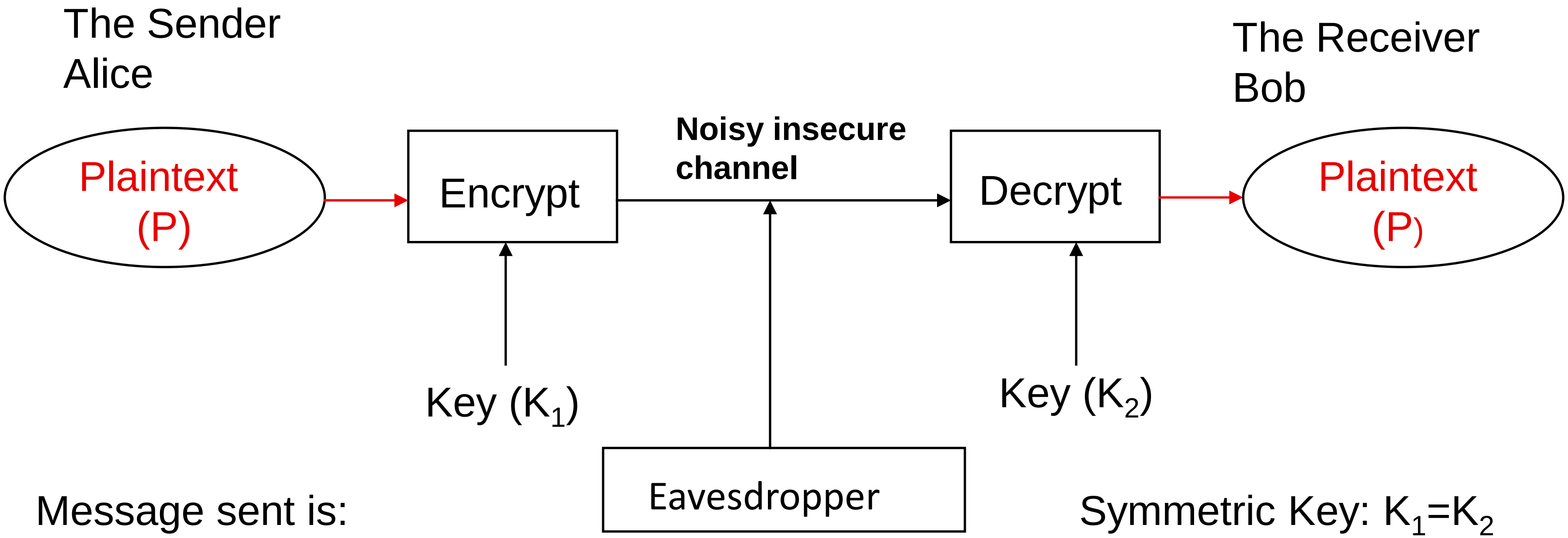
Block Ciphers

John Manferdelli
jmanfer@microsoft.com
JohnManferdelli@hotmail.com

This material is provided without warranty of any kind including, without limitation, warranty of non-infringement or suitability for any purpose. This material is not guaranteed to be error free and is intended for instructional use only

The wiretap channel: “In the beginning”



The Sender
Alice

Plaintext
(P)

Encrypt

Key (K_1)

Noisy insecure channel

Eavesdropper

Decrypt

Key (K_2)

The Receiver
Bob

Plaintext
(P)

Message sent is:
$C= E_{K1}(P)$
Decrypted as:
$P=D_{K2}(C)$
P is called plaintext.
C is called ciphertext.

Symmetric Key: $K_1=K_2$
Public Key: $K_1 \neq K_2$
K_1 is publicly known
K_2 is Bob’s secret

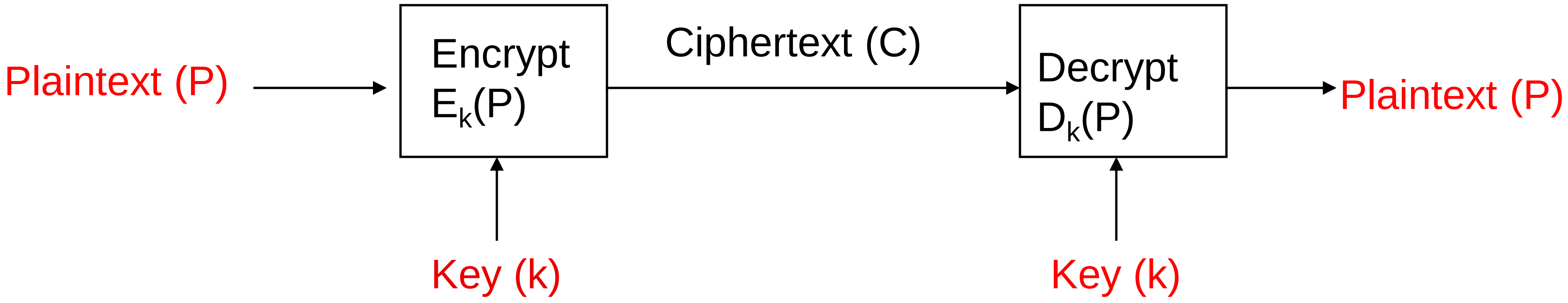
Cryptography and adversaries

- Cryptography is computing in the presence of an adversary.
- What do you want to protect?
- Against who?
- Under what circumstances?
- An adversary is characterized by:
  - Talent
  - Access to information
    - Probable plaintext attacks.
    - Known plaintext/ciphertext attacks.
    - Chosen plaintext attacks.
    - Adaptive interactive chosen plaintext attacks (oracle model).
  - Computational resources

Computational strength of adversary

- Infinite - Perfect Security
  - Information Theoretic
  - Doesn’t depend on computing resources or time available
- Polynomial
  - Asymptotic measure of computing power
  - Indicative but not dispositive
- Realistic
  - The actual computing resources under known or suspected attacks.
  - This is us, low brow.

Symmetric ciphers



- Encryption and Decryption use the same key.
  - The transformations are simple and fast enough for practical implementation and use.
  - Two major types:
    - Stream ciphers: bit at a time
    - Block ciphers: n bits → n bits
  - Examples: DES, AES, RC4, A5, Enigma, SIGABA, etc.

Cipher Requirements

- WW II
  - Universally available (simple, light instrumentation) – interoperability.
  - Compact, rugged: easy for people (soldiers) to use.
  - Kerckhoff’s Principle: Security in key only: We assume that the attacker knows the complete details of the cryptographic algorithm and implementation
  - Adversary has access to some corresponding plain and cipher-text
- Now
  - Adversary has access to unlimited cipher-text and lots of chosen text.
  - Implementation in digital devices (power/speed) paramount.
  - Easy for computers to use.
  - Resistant to ridiculous amount of computing power.

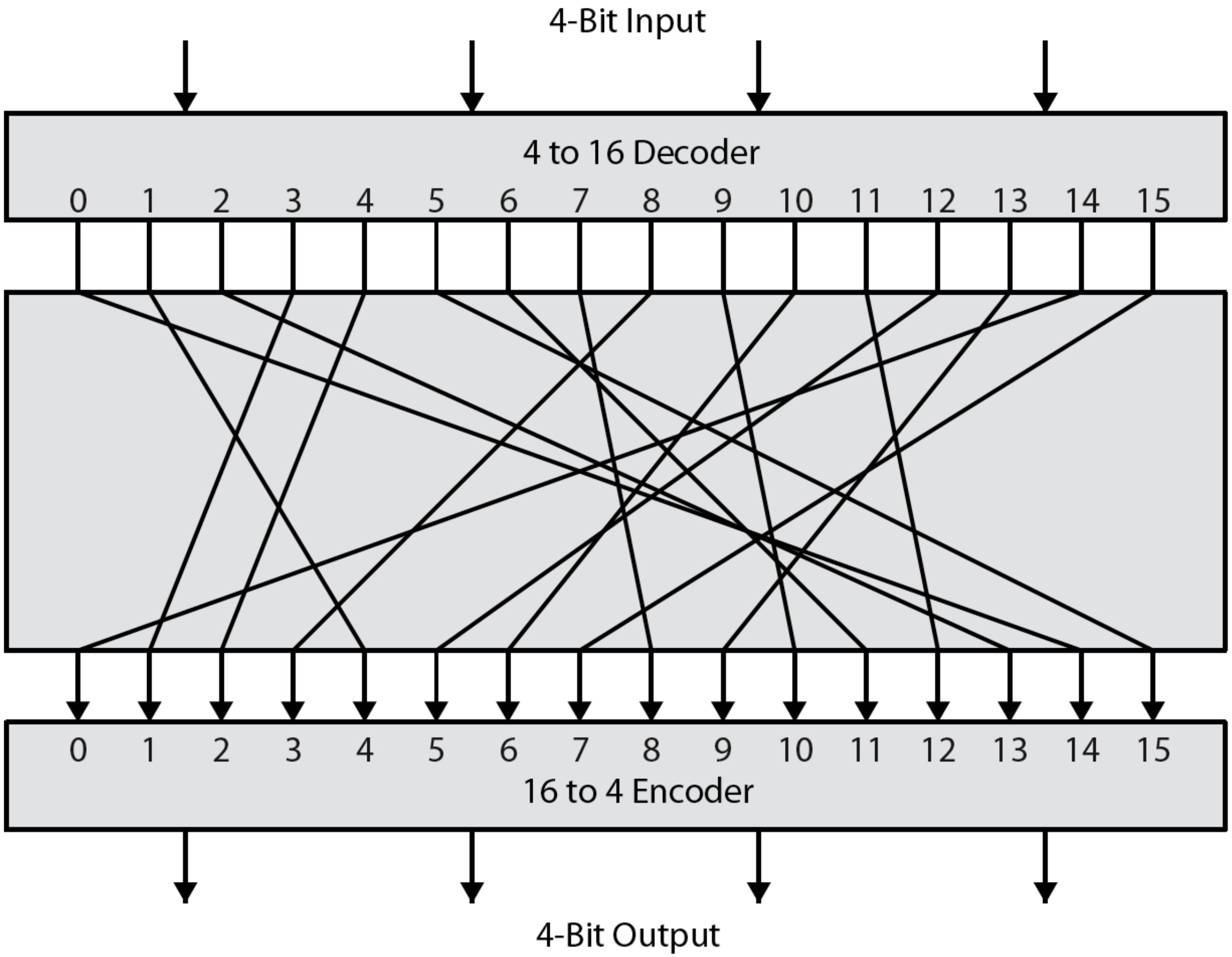
Practical attacks

- Exhaustive search of theoretical key space.
- Exhaustive search of actual key space as restricted by poor practice.
- Exploiting bad key management or storage.
- Stealing keys.
- Exploiting encryption errors.
- Spoofing (ATM PIN).
- Leaking due to size, position, language choice, frequency, inter-symbol transitions, timing differences, side channels..

Mathematical view of block ciphers

- $E(k, x) = y$.
- $E: GF(2)^m x GF(2)^n \longrightarrow GF(2)^n$, often $m=n$.
- $E(k,x)$ is a bijection in second variable.
- $E(k, \cdot)$ in S_N, $N= 2^n$. In other words, k selects a permutation from S_N. If $n=64$, $N=2^{64}$ and $|S_N|= 2^{64}!$ which is enormous
- Each bit position is a balanced boolean function.
- E (and its inverse) should be easy to compute if you know k but not if you don’t.

What is a block cipher



Iterated key dependant transformations

- Building an unpredictable or random permutation is easy if you’re allowed to use enormous keys.
- Each bit position must be a horribly complicated function of key and input to defeat cryptanalysis
- Lots of constraints must be satisfied (bijection, balance, …)
- How do we do this?
- Use a simple (key dependant) transformation (called a “round”) and apply it many (~n) times.
- The simple transformation must change for each round otherwise $E_k(x) = \sigma_k(x)^r$ which is not safe.
- Easiest way to do this is to make the simple transformation depend on different portions of the key in each round. This is called a “key schedule”.

