Cryptanalysis

Block Ciphers 2

John Manferdelli
JohnManferdelli@hotmail.com

Differential Cryptanalysis of DES

How input differentials affect output

- Expansion Matrix

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

Out	1	2	3	4
1	4,4	2,3	5,4	6,1
2	8,1	3,4	7,4	5,1
3	1,1	4,3	6,3	7,2
4	2,1	5,2	8,3	3,2
5	1,2	2,4	6,4	4,2
6	8,4	7,3	1,3	3,1
7	5,3	4,1	8,2	2,2
8	6,2	3,3	1,4	7,1

- After P
  - On average 1 bit difference affects 3 S boxes in next round after expansion.

How input differentials affect output

- Expansion Matrix

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

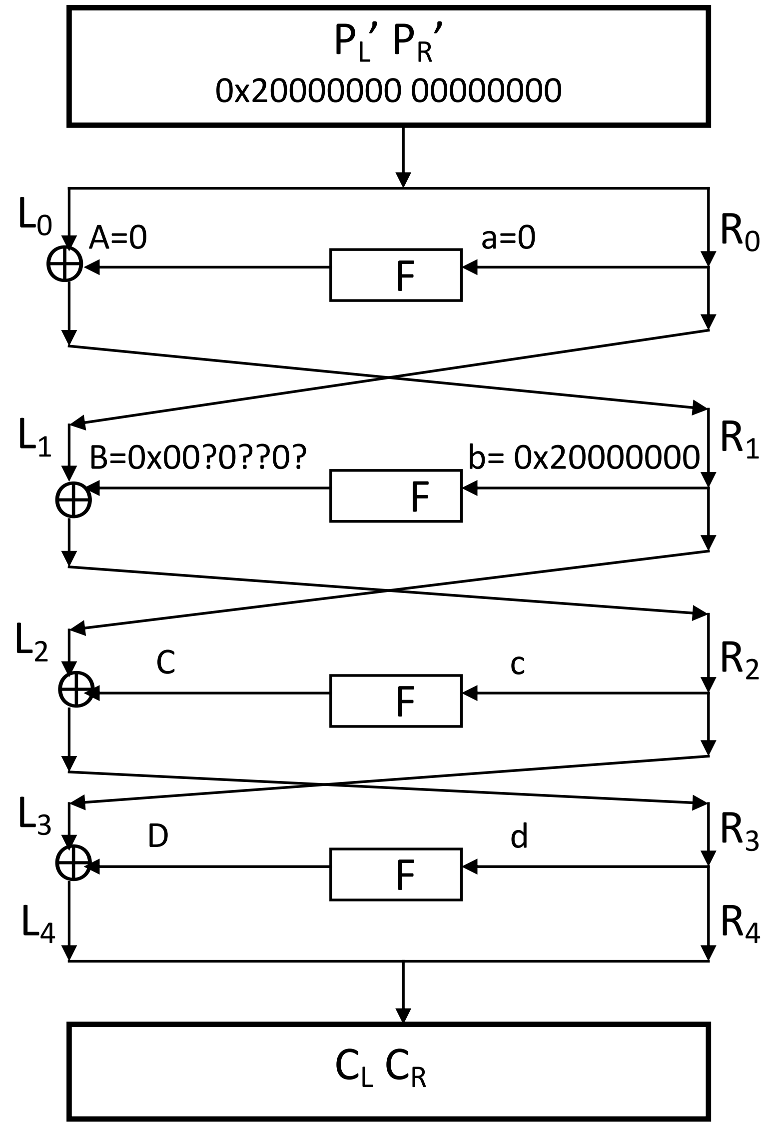
P	1	2	3	4
1	16	7	20	21
2	29	12	28	17
3	1	15	23	26
4	5	18	31	10
5	2	8	24	14
6	32	27	3	9
7	19	13	30	6
8	22	11	4	25

Out	1	2	3	4
1	4	2	5	6
2	8	3	7	5
3	1	4	6	7
4	2	5	8	3
5	1	2	6	4
6	8	7	1	3
7	5	4	8	2
8	6	3	1	7

- After P
  - Affected by box

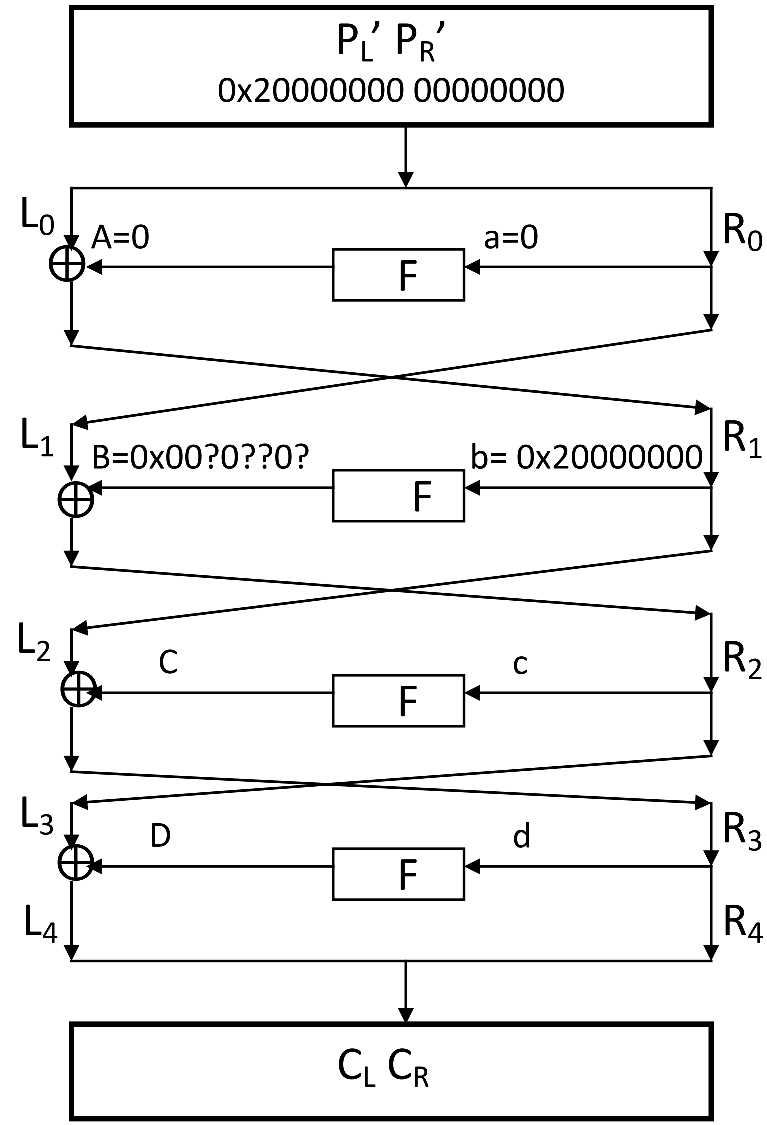
DC of DES, 4 rounds - 1

- Input differential: 0x20000000 00000000
- A'= 0, a'=0; b'= 0x20000000, B' is affected (at most) as mask=0x00808202=P(f0000000) since only the first S box is non-zero
- d'= C_R' is known
- D'= C_L'⊕B' is known in 28 bits (all but the mask positions: 0x00808202)
- S/N= pk/(λγ), is the ratio of discarded pairs to all pairs, is the number of keys suggested by a pair. Remember only about .8 of xor output patterns are possible.
- Bits that leave all S-boxes but S_1 are valid.
- Weighted probabilities (next slide)
- For each S box, try all 2^6 keys and bump counts for each key which matches the differential, d'→D'.

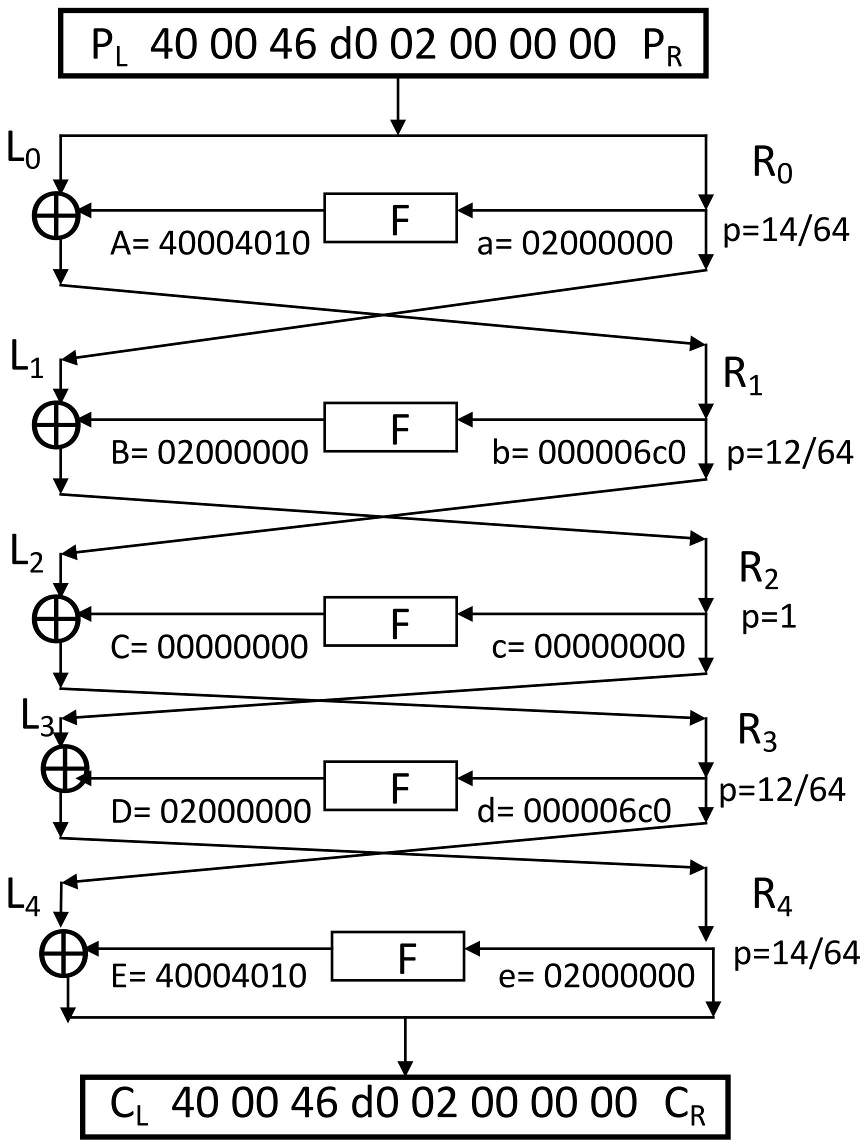


DC of DES, 4 rounds - 2

- For Sbox 1:
  - 0x04→0x3, p= 6/64 (0x00000202)
  - 0x04→0x5, p= 10/64 (0x00800002)
  - 0x04→0x6, p= 10/64 (0x00800200)
  - 0x04→0x7, p= 6/64 (0x00800202)
  - 0x04→0x9, p= 4/64 (0x00008002)
  - 0x04→0xa, p= 6/64 (0x00008200)
  - 0x04→0xb, p= 4/64 (0x00008202)
  - 0x04→0xc, p= 2/64 (0x00808000)
  - 0x04→0xd, p= 8/64 (0x00808002)
  - 0x04→0xe, p= 6/64 (0x00808200)
  - 0x04→0xf, p= 2/64 (0x00808202)



