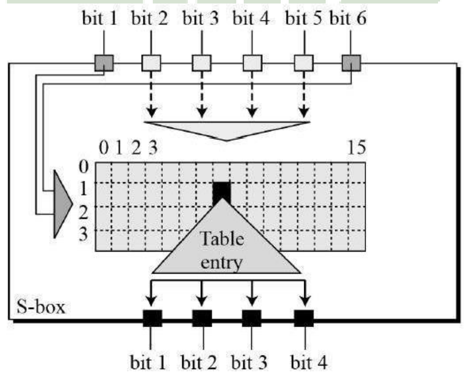
c. Subtitusi Tabel S-Box

Proses ini mensubtitusikan 8 buah S-Box. DES menggunakan 8 S-Box, masing-masing dengan input 6-bit dan output 4-bit. Ilustrasi subtitusi S-Box sebagai gambar 2.10berikut :



Gambar 2.10 Proses Subtitusi S-Box DES
Sumber : Mohtashim, 2014

Aturan ilustasi S-Box pada DES sebagai gambar 2.11 di bawah ini:



Gambar 2.11 Aturan Proses Subtitusi S-Box
Sumber : Mohtashim, 2014

Ada total delapan tabel S-Box. *Output* dari kedelapan S-Box ini kemudian digabungkan menjadi bagian 32 bit.Berikut nilai dari kedelapan S-Box DES (Hertel, 2021) :

Tabel 2.5 Nilai 8 S-Box DES

S1															
14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8

4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13
S2															
15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
3	13	14	7	15	2	8	14	12	0	1	10	6	9	11	5
0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9
S3															
10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
13	7	0	9	3	4	5	10	2	8	5	14	12	11	15	1
13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12
S4															
7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14
2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	16
4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3
S6															
12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13
S7															
4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12

Tabel 2.5 Lanjutan Nilai 8 S-Box DES

S1															
S8															
13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	1

Sumber : Hertel, 2021

d. Permutasi P-Box

Output 32 bit dari S-box kemudian mengalami permutasi dengan tabel P-Box.Gambar tabel P-Box :

16	07	20	21	29	12	28	17
01	15	23	26	05	18	31	10
02	08	24	14	32	27	03	09
19	13	30	06	22	11	04	25

Gambar 2.12 Tabel P-Box DES
Sumber : Mohtashim, 2014

2.4 File PDF

PDF *(Portable Document Format)* adalah jenis format dokumen atau berkas untuk keperluan pertukaran dokumen digital yang dibuat oleh *Adobe System* pada tahun 1993. Awalnya format ini digunakan untuk kepentingan pertukaran dokumen digital, namun sekarang format PDF juga banyak digunakan untuk presentasi dokumen dua dimensi yang terdiri dari huruf, teks, grafik vector dan citra (Maxmanroe, 2021).

Sesuai dengan pengertian PDF di atas, format ini sering digunakan dalam administrasi perkantoran karena kemudahannya. Meskipun pada awal rilis format ini kurang diminati karena kalah bersaing dengan pendahulunya yakni *MicrosoftOffice*, namun karena memiliki fitur yang memudahkan penggunanya untuk mengarsipkan dokumen membuat PDF saat ini semakin banyak digunakan.

Mengacu pada pengertian PDF *(Portable Document Format)* di atas, maka fungsi PDF adalah untuk membuat dan menyimpan sebuah dokumen yang berisi teks, gambar, dan link, dengan format khusus yang hanya dapat dibuka di komputer yang memiliki program khusus (Maxmanroe, 2021).

2.5 Tabel ASCII

American Standard Code for Information Interchange, atau kode ASCII, dibuat pada tahun 1963 oleh Komite "*American Standards Association*" atau "ASA", agensi tersebut mengubah namanya pada tahun 1969 oleh

"*AmericanNational Standards Institute*" atau "ANSI" sebagaimana adanya dikenal sejak (Nguyen, 2021).

Adapun tabel ascii keseluruhan dapat dilihat pada gambar di bawah ini :

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 2.13 Tabel Ascii 0-127
Sumber : Nguyen, 2021