Block ciphers -review

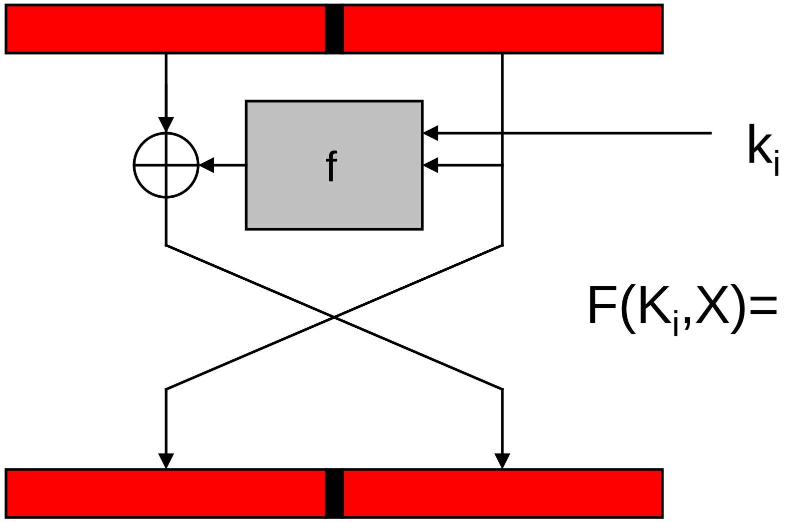
- Complicated keyed invertible functions constructed from iterated elementary rounds.

Characteristics:

- *Fast*
- *Data encrypted in fixed “block sizes” (64,128,256 bit blocks are common).*
- *Key and message bits non-linearly mixed in cipher-text*

Horst Feistel to the rescue!

Graphic courtesy of Josh Benaloh

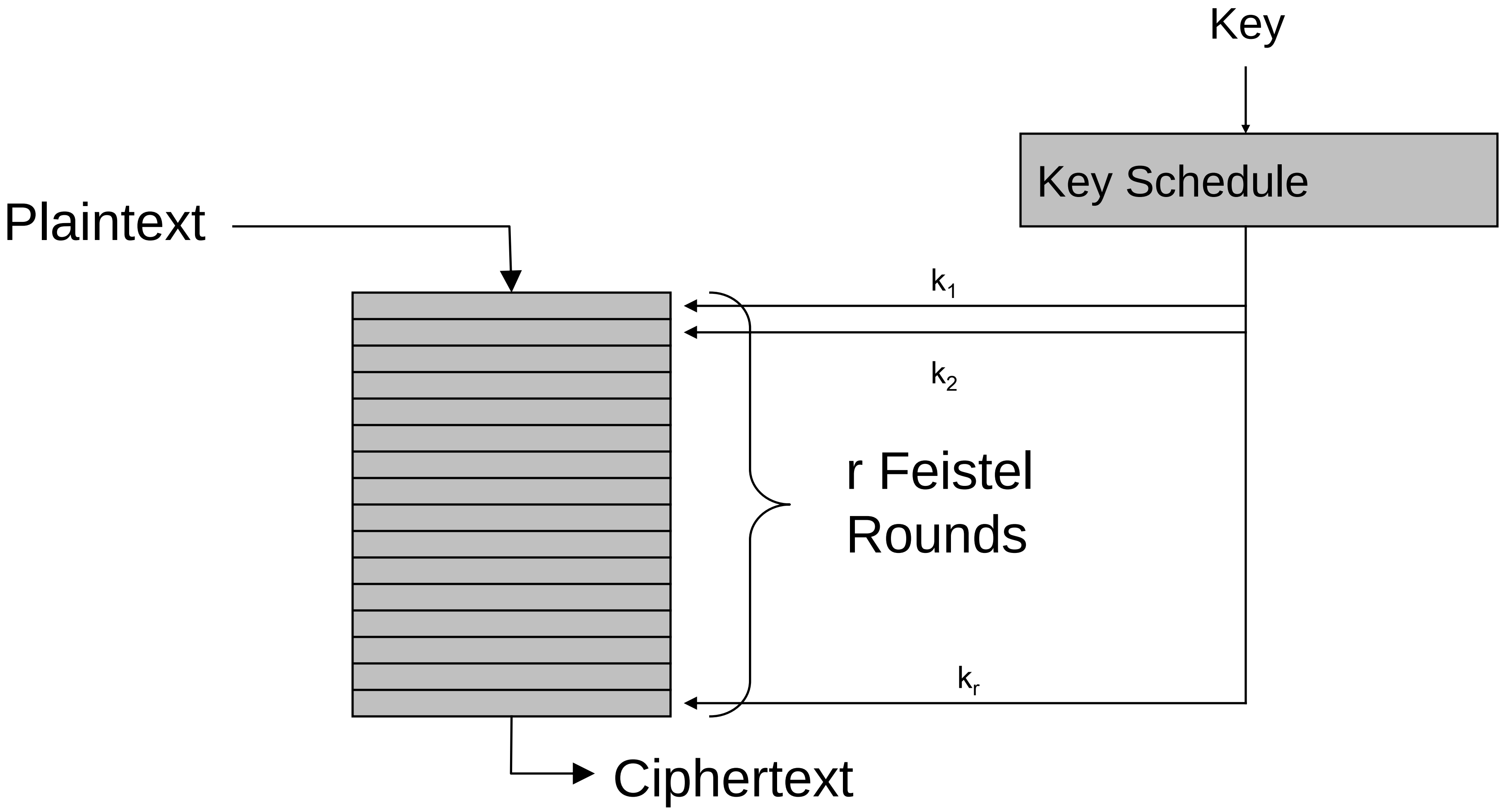


$F(K_i,X)$= non-linear function

Note: If $\sigma_i(L,R) = (L \oplus f(E(R) \oplus k_i), R)$ and $\tau(L, R) = (R, L)$, this round is $\tau\sigma_i(L, R)$.

To invert: swap halves and apply same transform with same key: $\sigma_i\tau\tau\sigma_i(L,R) = (L,R)$.

Iterated Feistel Cipher



Data Encryption Standard

- Federal History
  - 1972 study.
  - RFP: 5/73, 8/74.
  - NSA: S-Box influence, key size reduction.
  - Published in Federal Register: 3/75.
  - FIPS 46: January, 1976.
- *DES*
  - Descendant of Feistel’s Lucifer.
  - Designers: Horst Feistel, Walter Tuchman, Don Coppersmith, Alan Konheim, Edna Grossman, Bill Notz, Lynn Smith, and Bryant Tuckerman.
- Brute Force Cracking
  - EFS DES Cracker: $250K, 1998. 1,536 custom chips. Can brute force a DES key in days.
  - Deep Crack and distributed net break a DES key in 22.25 hours.

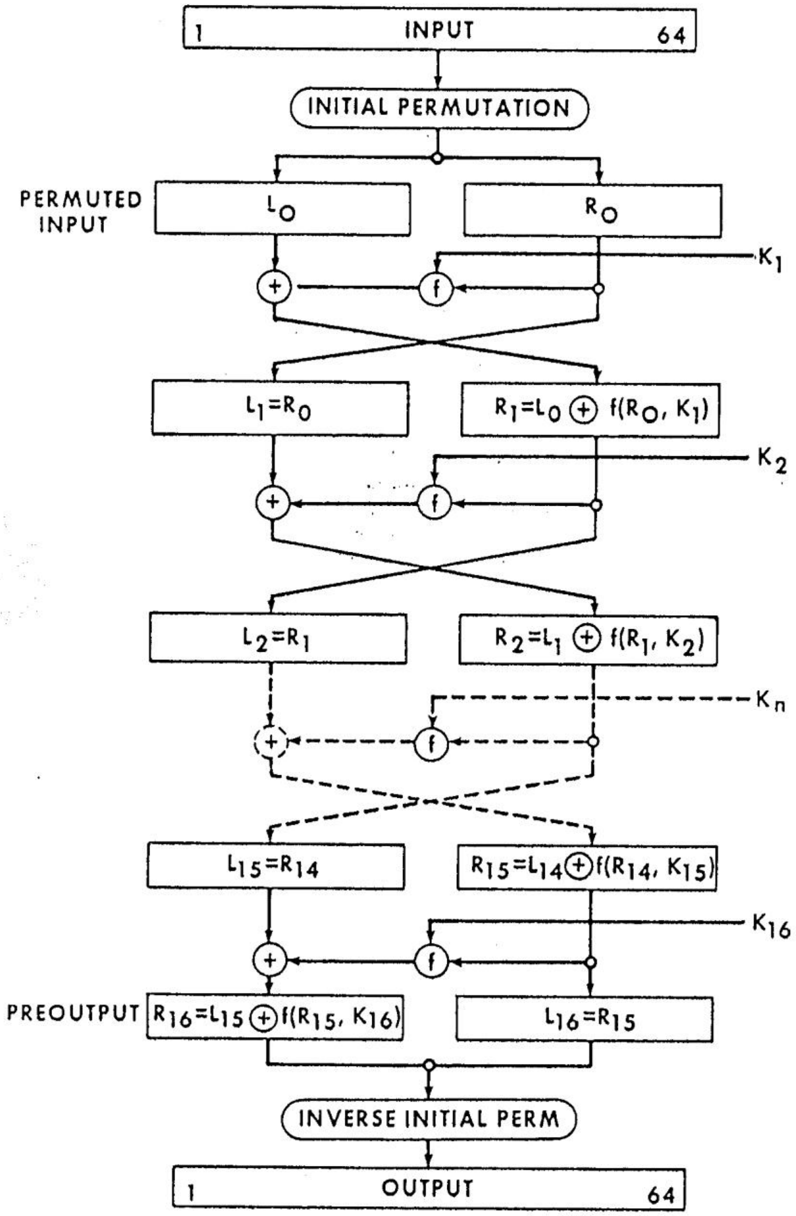
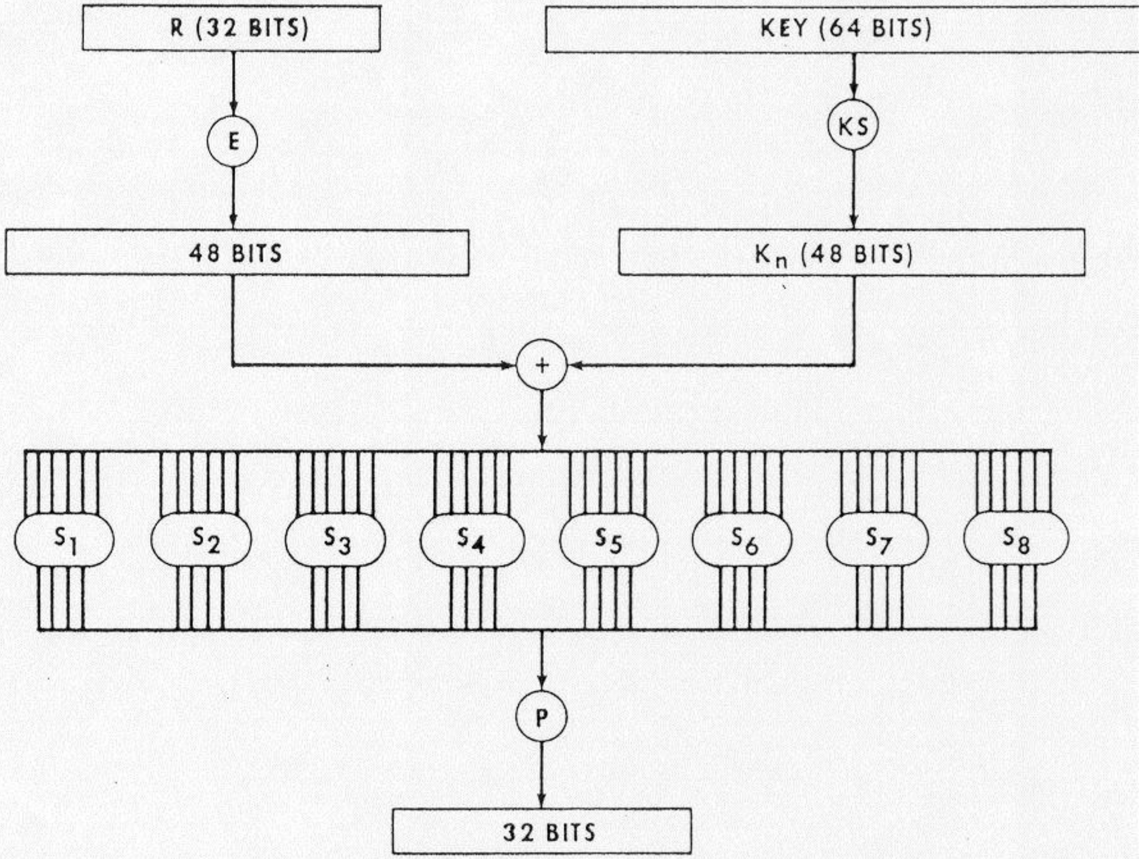


Figure 5.1. Electronic Codebook (ECB) Mode—Enciphering Computation.



