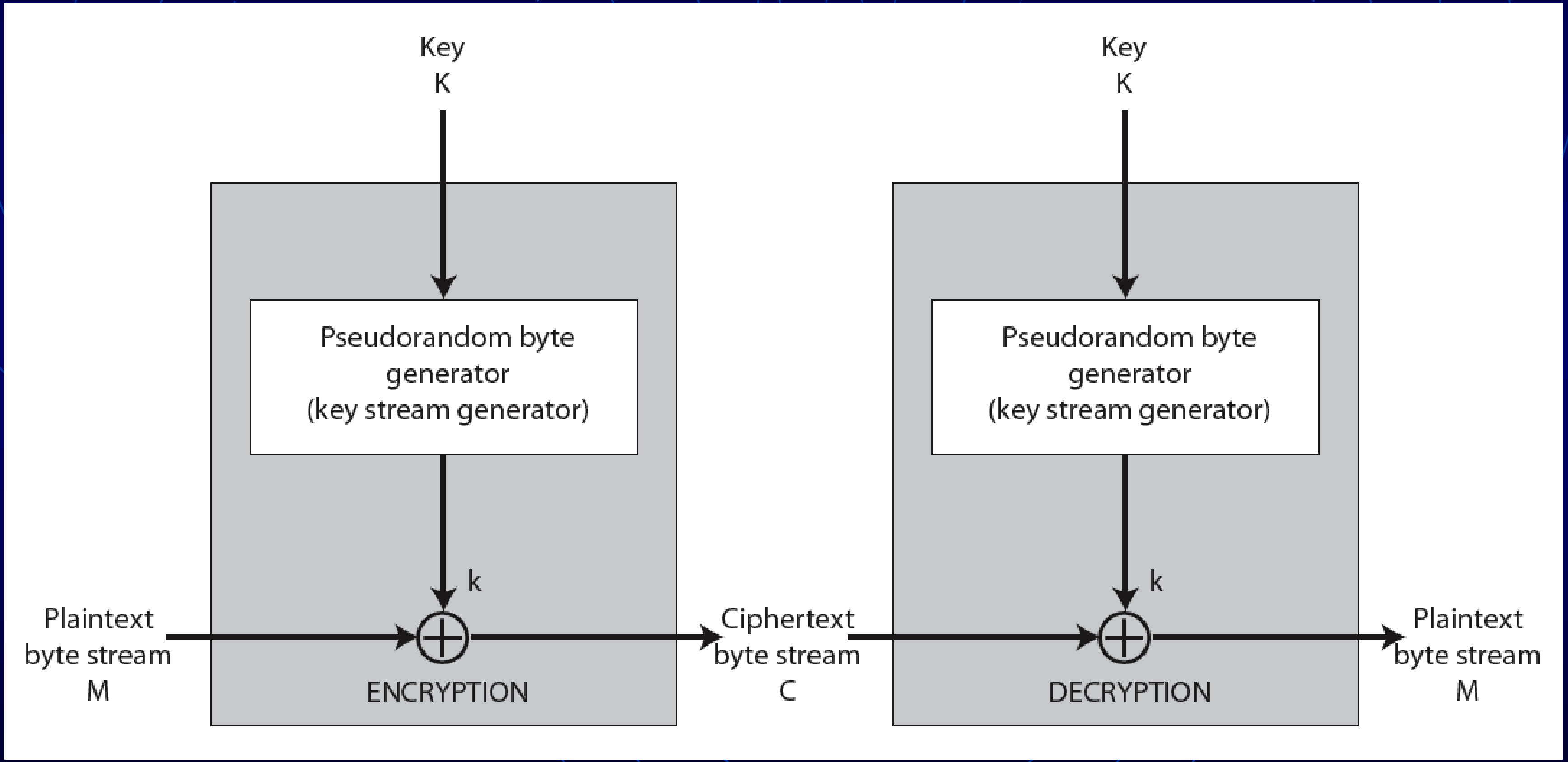
Stream Ciphers

Stream Ciphers

- Process message bit by bit (as a stream)
- Have a pseudo random **keystream**
- Combined (XOR) with plaintext bit by bit
- Randomness of **stream key** completely destroys statistically properties in message
  - $C_i = M_i \text{ XOR } StreamKey_i$
- But must never reuse stream key
  - otherwise can recover messages