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
128	80	Ç	160	A0	á	192	C0	└	224	E0	α
129	81	ü	161	A1	í	193	C1	┴	225	E1	ß
130	82	é	162	A2	ó	194	C2	┬	226	E2	Γ
131	83	â	163	A3	ú	195	C3	├	227	E3	π
132	84	ä	164	A4	ñ	196	C4	─	228	E4	Σ
133	85	à	165	A5	Ñ	197	C5	┼	229	E5	σ
134	86	å	166	A6	ª	198	C6	╞	230	E6	µ
135	87	ç	167	A7	º	199	C7	╟	231	E7	τ
136	88	ê	168	A8	¿	200	C8	╚	232	E8	Φ
137	89	ë	169	A9	⌐	201	C9	╔	233	E9	Θ
138	8A	è	170	AA	¬	202	CA	╩	234	EA	Ω
139	8B	ï	171	AB	½	203	CB	╦	235	EB	δ
140	8C	î	172	AC	¼	204	CC	╠	236	EC	∞
141	8D	ì	173	AD	¡	205	CD	═	237	ED	φ
142	8E	Ä	174	AE	«	206	CE	╬	238	EE	ε
143	8F	Å	175	AF	»	207	CF	╧	239	EF	∩
144	90	É	176	B0	░	208	D0	╨	240	F0	≡
145	91	æ	177	B1	▒	209	D1	╤	241	F1	±
146	92	Æ	178	B2	▓	210	D2	╥	242	F2	≥
147	93	ô	179	B3	│	211	D3	╙	243	F3	≤
148	94	ö	180	B4	┤	212	D4	╘	244	F4	⌠
149	95	ò	181	B5	╡	213	D5	╒	245	F5	⌡
150	96	û	182	B6	╢	214	D6	╓	246	F6	÷
151	97	ù	183	B7	╖	215	D7	╫	247	F7	≈
152	98	ÿ	184	B8	╕	216	D8	╪	248	F8	≈
153	99	Ö	185	B9	╣	217	D9	┘	249	F9	·
154	9A	Ü	186	BA	║	218	DA	┌	250	FA	·
155	9B	¢	187	BB	╗	219	DB	█	251	FB	√
156	9C	£	188	BC	╝	220	DC	▄	252	FC	ⁿ
157	9D	¥	189	BD	╜	221	DD	▌	253	FD	²
158	9E	₧	190	BE	╛	222	DE	▐	254	FE	■
159	9F	ƒ	191	BF	┐	223	DF	▀	255	FF	

Gambar 2.14 Tabel Ascii128-255
Sumber : Nguyen, 2021

2.6 PHP

PHP (*Hypertext Preprocessor*) adalah bahasa *script* yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk membuat program situs web dinamis. PHP dapat digunakan dengan gratis (*free*) dan bersifat *OpenSource*. PHP dirilis dalam lisensi PHP *license*. Untuk membuat program PHP kita diharuskan untuk menginstal *webserver*terlebih dahulu. (Ayu & Permatasari, 2018).

Beberapa keunggulan yang dimiliki oleh produk PHP diantaranya:

1. Pemograman yang *opensource*.
2. Tingkat akses yang lebih cepat.
3. Tingkat *lifecycle* yang cepat sehingga selalu mengikuti perkembangan teknologi internet.
4. Keamanan yang tinggi.
5. Mampu berjalan di beberapa server yang ada misalnya *Apache*, *Microsoft* IIS dan lainnya.

6. Mendukung akses ke beberapa *database* yang sudah ada, baik yang bersifat gratis ataupun komersil (Abdurahman & Prasetyo, 2016).

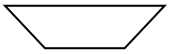
2.7 Flowchart

Untuk menggambarkan sebuah algoritma terstruktur dan mudah dipahami oleh orang lain (khususnya programer yang bertugas mengimplementasikan program), maka diperlukan alat bantu berupa diagram alir (*flowchart*). *Flowchart* menggambarkan urutan logika dari suatu prosedur pemecahan masalah, sehingga *flowchart* merupakan langkah-langkah penyelesaian masalah yang dituliskan dalam simbol-simbol tertentu. diagram alir ini akan menunjukan alur didalam program secara logika. Diagram alir ini selain dibutuhkan sebagai alat komunikasi juga diperlukan sebagai alat dokumentasi ((Ariyus, 2008)). Sebelum lebih jauh memahami komponen-komponen diagram alir, maka perlu kiranya disampaikan aturan-aturan dalam perancangan diagram alir tersebut, yaitu :

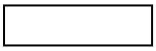
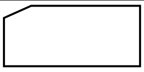
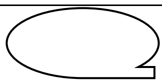
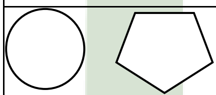
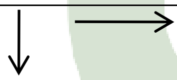
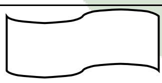
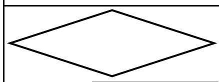
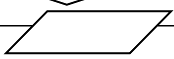
1. Diagram alir digambarkan dengan orientasi dari atas dari atas kebawah dan dari kiri kekanan.
2. Setiap kegiatan atau proses dalam diagram alir harus dinyatakan secara eksplisit.
3. Setiap digram alir harus dimulai dari satu *start state* dan berakhir pada satu atau lebih terminal akhir atau terminator atau halt state.
4. Gunakan *connector* dari *off-pege connector state* dengan label yang sama untuk menunjukkan keterhubungan antarpath algoritma yang terputus atau terpotong, misalnya sebagai akiat pindah atau ganti halaman.