K_n CHANGES FOR N=1, 2...16

E=E FUNCTION
KS=KEY SCHEDULE

Figure 5.2. Electronic Codebook (ECB) Mode—Calculation of f(R,K).

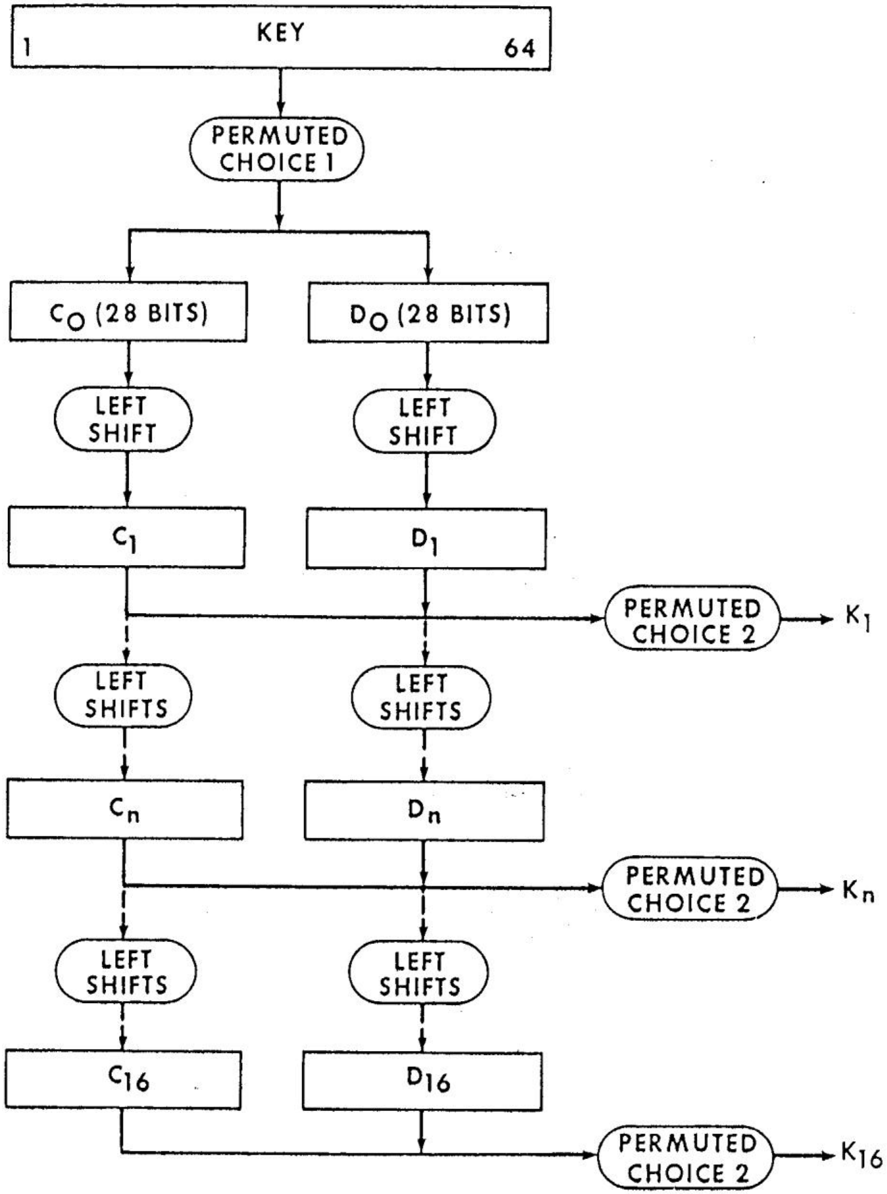


Figure 5.3. Electronic Codebook (ECB) Mode—Key Schedule (KS) Calculation.

DES Described Algebraically

- □$\sigma_i(L,R) = (L \oplus f(E(R) \oplus k_i), R)$
  - k_i is 48 bit sub-key for round i.
  - $f(x) = P(S_1 S_2 S_3 \ldots S_8(x))$. Each S –box operates on 6 bit quantities and outputs 4 bit quantities.
  - P permutes the resulting 32 output bits.
- $\tau(L, R) = (R, L)$.
- Each round (except last) is $\tau\sigma_{i.}$
- Note that $\tau\tau$□ □τ□□ □□□ □$\sigma_i\,\sigma_i = \sigma_i^2$.
- Full DES is: $DES_K(x) = IP^{-1}\ \sigma_{16}\tau$□□□□$\sigma_3\tau$□$\sigma_2\tau\sigma_1\ IP(x)$.
- So its inverse is: $DES_K^{-1}(x) = IP^{-1}\ \sigma_1\tau$□□□□$\sigma_{14}\tau\sigma_{15}\tau\sigma_{16}\ IP(x)$.

DES Key Schedule

Key schedule round 1

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

Key schedule round 2

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

What can go wrong

- Key space is too small
- $E_k(x) = \rho_r\ \rho_{r-1} \ldots \rho_1$, all linear in the key bits.
  - Resulting transformation is linear
  - It’s easy to solve the resulting linear equations
- $E_k(x)$ decomposible into transformations with independent key bits
  - $E_{k1||k2}(x) = E'_{k1}(x)||E''_{k2}(x)$
- $E_k(x)$ should “look” like a random permutation and the effect of k should “look” like it picks the random permutations unpredictibly

DES Attacks: Exhaustive Search

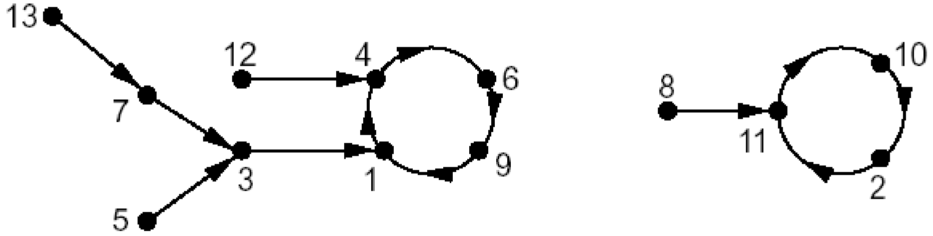
- Symmetry DES(**k**$\oplus$**1**, **x**$\oplus$**1**)=DES(**k, x**)$\oplus$**1**
- Suppose we know plain/cipher text pair (p,c)

```
for(k=0;k<2^56;k++) {
  if(DES(k,p)==c) {
      printf("Key is %x\n", k);
      break;
      }
}
```

- Expected number of trials (if k was chosen at random) before success: 2^{55}

Random mappings

- Let F_n denote all functions (mappings) from a finite domain of size *n* to a finite co-domain of size *n*
- Every mapping is equally likely to be chosen, $|F_n|$ = n^n the probability of choosing a particular mapping is 1/ n^n
- *Example*▷ *f* : ¶1˓ 2˓ …˓ 13♢ → ¶1˓ 2˓ …˓ 13♢



Graphic by Maithili Narasimha

- As n tends to infinity, the following are expectations of some parameters associated with a random point in {1, 2, …, n℘ and a random function from $\mathcal{F}_n$: (i) tail length: √(πn◁8) (ii) cycle length: √(π n◁8) (iii) rho-length: √ (π n◁2↽—▷

Time memory trade off (“TMTO”)

- If we can pre-compute a table of (k, E_k(x)) for a fixed x, then given corresponding (x,c) we can find the key in O(1) time.
- Trying random keys takes O(N) time (where N, usually, 2^k, is the number of possible keys)
- Can we balance “memory” and “time” resources?
- It is not a 50-50 proposition. Hellman showed we could cut the search time to O($N^{(1/2)}$) by pre-computing and storing O($N^{(1/2)}$) values.

Group theory and DES

- What is the minimum length of a product of involutions from a fixed set required to generate S_n?
- What does this have to do with the number of rounds in a cipher?
- How does this affect the increased security by "enciphering twice" with different keys?
- **Theorem** (Coppersmith and Grossman): If $\sigma_K(L,R)= (L\oplus f(E(R)\oplus K , R)$, $<\tau \square \sigma_K>= A_N$, $N= 2^n$.
- Note (Netto): If a and b are chosen at random from S_n there is a good chance (~¾) that $<a,b>= A_n$ or S_n .

Weak Keys

- DES has:
  - Four weak keys k for which $E_k(E_k(m))= m$.
  - Twelve semi-weak keys which come in pairs k_1 and k_2 and are such that $E_{k1}(E_{k2}(m))= m$.

  - Weak keys are due to “key schedule” algorithm
- How they arise:
  - A 28 bit quantity has potential symmetries of period 1, 2, 4, 7, and 14.
  - Suppose each of C_0 and D_0 has a symmetry of period 1; for example C_0 =0x0000000, D_0= 0x1111111. We can easily figure out a master key (K) that produces such a C_0 and D_0.

Feistel Ciphers defeat simple attacks

- After 4 to 6 rounds to get flat statistics.
- Parallel system attack
  - Solve for key bits or constrain key bits

$$k_{i(1)}= a_{11}(K)p_1\, c_1 + a_{12}(K)p_2\, c_1 +\ldots+ a_{1N}(K)p_n c_n$$
$$\ldots \qquad \ldots \qquad \ldots \qquad \ldots$$
$$k_{i(m)}= a_{m1}(K)p_1\, c_1 + a_{m2}(K)p_2\, c_1 +\ldots+ a_{mN}(K)p_n c_n$$

- Solving Linear equations for coefficients determining cipher

$$c_1= f_{11}(K)p_1 + f_{12}(K)p_2 +\ldots+ f_{1n}(K)p_n$$
$$c_2= f_{21}(K)p_1 + f_{22}(K)p_2 +\ldots+ f_{2n}(K)p_n$$
$$\ldots \qquad \ldots \qquad \ldots \qquad \ldots$$
$$c_m= f_{m1}(K)p_1 + f_{m2}(K)p_2 +\ldots+ f_{mn}(K)p_n$$

- Even a weak round function can yield a strong Feistel cipher if iterated sufficiently.
  - Provided it’s non-linear

The sophisticated attacks

- Exhaustive search
- Differential cryptanalysis
  - Differentials
- Linear Cryptanalysis
  - Linear approximations

Polynomial representation

- If f is boolean function on n variables $x_1, x_2, \ldots, x_n$ and **a**=(a1, a2, …, an)

  then $f(x_1, x_2, \ldots, x_n)=$□$\sum_{\mathbf{a}} g(\mathbf{a})\, x_1^{a1}\, x_2^{a2} \ldots, x_n^{an}$ where $g(\mathbf{a}) = \sum_{\mathbf{b<a}} f(b_1, b_2, \ldots, b_n)$. Here **b<a** means the binary representation of b does not have a 1 unless there is a corresponding 1 in the representation of a.