DC of DES, 5 rounds

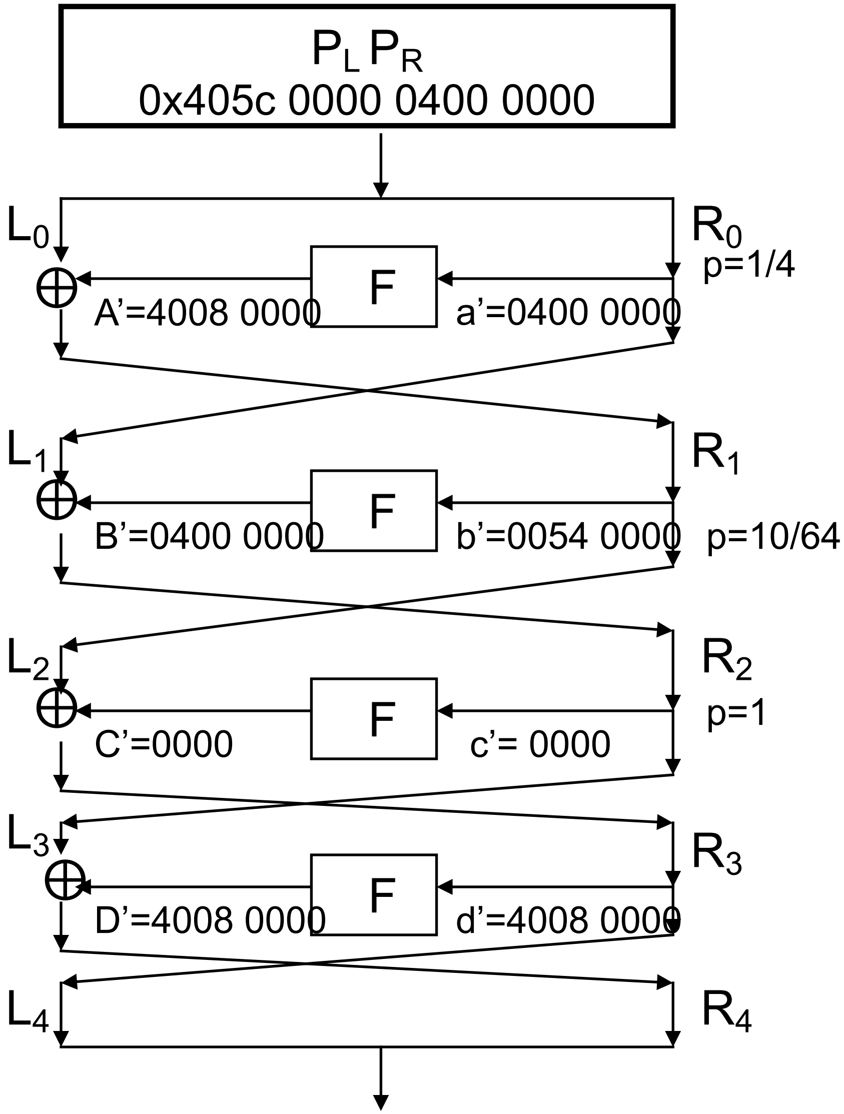


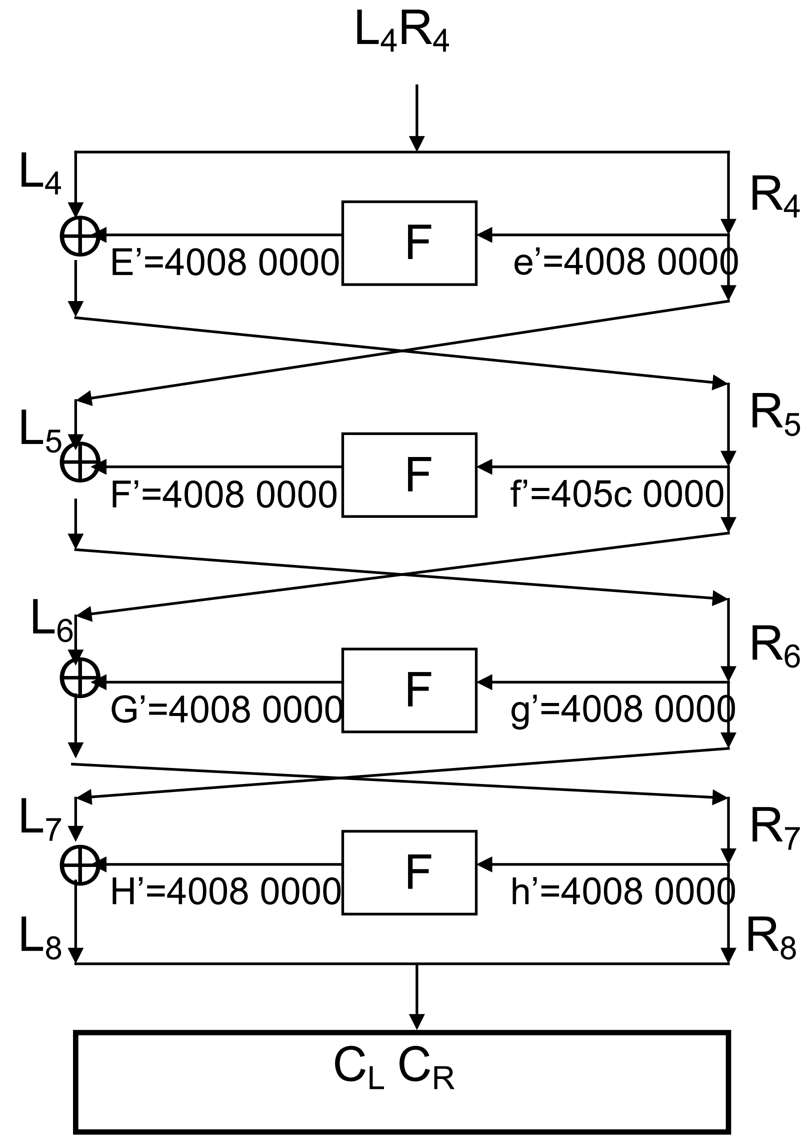
- $(P_1, P_2) \rightarrow (C_1, C_2)$ gives information about K_5 in S_2.

- $C_L \oplus L_4 = E$
- $L_2 \oplus L_4 = 0$
- $L_2 \oplus L_0 = A$
- So, $L_0 \oplus L_4 = E \oplus A$

- 02000000 → 40004010, p=14/64
- 000006c0 → 02000000, p=12/64
- Need 3-5 right pairs
- Pr[wrong pair]= 2^{-64}
- Expected # of wrong pairs is $m2^{-64}$

DC of DES, 8 rounds - 1





DC of DES, 8 rounds - 2

- Requires 25,000 cipher texts. Finds 30 bits in K_8.
- Uses 5 round differential 405c 0000 0400 0000 → 405c 0000 0400 0000 for five rounds, p= 1/10485.76.
- $f' = d' \oplus E' = b' \oplus A' = L'$, $H' = l' \oplus g' = l' \oplus e' \oplus F'$
- $S/N = 2^{30}/(4^5 \cdot 10485.76) = 100$
- 4008 0000= P(0a00 0000), 0400 0000=P(0010 0000)
- $S/N = 2^{30}/(4^5 \cdot 10485.76) = 100$ for 30 bits --- too many counters.
- Reduce to 24 bit search with enhanced probability.
  - e' → E'=P(0W 00 00 00)= X0 0Y Z0 00=f' = X0 5V Z0 00.
  - W ε{1,2,3,8,9,a,b}, Xε{0,4}, Yε{0,8}, Zε{0,4}. $V = Y \oplus 4$.
  - Z=0, 0400 0000→4008 0000, p=1/4, all others Z=4, p=20/64
  - $p_{e' \rightarrow E'} = 1/4 + .8(20/64) = 1/2$
  - Pr(24 bit, differential)= $[(16 \cdot 10 \cdot 16)/64^3] \cdot [(16 \cdot 10 \cdot 32)/64^3] = 1/5243$

DC of DES, 8 rounds - 3

- For enhanced probability, 24 bits, find keys in S_2, S_6, S_7, S_8.
- e'=0400 0000 $\rightarrow$ E'=P(0w 00 00 00)= x0 0y z0 00= f' = x0 5v z0 00.
- S/N= $2^{24}/(4^4 \cdot .8 \cdot 5243)$= 15.6
- Alternatively use 18 bit count (S_6, S_7, S_8), requiring 150,000 pairs with S/N= 1.2 followed by 12 bits.
- These keys allow us to calculate 20 bits of H, H*.
- Can use this to complete K_8 (48 bits).
- Final 8 bits from exhaustive search.

DC of DES, 8 rounds - 4

- 18 bits of key, 150,000 pairs from S_2, S_6, S_7, S_8
    1. Set up 2^{18} counter
    2. Preprocess S_I, $S_I' \rightarrow S_O'$.
    3. For each cipher text pair
        a. Calculate $S_{EH}'=S_{Ih}'$, $S_{Oh'}'$ for S_2, S_5, S_6, S_7, S_8
        b. For each of S_2, S_5, S_6, S_7, S_8, check is $S_{ih}' \rightarrow S_{Oh}'$ is not satisfied for any S-box. If so, discard.
        c. For S_6, S_7, S_8, get all S_{Ih} which are possible for $S_{ih} \rightarrow S_{Ih}$. Calculate $S_{Kh} = S_{Ih} \oplus S_{Eh}$
    4. Get entry of maximal count

Full Differential Attack on DES

- Use 0 → 0 and concatenated 2R characteristic with $p = \frac{1}{234}$ to get 13th round with $p=2^{-47.2}$.
- Want 1960 0000 0000 0000
- Candidate in round 16 has 20 ciphertexts with 0, use 2^{24}
- 2^{-20} of these
- Additional filter: 3 xors can only produce 15 outputs
- Survival rate: .0745, get 1.19 for $2^{35.2}$ structures
- Rate of values not discarded in round 16 is $2^{-32}/(4/5)^8$
- This gives 1.19x.84 =1 key

Summary of DES DC Attacks

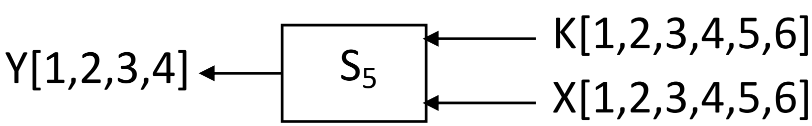
# Rounds	# Pairs needed	# Pairs used	# bits found	# chrtstcs	p	S/N	l	g
4	2^3	2^3	24	1	1	16		
6	2^7	2^7	30	3	1/16	2^{16}		
8	2^{15}	2^{13}	30	5	1/104656	15.6		
8	2^{17}	2^{13}	30	5	1/104656	1.2		
8	2^{20}	2^{19}	30	5	1/55000	1.5		
9	2^{25}	2^{24}	30	6	10^{-6}	1.0		
9	2^{26}	2^8	48	7	10^{-24}	2^{23}		

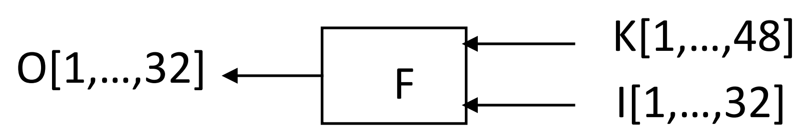
- For simple attacks

Linear Cryptanalysis of DES

One round linear constraint

- $S_5(x_1 \oplus k_1, x_2 \oplus k_2, x_3 \oplus k_3, x_4 \oplus k_4, x_5 \oplus k_5, x_6 \oplus k_6) \oplus x_2 = k_2 \oplus 1$, p=52/64
- Output of F from S_5 is permuted (by P) into positions 3,8,14,25 of round output, O.
- Input to S_5 for F comes from bits 16,17,18,19,20,21 of round input, I (after expansion).
- Key bits for S_5 are from bits 25,26,27,28,29,30 of the round key, K.
- After renaming input, output and key bits in this way, the constraint becomes $O[3,8,14,25] \oplus I[17] = K[26] \oplus 1$.





Matsui's Per Round Constraints

	SBx	Sbox Equation	w	ht(w)	Prob	Round Equation
A	5	X[2]⊕Y[1,2,3,4]= K[2]⊕1	40_8	40	12/64	X[17]⊕Y[3,8,14,25]=K[26]
B	1	X[2,3,5,6]⊕Y[2]= K[2,3,5,6]⊕1	27_8	20	22/64	X[1,2,4,5]⊕Y[17]=K[2,3,5,6]
C	1	X[4]⊕Y[2]= K[4]⊕1	4_8	4	30/64	X[3]⊕Y[17]=K[4]
D	5	X[2]⊕Y[1,2,3]= K[2]	10_8	20	42/64	X[17]⊕Y[8,14,25]=K[26]
E	5	X[1, 5]⊕Y[1,2,3]= K[1,5]⊕1	22_8	32	16/64	X[16,20]⊕Y[8,14,25]=K[25,29]

Ht(w) is (unnormalized) Hadamard weight. Note that a-d=ht(w) and a+d=2^n so a= (2^n+ht(w))/2 where a= # places linear appx agrees and d= # places linear appx disagrees.

Matsui: Linear Cryptanalysis Method for DES Cipher. Eurocrypt, 98. By the way, Matsui's bit numbering scheme differs from ours.

S-Box constraints

- S-1, Y[4]:

```
w :  000 001 002 003 004 005 006 007 008 009 010 011 012 013 014 015
ht:  000 000 004 004 -04 004 000 008 -08 000 004 -04 004 -12 000 000
w :  016 017 018 019 020 021 022 023 024 025 026 027 028 029 030 031
ht:  -04 -04 -08 -08 -08 000 -12 -04 -04 004 000 -08 008 -08 -04 -04
w :  032 033 034 035 036 037 038 039 040 041 042 043 044 045 046 047
ht:  000 000 -04 -04 -04 004 -08 000 -08 000 -04 020 -12 004 008 008
w :  048 049 050 051 052 053 054 055 056 057 058 059 060 061 062 063
ht:  004 004 008 008 -16 -08 -12 -04 020 -04 000 -08 000 -16 -04 028
```

- S5, Y[1 2 3 4]:

```
w :  000 001 002 003 004 005 006 007 008 009 010 011 012 013 014 015
ht:  000 000 008 008 000 -08 000 008 -08 008 000 000 008 000 008 000
w :  016 017 018 019 020 021 022 023 024 025 026 027 028 029 030 031
ht:  040 -08 000 000 000 -08 000 -08 008 008 000 000 000 -08 000 008
w :  032 033 034 035 036 037 038 039 040 041 042 043 044 045 046 047
ht:  000 000 024 008 000 008 000 008 000 000 -08 -08 000 -08 000 008
w :  048 049 050 051 052 053 054 055 056 057 058 059 060 061 062 063
ht:  -08 -08 000 000 000 -08 000 008 000 000 008 -08 -08 000 -08 000
```