Stream Cipher Structure



Stream Cipher Properties

- Some design considerations are:
  - long period with no repetitions
  - statistically random
  - depends on large enough key
  - large linear complexity
- Properly designed, can be as secure as a block cipher with same size key
- But usually simpler & faster

RC4

- A proprietary cipher owned by RSA DSI
- Another Ron Rivest design, simple but effective
- Variable key size, byte-oriented stream cipher
- Widely used (web SSL/TLS, wireless WEP/WPA)
- Key forms random permutation of all 8-bit values
- Uses that permutation to scramble input info processed a byte at a time

RC4 Key Schedule

- Starts with an array S of numbers: 0..255
- use key to well and truly shuffle
- S forms **internal state** of the cipher

```
S initialization
for i = 0 to 255 do
   S[i] = i
   T[i] = K[i mod keylen])// extend key
S Permutation
   j = 0
   for i = 0 to 255 do
   j = (j + S[i] + T[i]) (mod 256)
   swap (S[i], S[j])
```

RC4 Encryption

- Encryption continues shuffling array values
- Sum of shuffled pair selects "stream key" value from permutation
- XOR S[t] with next byte of message to en/decrypt

```
i = j = 0
for each message byte Mi
    i = (i + 1) (mod 256)          // rotate thru S
    j = (j + S[i]) (mod 256)       // update swap location
    swap(S[i], S[j])               // update permutation S
    t = (S[i] + S[j]) (mod 256)    // pick key index
    Ci = Mi XOR S[t]               // encrypt
```

RC4 Overview

initialize permutation

shuffle permutation

extend key

update permutation

RC4 Example

- Plaintext P = [1,2,2,2]
- K = [1,3]
- S = [1,2,3,4]

  0 1 2 3 (index)
- T = [1,3,1,3]

  0 1 2 3 (index)

```
for i = 0 to 3 do
    S[i] = i
    T[i] = K[i mod keylen])
    j = 0
for i = 0 to 3 do
    j = (j + S[i] + T[i])(mod 4)
    swap (S[i], S[j])
```

RC4 Example

j = 0; i = 0

    j = (j + S[0] + T[0])(mod 4)

    j = (0 + 1 + 1)(mod 4)

    j = 2 mod 4 = 2

swap (S[0], S[2])

    S = [3,2,1,4]

        0 1 2 3 (index)

j = 3; i = 2

    j = (j + S[2] + T[2])(mod 4)

    j = (3 + 1 + 1)(mod 4)

    j = 5 mod 4 = 1

swap (S[2], S[1])

    S = [3,1,4,2]

        0 1 2 3 (index)

j = 2; i = 1

    j = (j + S[1] + T[1])(mod 4)

    j = (2 + 2 + 3)(mod 4)

    j = 7 mod 4 = 3

swap (S[1], S[3])

    S = [3,4,1,2]

        0 1 2 3 (index)

j = 1; i = 3

    j = (j + S[3] + T[3])(mod 4)

    j = (1 + 2 + 3)(mod 4)

    j = 6 mod 4 = 2

swap (S[3], S[2])

    S = [3,1,2,4]

        0 1 2 3 (index)