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	1
1	0	0	0
0	1	0	1
1	1	0	0
0	0	1	1
1	0	1	0
0	1	1	0
1	1	1	1

- g(0,0,0)= f(0,0,0)=1
- g(0,1,0)=f(0,0,0)+f(0,1,0)=0
- g(1,0,0)=f(0,0,0)+f(1,0,0)=1
- g(1,1,0)=f(0,0,0)+f(1,0,0))+f(0,1,0))+f(1,1,0)=0
- g(0,0,1)=f(0,0,0)+f(0,0,1)=0
- g(0,1,1)=f(0,0,0)+f(0,0,1) +f(0,1,0)+f(0,1,1)=1
- g(0,0,1)= g(1,0,1)= g(0,1,1)= g(1,1,1)= 0

- $f(x_1, x_2, x_3)= 1+x_1+x_2\, x_3$

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+1
   35+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+2
   34+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+1
   45+13+1356+134+13456+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+1
   345+1256+124+1246+1245+123+12356+1234+12346
```

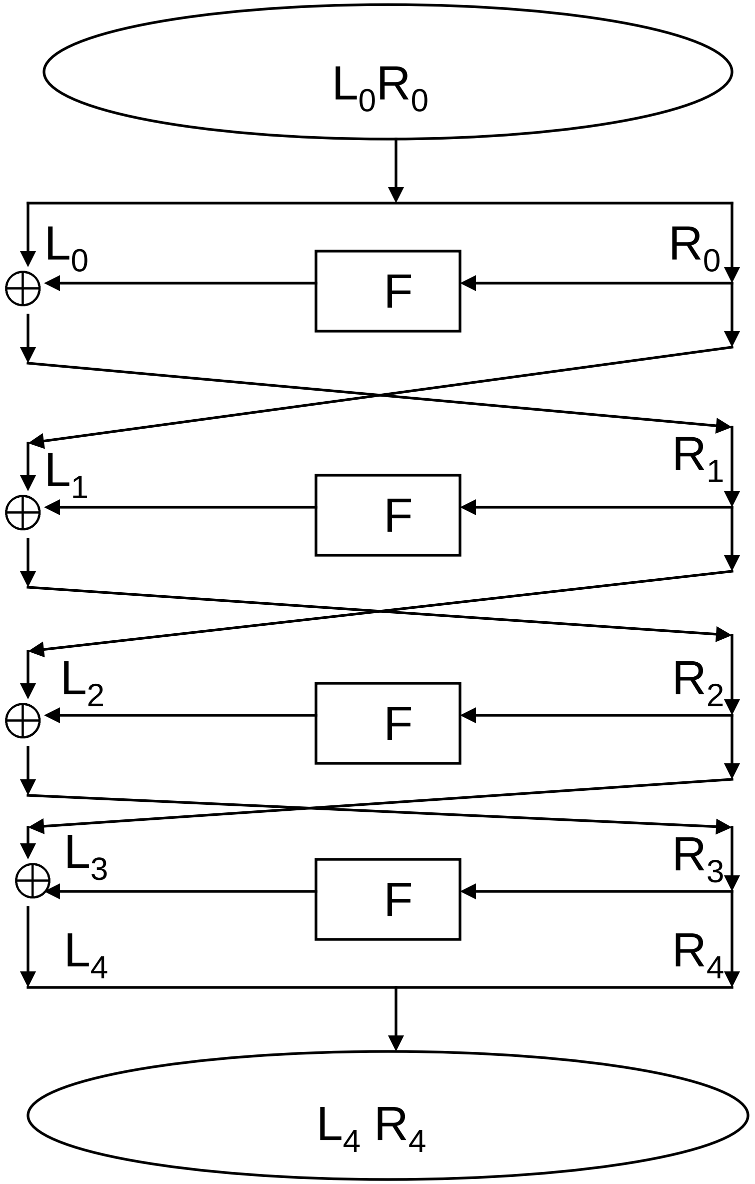
Legend: `C+6+56+46 means` $1 \oplus x_6 \oplus x_5 x_6 \oplus x_4 x_6$

Differential Cryptanalysis

- Let E and E* be inputs to a cipher and C and C* be corresponding outputs with $E\oplus E^*=E'$ and $C\oplus C^*=C'$.
- The notation E' → C', p means the "input xor", E' produces the "output xor" C' with probability p. Not all input/output xors and possible and the distribution is uneven. This can be used to find keys. E' → C', p is called a *characteristic*.
- Notation: $D_j(x',y')= \{u: S_j(u)\oplus S_j(u\oplus x')= y'\}$. $k_j\in x\oplus D_j(x',y')= \square_j(x,x',y')$. test($E_j$, E_j^*,C_j')= $\tau_j(E_j,E_j\oplus E_j^{*\prime}, C_j')$
- For the characteristic 0x34→d in S-box 1 from inputs$1\oplus 35=34$, $D_1(34,d)$= {06, 10, 16, 1c, 22, 24, 28, 32} and $k_j\in$ {7, 10, 17, 1d, 23, 25, 29, 33}= $1\oplus D_1(34,d)$

Simplified DES

- $L_{i+1}= R_i$, each 6 bits.
- $R_{i+1}= L_i \oplus f(R_i,K_i)$
- K is 9 bits.
- $E(x)= (x_1\ x_2\ x_4\ x_3\ x_4\ x_3\ x_5\ x_6)$
- S_1
  - `101 010 001 110 011 100 111 000`
  - `001 100 110 010 000 111 101 011`
- S_2
  - `100 000 110 101 111 001 011 010`
  - `101 011 000 111 110 010 001 100`
- K_i is 8 bits of K starting at i^th^ bit.

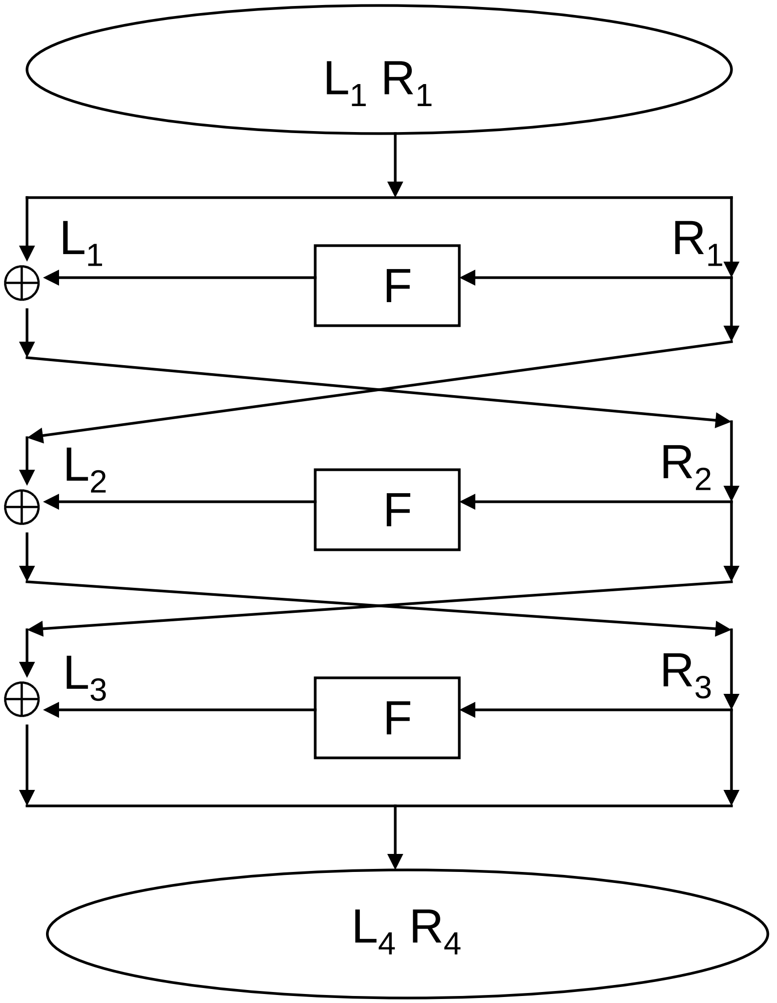


Differential Cryptanalysis – 3 rounds

- $R_4 \oplus R_1 = f(k_3,R_2)$. .... (1)
- $L_4 \oplus L_3 = f\,(k_4,R_3)$. .... (2)
- $R_4=R_3$, $L_2=R_1$, $L_3=R_2$.

- 1&2 → $L_4 \oplus L_3 \oplus R_2 \oplus L_1 = f(k_2,R_1) \oplus f(k_4,R_3)$.
- $L_3=R_2$ → $L_4 \oplus L_1 = f(k_2,R_1) \oplus f(k_4,R_3)$.

- $L_4 \oplus L_1 = f(k_2,R_1) \oplus f(k_4,R_3)$. ....(3)
- $L_4* \oplus L_1* = f(k_2,R_1*) \oplus f(k_4,R_3*)$. ....(4)
- 3&4 → $L_4' \oplus L_1' = f(k_2,R_1^*) \oplus f(k_4,R_3^*) \oplus f(k_2,R_1^*) \oplus f(k_4,R_3^*)$.
- $R_1=R_1^*$ → $L_4' \oplus L_1' = f(k_4,R_3) \oplus f(k_4,R_3^*)$.



Differential Cryptanalysis – 3 rounds

L_1, R_1 : 000111 011011

L_1*, R_1*: 101110 011011

L_1', R_1': 101001 000000

L_4, R_4 : 100101 000011

L_4*, R_4*: 011000 100100

L_4', R_4': 111101 100111

$E(R_4)$: 0000 0011

$E(R_4')$: 1010 1011

$L_4' \oplus L_1'$: 111 101$\oplus$□101 001= 010 100.

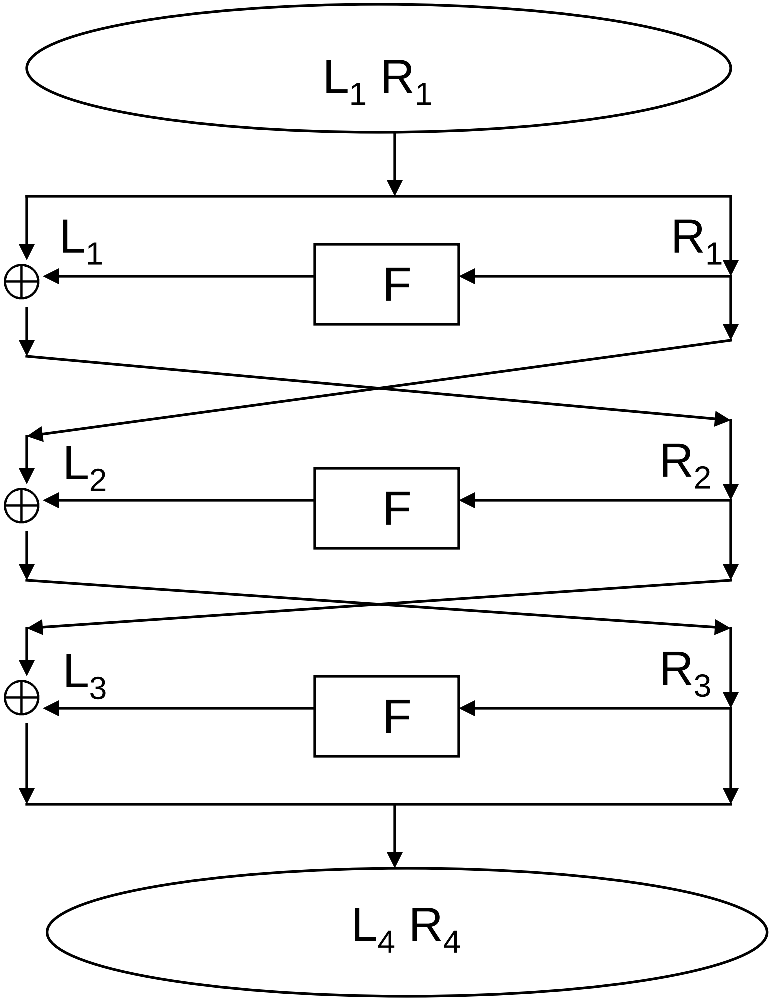
S_1': 1010 → 010(1001,0011).

S_2': 1011 → 100(1100,0111).

$(E(R_4 \oplus k_4)_{1..4}$=1001|0011, k_4= 1001|0011.

$(E(R_4) \oplus k_4)_{5..8}$= 1100|0111,$k_4$= 1111|0100.

K= 00x001101