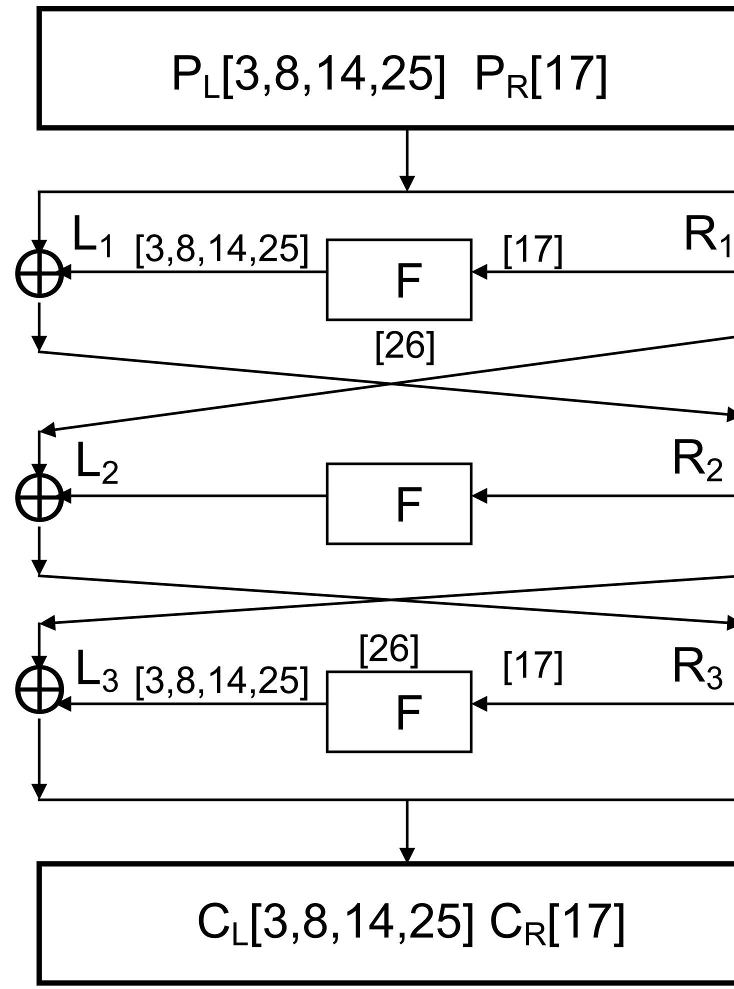
S-Box constraints to round constraints

- S-Box output bit use

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

LC of DES, 3 rounds - 1

- Input at round 1 to activate S_5 constraint is
  - $P_R[17]$.
- Output at round 1 for constraint is
  - $O[3,8,14,25] = P_L[3,8,14,25] \oplus R_2[3,8,14,25]$ which holds with probability 52/64.
- Key bits are $K_1[26]$ and $K_3[26]$.
- First round thus yields
  - $P_L[3,8,14,25] \oplus R_2[3,8,14,25] \oplus P_R[17] = K_1[26] \oplus 1$
- Similarly using the same S_5 relation, round 3 is
  - $C_L[3,8,14,25] \oplus R_2[3,8,14,25] \oplus C_R[17] = K_3[26] \oplus 1$, which holds with probability 52/64.
- Adding we get
  - $P_L[3,8,14,25] \oplus C_L[3,8,14,25] \oplus P_R[17] \oplus C_R[17] = K_1[26] \oplus K_3[26]$.
- This holds with probability
- $p = (52/64)^2 + (12/64)^2 = .6953$

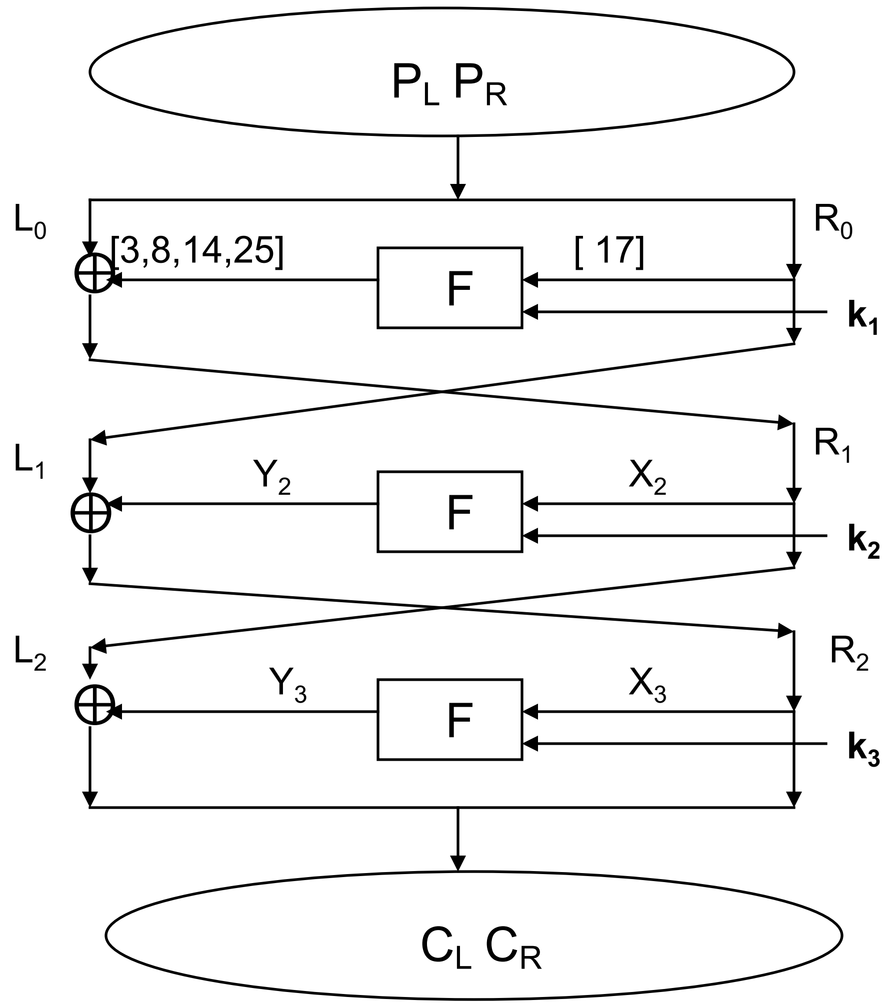


Evaluating experimental outcome

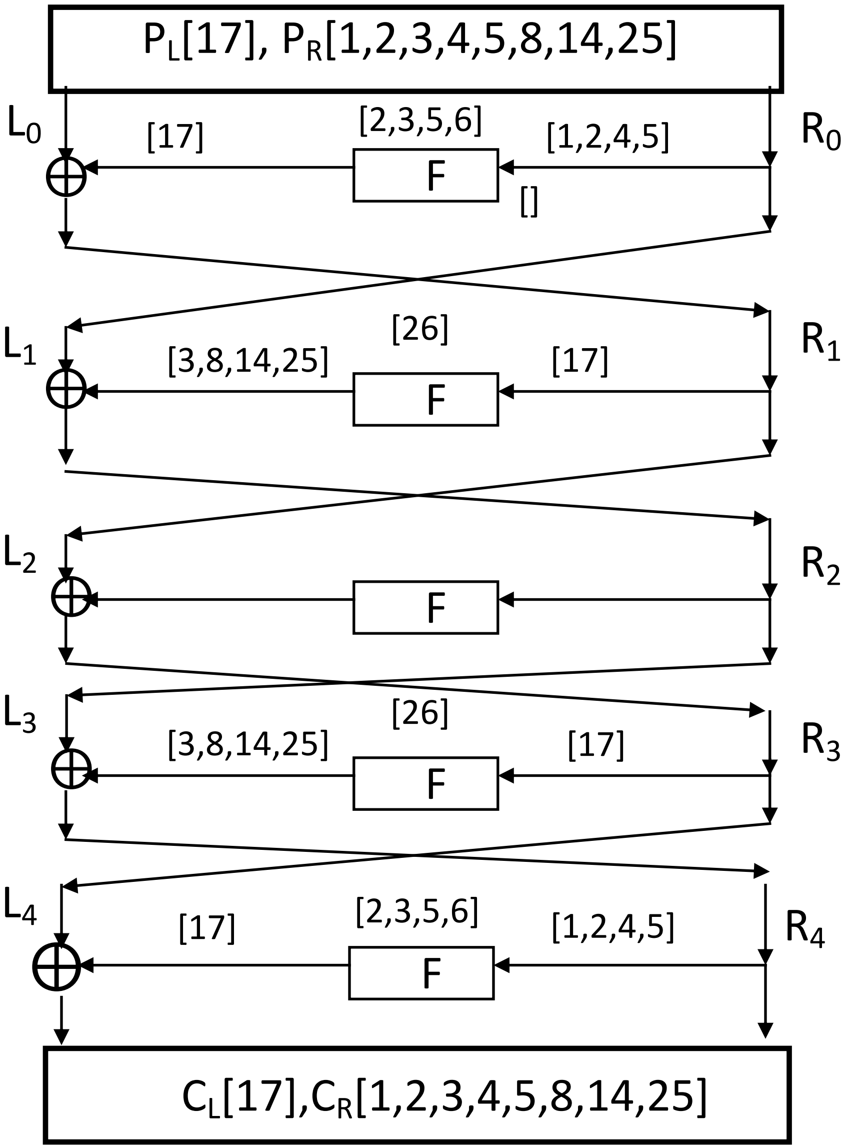
- Suppose an affine constraint $P[j_1, \ldots, j_m] \oplus C[l_1, \ldots, l_{m'}] = K[k_1, \ldots, k_{m''}]$ holds with probability p. Put $\mathbf{x}=(x_1, \ldots, x_n)$ where $x_i= P_i[j_1, \ldots, j_m] \oplus C_i[l_1, \ldots, l_{m'}]$ for the observed sequence (P_i, C_i) of corresponding plain and cipher text. $\mathbf{x}$ is sampled from one of two populations: one with $K[k_1, \ldots, k_{m''}] =0$ and one with $K[k_1, \ldots, k_{m''}] =1$. We assume that the choice of population 1 or population 2 is made at random prior to observation of (P_i, C_i).
- If $\mathbf{x}$ is sampled from the first population (q=0), $Pr(x_i|q=0)= p$ while if $\mathbf{x}$ is sampled from the second population (q=1), $Pr(x_i|q=1)= q= 1-p$.
- Denoting $p_0= Pr(q=0|\mathbf{x})$ and $p_1= Pr(q=1|\mathbf{x})$, from Bayes Theorem, we obtain $p_0= Pr(q=0|\mathbf{x}) = Pr(\mathbf{x}|q=0) \cdot Pr(q=0)/Pr(\mathbf{x})$ while $p_1= Pr(q=1|\mathbf{x}) = Pr(\mathbf{x}|q=1) \cdot Pr(q=1)/Pr(\mathbf{x})$.
- $Pr(q=0)=Pr(q=1)= 1/2$. Suppose we observe a 0's in $\mathbf{x}$ and b 1's (a+b=n), then $Pr(\mathbf{x}|q=0)= {}_nC_a\, p^a q^b$ and similarly, $Pr(\mathbf{x}|q=1)= {}_nC_a\, q^a p^b$, while $Pr(\mathbf{x})= {}_nC_a\, (1/2)^a\, (1/2)^b= 2^{-n}\, {}_nC_a$.
- So $p_0= 2^{n-1}p^aq^b$ and $p_1= 2^{n-1}q^ap^b$.
- Thus, $p_0/p_1= (p/q)^a(q/p)^b$.

LC of DES, 3 rounds - 2

- $P_R[17] \oplus P_L[3,8,14,25] \oplus C_L[3,8,14,25] \oplus C_R[17] = K_1[26] \oplus K_3[26]$
- Recall p= .6953 so q= .3047.
- If we observe a 0's in x and b 1's, the previous result gives:
  $Pr(q=0|\mathbf{x})/Pr(q=1|\mathbf{x}) = (p/q)^a(q/p)^b$.
- Equivalently, if a>b,
  $Pr(q=0|\mathbf{x})/Pr(q=1|\mathbf{x}) = (p/q)^{a-b} \cong (7/3)^{a-b}$.
- So, if, for example, a-b=5, $p_0 \cong .99$.



LC of DES, 5 rounds - 1

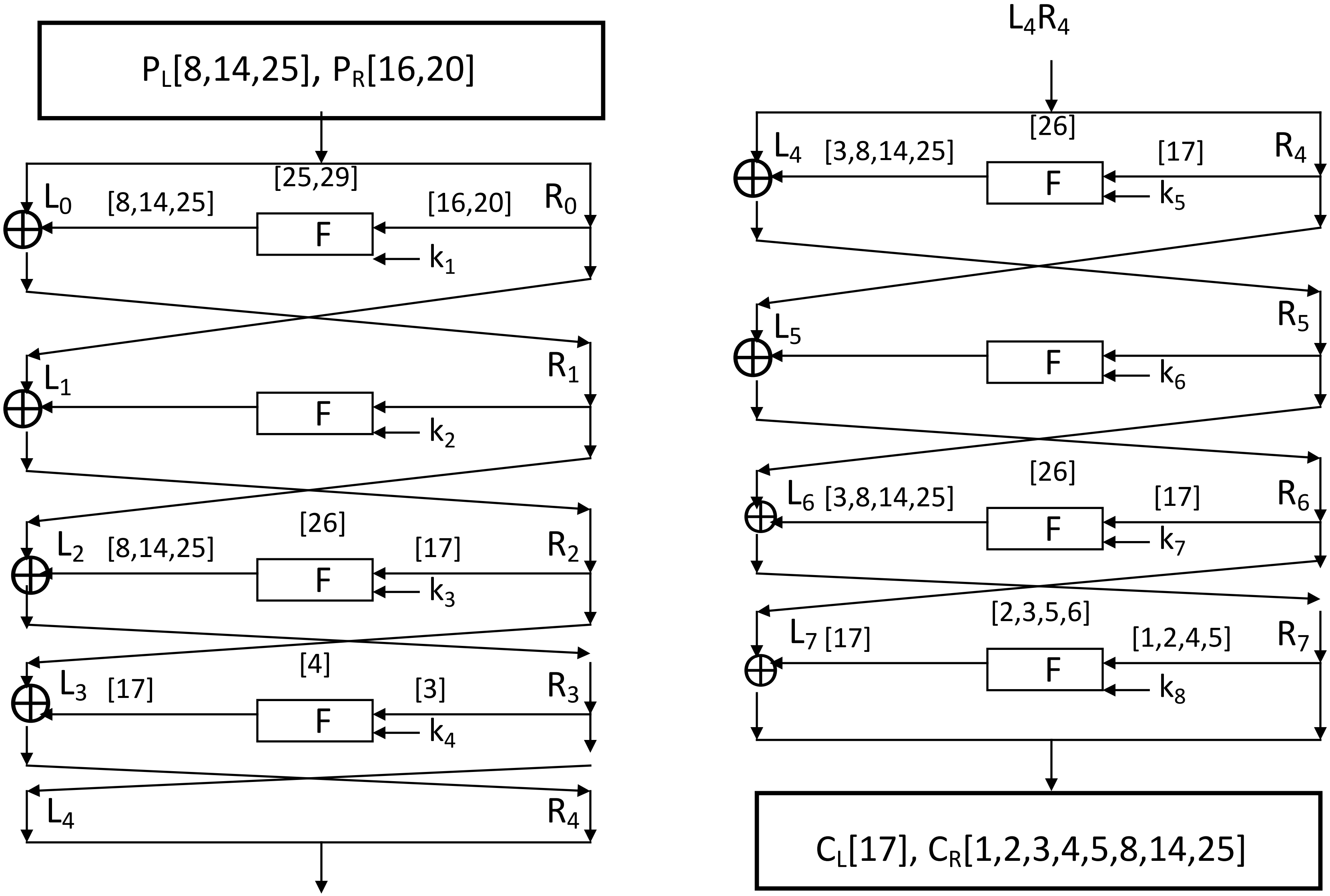


LC of DES, 5 rounds - 2

1. $P_L[17] \oplus R_1[17] = K_1[2,3,5,6] \oplus P_R[1,2,4,5] \oplus 1$ .....(Eq B)
2. $P_R[3,8,14,25] \oplus R_2[3,8,14,25] = K_2[26] \oplus R_1[17] \oplus 1$ .....(Eq A)
3. $R_2[3,8,14,25] \oplus C_R[3,8,14,25] = K_4[26] \oplus C_R[17] \oplus 1$ .....(Eq A)
4. $C_L[17] \oplus R_3[17] = K_5[2,3,5,6] \oplus C_R[1,2,4,5] \oplus 1$ .....(Eq B)

- Adding yields:
  $P_L[17] \oplus P_R[1,2,3,4,5,8,14,25] \oplus C_L[17] \oplus C_R[1,2,3,4,5,8,14,25] =$
  $K_1[2,3,5,6] \oplus K_2[26] \oplus K_4[26] \oplus K_5[2,3,5,6]$
- This holds with probability:
  $p = p_B^2 p_A^2 + p_B^2 q_A^2 + p_A^2 q_B^2 + 4(q_A p_B q_B p_A) + q_B^2 q_A^2 \cong .519 = .5 + 1.22\text{x}2^{-6}$,
  where $q_i = 1 - p_i$. $p/q = 1.07927$..
- Suppose we decide, based on an excess (e), of LHS values. Odds of right answer is $r = (p/q)^e$. For example, if e= 64, r $\cong$131.92.