RC4 Encryption Example

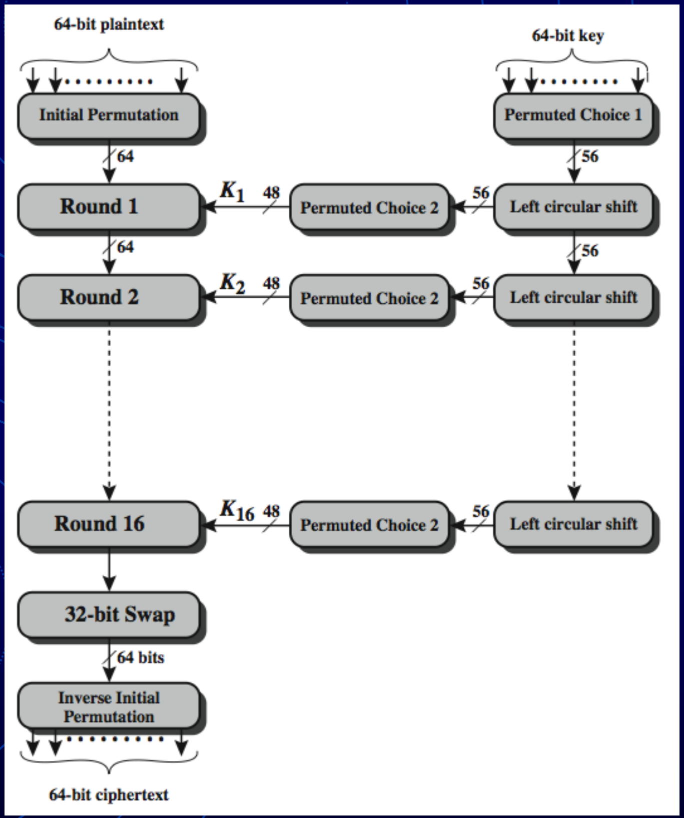
```
S = [3,1,2,4]
     0 1 2 3 (index)
i = j = 0
for each message byte Mi
    i = (i + 1) (mod 4)
    j = (j + S[i]) (mod 4)
    swap(S[i], S[j])
    t = (S[i] + S[j]) (mod 4)
    Ci = Mi XOR S[t]
```

```
P = [1,2,2,2]
     0 1 2 3 (index)
i = j = 0
for each message byte M0
i = (0 + 1) (mod 4) = 1
    j = (0 + S[1]) (mod 4)
    j = (0 + 1) (mod 4) = 1
    swap(S[1], S[1])
 S = [3,1,2,4]
      0 1 2 3 (index)
    t = (S[1] + S[1]) (mod 4)
    t = (1 + 1) (mod 4) = 2
    C0 = M0 XOR S[1]
    C0 = 1 XOR 1
    C0 = 0001 XOR 0001 = 0000
```

Differential Cryptanalysis

Introduction

- **Differential Cryptanalysis** can be successfully used to cryptanalyse the DES with an effort on the order of 2^{47} encryptions, requiring 2^{47} **chosen plaintexts**.
- 2^{47} is certainly significantly less than 2^{56}
- DES key length is 56 bits, there are 2^{56} possible keys, which is approximately 7.2*1016 keys. Thus a **brute-force attack** appeared impractical.



Key Size (bits)	Cipher	Number of Alternative Keys	Time Required at 10^9 Decryptions/s
56	DES	$2^{56} \approx 7.2 \times 10^{16}$	2^{55} ns = 1.125 years

- **Differential cryptanalysis** :-

1. In an ideally randomizing cipher, the probability that a particular output difference ΔY occurs given a particular input difference ΔX is $1/2^n$ where n is the number of bits of X.
2. Non-random changes to the ciphertext may signify a weakness in the encryption scheme.
3. Attacker may gain information about what was encrypted or how it was encrypted by monitoring data changes.