Comments on Differential Cryptanalysis of full DES

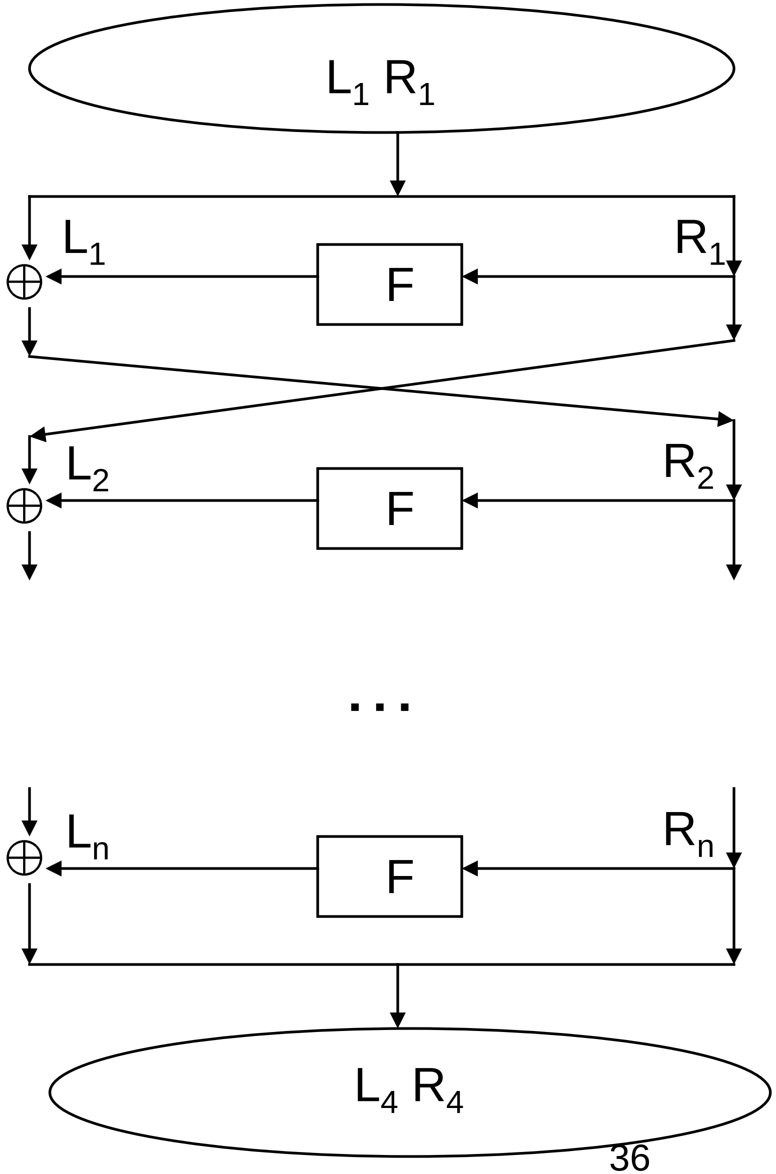
# Rounds	Needed pairs	Analyzed Pairs	Bits Found	# Char rounds	Char prob	S/N	Chosen Plain
4	2^3	2^3	42	1	1	16	2^4
6	2^7	2^7	30	3	1/16	2^{16}	2^8
8	2^{15}	2^{13}	30	5	1/10486	15.6	2^{16}
16	2^{57}	2^5	18	15	$2^{-55.1}$	16	2^{58}

DES S-Box Design Criteria

- No S-box is linear or affine function of its input.
- Changing one bit in the input of an S-Box changes at least two output bits.
- S-boxes were chosen to minimize the difference between the number of 1’s and 0’s when any input bit is held constant.
- S(X) and S(X$\oplus$001100) differ in at least 2 bits
- S(X)$\neq$S(X$\oplus$11xy00)

1R Differential attack

- Trial decode last round with all possible subkeys, see if differential holds.



Linear Cryptanalysis

- Basic idea:
  - Suppose $\alpha_i(P) \oplus \beta_i(C) = \gamma_i(k)$ holds with γ_i□linear, for i= 1, 2, …, m.
  - Each equation imposes a linear constraint and reduces key search by a factor of 2.
  - Guess (n-m-1) bits of key. There are $2^{(n-m-1)}$. Use the constraints to get the remaining keys.

- Can we find linear constraints in the “per round” functions and knit them together?
- No! Per Round functions do not have linear constraints.

Linear Cryptanalysis

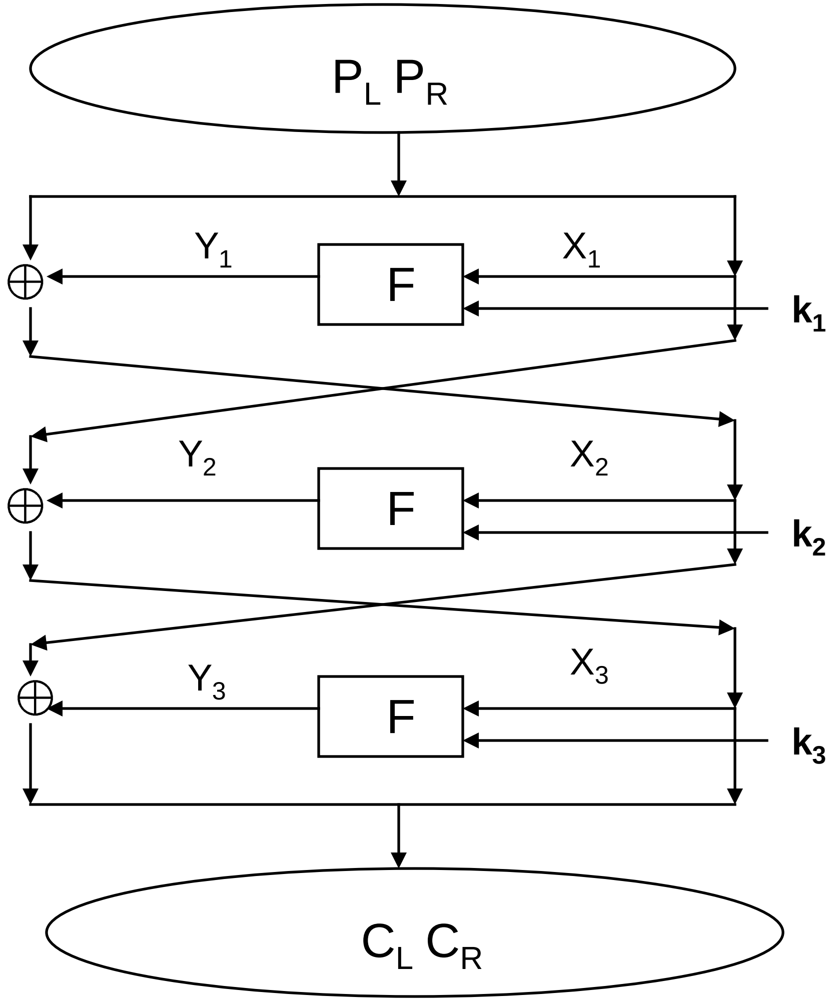
- Next idea
  - Can we find $\alpha(P)\oplus\beta(C)= \gamma(k)$ which holds with γ linear, with probability p?
  - Suppose $\alpha(P)\oplus\beta(C)= \gamma(k)$, with probability p>.5.
  - Collect a lot of plain/cipher pairs.
  - Each will “vote” for $\gamma(k)=0$ or $\gamma(k)=1$.
  - Pick the winner.

- p= $1/2+\in$ requires $c\in^{-2}$ texts (we’ll see why later).
- $\in$ is called “bias”.

Linear Cryptanalysis Notation

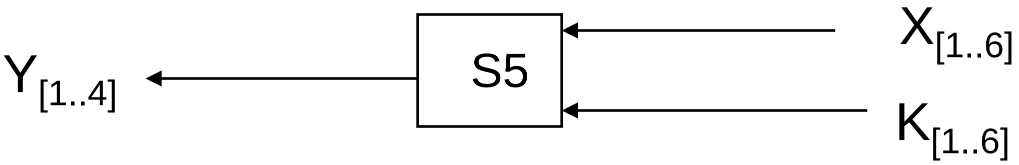
- Matsui numbers bits from right to left, rightmost bit is bit 0. FIPS (and everyone else) goes from left to right starting at 1. I will use the FIPS conventions. To map Matsui positions to everyone else's:
  - M(i)= 64-EE(i). For 32 bits make the obvious change.

- Matsui also refers to the two portions of the plaintext and cipher-text as (P_H, P_L), (C_H, C_L), we'll stick with (P_L, P_R), (C_L, C_R).



Linear and near linear dependence

- Here is a linear relationship over GF(2) in S5 that holds with probability 52/64 (from $NS_5(010000,1111)= 12$:



- $X[2]\oplus Y[1]\oplus Y[2]\oplus Y[3]\oplus Y[4]=K[2]\oplus 1$.
- Sometimes written: $X[2]\oplus Y[1,2,3,4]=K[2]\oplus 1$□

- You can find relations like this using the “Boolean Function” techniques we describe a little later

- After applying P, this becomes

  $X[17]\oplus F(X,K)[3,8,14,25]= K[26]\oplus 1$

Linear Cryptanalysis of 3 round DES

$X[17] \oplus Y[3,8,14,25] = K[26] \oplus 1$, p= 52/64

- Round 1

$X_1[17] \oplus Y_1[3,8,14,25] = K_1[26] \oplus 1$

$P_R[17] \oplus P_L[3,8,14,25] \oplus R_1[3,8,14,25] =$
$K_1[26] \oplus 1$

- Round 3

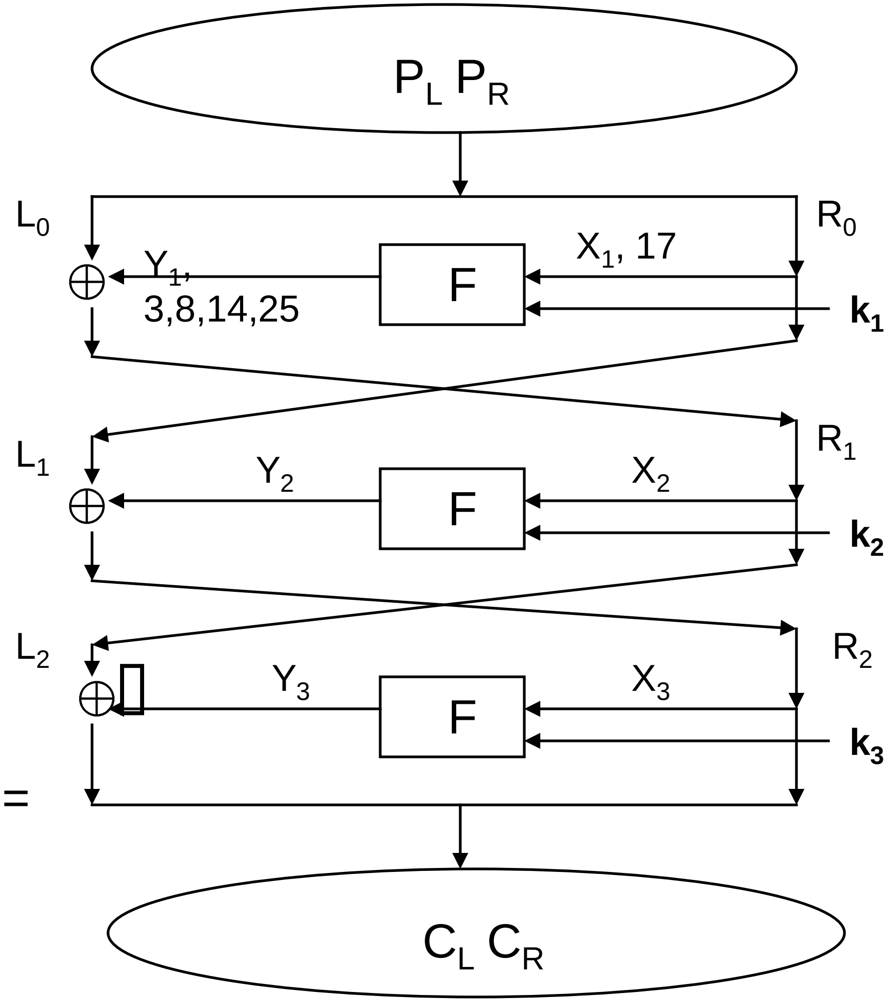
$X_3[17] \oplus Y_3[3,8,14,25] = K_3[26] \oplus 1$

$R_1[3,8,14,25] \oplus C_L[3,8,14,25] \oplus C_R[17] =$
$K_3[26] \oplus 1$

- Adding the two get:

$P_R[17] \oplus P_L[3,8,14,25] \oplus C_L[3,8,14,25] \oplus C_R[17] =$