LC of DES, 8 rounds - 1



LC of DES, 8 rounds - 2

1. $P_L[8,14,25] \oplus R_1[8,14,25] = K_1[25, 29] \oplus P_R[16,20] \oplus 1$ ......(Eq E)
2. $R_1[8,14,25] \oplus R_3[8,14,25] = K_3[26] \oplus R_2[17]$ ......(Eq D)
3. $R_3[3,8,14,25] \oplus R_5[3,8,14,25] = K_5[26] \oplus R_4[17]$ ......(Eq A)
4. $R_2[17] \oplus R_4[17] = K_4[4] \oplus R_3[3] \oplus 1$ ......(Eq C)
5. $R_5[3,8,14,25] \oplus R_7[3,8,14,25] = K_7[26] \oplus R_6[17]$ ......(Eq A)
6. $C_L[17] \oplus R_6[17] = K_8[2,3,5,6] \oplus C_R[1,2,4,5] \oplus 1$ ......(Eq B)

- $P_L[8,14,25] \oplus P_R[16,20] \oplus C_R[1,2,3,4,5,8,14,25] \oplus C_L[17] =$
  $K_1[25,29] \oplus K_3[26] \oplus K_4[4] \oplus K_5[26] \oplus K_7[26] \oplus K_8[2,4,5,6] \oplus 1.$
- This holds with probability: $p \cong 0.500596 = .50 + 1.22 \times 2^{-11}$.

15 Round Linear Approximation

Pattern: E-DCA-ACD-DCA-A. Note $L_i = R_{i-1}$, $L_i \oplus R_{i+1} = L_i \oplus L_{i+2}$.

1	$P_L[8,14,25] \oplus R_2[8,14,25] \oplus P_R[16,20]$	$= K_1[23,25]$
3	$L_3[8,14,25] \oplus R_4[8,14,25] \oplus R_3[17]$	$= K_3[26]$
4	$L_4[17] \oplus R_5[17] \oplus R_4[3]$	$= K_4[4]$
5	$L_5[3,8,14,25] \oplus R_6[3,8,14,25] \oplus R_5[17]$	$= K_5[26]$
7	$L_7[3,8,14,25] \oplus R_8[3,8,14,25] \oplus R_7[17]$	$= K_7[26]$
8	$L_8[17] \oplus R_9[17] \oplus R_8[3]$	$= K_8[4]$
9	$L_9[8,14,25] \oplus R_{10}[8,14,25] \oplus R_9[17]$	$= K_9[26]$
11	$L_{11}[8,14,25] \oplus R_{12}[8,14,25] \oplus R_{11}[17]$	$= K_{11}[26]$
12	$L_{12}[17] \oplus R_{13}[17] \oplus R_{12}[3]$	$= K_{12}[4]$
13	$L_{13}[3,8,14,25] \oplus R_{14}[3,8,14,25] \oplus R_{13}[17]$	$= K_{13}[26]$
15	$L_{15}[3,8,14,25] \oplus C_L[3,8,14,25] \oplus C_R[17]$	$= K_{15}[26]$

15 Round Linear Approximation

Adding and canceling:

- $P_L[8,14,25] \oplus P_R[16,20] \oplus C_L[3,8,14,25] \oplus C_R[17] =$
  $K_1[23,25] \oplus K_3[26] \oplus K_4[4] \oplus K_5[26] \oplus K_7[26] \oplus K_8[4]$
  $\oplus K_9[26] \oplus K_{11}[26] \oplus K_{12}[4] \oplus K_{13}[26] \oplus K_{15}[26]$

which holds (Piling-up Lemma) with the indicated probability.

Full Linear Attack on DES

- Linear cryptanalysis can be accomplished with ~2^{43} known plaintexts, using a more sophisticated estimation 14 round approximation
  - For each 48 bit last round sub-key, decrypt cipher-text backwards across last round for all sample cipher-texts
  - Increment count for all sub-keys whose linear expression holds true to the penultimate round
  - This is done for the first and last round yielding 13 key bits each (total: 26)
- Here they are:

$$P_R[8,14,25] \oplus C_L[3,8,14,25] \oplus C_R[17] = K_1[26] \oplus K_3[4] \oplus K_4[26] \oplus K_6[26] \oplus K_7[4] \oplus K_8[26] \oplus K_{10}[26] \oplus K_{11}[4] \oplus K_{12}[26] \oplus K_{14}[26]$$

with probability **½ -1.19x2^{-21}**

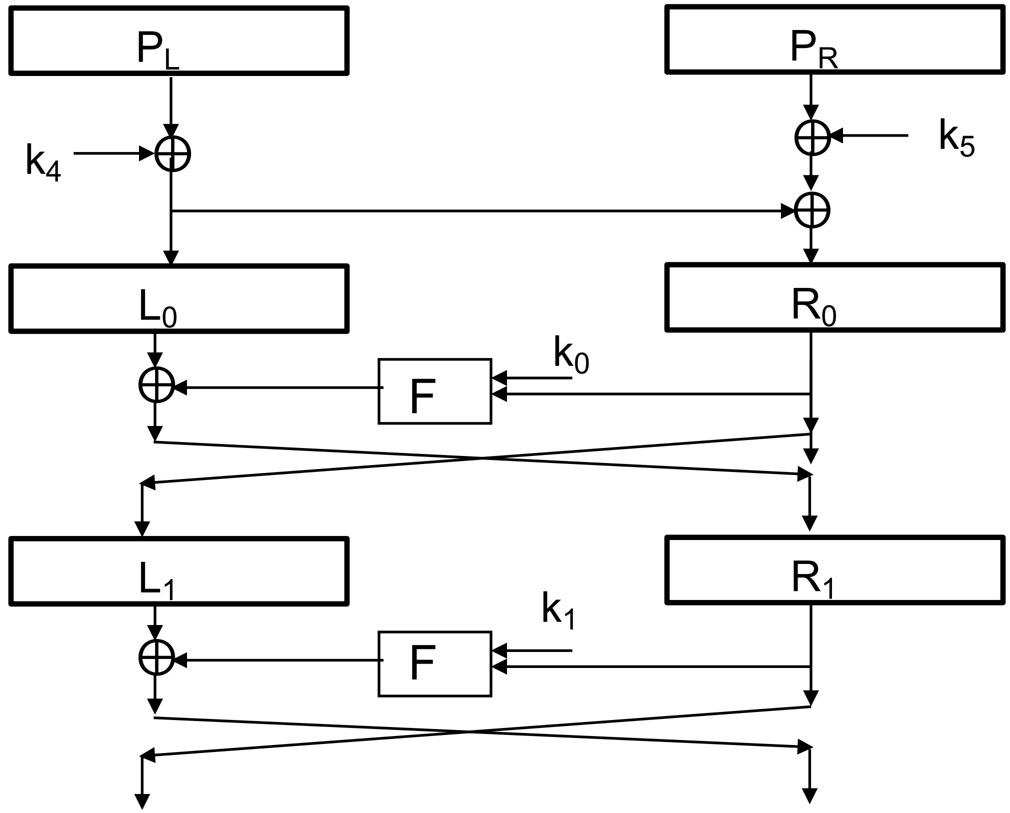
$$C_R[8,14,25] \oplus P_L[3,8,14,25] \oplus P_R[17] = K_{13}[26] \oplus K_{12}[24] \oplus K_{11}[26] \oplus K_9[26] \oplus K_8[24] \oplus K_7[26] \oplus K_5[26] \oplus K_4[4] \oplus K_3[26] \oplus K_1[26]$$

with probability **½ -1.19x2^{-21}**

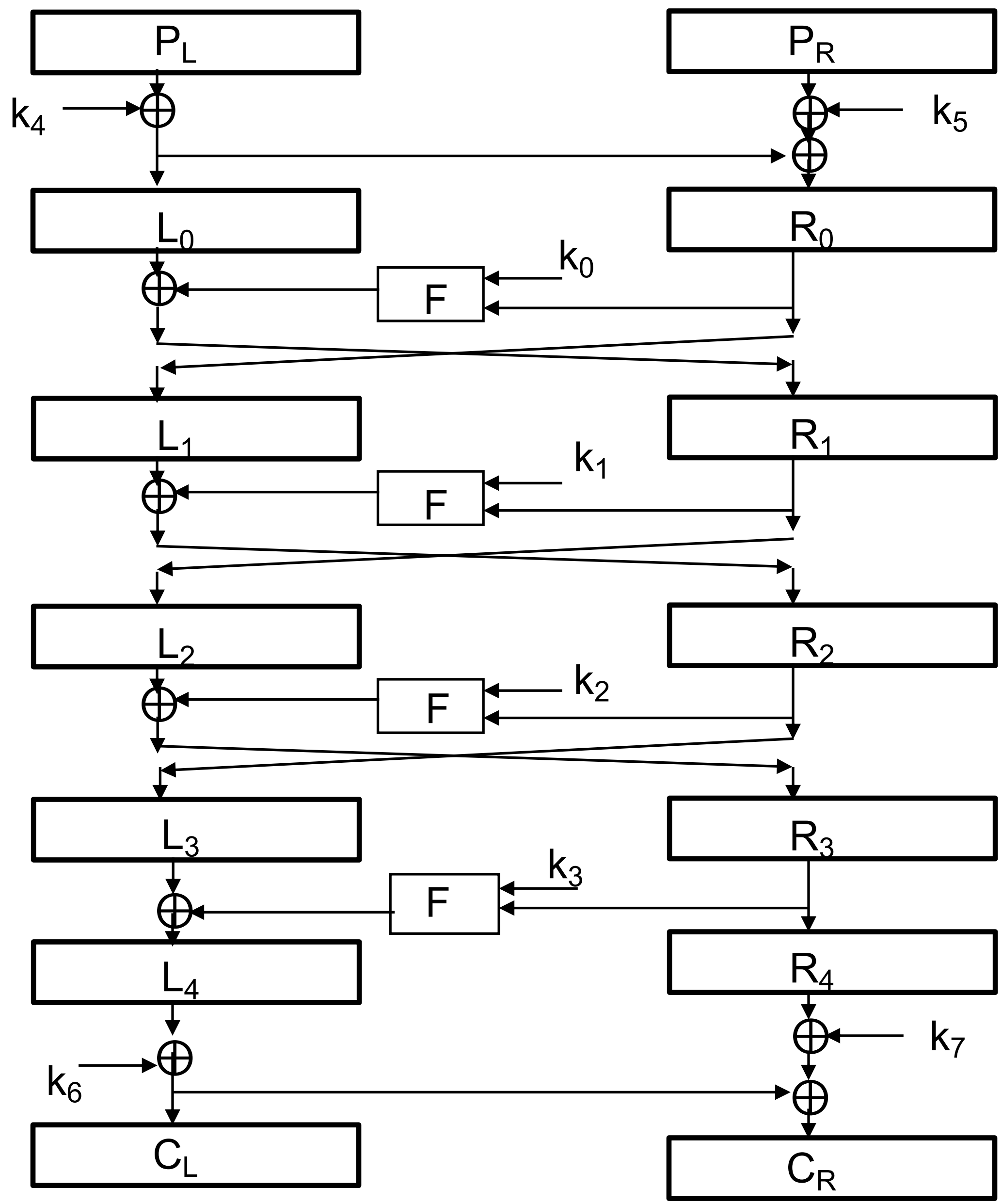
FEAL
(A fortunate mistake)

FEAL-4

- Four round Feistel cipher with a 64-bit block and 64-bit key
- Plaintext: P, Cipher-text: C
- Round function: F
- 32-bit sub-keys: $K_0, K_1, \ldots, K_7$
- Most important failed cipher: showed the power of differential cryptanalysis and linear cryptanalysis