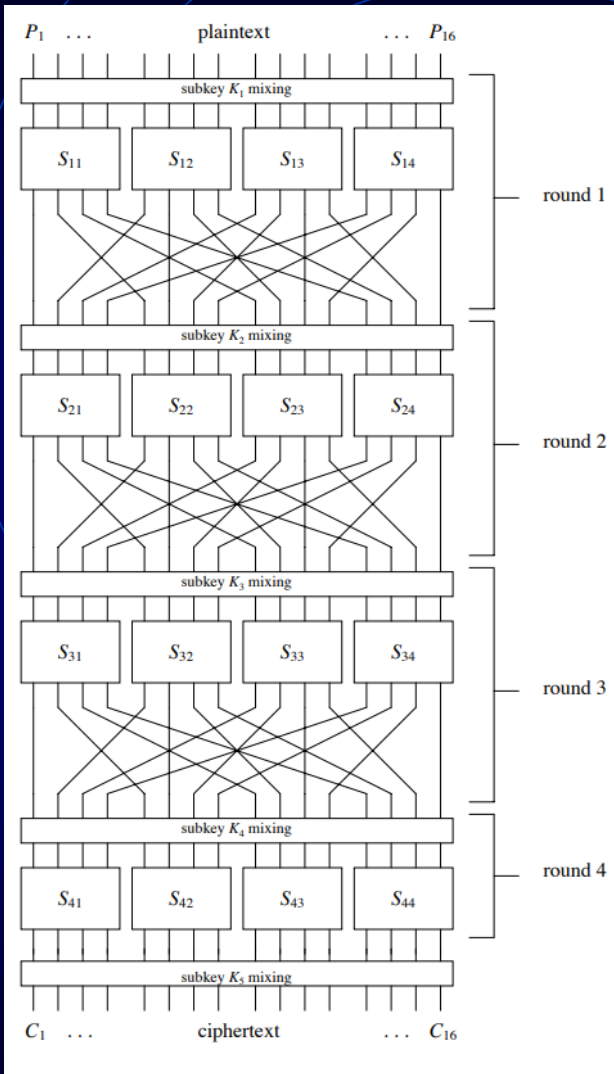
consider a system with input X = [X1 X2 ... Xn] and output Y = [Y1 Y2 ... Yn]

Differential Cryptanalysis

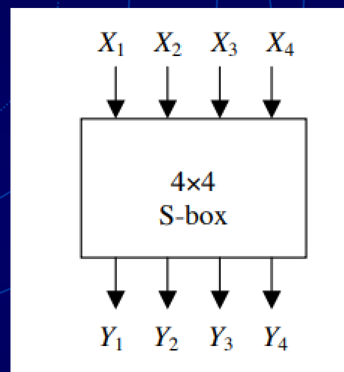
Differential cryptanalysis

- Differential cryptanalysis seeks to exploit a scenario where a particular Δ Y occurs given a particular input difference Δ X with a very high probability P_D (i.e., much greater than 1/2n).
- The pair (Δ X, Δ Y) is referred to as a differential.
- Differential cryptanalysis is a chosen plaintext attack, meaning that the attacker is able to select inputs and examine outputs in an attempt to derive the key.
- For differential cryptanalysis, the attacker will select pairs of inputs, X_1 *and* X_2 , to satisfy a particular Δ X, knowing that for that Δ X value, a particular Δ Y value occurs with high probability.

A basic Substitution-Permutation Network (SPN).



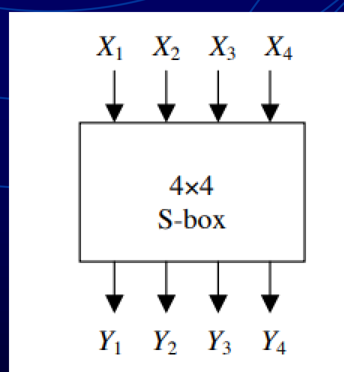
input	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
output	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7



X_1	X_2	X_3	X_4	Y_1	Y_2	Y_3	Y_4
0	0	0	0	1	1	1	0
0	0	0	1	0	1	0	0
0	0	1	0	1	1	0	1
0	0	1	1	0	0	0	1
0	1	0	0	0	0	1	0
0	1	0	1	1	1	1	1
0	1	1	0	1	0	1	1
0	1	1	1	1	0	0	0
1	0	0	0	0	0	1	1
1	0	0	1	1	0	1	0
1	0	1	0	0	1	1	0
1	0	1	1	1	1	0	0
1	1	0	0	0	1	0	1
1	1	0	1	1	0	0	1
1	1	1	0	0	0	0	0
1	1	1	1	0	1	1	1