$K_1[26] \oplus K_3[26]$

Thus holds with p= $(52/64)^2 + (12/64)^2 = .66$



Piling up Lemma

- Let X_i $(1 \leqq i \leqq n)$ be independent random variables whose values are 0 with probability p_i. Then the probability that $X_1 \oplus X_2 \oplus \cdots \oplus X_n = 0$ is

$$\frac{1}{2} + 2^{n-1} \prod_{[1,n]} (p_i - 1/2)$$

Proof:

By induction on n. It's tautological for n=1.

Suppose $\Pr[X_1 \oplus X_2 \oplus \cdots \oplus X_{n-1} = 0] = q = \frac{1}{2} + 2^{n-2} \prod_{[1,n-1]} (p_i - 1/2)$.

Then $\Pr[X_1 \oplus X_2 \oplus \cdots \oplus X_n = 0] = qp_n + (1-q)(1-p_n) = \frac{1}{2} + 2^{n-1} \prod_{[1,n]} (p_i - 1/2)$ as claimed.

Linear Cryptanalysis of full DES

- Can be accomplished with $\sim 2^{43}$ known plaintexts, using a14 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 $-1.19\text{x}2^{-21}$

$C_R[8,14,25]\oplus P_L[3,8,14,25]\oplus P_R[17]= K_{13}[26]\oplus K_{12}[24]\oplus K_{11}[26]\oplus$
$K_9[26] \oplus K_8[24]\oplus K_7[26]\oplus K_5[26]\oplus K_4[4]\oplus K_3[26]\oplus K_1 [26]$
with probability ½ $-1.19\text{x}2^{-21}$

Estimating cost of Linear attack

- Let X be the random variable representing the number of “1’s” resulting from an approximate linear relation of bias q.
- Linear attack is successful if for n trials, X>N/2
- What is Pr(X>N/2)? X is normally distributed as X~N(□□□□), where □=N/2+Nq and □ = N$^{1/2}$/2. N~O(q^{-2})

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:

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

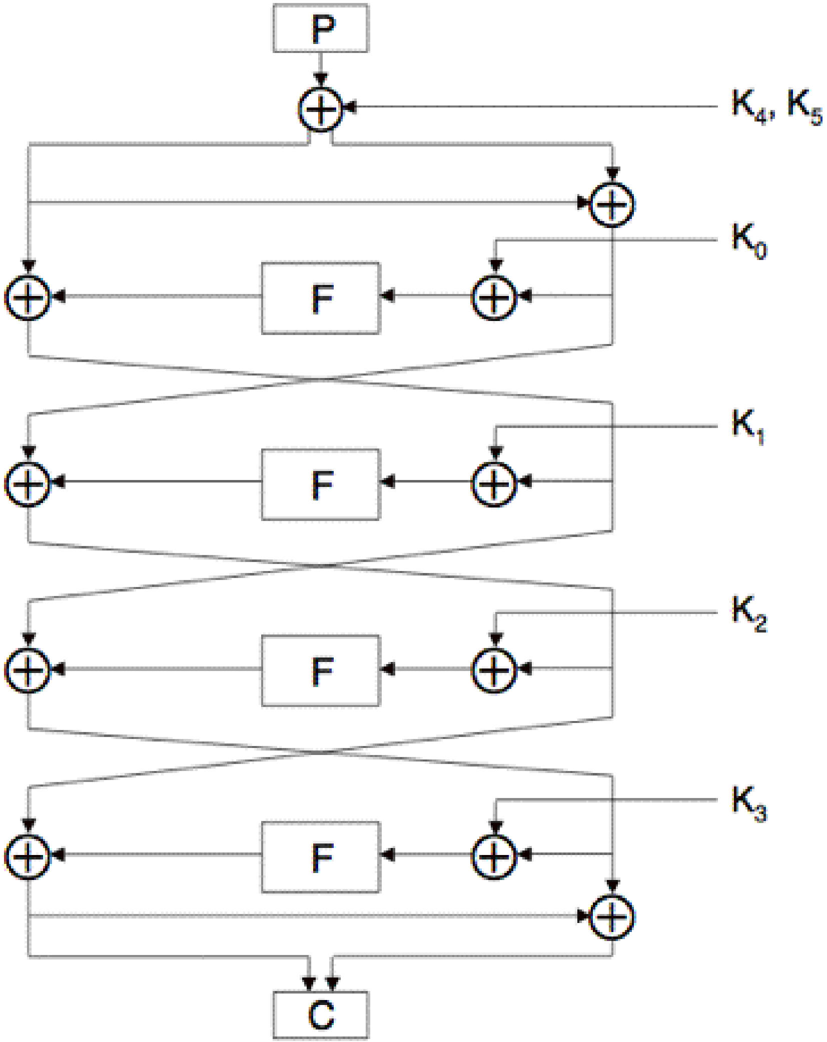
with probability **$\frac{1}{2}$ -1.19x2^{-21}**

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

with probability **$\frac{1}{2}$ -1.19x2^{-21}**

FEAL-4 Cipher

- 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, …, K_5
- Most important failed cipher: showed the power of differential cryptanalysis and linear cryptanalysis



Slide adapted from Mark Stamp

FEAL-4 Round Function

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

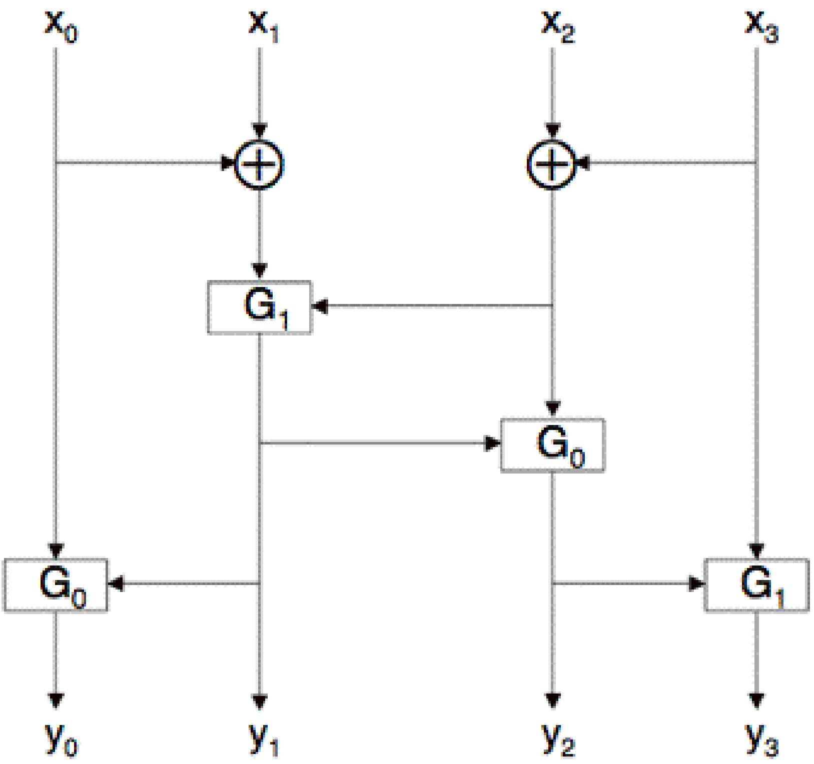


Diagram from Mark Stamp

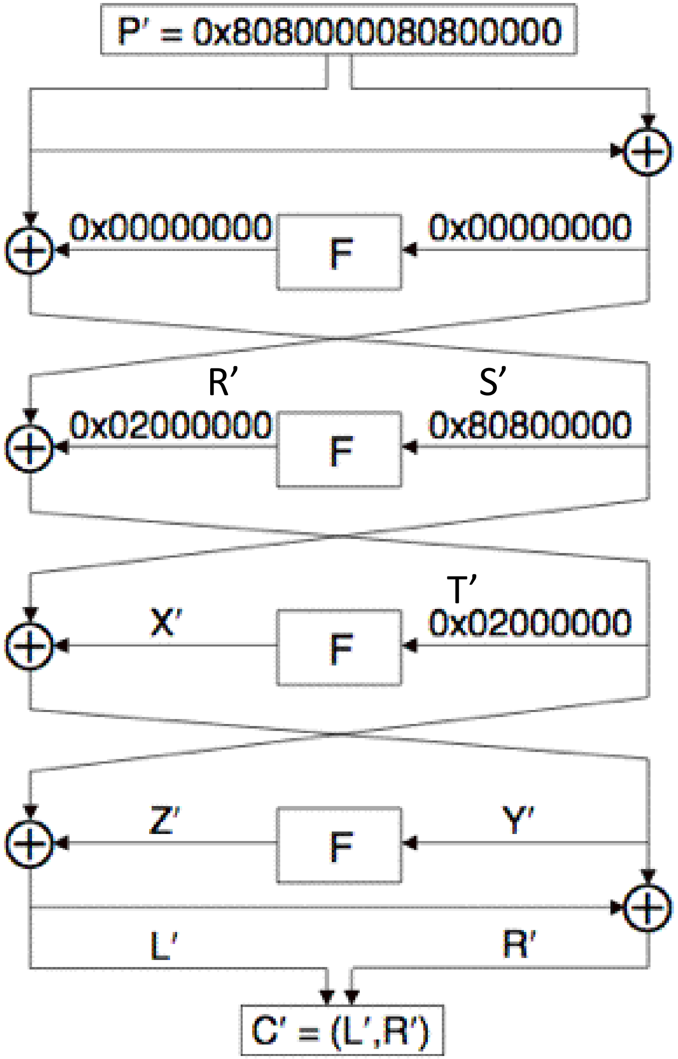
FEAL-4 Key Schedule

- $F_K(a_0||a_1||a_2||a_3, b_0||b_1||b_2||b_3)= c_0||c_1||c_2||c_3$ by
  - $d_1= a_0 \oplus a_1$
  - $d_2= a_2 \oplus a_3$
  - $c_1= G_1(d_1,a_2 \oplus b_0)$
  - $c_2= G_0(d_2,c_1 \oplus b_1)$
  - $c_0= G_0(a_0,c_1 \oplus b_2)$
  - $c_3= G_1(a_3,c_2 \oplus b_3)$
- $K_{-2}= 0$
- $K_{-1}= K_L$
- $K_0= K_R$
- $K_i= f_K(K_{i-2}, K_{i-1} \oplus K_{i-3})$

Slide adapted from Mark Stamp

FEAL-4 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 putative Z_0, Z_1
  - Note: $Z_i = F(Y_i \oplus K_3)$
- Compare true Z' to putative Z'



Slide adapted from Mark Stamp

FEAL-4 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

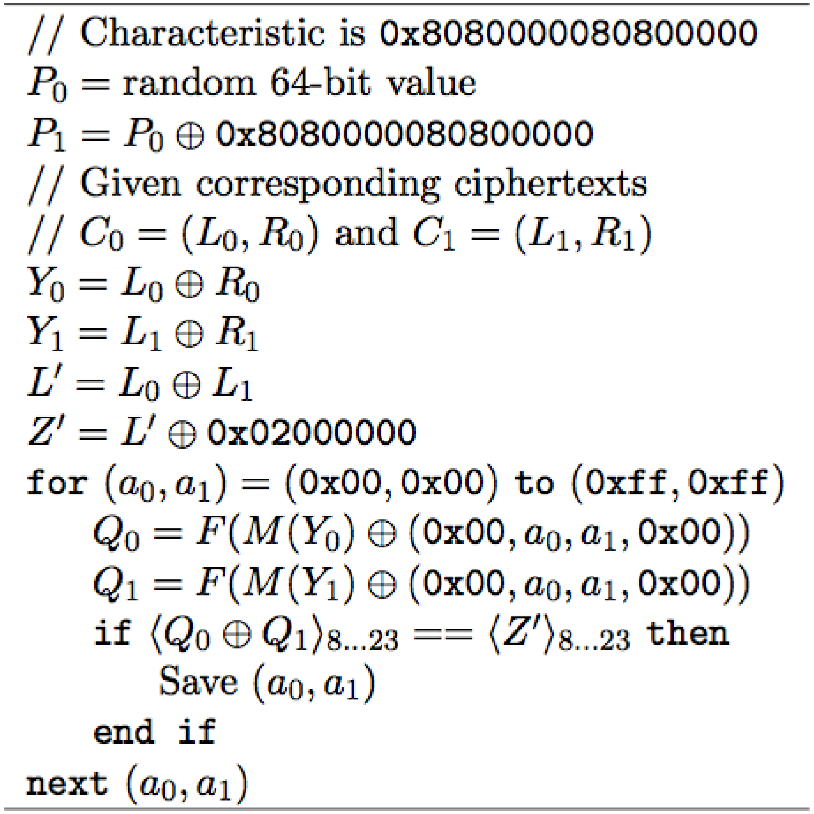
    $M(A) = (z, a_0 \oplus a_1, a_2 \oplus a_3, z)$, where z is all-zero byte
  - For all possible $A=(z, a_0, a_1, z)$, compute

    $Q_0 = F(M(Y_0) \oplus A)$ 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

Slide adapted from Mark Stamp

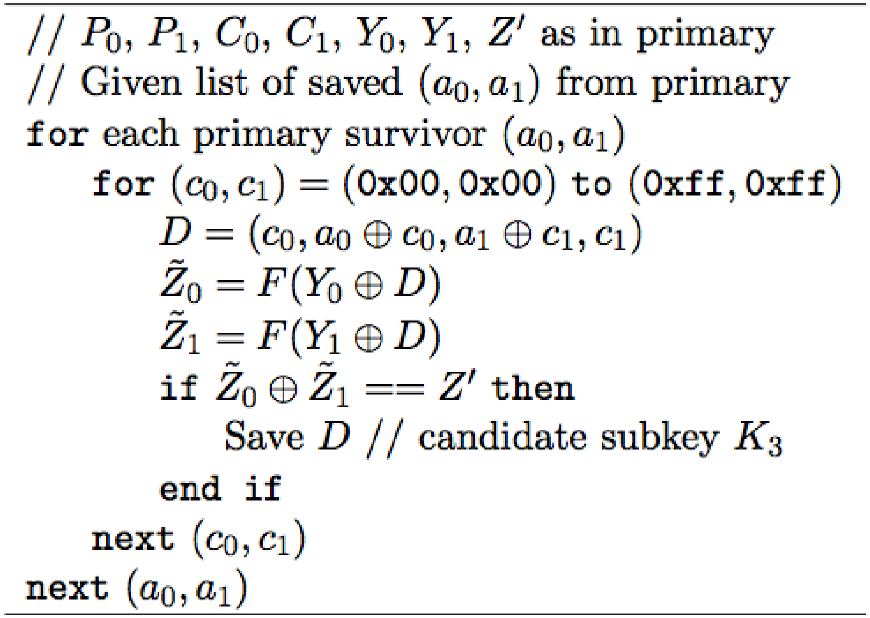
FEAL-4 Differential Attack

- Primary for K_3