Original FEAL-4



FEAL-4 Round Function

- $G_0(a,b) = (a+b \pmod{256}) \lll 2$
- $G_1(a,b) = (a+b+1 \pmod{256}) \lll 2$
  where “<<<” is left cyclic shift (rotation)
- $F(x_0,x_1,x_2,x_3) = (y_0,y_1,y_2,y_3)$ where
  1. $y_1 = G_1(x_0 \oplus x_1,\ x_2 \oplus x_3)$
  2. $y_0 = G_0(x_0,\ y_1)$
  3. $y_2 = G_0(y_1,\ x_2 \oplus x_3)$
  4. $y_3 = G_1(y_2,\ x_3)$

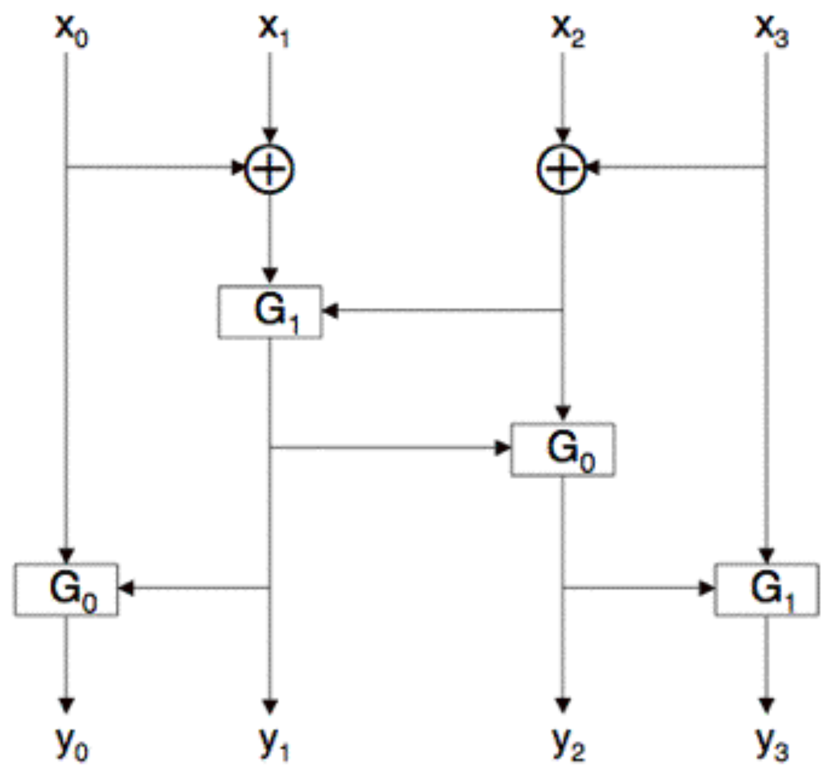
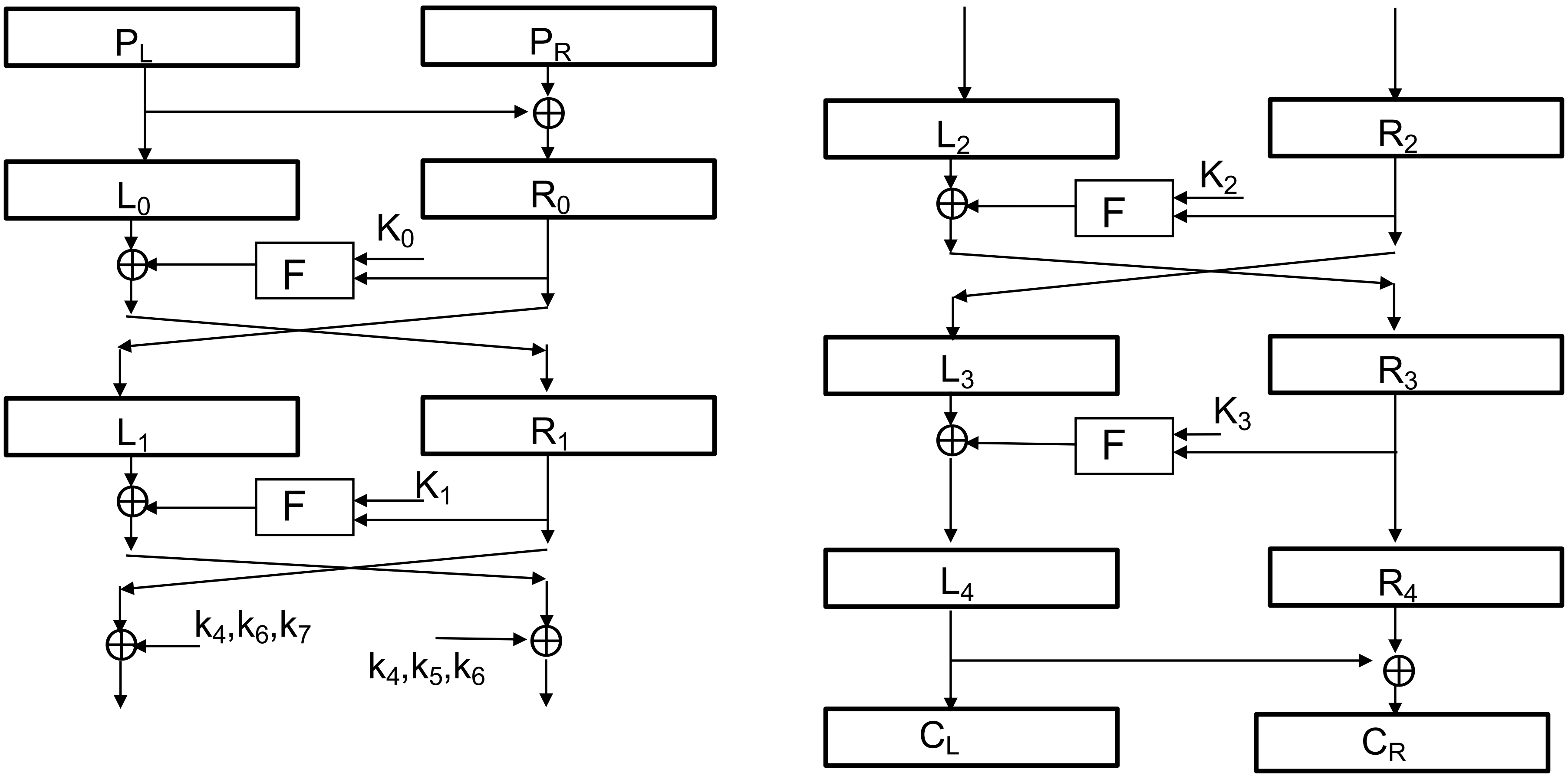


Diagram from Mark Stamp

FEAL-4 Key Schedule

- $F_K(a_0||a_1||a_2||a_3, b_0||b_1||b_2||b_3)= c_0||c_1||c_2||c_3$ by
  - $d_1= a_0 \oplus a_1$
  - $d_2= a_2 \oplus a_3$
  - $c_1= G_1(d_1, a_2 \oplus b_0)$
  - $c_2= G_0(d_2, c_1 \oplus b_1)$
  - $c_0= G_0(a_0, c_1 \oplus b_2)$
  - $c_3= G_1(a_3, c_2 \oplus b_3)$
- $k_{-2}= 0$
- $k_{-1}= k_L$
- $k_0= k_R$
- $k_i= f_K(k_{i-2},k_{i-1} \oplus k_{i-3})$

Refactored FEAL-4



$K_0 = k_0+k_4+k_5$
$K_1 = k_1+k_4$

$K_3 = k_3+k_6+k_7$
$K_2 = k_2+k_6$

Refactored FEAL-4 Equations

- $K_0 = k_0+k_4+k_5$, $K_1 = k_1+k_4$, $K_2=k_2+k_6$, $K_3=k_3+k_6+k_7$
- $K_4 = k_4+k_5+k_6$, $K_5 = k_4+k_6+k_7$
- $L_1 = P_L+P_R$, $R_1 = P_L+f(P_L+P_R+K_0)$
- $L_2 = R_1+K_5$, $R_2 = L_1+K_4+f(R_1+K_1)$
- $L_3 = R_2$, $R_3 = L_2+f(R_2+K_2)$
- $C_L = L_3+f(R_3+K_3)$, $C_R = C_L+R_3$
- Substituting,
  - $C_L = P_L+P_R+ k_4+k_5+k_6+f(P_L+k_4+k_1+f(P_L+P_R+k_4+k_5+k_0))$
  - $C_R = C_L+$

    $(P_L+k_4)+k_6+k_7+f(P_L+P_R+k_4+k_5+k_0)+$

    $f(P_L+P_R+k_4+k_5+k_2+f(P_L+k_4+k_1+f(P_L+P_R+k_4+k_5+k_0)))$

FEAL-4 Basic Differential Attack

- If $A_0 \oplus A_1 = 0$ then $F(A_0) = F(A_1)$, p=1.
- If $A_0 \oplus A_1$= 0x80800000 then $F(A_0) \oplus F(A_1)$= 0x02000000, p=1
- Choose (P_0, P_1):
- $P_0 \oplus P_1$=0x8080000080800000
- $P' = P_0 \oplus P_1$, $C' = C_0 \oplus C_1$
- L'=0x02000000$\oplus Z'$, Y'=0x80800000 $\oplus X'$
- For $C = (L,R)$ we have $Y = L \oplus R$
- Solve for sub-key K_3: Z'= 0x02000000$\oplus L'$
- Compute $Y_0 = L_0 \oplus R_0$, $Y_1 = L_1 \oplus R_1$
- Guess K_3 and compute guessed Z_0, Z_1
  - Note: $Z_i = F(Y_i \oplus K_3)$
- Compare true Z' to guessed Z'

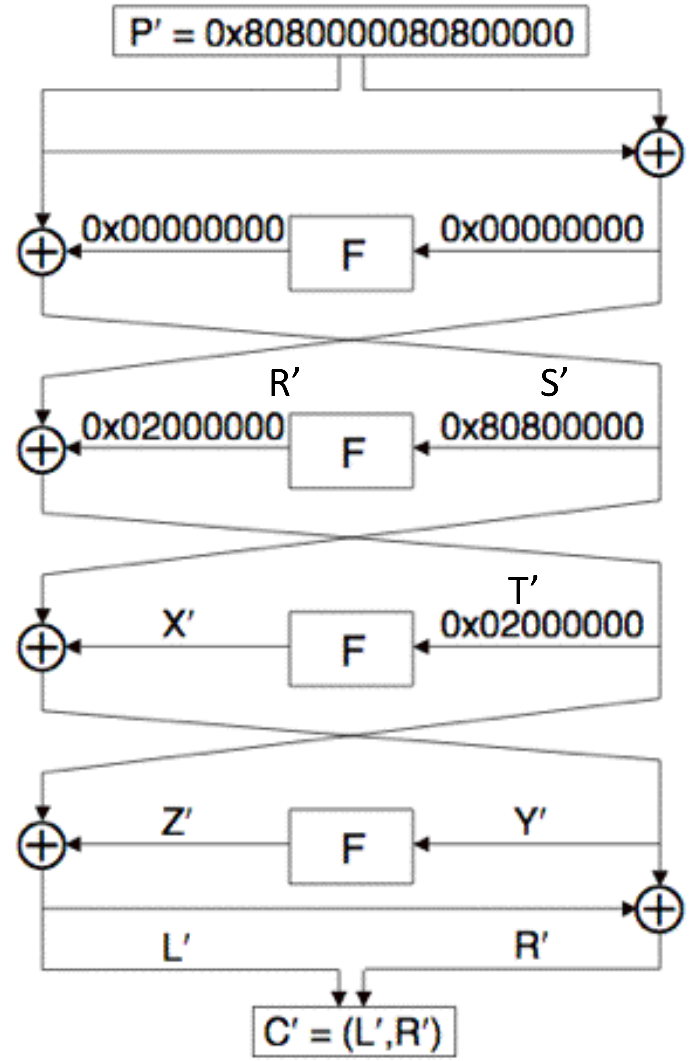


Diagram from Mark Stamp

FEAL-4 Improved Differential Attack

- Using 4 chosen plaintext pairs
  - Work is of order 2^{32}
  - Expect one K_3 to survive
- Can reduce work to about 2^{17}
  - For 32-bit word $A=(a_0,a_1,a_2,a_3)$, define $M(A) = (z, a_0 \oplus a_1, a_2 \oplus a_3, z)$, where z is all-zero byte
  - For all possible $A=(z, a_0, a_1, z)$, compute $Q_0= F(M(Y_0) \oplus A)$ and $Q_1= F(M(Y_1) \oplus A)$
  - Can be used to find 16 bits of K_3
- When $A = M(K_3)$, we have $\langle Q_0 \oplus Q_1 \rangle_{8\ldots23} = \langle Z' \rangle_{8\ldots23}$ where $\langle X \rangle_{i\ldots j}$ is bits i thru j of X. Can recover K_3 with about 2^{17} work
- Once K_3 is known, can successively recover K_2,K_1,K_0 and finally K_4,K_5
- Second characteristic: 0xa200 8000   0x2280 8000

FEAL-4 Differential Attack

- Primary for K_3

```
// Characteristic is 0x8080000080800000
P0 = random 64-bit value
P1 = P0 ⊕ 0x8080000080800000
// Given corresponding ciphertexts
// C0 = (L0, R0) and C1 = (L1, R1)
Y0 = L0 ⊕ R0
Y1 = L1 ⊕ R1
L' = L0 ⊕ L1
Z' = L' ⊕ 0x02000000
for (a0, a1) = (0x00, 0x00) to (0xff, 0xff)
    Q0 = F(M(Y0) ⊕ (0x00, a0, a1, 0x00))
    Q1 = F(M(Y1) ⊕ (0x00, a0, a1, 0x00))
    if ⟨Q0 ⊕ Q1⟩8...23 == ⟨Z'⟩8...23 then
        Save (a0, a1)
    end if
next (a0, a1)
```