Distribution of the S-Box output XOR for the input XOR = 1011

X	Y
0000	1110
0001	0100
0010	1101
0011	0001
0100	0010
0101	1111
0110	1011
0111	1000
1000	0011
1001	1010
1010	0110
1011	1100
1100	0101
1101	1001
1110	0000
1111	0111



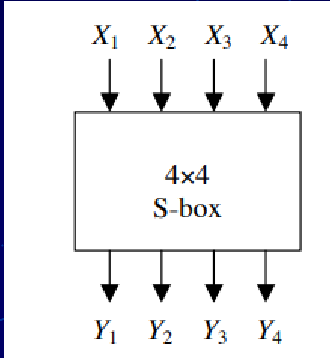
x	x^*	y	y^*	y'
0000	1011	1110	1100	0010
0001	1010	0100	0110	0010
0010	1001	1101	1010	0111
0011	1000	0001	0011	0010
0100	1111	0010	0111	0101
0101	1110	1111	0000	1111
0110	1101	1011	1001	0010
0111	1100	1000	0101	1101
1000	0011	0011	0001	0010
1001	0010	1010	1101	0111
1010	0001	0110	0100	0010
1011	0000	1100	1110	0010
1100	0111	0101	1000	1101
1101	0110	1001	1011	0010
1110	0101	0000	1111	1111
1111	0100	0111	0010	0101

considering input pairs (X', X'') such that $X' \oplus X'' = \Delta X$

$(X', X''=X' \oplus \Delta X)$

ΔX values of 1011 (hex B), 1000 (hex 8), and 0100 (hex 4)

X	Y
0000	1110
0001	0100
0010	1101
0011	0001
0100	0010
0101	1111
0110	1011
0111	1000
1000	0011
1001	1010
1010	0110
1011	1100
1100	0101
1101	1001
1110	0000
1111	0111



ΔY		
$\Delta X = 1011$	$\Delta X = 1000$	$\Delta X = 0100$
0010	1101	1100
0010	1110	1011
0111	0101	0110
0010	1011	1001
0101	0111	1100
1111	0110	1011
0010	1011	0110
1101	1111	1001
0010	1101	0110
0111	1110	0011
0010	0101	0110
0010	1011	1011
1101	0111	0110
0010	0110	0011
1111	1011	0110
0101	1111	1011

$\Delta Y = 0010$ for $\Delta X = 1011$ is 8 out of 16 possible values (i.e., a probability of 8/16)

$\Delta Y = 1011$ given $\Delta X = 1000$ is 4 out of 16

$\Delta Y = 1010$ given $\Delta X = 0100$ is 0 out of 16.

If the S-box could be "ideal" the number of occurrences of difference pair values would all be 1 to give a probability of 1/16 of the occurrence of a particular ΔY value given ΔX.

Input Difference		Output Difference															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	0	0	2	0	0	0	2	0	2	4	0	4	2	0	0
	2	0	0	0	2	0	6	2	2	0	2	0	0	0	0	2	0
	3	0	0	2	0	2	0	0	0	0	4	2	0	2	0	0	4
	4	0	0	0	2	0	0	6	0	0	2	0	4	2	0	0	0
	5	0	4	0	0	0	2	2	0	0	0	4	0	2	0	0	2
	6	0	0	0	4	0	4	0	0	0	0	0	0	2	2	2	2
	7	0	0	2	2	2	0	2	0	0	2	2	0	0	0	0	4
	8	0	0	0	0	0	0	2	2	0	0	0	4	0	4	2	2
	9	0	2	0	0	2	0	0	4	2	0	2	2	2	0	0	0
	A	0	2	2	0	0	0	0	0	6	0	0	2	0	0	4	0
	B	0	0	8	0	0	2	0	2	0	0	0	0	0	2	0	2
	C	0	2	0	0	2	2	2	0	0	0	0	2	0	6	0	0
	D	0	4	0	0	0	0	0	4	2	0	2	0	2	0	2	0
	E	0	0	2	4	2	0	0	0	6	0	0	0	0	0	2	0
	F	0	2	0	0	6	0	0	0	0	4	0	2	0	0	2	0