```
// Characteristic is 0x8080000080800000
$P_0$ = random 64-bit value
$P_1 = P_0 \oplus$ 0x8080000080800000
// Given corresponding ciphertexts
// $C_0 = (L_0, R_0)$ and $C_1 = (L_1, R_1)$
$Y_0 = L_0 \oplus R_0$
$Y_1 = L_1 \oplus R_1$
$L' = L_0 \oplus L_1$
$Z' = L' \oplus$ 0x02000000
for $(a_0, a_1)$ = (0x00, 0x00) to (0xff, 0xff)
    $Q_0 = F(M(Y_0) \oplus (\texttt{0x00}, a_0, a_1, \texttt{0x00}))$
    $Q_1 = F(M(Y_1) \oplus (\texttt{0x00}, a_0, a_1, \texttt{0x00}))$
    if $\langle Q_0 \oplus Q_1 \rangle_{8\ldots 23} == \langle Z' \rangle_{8\ldots 23}$ then
        Save $(a_0, a_1)$
    end if
next $(a_0, a_1)$
```

- Secondary for K_3



```
// $P_0, P_1, C_0, C_1, Y_0, Y_1, Z'$ as in primary
// Given list of saved $(a_0, a_1)$ from primary
for each primary survivor $(a_0, a_1)$
    for $(c_0, c_1)$ = (0x00, 0x00) to (0xff, 0xff)
        $D = (c_0, a_0 \oplus c_0, a_1 \oplus c_1, c_1)$
        $\tilde{Z}_0 = F(Y_0 \oplus D)$
        $\tilde{Z}_1 = F(Y_1 \oplus D)$
        if $\tilde{Z}_0 \oplus \tilde{Z}_1 == Z'$ then
            Save $D$ // candidate subkey $K_3$
        end if
    next $(c_0, c_1)$
next $(a_0, a_1)$
```

- Assuming only one chosen plaintext pair

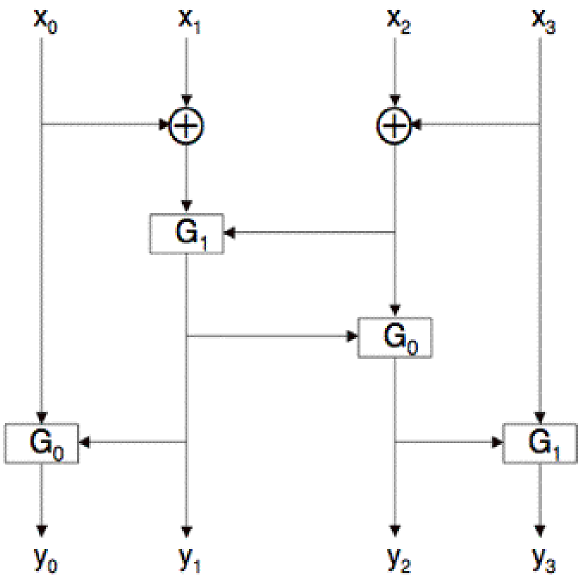
Slide adapted from Mark Stamp

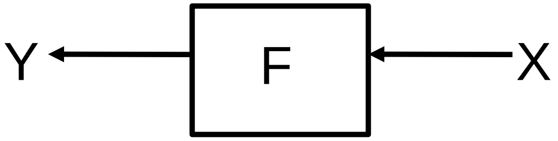
FEAL-4 Linear Attack

- X = X[0], …, X[31]), Y=F(X). Notation: X[i,j]= X[i]$\oplus$X[j]
- (a$\oplus$b)[7] = (a+b(mod 256))[7], so
- G_0(a,b)[5] = (a$\oplus$b)[7]
- (a$\oplus$b$\oplus$1)[7] = (a+b+1(mod 256))[7], so
- G_1(a,b)[5] = (a$\oplus$b$\oplus$1)[7]
- Since y_1= $G_1(x_0 \oplus x_1,\ x_2 \oplus x_3)$,
- Y[13]=y_1[5]=x_0[7]$\oplus$$x_1$[7]$\oplus$$x_2$[7]$\oplus$$x_3$[7]$\oplus$1=X[7,15,23,31]$\oplus$1
- Since $y_0=G_0(x_0,\ y_1)$, Y[5]=y_0[5]=y_1[7]$\oplus$$x_0$[7] =Y[15]$\oplus$X[7]
- Since $y_2=G_0(y_1,\ x_2 \oplus x_3)$,
  Y[21]=y_2[5]=y_1[7]$\oplus$$x_2$[7]$\oplus$$x_3$[7] = Y[15]$\oplus$X[23,31]
- Since $y_3=G_1(y_2,x_3)$,
  Y[29]=y_3[5]=y_2[7]$\oplus$$x_3$[7]$\oplus$1= Y[23]$\oplus$X[31]$\oplus$1

Y=F(X)

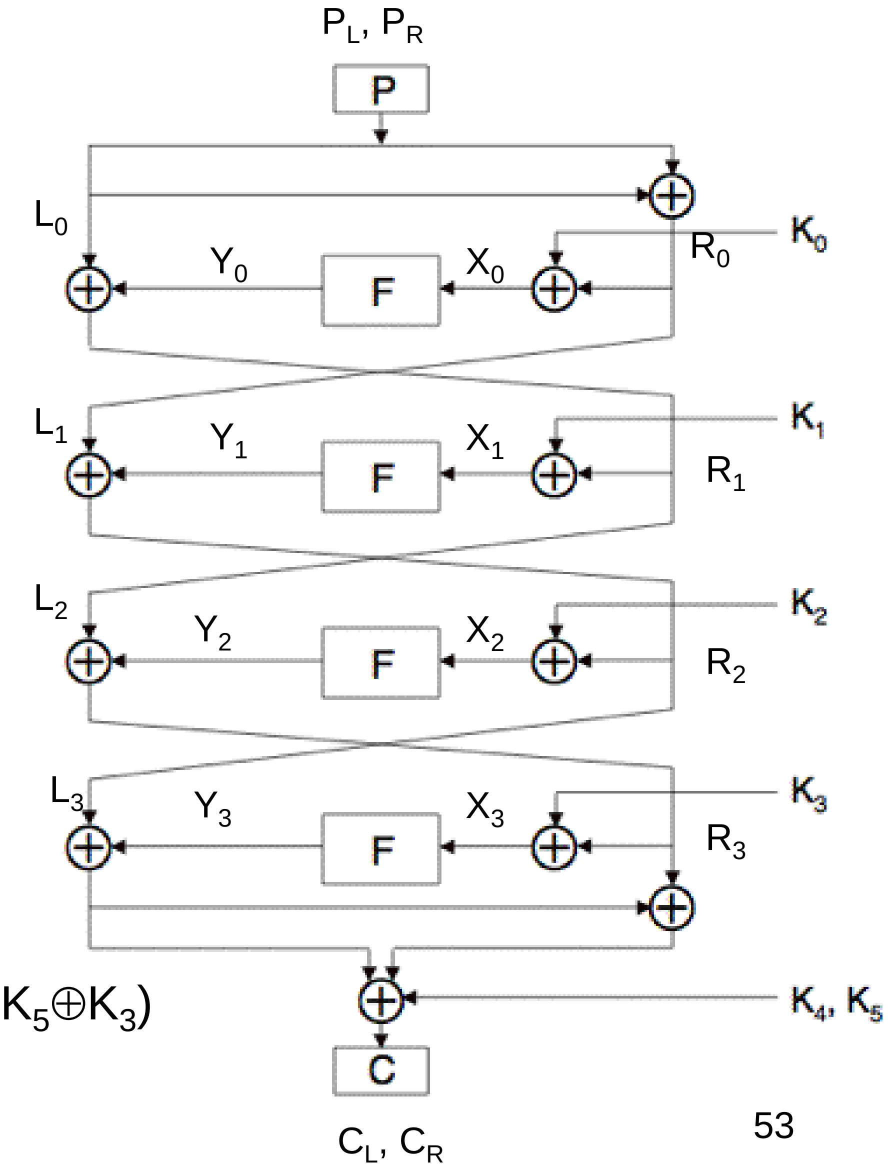
- $Y=(y_0,\ y_1,\ y_2,\ y_3)$
- $X=(x_0,\ x_1,\ x_2,\ x_3)$





FEAL-4 Linear Attack

- $L_0 = P_L,\ R_0 = P_L \oplus P_R$
- $Y_0 = F(R_0 \oplus K_0),\ R_1 = L_0 \oplus Y_0,\ L_1 = R_0$
- $Y_1 = F(R_1 \oplus K_1),\ R_2 = L_1 \oplus Y_1,\ L_2 = R_1$
- $Y_2 = F(R_2 \oplus K_2),\ R_3 = L_2 \oplus Y_2,\ L_3 = R_2$
- $Y_3 = F(R_3 \oplus K_3)$
- $C_L = L_3 \oplus Y_3 \oplus K_4,\ C_R = C_L \oplus R_3 \oplus K_5$
- $C_L = L_1 \oplus Y_1 \oplus Y_3 \oplus K_4 = P_L \oplus P_R \oplus Y_1 \oplus Y_3 \oplus K_4$
- So $C_L \oplus P_L \oplus P_R \oplus K_4 = Y_1 \oplus Y_3$
- $C_L \oplus P_L \oplus P_R \oplus K_4 = F(R_1 \oplus K_1) \oplus F(R_3 \oplus K_3)$
- $C_L \oplus P_L \oplus P_R \oplus K_4 = F(L_0 \oplus Y_0 \oplus K_1) \oplus F(R_3 \oplus K_3)$
- Since $R_3 = C_L \oplus C_R \oplus K_5$, and $L_0 = P_L$
- $C_L \oplus P_L \oplus P_R \oplus K_4 = F(P_L \oplus Y_0 \oplus K_1) \oplus F(C_L \oplus C_R \oplus K_5 \oplus K_3)$



FEAL-4 Linear Attack

- We’ve show
  1. $C_L \oplus P_L \oplus P_R \oplus K_4 = F(P_L \oplus Y_0 \oplus K_1) \oplus F(C_L \oplus C_R \oplus K_5 \oplus K_3)$,
  2. $Y_0 = F(R_0 \oplus K_0) = F(P_L \oplus P_R \oplus K_0)$
  3. $Y[13]=X[7,15,23,31] \oplus 1$
  4. $Y[5] = Y[15] \oplus X[7]$
  5. $Y[21]=Y[15] \oplus X[23,31]$
  6. $Y[29] \oplus Y[23] = X[31] \oplus 1$
- From 1,
  7. $(C_L \oplus P_L \oplus P_R \oplus K_4)[23,29] = F(P_L \oplus Y_0 \oplus K_1)[23,29] \oplus F(C_L \oplus C_R \oplus K_5 \oplus K_3)[23,29]$