- Secondary for K_3

```
// P0, P1, C0, C1, Y0, Y1, Z' as in primary
// Given list of saved (a0, a1) from primary
for each primary survivor (a0, a1)
    for (c0, c1) = (0x00, 0x00) to (0xff, 0xff)
        D = (c0, a0 ⊕ c0, a1 ⊕ c1, c1)
        Z̃0 = F(Y0 ⊕ D)
        Z̃1 = F(Y1 ⊕ D)
        if Z̃0 ⊕ Z̃1 == Z' then
            Save D // candidate subkey K3
        end if
    next (c0, c1)
next (a0, a1)
```

Slide adapted from Mark Stamp

- Assuming only one chosen plaintext pair

FEAL-4 Linear Attack

- Now we'll use linear cryptanalysis to break Feal-4.
- We will actually break the equivalent refactored FEAL-4 in the end.
- Notation: let Y=F(X). We use X[i,j] to denote $X[i] \oplus X[j]$
- Using the definition of F, we will see (next slide) that the following linear constraints hold with probability 1. These are called the F-constraints.
  1. Y[13] = X[7, 15, 23, 31] + 1
  2. Y[5, 15] = X[7]
  3. Y[15, 21] = X[23, 31]
  4. Y[23, 29] = X[31] + 1

FEAL-4 Constraint Derivation

Y=F(X)

- Y=(y_0, y_1, y_2, y_3)
- X=(x_0, x_1, x_2, x_3)

- $(a \oplus b)[7] = (a+b(\text{mod } 256))[7]$, so
- $G_0(a,b)[5] = (a \oplus b)[7]$, similarly, $G_1(a,b)[5] = (a \oplus b \oplus 1)[7]$
- $y_1 = G_1(x_0 \oplus x_1, x_2 \oplus x_3) \rightarrow Y[13] = y_1[5] = x_0[7] \oplus x_1[7] \oplus x_2[7] \oplus x_3[7] \oplus 1 = X[7,15,23,31] \oplus 1$
- $y_0 = G_0(x_0, y_1) \rightarrow Y[5] = y_0[5] = y_1[7] \oplus x_0[7] = Y[15] \oplus X[7]$
- $y_2 = G_0(y_1, x_2 \oplus x_3) \rightarrow$
  $Y[21] = y_2[5] = y_1[7] \oplus x_2[7] \oplus x_3[7] = Y[15] \oplus X[23,31]$
- $y_3 = G_1(y_2, x_3) \rightarrow$
  $Y[29] = y_3[5] = y_2[7] \oplus x_3[7] \oplus 1 = Y[23] \oplus X[31] \oplus 1$

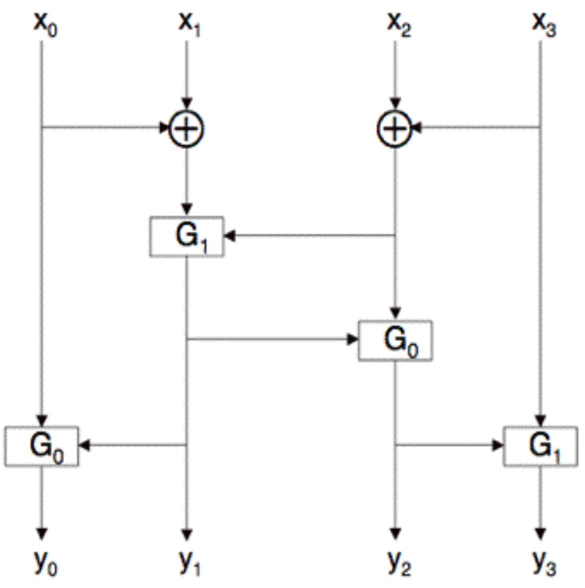


Diagram from Mark Stamp

FEAL-4 Linear Attack Equations

- Adapting the F constraint equations for each round, we get:
  - $Y_0= F(R_0 \oplus k_0)$, $R_1= L_0 \oplus Y_0$, $L_1= R_0$
  - $Y_1= F(R_1 \oplus k_1)$, $R_2= L_1 \oplus Y_1$, $L_2= R_1$
  - $Y_2= F(R_2 \oplus k_2)$, $R_3= L_2 \oplus Y_2$, $L_3= R_2$
  - $Y_3= F(R_3 \oplus k_3)$
- Looking at the original FEAL-4 diagram (using "+" instead of "$\oplus$"), we get
  - $L_4 = R_2+Y_3$ and $R_2 = R_0+Y_1$, "adding" these gives
  - $L_4+R_0 = Y_1+Y_3$, or
  - $L_4+R_0 = F(R_1+k_1)+F(R_4+k_3)$
- Since $R_1 = L_0+F(R_0+k_0)$, we have finally
  - $L_4+R_0 = F(R_4+k_3)+F(L_0+F(R_0+k_0)+k_1)$
- Note that $L_0= P_L+k_4$, $R_0= P_L+P_R+k_4+k_5$, $L_4= C_L+k_6$ and $R_4= C_L+C_R+k_6+k_7$, so we get
  - $C_L+P_L+P_R+k_4+k_5 = F(P_L+k_4+F(C_L+C_R+k_4+k_5+k_0)+k_1) + F(C_L+C_R+k_6+k_7+k_3)$

FEAL-4 Linear Attack using refactored FEAL-4

- Now we can explain why we refactored FEAL-4.
- If we knew L_0, R_0, L_4, R_4, we could mount a standard linear attack on FEAL-4. Because of the "whitening" keys, k_4, k_5, k_6, k_7, the first and last inputs to F are unknown.
- However, if we use the round key $K_0 = k_0+k_4+k_5$, for the first round key and $K_3 = k_3+k_6+k_7$ for the last round key, we can express the inputs to F in the first and last rounds in terms of P_L, P_R, C_L, C_R, K_0, and K_3. This allows us to find K_0, and K_1.
- We can then use K_0 and K_3 to find K_2 and K_3
- Knowing K_0, K_3, K_2, and K_3 allows us to compute the intermediate keys $k_4+k_6+k_6$ and $k_4+k_6+k_7$ for refactored FEAL4.

FEAL-4 Linear Attack

- $(C_L+P_L+P_R)+k_4+k_5= F(P_L+F(R_0+K_0)+K_1)+F(C_L+C_R+K_3)$
- From F-constraint 4,
  - $F(C_L+C_R+K_3)[23,29]= (C_L+C_R+K_3)[31]+1$
  - $F(P_L+F(R_0+K_0)+K_1)[23,29]= (P_L+ F(P_L+ P_R+ K_0)+K_1))[31]+1$
- Rearranging, we get "Equation A:"

$$K_3[31]+K_1[31]+(k_4+k_5)[23,29] = (C_L+P_L+P_R)[23,29]+P_L[31] +(C_L+C_R)[31]+F(P_L+P_R+K_0)[31]$$

- The attack consists of guessing K_0 and computing

$$h_A(P,C)= (C_L+P_L+P_R)[23,29]+P_L[31]+(C_L+C_R)[31]+F(P_L+P_R+K_0)[31]$$

  for a number of corresponding (P_L,P_R), (C_L,C_R).
- If the guessed K_0 is right, $h_A(P,C)$ will have the same value for each corresponding pair of plain-text and cipher-text.

Computing the Final Equations - A

- Remember, Equation A gave us
  $h_A(P,C)= (C_L+P_L+P_R)[23,29]+P_L[31]+(C_L+C_R)[31]+F(P_L+P_R+K_0)[31]$
- It was derived from
  - $(L_4+R_0)[23,29] = Y_1[23,29] + Y_3[23, 29]$.
  - $Y_1[23, 29] = F(R_1+k_1)[31]+1$, and $R_1[31] = L_0[31]+F(R_0+K_0)[31]$, giving
  - $Y_1[23, 29] = (L_0[31] + F(R_0+K_0)+k_1)[31] +1$
  - $Y_3[23,29] = (R_4+K_3)[31]+1$
- Combining, we got
  - $h_A(P,C) = f(K_i) = (L_4+R_0)[23,29] + (R_4+L_0+F(R_0+K_0))[31]$

Computing the Final Equations - B

- Analogously,
  - $(L_4+R_0)[13] = Y_1[13] + Y_3[13]$
  - $Y_1[13] = F(R_1+K_2)[13]+1 = (R_1+K_2)[7, 15, 23, 31]+1$
  - $R_1[7, 15, 23, 31] = (L_0[7, 15, 23, 31] + F(R_0+K_0))[7, 15, 23, 31]$, so
  - $Y_1[13] = (L_0[7, 15, 23, 31] + F(R_0+K_0))[7, 15, 23, 31] + K_2[7, 15, 23, 31]+1$
  - $Y_3[13] = F(R_4+K_3)[13]+1 = (R_4+K_3)\ [7, 15, 23, 31]+1$
  - $(L_4+R_0)[13] = (L_0[7, 15, 23, 31] + F(R_0+K_0))[7, 15, 23, 31]+$
    $K_2[7, 15, 23, 31]+(R_4+K_3)\ [7, 15, 23, 31]$
- This yields

– $h_B(P,C)= (C_L+P_L+P_R)[13]+(P_L+(C_L+C_R)+F(P_L+P_R+K_0))[7, 15, 23, 31]$

Computing the Final Equations - C

- Similarly
  - $(L_4+R_0)[5, 15] = Y_1[5, 15] + Y_3[5, 15]$
  - $Y_1[5, 15] = F(R_1+K_2)[5, 15] +1 = (R_1+K_2)[7]$
  - $R_1[7] = (L_0[7] + F(R_0+K_0))[7]$, so
  - $Y_1[5, 15] = (L_0[7]+F(R_0+K_0))[7] + K_2[7]$
  - $Y_3[5, 15] = F(R_4+K_3)[5, 15] = (R_4+K_3) [7]$
  - $(L_4+R_0)[5, 15] = (L_0[7]+F(R_0+K_0))[7]+K_2[7]+(R_4+K_3) [7]$

- This gives
  - $h_C(P,C)= (C_L+P_L+P_R)[5, 15]+(P_L+(C_L+C_R)+F(P_L+P_R+K_0))[7]$

Computing the Final Equations - D

- From $Y[15, 21] = X[23, 31]$
  - $(L_4+R_0)[15, 21] = Y_1[15, 21] + Y_3[15, 21]$
  - $Y_1[15, 21] = F(R_1+K_2)[15, 21]+1 = (R_1+K_2)[23, 31]$
  - $R_1[23, 31] = (L_0+ F(R_0+K_0))[23, 31]$, so
  - $Y_1[15, 21] = (L_0 + F(R_0+K_0))[23, 31] + K_2[23, 31]$
  - $Y_3[15, 21] = F(R_4+K_3)[15, 21] = (R_4+K_3) [23, 31]$
  - This gives
  - $(L_4+R_0)[15, 21] = (L_0+ F(R_0+K_0))[23, 31] + K_2[23, 31] + (R_4+K_3) [23, 31]$
- This gives
  - $h_D(P,C)= (C_L+P_L+P_R)[15, 21]+(P_L+(C_L+C_R)+F(P_L+P_R+K_0))[23, 31]$

Computing the Final Equations - E

- We will use one more constraint. Adding all four round constraints, we get
  - $(L_4+R_0)[5,13,21] = Y_1[5,13,21]+Y_3[5,13,21] = F(R_1+K_1)\ [5,13,21] + F(R_4+K_3)\ [5,13,21]$
  - $F(R_4+K_3)\ [5,13,21] = (R_4+K_3)\ [15]+1$ and since $R_1 = L_0+F(L_0+Y_0+K_0)$,
  - $F(R_1+K_1)\ [5,13,21] = F(L_0+F(L_0+Y_0+K_0)+K_1)= (L_0+F(L_0+Y_0+K_0)+K_1)[15]+1$
- This gives
  - $h_E(P,C)= (C_L+P_L+P_R)[5,13,21]+P_L[15]+(C_L+C_R)[15]+F(P_L+P_R+K_0)[15]$
- Putting $P_L+P_R+K_0 = (x_0, x_1, x_2, x_3)$, we note that $F(P_L+P_R+K_0)[15]$ is only dependent on $(x_0 \oplus x_1, x_2 \oplus x_3)$

- Similar relations hold looking at FEAL-4 as a decryption algorithm. These constraints are summarized in the next two slides.

FEAL-4 Summary of invariants

Name	First Round Equation	Key bits affecting outcome
A	$h_A(P,C)=(C_L+P_L+P_R)[23,29]+P_L[31]+$ $(C_L+C_R)[31]+F(P_L+P_R+K_0)[31]$	
B	$h_B(P,C)= (C_L+P_L+P_R)[13]+$ $(P_L+(C_L+C_R)+F(P_L+P_R+K_0))[7, 15, 23, 31]$	
C	$h_C(P,C)= (C_L+P_L+P_R)[5, 15]+$ $(P_L+(C_L+C_R)+F(P_L+P_R+K_0))[7]$	
D	$h_D(P,C)= (C_L+P_L+P_R)[15, 21]+$ $(P_L+(C_L+C_R)+F(P_L+P_R+K_0))[23, 31]$	
E	$h_E(P,C)=(C_L+P_L+P_R)[5,13,21]+P_L[15]+$ $(C_L+C_R)[15]+F(P_L+P_R+K_0)[15]$	9,…,15; 17,…,23

FEAL-4 Summary of invariants

Name	Fourth Round Equation	Key bits affecting outcome
A	$h_A'(P,C)=(P_L+C_L+C_R)[23,29]+(C_L+(P_L+P_R))[31]+F(C_L+C_R+K_3)[31]$	
B	$h_B'(P,C)=(P_L+C_L+C_R)[13]+(C_L+(P_L+P_R))[7, 15, 23, 31]+F(C_L+C_R+K_3))[7, 15, 23, 31]$	
C	$h_C'(P,C)=(P_L+C_L+C_R)[5, 15]+(C_L+(P_L+P_R)[7]+F(C_L+C_R+K_3))[7]$	
D	$h_D'(P,C)=(P_L+C_L+C_R)[15, 21]+(C_L+(P_L+P_R))+F(C_L+C_R+K_3))[23, 31]$	
E	$h_E'(P,C)=(P_L+C_L+C_R)[5,13,21]+(C_L+(P_L+P_R))[15]+F(C_L+C_R+K_3)[15]$	9,…,15; 17,…,23

Strategy for FEAL-4 Linear Attack

- We use $h_E(P,C)$ to estimate the xor of the first two and last two bytes of K_0 and R_0 to estimate the xor of the two halves of K_0 (see slide 47) then we use $h_{A,}$ …, h_D to find K_0.
- Next, we use $h_E'(P,C)$ to estimate the xor of the first two and last two bytes of K_3 and R_4 then we use $h_A'{}_{,}$ …, h_D' to find K_3.
- Next compute candidate K_1's; for successful candidates, compute
  - $k_4+k_5+k_6 = F(P_L+F(P_L+P_R+K_0)+K_1) + F(C_L+C_R+K_3) + (P_L+P_R+C_L)$
- Analogously, for round 3, compute candidate K_2's; for successful, candidates compute
  - $k_4+k_6+k_7 = F(C_L+F(C_L+C_R+K_3)+K_2) + F(P_L+P_R+K_0) +(C_L+C_R+P_L)$
- The "vanilla" attack of guessing K_0, also works but our modified attack is much faster --- on the order of 2^{16}, which is peanuts.