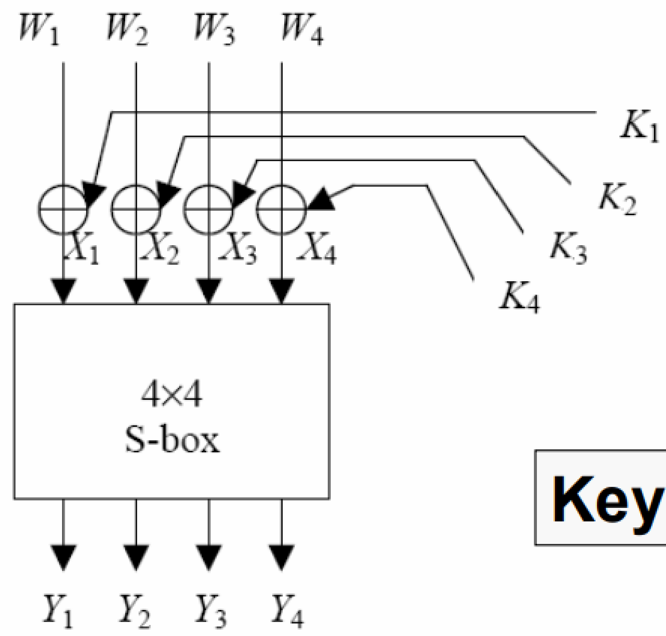
S_{12}: $\Delta X = \mathrm{B} \rightarrow \Delta Y = 2$ with probability 8/16

S_{23}: $\Delta X = 4 \rightarrow \Delta Y = 6$ with probability 6/16

S_{32}: $\Delta X = 2 \rightarrow \Delta Y = 5$ with probability 6/16

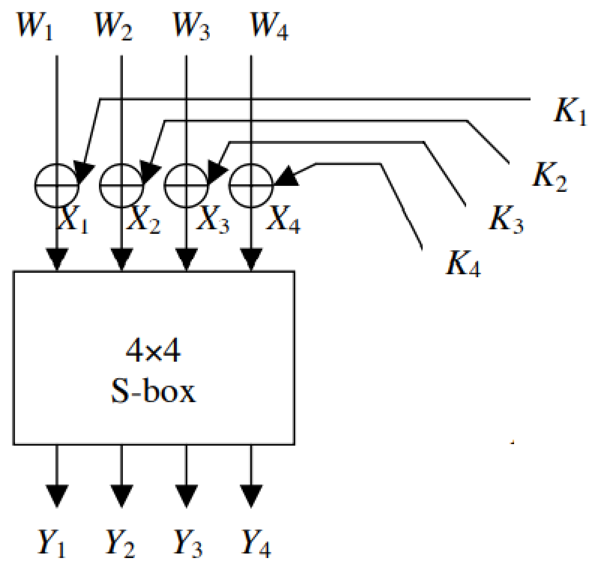
S_{33}: $\Delta X = 2 \rightarrow \Delta Y = 5$ with probability 6/16

Effect of the key on the Differential



Keyed S-Box

- **The Key has no effect on the XOR because it is mixed using XOR function, which is also used to compute the XOR**