- From 6,
  8. $F(P_L \oplus Y_0 \oplus K_1)[23,29] = (P_L \oplus Y_0 \oplus K_1)[31] \oplus 1$
  9. $F(C_L \oplus C_R \oplus K_5 \oplus K_3)[23,29] = (C_L \oplus C_R \oplus K_5 \oplus K_3)[31] \oplus 1$
- Adding 8 and 9,
  10. $(C_L \oplus P_L \oplus P_R \oplus K_4)[23,29] = (P_L \oplus Y_0 \oplus K_1)[31] \oplus (C_L \oplus C_R \oplus K_5 \oplus K_3)[31]$

Slide adapted from Mark Stamp

FEAL-4 Linear Attack

- From the last slide,
- $(C_L \oplus P_L \oplus P_R \oplus K_4)[23,29] = (P_L \oplus Y_0 \oplus K_1)[31] \oplus (C_L \oplus C_R \oplus K_5 \oplus K_3)[31]$, so
- $K_4[23,29] \oplus (K_1 \oplus K_5 \oplus K_3)[31] =$

  $(C_L \oplus P_L \oplus P_R)[23,29] \oplus P_L[31] \oplus Y_0[31] \oplus (C_L \oplus C_R)[31] =$

  $(C_L \oplus P_L \oplus P_R)[23,29] \oplus P_L[31] \oplus (C_L \oplus C_R)[31] \oplus F(P_L \oplus P_R \oplus K_0)[31]$
- The left hand side is a constant for fixed key.
- The attack consists of guessing K_0 and computing

  $h(P,C) = (C_L \oplus P_L \oplus P_R)[23,29] \oplus P_L[31] \oplus (C_L \oplus C_R)[31] \oplus F(P_L \oplus P_R \oplus K_0)[31]$

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

FEAL-4 Linear Attack - Improvement

- Possible to improve on linear attack
  - Put K_0’= $((K_0)_{0,\ldots,7} \oplus (K_0)_{8,\ldots,15}, (K_0)_{16,\ldots,23} \oplus (K_0)_{24,\ldots,31})$
  - Consider reduced cipher to get a new relation
  - h’(P,C)= $(C_L \oplus P_L \oplus P_R)[5,13,21] \oplus P_L[15] \oplus (C_L \oplus C_R)[15] \oplus F(P_L \oplus P_R \oplus K_0)[15]$
  - h’(P,C) depends only on bits 0,9,…,15,17,…,23 of K_0
  - Find these 12 bits of K_0 first, then the remaining 20 can be found using similar approximations and exhaustive search.

DESX and whitening

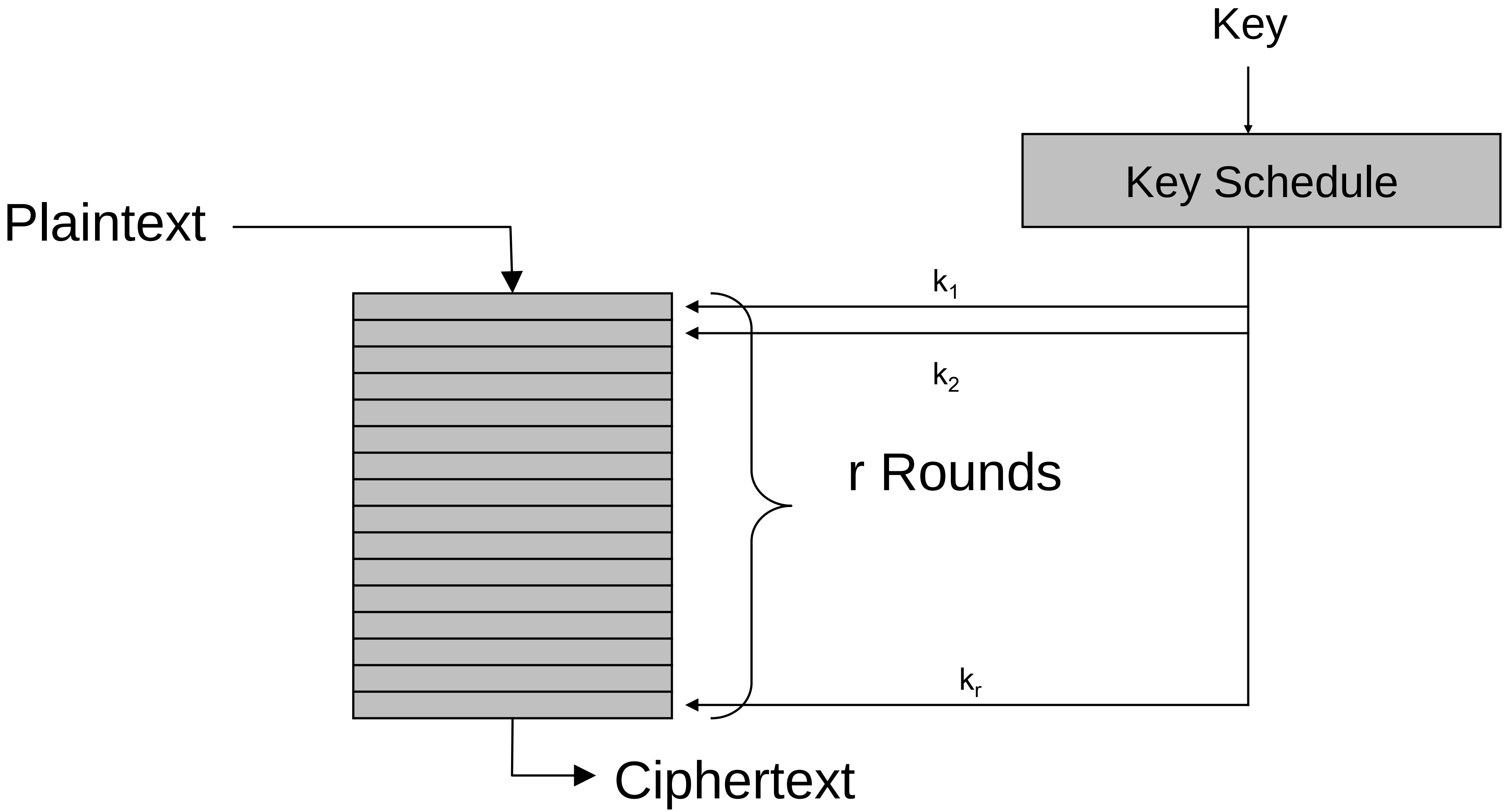
- Attacks like differential and linear cryptanalysis are easier since we can direct observe the input to the first round and output of the last round directly.
- Rivest and Killian:
- $\text{DESX}(k_1,k_2,k_3,x)= k_3\oplus\text{DES}(k_1, k_2\oplus x)$
- Strategy adopted by almost all the AES participants.

AES History

- Call for DES successor 1/97
- Nine Submissions
  - CAST-256, CRYPTON, DEAL, DFC (cipher), E2, FROG, HPC, LOKI97, MAGENTA, MARS, RC6, Rijndael, SAFER+, Serpent, and Twofish.
- Finalists
  - MARS, RC6, Rijndael, Serpent, and Twofish
- And
- the winner is Rijndael: FIPS 197 published 11/2001

- Good References:
  - Daemen and Rijimen, The Design of Rijndael. Springer.
  - Ferguson et. al., The Twofish Encryption Algorithm. Wiley.
  - Tons of contemporaneous material, thesis, etc. Almost all on WWW.

AES



AES Requirements

- 128, 192, 256 bit keys
- Algorithms will be judged on the following factors:
  - Actual security of the algorithm compared to other submitted algorithms (at the same key and block size).
  - The extent to which the algorithm output is indistinguishable from a random permutation on the input block.
  - Soundness of the mathematical basis for the algorithm’s security.
  - Other security factors raised by the public during the evaluation process, including any attacks which demonstrate that the actual security of the algorithm is less than the strength claimed by the submitter.
  - Claimed attacks will be evaluated for practicality.
- Key agility (NSA): “Two blocks encrypted with two different keys should not take much more time than two blocks encrypted with the same key.

End