Tujuan dari *flowchart* adalah untuk menggamarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi dan jelas menggunakan simbol-simbol yang standar. Simbol-simbol yang digunakan untuk menggambarkan algoritma dalam bentuk diagram alir dan kegunaan dari simbol-simbol yang bersangkutan. Adapun simbol-simbol dan kegunaannya sebagai berikut.

Tabel 2.6 Tabel simbol-simbol *flowchart*

Simbol	**Maksud**	**Kegunaan**
	Kegiatan manual	Menunjukan Perkerjaan manual

Tabel 2.6 Tabel simbol-simbol *flowchart*(**Lanjutan**)

Simbol	**Maksud**	**Kegunaan**
	Proses	Menunjukan kegiatan atau proses dan opersi program computer
	Dokumen	Menunjukan dokumen input dan output baik untuk proses manual, mekanik, komputer.
	Kartu prolok	Menunjukan input atau output kartu plong
	Pita magnetic	Menunjukan input atau output menggunakan pita magnetic
	Disket	Menunjukan input atau output menggunakan disket.
	Display	Menunjukan output yang ditampilkan ke monitor
	Penghubung	Menunjukan penjelasan dan penghubung ke halaman yang masih sama atau kehalaman lain.
	Garis alir	Menunjukan arus dan proses
	Pita kertas	Menunjukan input atau output menggunakan pita kertas berlubang
	Start	Awal/akhir program
	Inisialisai	Menginisialisasikan input kedalam bentuk variabel untuk diteruskan kedalam proses
	Keputusan	angkah pengambilan keputusan
	Input/output	Memasukan data /menunjukan hasil dari suatu proses

Sumber : (Nugroho, 2005)

2.7 Penelitian Terkait

Adapun penelitian terkait judul yang diambil dapat diliat pada tabel dibawah ini:

Tabel 2.7 Penelitian Terkait

No	Penelitian	Asal&Tahun	Judul	Kesimpulan
1	Deny Adhar	Universitas Potensi Utama,2019	Implementasi Algoritma Des (Data Encryption Standard) Pada Enkripsi Dan Deskripsi Sms Berbasis Android	DES merupakan algoritma penyandian simetris, dimana untuk proses enkripsi dan dekripsi pesan menggunakan kunci yang sama. Jadi, walaupun seorang criptanalis rmengerti dengan baik algoritma yang digunakan untuk menyandikan pesan tersebut, tapi kalau tidak tahu kunci yang digunakan, maka tidak akan dapat mendekripsi pesan SMS tersebut.
2	Esti Rahmawati Agustina , Agus Kurniati	UPN ”Veteran” Yogyakarta, 2009	Pemanfaatan kriptografi dalam mewujudkan keamanan informasi pada e-voting di indonesia	pemanfaatan kriptografi dalam mewujudkan keamanan informasi pada e-voting di Indonesia yaitu dapat mendukung aspek-aspek keamanan informasi meliputi kerahasiaan, integritas data, otentikasi dan nir penyangkalan. Sehingga sebuah protokol kriptografi dapat memenuhi requirement dasar e-voting. Namun, terdapat satu aspek dalam keamanan informasi yaitu

				ketersediaan yang tidak dapat didukung oleh kriptografi.
3	Rifkie Primartha	Universitas Sriwijaya,2011	Penerapan enkripsi dan dekripsi file menggunakan algoritma Data Encryption Standard (DES).	Dengan adanya aplikasi kriptografi yang dikembangkan berdasarkan algoritma DES, maka data-data penting dapat diamankan (dienkripsi) ketika hendak dikirim melalui media internet. Proses enkripsi dan dekripsi file maupun teks, pada prinsipnya memiliki mekanisme proses yang sama.
4	Neti Rusri Yanti, Alimah, Desi Afrida Ritonga	a STMIK Budidarma Medan,2018	Implementasi Algoritma Data Encryption Standard Pada Penyandian Record Database.	Tingkat keamanan dari yang dibuat cukup aman karena algoritma DES memiliki panjang kunci yang besar. Kunci internal yang berjumlah 56 bit didapatkan dari kunci eksternal yang berjumlah 64 bit
5	Nurmarlina Siregar	STMIK Budi Darma,2019	Perancangan aplikasi keamanan pesan teks dengan menggunakan algoritma triple des	Proses enkripsi dan dekripsi suatu pesan teks dengan algoritma triple DES dilakukan dengan cara mengimplementasikan algoritma DES sebanyak 3 kali, sesuai dengan pemilihan kuncinya dan urutan proses yang dipilih yaitu algoritma triple DES sederhana.