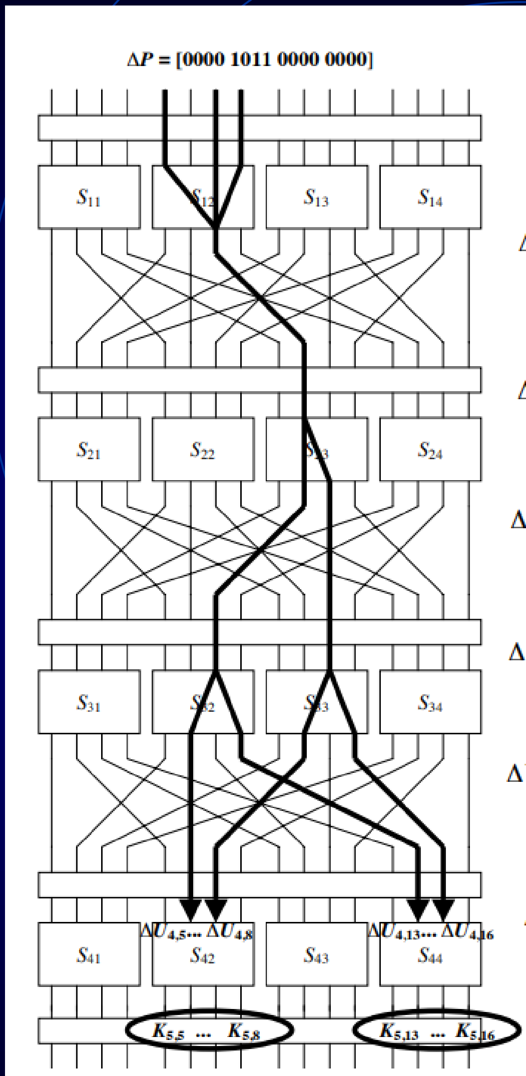
$$\Delta W = [W_1' \oplus W_1'' \quad W_2' \oplus W_2'' \quad \ldots \quad W_n' \oplus W_n'']$$

where $W' = [W_1' \ W_2' \ldots W_n']$ and $W'' = [W_1'' \ W_2'' \ldots W_n'']$ represent the two input values.

Since the key bits remain the same for both W' and W'',

$$\begin{aligned} \Delta W_i &= W_i' \oplus W_i'' = (X_i' \oplus K_i) \oplus (X_i'' \oplus K_i) \\ &= X_i' \oplus X_i'' = \Delta X_i \end{aligned}$$

since $K_i \oplus K_i = 0$.



$\Delta P = \Delta U_1 =$ [0000 1011 0000 0000]

$\Delta V_1 =$ [0000 0010 0000 0000]

$\Delta U_2 =$ [0000 0000 0100 0000]

$\Delta V_2 =$ [0000 0000 0110 0000]

$\Delta U_3 =$ [0000 0010 0010 0000]

$\Delta V_3 =$ [0000 0101 0101 0000]

$\Delta U_4 =$ [0000 0110 0000 0110]

- In S_2^1, $R_p(1011, 0010) = 1/2$
- In S_3^2, $R_p(0100, 0110) = 3/8$
- In S_2^3, $R_p(0010, 0101) = 3/8$
- In S_3^3, $R_p(0010, 0101) = 3/8$

$$R_p(\texttt{0000 1011 0000 0000}, \texttt{0000 0101 0101 0000}) = \frac{1}{2} \times \left(\frac{3}{8}\right)^3 = \frac{27}{1024}.$$

Hence it follows that
if x'=0000 1011 0000 0000,
then
(u4)'=0000 0110 0000 0110
with a probability of 27/1024.

Linear Cryptanalysis

- Obtain linear approximation(s) of the cipher relating P,K,C

$$\oplus_{i\in X,}\ p_i \oplus_{j\in Y} c_j = \oplus_{g\in Z}\ k_g$$

which occur with probability pr = ½ + e for

max bias $-½ \leq e_i \leq ½$.

- Encrypt random P's to obtain C's and compute k_g's.
- The attacker has a lot of plaintext-ciphertext pairs **(known plaintext attacks)**.

Linear Cryptanalysis

The Piling-up

- Suppose X_1, X_2,… are independent random variables from {0,1}. And

$$\Pr[X_i = 0] = p_i\,, \qquad i = 1,2,\ldots \text{ Hence,}$$
$$\Pr[X_i = 1] = 1 - p_i\,, \quad i = 1,2,\ldots$$

- The independence of X_i, X_j implies

$$\Pr[X_i = 0, X_j = 0] = p_i p_j$$
$$\Pr[X_i = 0, X_j = 1] = p_i(1 - p_j)$$
$$\Pr[X_i = 1, X_j = 0] = (1 - p_i) p_j$$
$$\Pr[X_i = 1, X_j = 1] = (1 - p_i)(1 - p_j)$$