FEAL-4 Linear Attack in gory detail

- Remember $k_4+k_5+k_6 = F(P_L+F(P_L+P_R+K_0)+K_1)+F(C_L+C_R+K_3)+(P_L+P_R+C_L)$
  - If $X= P_L+F(P_L+P_R+K_0)$, $Y= F(C_L+C_R+K_3)$ and $Z= P_L+P_R+C_L$. Note that X, Y and Z are known once we know K_0 and K_3.
  - $k_4+k_5+k_6= Z+Y+F(X+K_1)$.
  - Guess $K_1[0,1]$, $K_1[2,3]$ and compute $X[0,1]$, $X[2,3]$, we can test the guess by checking that $(Z+Y+F(X+K_1))[8,9,\ldots 15]$ remains constant over a set of plain/cipher pairs. This requires 2^{16} time.
  - Next, guess $K_1[0]$, $K_1[3]$ and again confirm the guess by checking that $(Z+Y+F(X+K_1))$ is constant.
  - Now that we know K_1, can compute $k_4+k_5+k_6= Z+Y+F(X+K_1)$.
- By looking at the corresponding FEAL-4 decryption, we get K_2 in exactly the same way as well as the other invariants r intermediate key, $k_4+k_6+k_7$.
- Finally, we check the complete set of guesses to confirm all the sub-keys are right.
- The entire automated attack runs in about 1 second on my MAC using 128 pairs of corresponding plain and cipher text.

Automated attack

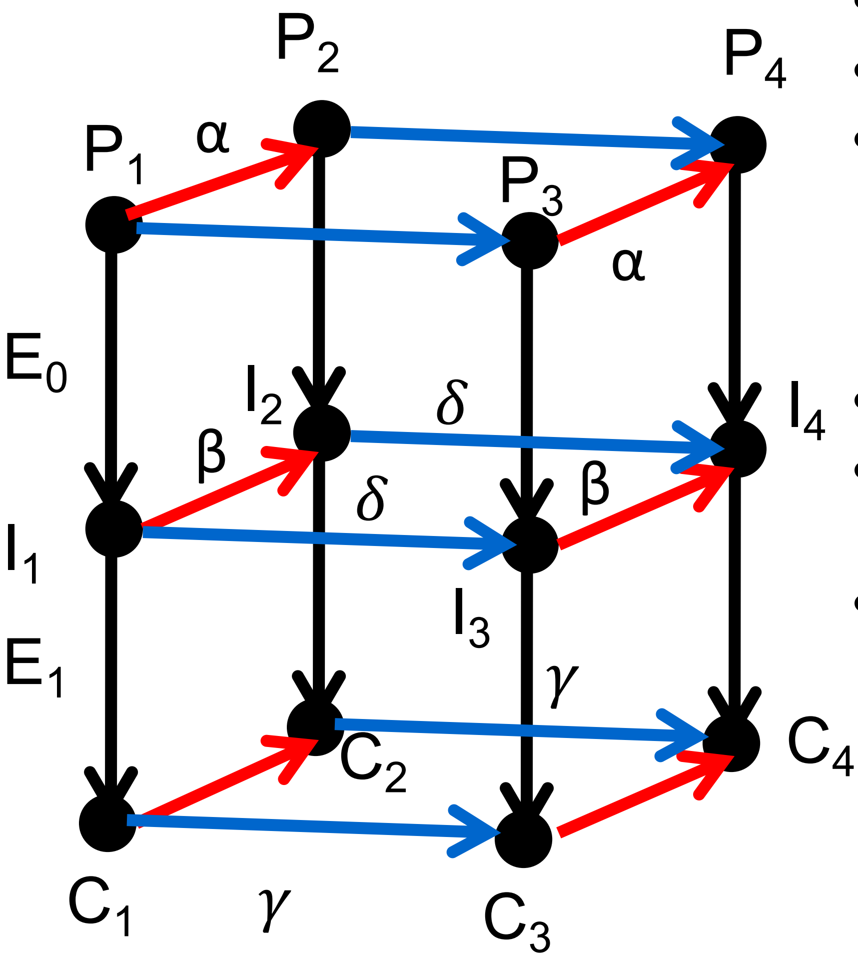
```
./new_feal4.exe -preparecorrespondingtext 1234567890abcdef
23234545ababcdcd 2048 feal.in1 feal.in2
Key schedule
  k0       : 90abcdef
  k1       : 32b729f8
  k2       : ada42552
  k3       : d26ad875
  k4       : ed3f65e8
  k5       : 5f452e24
  k6       : 14ee3941
  k7       : dbcb9075
  k0+k4+k5 : 22d18623
  k1+k4    : df884c10
  k2+k6    : b94a1c13
  k3+k6+k7 : 1d4f7141
  k4+k5+k6 : a694728d
  k4+k6+k7 : 221accdc
```

Automated attack

```
./new_feal4.exe -linearattack feal.in1 feal.in2
256 pairs examined
Plain: a1b24026 54a3e397, Cipher: c259fa58 99a44084
Plain: 44392b89 3e28d016, Cipher: b01696d4 59d70a09
…
…

Final check
  Round 1 trial key: 22d18623
  Round 2 trial key: df884c10
  Round 3 trial key: b94a1c13
  Round 4 trial key: 1d4f7141
  k4k5k6  trial key: a694728d
  k4k6k7  trial key: 221accdc
  succeeded
```

Boomerang Attack



- E_0: $\alpha \rightarrow \beta$ with probability, p.
- E_1: $\delta \rightarrow \gamma$ with probability, q.
- For each pair (P_1, P_2) with E_0: $\alpha \rightarrow \beta$, obtain (C_1, C_2) and compute $C_3 = C_1 \oplus \gamma$ and $C_4 = C_2 \oplus \gamma$. Request the decryption of (C_3, C_4) as (P_3, P_4).
- Probability that $P_3 \oplus P_4 = \alpha$, is p^2q^2.
- For random permutation, the probability that $P_3 \oplus P_4 = \alpha$, is 2^{-n}.
- Can also be mounted for all possible β's and γ's as long as $\beta^1\gamma$, with $p^2 = [\sum_{\beta,\alpha\rightarrow\beta} Pr^2(\alpha\rightarrow\beta)]^{1/2}$, $q^2 = [\sum_{\gamma,\gamma\rightarrow\delta} Pr^2(\gamma\rightarrow\delta)]^{1/2}$

End

DES Data

S Boxes as Polynomials over GF(2)

```
1,1:
   56+4+35+2+26+25+246+245+236+2356+16+15+156+14+146+145+13+135+134+1346+1345+
   13456+125+1256+1245+123+12356+1234+12346
1,2:
   C+6+5+4+45+456+36+35+34+346+26+25+24+246+2456+23+236+235+234+2346+1+15+156+
   134+13456+12+126+1256+124+1246+1245+12456+123+1236+1235+12356+1234+12346
1,3:
   C+6+56+46+45+3+35+356+346+3456+2+26+24+246+245+236+16+15+145+13+1356+134+13
   456+12+126+125+12456+123+1236+1235+12356+1234+12346
1,4:
   C+6+5+456+3+34+346+345+2+23+234+1+15+14+146+135+134+1346+1345+1256+124+1246
   +1245+123+12356+1234+12346

2,1: C+4+456+3+36+35+26+245+2456+235+2356+1+16+156+1456+13+136+135+1356+12+
     125+1256+1246+1236+12356
2,2: C+5+4+35+34+346+345+2+256+246+2456+236+1+156+145+13+135+134+
     1346+1345+12+126+125+124+1246+12456+123+1235+12356+1234
2,3: C+6+5+4+456+36+3456+2+24+246+23+1+1245+12456+1235+12356
2,4: C+6+5+45+3+26+24+245+23+236+1+156+145+1456+1356+126+1256+1245+12456+
     123+1236
```

Legend: `C+6+56+46` means $1 \oplus x_6 \oplus x_5 x_6 \oplus x_4 x_6$

S boxes as polynomials

```
3,1: 6+4+45+35+2+1+16+15+146+145+13+135+12+126+125+1256+123+1236+1235+12346
3,2: C+6+5+4+46+456+36+35+356+34+346+345+3456+2+25+256+24+245+23+236+
     234+2346+1+16+14+146+145+1456+135+1356+1346+13456+126+125+
     1256+124+1246+12456+1234+12346
3,3: 6+46+45+456+3+35+26+25+256+24+246+23+236+235+2356+234+1+1456+
     13456+12+126+125+1256+124+123+1236+1235+12356+1234
3,4: C+5+46+45+456+3+35+34+3456+2+24+245+2456+235+2356+234+16+14+146+
     145+1456+13+1356+134+13456+12+124+1245+12456+123+1234

4,1: C+56+4+46+45+3+3456+26+25+256+245+2456+23+236+2346+1+16+156+
     146+1456+13+136+135+13456+12+125+124+1245+123+1236+12356+1234
4,2: C+6+5+56+46+45+3+345+3456+2+26+256+2456+236+234+2346+16+15+
     156+14+146+145+1456+136+135+1345+13456+12+125+124+1245+1236+1235+
     12356+1234
4,3: C+56+46+45+456+3+36+35+2+26+256+2456+23+2356+234+2346+1+15+156+
     146+135+1356+1346+13456+1256+124+1245+12356+1234
4,4: 6+5+56+4+46+456+36+35+26+25+256+245+2456+23+235+2356+2346+1+
     156+14+146+1356+134+1346+1345+13456+125+1256+124+1245+1235+12356+1234
```

S boxes as polynomials

```
5,1: 56+45+3+36+35+356+346+345+3456+26+25+256+24+246+2456+235+16+14+
     145+13+136+1346+1345+13456+12+126+125+1256+124+1245+123+1236+1235+
     12356+1234
5,2: C+5+56+4+46+45+36+35+34+346+345+3456+2+25+256+246+245+235+2356+234+
     2346+1+16+156+14+145+13+136+135+134+1346+1345+13456+126+125+124+
     12456+123+12356+1234+12346
5,3: 6+5+4+3+36+356+346+3456+24+236+2346+1+156+145+1456+1345+126+1246+
     123+1236+1234+12346
5,4: 6+5+56+46+45+36+34+346+345+3456+2+24+246+245+236+2356+15+156+146+
     13+136+1356+1345+1256+124+1246+1245+12456+1236+1234

6,1: 5+456+3+34+346+345+3456+24+2456+23+234+2346+1+16+145+1456+135+134+
     1346+1345+13456+126+1246+12456+1236
6,2: 6+4+456+35+256+245+23+235+16+15+1456+13+136+135+1356+12+1245+
     12456+123+12356
6,3: C+6+5+4+3+35+345+2+24+2456+1+145+1456+13+136+1356+1345+1245+123+
     1236+1235+12356+12346
6,4: C+5+56+46+45+456+36+356+34+346+345+3456+2+23+2346+16+15+156+146+1456+
     13+136+135+1356+1246+12456+1236+12356+12346
```

S boxes as polynomials

7,1: 6+5+45+3+34+345+2+246+2456+23+1+146+1456+1346+13456+1256+1246+1236

7,2: 5+56+4+45+456+3+36+346+3456+2+245+2456+2346+16+15+156+13+135+1356+
1346+13456+124+1245+123+1236+1235+12356+12346

7,3: C+5+4+3456+2+26+24+2456+23+1+16+14+13+1345+12+1246+12456+1236+1234

7,4: 6+5+3+345+3456+24+23+236+234+2346+16+15+156+14+1456+136+135+1345+
13456+12+124+1245+123+1236+1235+1234+12346

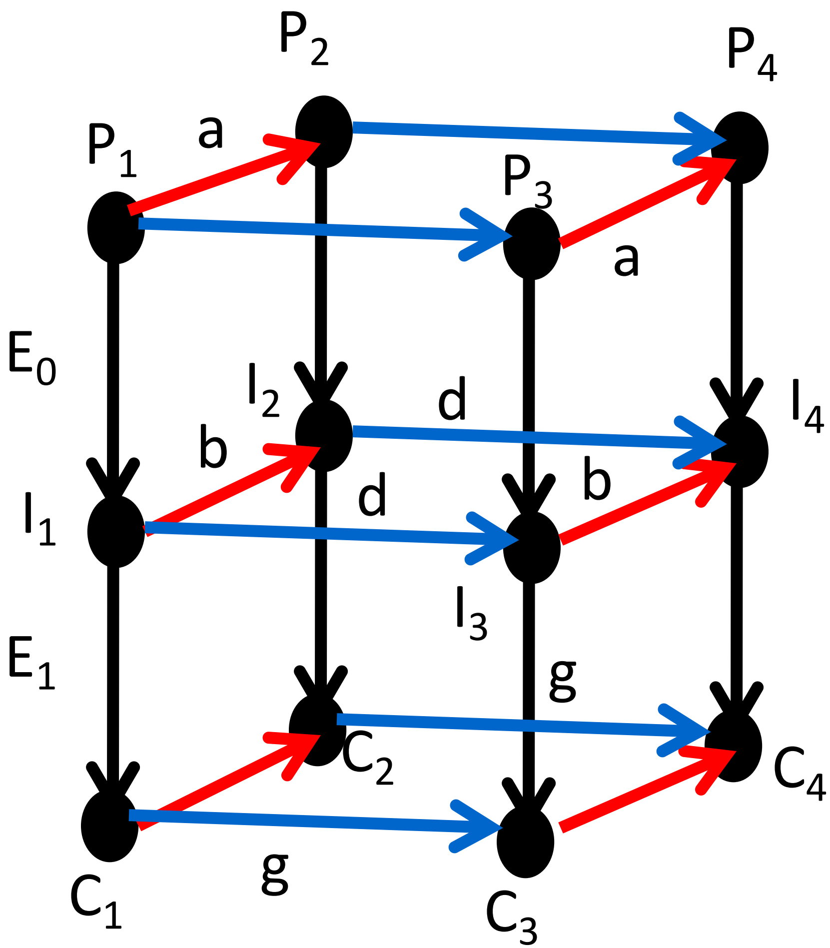
8,1: C+5+56+4+46+45+3+356+346+3456+2+256+245+236+16+15+1456+13+135+1356+
1346+1256+124+1246+1245+123+1235+12356+12346

