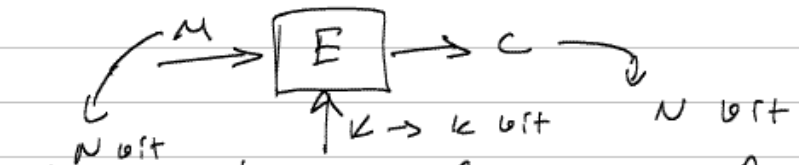
Note Title 8/28/2006

$E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$



$E^{-1} : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$

function $\mathrm{DES}_K(M)$ // $|K| = 56$ and $|M| = 64$
  $(K_1, \ldots, K_{16}) \leftarrow KeySchedule(K)$ // $|K_i| = 48$ for $1 \le i \le 16$
  $M \leftarrow IP(M)$
  Parse M as $L_0 \| R_0$ // $|L_0| = |R_0| = 32$
  for $r = 1$ to 16 **do**
    $L_r \leftarrow R_{r-1}$; $R_r \leftarrow f(K_r, R_{r-1}) \oplus L_{r-1}$
  $C \leftarrow IP^{-1}(L_{16} \| R_{16})$
  return C

Feistel round

critical parts

Figure 2.1: The DES blockcipher. The text and other figures describe the subroutines *KeySchedule*, f, IP, IP^{-1}.

IP

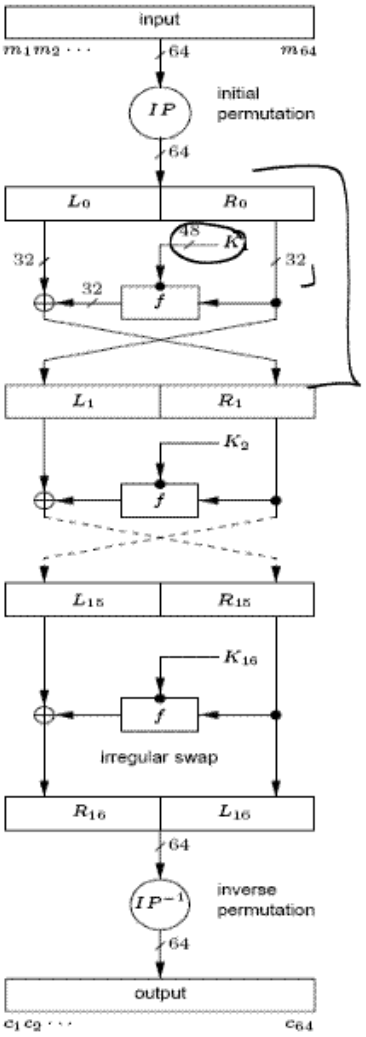
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

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

Figure 2.2: Tables describing the DES initial permutation IP and its inverse IP^{-1}.

(a) twisted ladder



$R_i \leftarrow L_{i-1} \oplus f(K_i, R_{i-1})$

$L_i \leftarrow R_{i-1}$

```
function f(J, R)    // |J| = 48 and |R| = 32
    R ← E(R) ; R ← R ⊕ J
    Parse R as R_1 ‖ R_2 ‖ R_3 ‖ R_4 ‖ R_5 ‖ R_6 ‖ R_7 ‖ R_8    // |R_i| = 6 for 1 ≤ i ≤ 8
    for i = 1, ..., 8 do
        R_i ← S_i(R_i)    // Each S-box returns 4 bits
    R ← R_1 ‖ R_2 ‖ R_3 ‖ R_4 ‖ R_5 ‖ R_6 ‖ R_7 ‖ R_8    // |R| = 32 bits
    R ← P(R)
    return R
```

Figure 2.3: The f-function of DES. The text and other figures describe the subroutines used.

E

32	1	2	3	4	5
4	5	6	7	8	9
8	9	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	30	31	32	1

P

16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

Figure 2.4: Tables describing the expansion function E and final permutation P of the DES f-function.

$S_2(011010)$

01 | 1010 | 10

output 12

S_1

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

S_2

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

S_3

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

S_4

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

S_5

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

S_6

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

S_7

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

S_8

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

Figure 2.5: The DES S-boxes.

Algorithm *KeySchedule*(K) // $|K| = 56$
$K \leftarrow$ *PC*-1(K)
Parse K as $C_0 \| D_0$
for $r = 1, \ldots, 16$ **do**
  if $r \in \{1, 2, 9, 16\}$ **then** $j \leftarrow 1$ **else** $j \leftarrow 2$ **fi**
  $C_r \leftarrow leftshift_j(C_{r-1})$; $D_r \leftarrow leftshift_j(D_{r-1})$
  $K_r \leftarrow$ *PC*-2$(C_r \| D_r)$
return$(K_1, \ldots, K_{16})$

Figure 2.6: The key schedule of DES. Here $leftshift_j$ denotes the function that rotates its input to the left by j positions.

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

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

Figure 2.7: Tables describing the *PC*-1 and *PC*-2 functions used by the DES key schedule of Fig. 2.6.

Cryptanaylrss

Frx blockcrpher wrth given decriptron

First Goal : Key Recovery

For fixed key T, you are given

$(M_1, E_T(M_1)) \cdots (M_q, E_T(M_q))$

Known – Message Attack

You are given the messages

Chosen ciphertext Attack

E.g.

for { $i = 1$ to 2^k }

{

If ($\forall j \quad E_{T_i}(M_j) = C_j$)

then return T_i

}

worst case: $O(2^k)$ operations

Expected Running Time: $\sum_{i=1}^{2^k} i \cdot Pr[K = T_i]$

$= \sum_{i=1}^{2^k} i \cdot \frac{1}{2^k} = \frac{2^k \cdot (2^k + 1)}{2^{k+1}}$

$\approx 2^{k-1}$

2DES :

$2DES(K_1, K_2, M) = DES_{K_2}(DES_{K_1}(M))$

$2DES^{-1}(K_1, K_2, M) = DES_{K_1}^{-1}(DES_{K_2}^{-1}(M))$

$2^{56}, 2^{112}$

Given (M, C)

$DES_{K_1}(M) = DES_{K_2}^{-1}(C)$

for $T = 1$ to 2^k

$A[T] = DES_{T_T}(M)$

for $T = 1$ to 2^k

$B[T] = DES_{T_T}^{-1}(C)$

Find J, K $(A[J] = B[K])$

$\downarrow$

(T_J, T_K) Recovered Key

2^{57} DES, DES^{-1} operations

$DES_{K_3}(DES_{K_2}^{-1}(DES_{K_1}(M)))$

$= 3DES3(K_1, K_2, K_3, M)$

$$DES_{K_2}(DES_{K_1}^{-1}(DES_{K_2}(M))) = 3DES2(K_1, K_2, M)$$

$$DES(K, M) = 3DES3(K, K, K, M)$$

$$= DES_K(DES_K^{-1}(DES_K(M)))$$

$$DES(K, M) = 3DES2(K, K, M)$$

$$= DES_K(DES_K^{-1}(DES_K(M)))$$