Linear Cryptanalysis

- Now consider $X_i \oplus X_j$

$$\Pr[X_i \oplus X_j = 0] = p_i p_j + (1 - p_i)(1 - p_j)$$
$$\Pr[X_i \oplus X_j = 1] = p_i(1 - p_j) + (1 - p_i) p_j$$

- The **bias** of X_i is defined to be the quantity

$$\varepsilon_i = p_i - \tfrac{1}{2}$$

- And we have

$$-\tfrac{1}{2} \le \varepsilon_i \le \tfrac{1}{2},$$
$$\Pr[X_i = 0] = \tfrac{1}{2} + \varepsilon_i \,,$$
$$\Pr[X_i = 1] = \tfrac{1}{2} - \varepsilon_i \,.$$

Linear Cryptanalysis

Linear Approximations of S-boxes

- Consider an S-box $\pi_S : \{0,1\}^m \to \{0,1\}^n$
- Let the input m-tuple be $X=(x_1,\ldots,x_m)$. And the output n-tuple be $Y=(y_1,\ldots,y_n)$.
- We can see that

$$\Pr[X_1 = x_1,\ldots, X_m = x_m, Y_1 = y_1,\ldots, Y_n = y_n] = 0$$

if $(y_1,\ldots, y_n) \neq \pi_S(x_1,\ldots, x_m)$; and

$$\Pr[X_1 = x_1,\ldots, X_m = x_m, Y_1 = y_1,\ldots, Y_n = y_n] = 2^{-m}$$

if $(y_1,\ldots, y_n) = \pi_S(x_1,\ldots, x_m)$.

- Now we can compute the bias of the form

$$X_{i_1} \oplus \cdots \oplus X_{i_k} \oplus Y_{j_1} \oplus \cdots \oplus Y_{j_l}$$

using the formulas stated above.

Linear Cryptanalysis

- We use the S-box .

X_1	X_2	X_3	X_4	Y_1	Y_2	Y_3	Y_4	$X_2 \oplus X_3$	$Y_1 \oplus Y_3 \oplus Y_4$	$X_1 \oplus X_4$	Y_2	$X_3 \oplus X_4$	$Y_1 \oplus Y_4$
0	0	0	0	1	1	1	0	0	0	0	1	0	1
0	0	0	1	0	1	0	0	0	0	1	1	1	0
0	0	1	0	1	1	0	1	1	0	0	1	1	0
0	0	1	1	0	0	0	1	1	1	1	0	0	1
0	1	0	0	0	0	1	0	1	1	0	0	0	0
0	1	0	1	1	1	1	1	1	1	1	1	1	0
0	1	1	0	1	0	1	1	0	1	0	0	1	0
0	1	1	1	1	0	0	0	0	1	1	0	0	1
1	0	0	0	0	0	1	1	0	0	1	0	0	1
1	0	0	1	1	0	1	0	0	0	0	0	1	1
1	0	1	0	0	1	1	0	1	1	1	1	1	0
1	0	1	1	1	1	0	0	1	1	0	1	0	1
1	1	0	0	0	1	0	1	1	1	1	1	0	1
1	1	0	1	1	0	0	1	1	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1	0	1	0
1	1	1	1	0	1	1	1	0	0	0	1	0	1

Linear Cryptanalysis

- Consider $X_1 \oplus X_4 \oplus Y_2$ The probability that $X_1 \oplus X_4 \oplus Y_2 = 0$ can be determined by counting the number of rows in which $X_1 \oplus X_4 \oplus Y_2 = 0$, and then dividing by 16.
- It is seen that

  $$\Pr[X_1 \oplus X_4 \oplus Y_2 = 0] = \tfrac{1}{2}$$

  Hence, the bias is 0.
- If we instead analyze $X_3 \oplus X_4 \oplus Y_1 \oplus Y_4$, we find that the bias is –3/8.

□