8,2: 5+45+3+35+2+26+256+246+2456+236+2346+1+15+156+14+146+145+1456+135+
125+12456+1235+12356

8,3: C+6+5+4+35+2+25+24+245+23+156+14+146+13+135+1356+134+1346+125+124+
1245+123+1234+12346

8,4: C+6+5+46+456+3+34+346+26+25+256+24+246+245+234+2346+1+16+156+145+
1456+136+135+134+1346+1246+12456+1236+12356+1234+12346

Amplified Boomerang Attack



- Given plaintext pair $(P_1, P_2)(P_3, P_4))$
- For random permutations, the probability that $P_1 \oplus P_2 = P_3 \oplus P_4 = a$,
- E_0: $a \rightarrow b$ with probability, p.
- When both pairs satisfy $E_0(P_1) \oplus E_0(P_2) = E_0(P_3) \oplus E_0(P_4) = b$, $E_0(P_1) \oplus E_0(P_3) = (E_0(P_1) \oplus b) \oplus (E_0(P_3) \oplus b) = E_0(P_2) \oplus E_0(P_4)$.
- If $E_0(P_1) \oplus E_0(P_3) = E_0(P_2) \oplus E_0(P_4) = g$, each has a probability, q, to be a right pair wrt $g \rightarrow d$. $C_1 \oplus C_3 = C_2 \oplus C_4 = d$
- Pr(quartet becomes right quartet with difference a)= $(Np)^2/2$ quartets
- Expected number of right quartets is ${}_{Np}C_2 2^{-n} q^2$

Truncated Differentials

- A *truncated differential* predicts that the differences are restricted to some set. For example, in the description of the 2R-attack on 7-round DES for a right pair with respect to the 5-round characteristic, there are some cipher text bits with a zero difference for sure. This can be described as a 7-round truncated differential of DES with probability p=1/9511 that predicts the difference of 12 output bits.

- Truncated differentials can be used in the differential 1R- and 2R-attacks, to discard wrong pairs. Another application of truncated differentials is to define a distinguisher for the cipher (resulting in a key recovery attack at the end). For example, there is a 12-round truncated differential (in rounds 5–16) of Skipjack with probability 1 that predicts 16 bits of difference.

Rectangle Attack

- Given N pairs with difference a, pN pairs satisfy a→b.
- pN pairs satisfy a→b.
- ~$(Np)^2/2$ quartets that satisfy differentials.
- Given Np pairs $(P_1,P_2),(P_3,P_4)$, expected number of right quartets is ${}_{Np}C_2\ 2^{-n} q^2=N^2\ 2^{-n+1}\ (pq)^2$
- $E' = E_f \cdot E_1 \cdot E_0 \cdot E_b, Z_i=E_0(P_i))$
- Instead of just looking for g→d, look for any g'→d.

Rectangle Distinguisher

- $P_1 \oplus P_2 = P_3 \oplus P_4 = a$, $C_1 \oplus C_3 = C_2 \oplus C_4 = b$
- $Pr[(P_1,P_2),(P_3,P_4)$ is a right quartet$] = 2^{-n} \sum_{a,b} ([Pr(a \rightarrow a)\ Pr(b \rightarrow b)) \sum_{g} ([Pr(g \rightarrow d)\ Pr(g \oplus a \oplus b \rightarrow d))$
- $E' = E_f \cdot E_1 \cdot E_0 \cdot E_b$, $Z_i = E_0(P_i))$
- Steps
  1. Data collection
  2. Initialize
  3. Insert
  4. Generate Quartet
  5. Find and analyze quartets
  6. Count sub-keys

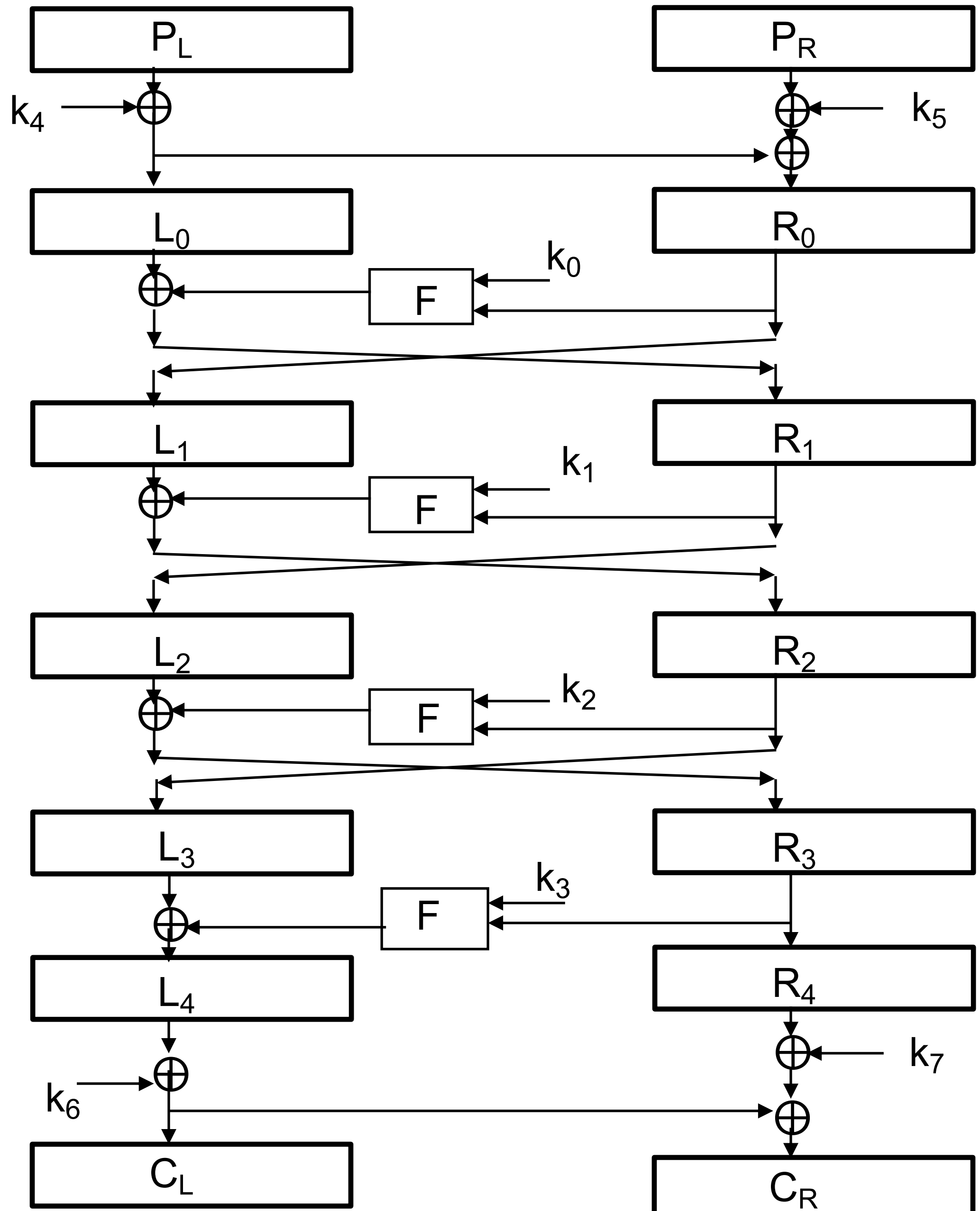
Bilinear Attack

- Let $L_r[0, 1, 2, . . . , n−1]$, $R_r[0, 1, 2, . . . , n−1]$ are the input to round r and $L_r[0, 1, 2, . . . , n − 1]$, $O_r[0, 1, 2, . . . , n−1]$ are the input (without key) and output to the round functions.
- If $\alpha \subseteq \{0, 1, 2, . . . , n−1\}$, define $L_r[\alpha] = \bigoplus_{s\in\alpha} L_r[s]$.
- Consider the bilinear $L_{r+1}[\beta]\cdot R_{r+1}[\alpha] \bigoplus R_r[\beta]\cdot L_r[\alpha] = L_r[\beta]\cdot O_r[\alpha]$.

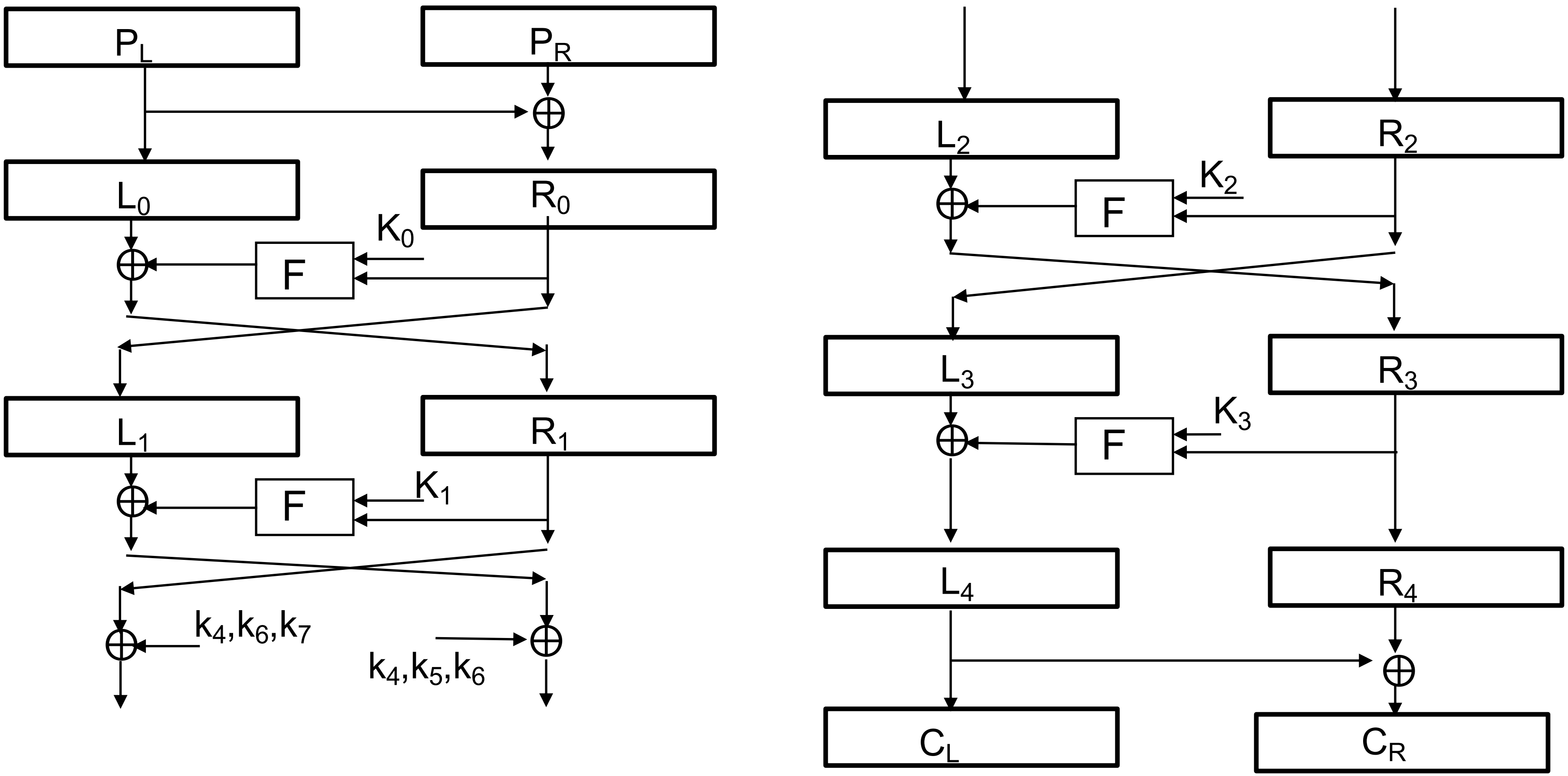
Slide Attack

- Let F be a per-round function.
- If $C = E_K(P) = F^m{}_K(P)$, $P,C \varepsilon GF(2)^n$ and $P' = F(P)$
- $C' = E(P') = F(C)$. To find slide pairs, let $a_F(P,C) = K$ which is easy to calculate. Store $2^n/2$ (and possibly less as in DES) pairs (P,C) if $a_F(P,C) = a_F(P',C')$, $P' = F_K(P)$ and $C' = F(C)$. By birthday collision, this will happen.
- Effective against rounds which implement weak permutations.

Original FEAL-4



Refactored FEAL-4



$K_0 = k_0+k_5+k_6$

$K_1 = k_1+k_4$

$K_3 = k_3+k_6+k_7$

$K_2 = k_2+k_6$