$S$ → sample space

$x \in S$ → Sample pt.

event- $E$

Card → $S = \{52\}$

$Prob(E) = \frac{m}{n}$ → event
$n$ → Total no. of elem

$\frac{M}{n}$ → $M$ → event occurring $M$ times

Sure event → $P(S) = 1$

Impossible

$0 \leq P(E) \leq 1$

Complementary

Add

$P(E \cup F) = P(E) + P(F) - P(E \cap F)$

Exclusive

$P(E \cup F) = P(E) + P(F)$

$\sum P(E) = 1$

multiplicative

$P(E \cap F) = P(F|E) \cdot P(E)$
$= P(E|F) \cdot P(F)$

$P(E|F) = \frac{P(E \cap F)}{P(F)}$

$P(E|F) = P(E)$

Independent

Binomial

$P(S) = p \quad P(F) = 1 - p$

$P(k \text{ success in } n \text{ trials}) = \binom{n}{k} p^k (1-p)^{n-k}$

Secure use

$k \geq m$ key used only on

$k \in K$ length $k \rightarrow l$

$L_i \leftarrow R_{i-1}$

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

$k_i \rightarrow$ round key $\rightarrow$ 48 bit- subarray

$f \rightarrow$ S-box flow

Symmetric Cipher - DES

* Feistal block Cipher

Stream Cipher → one bit / one byte encrypts

eg:- Vigenere, Vernam

One-time pad - Keystream - (Ki) long as plain text

* bit stream generator

block cipher → treated as whole

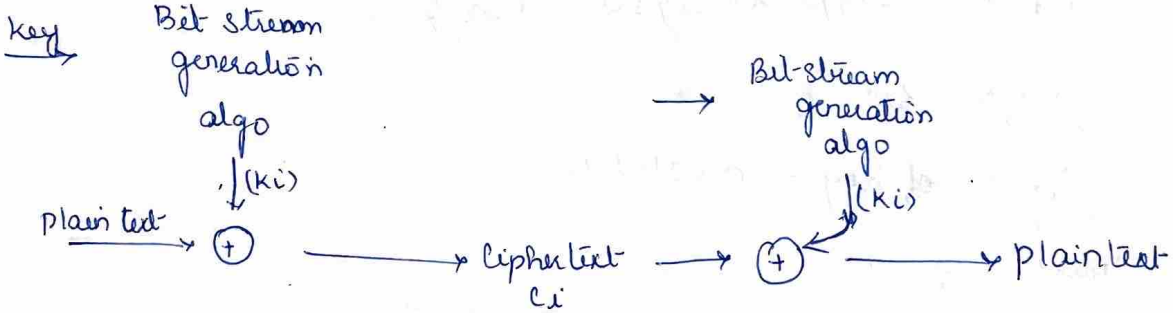
64 (or) 128 bits

broad appln

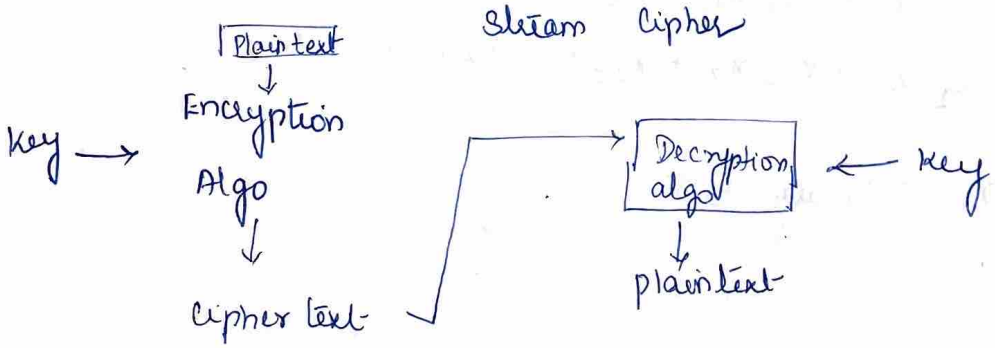
Motivation of Feistal Cipher structure

plain text n bits ⇒ Ciphertext n bits

$2^n$ different plaintext blocks → unique cipher text




Stream Cipher




* Reversible mapping / Irreversible mapping

$2^n!$ transformation

n - bit block substitution

Classical sub cipher for $n=4$

4 bit input produces → 16 possible o/p state

| plaintext | Ciphertext |
|---|---|
| 0000 | 1110 |
| 0001 | 0100 |
| 0010 | 1101 |
| 0011 | 0001 |
| 0100 | 0010 |
| ⋮ | |
| 1111 | 0111 |

Feistal refers → ideal block cipher

if small block size, $n=4$ is used → classical sub. cipher
vulnerable to statistical analyses

n large → cryptanalysis is difficult

4 x 16 = 64 bits
bit (Row)

length of key → $n \times 2^n$ bits

Linear eqn

$$y_1 = K_{11} x_1 + K_{12} x_2 + K_{13} x_3 + K_{14} x_4$$

$$y_2 = K_{21} x_1 + K_{22} x_2 + K_{23} x_3 + K_{24} x_4$$

$x_i$ → binary bits

$y$ →

Feistal Cipher

block cipher → prdt cipher

2 or more ciphers in sequence

$2^k$ instead $2^n!$

Alternate Substitution & Permutation

Diffusion & Confusion

Claude Shannon → to capture 2 basic building blocks

Diffusion → plaintext ⇒ Ciphertext

$M = m_1 m_2 m_3 \ldots$ characters by arg.

$$y_i = \left(\sum m_{n+i}\right) \bmod 26$$

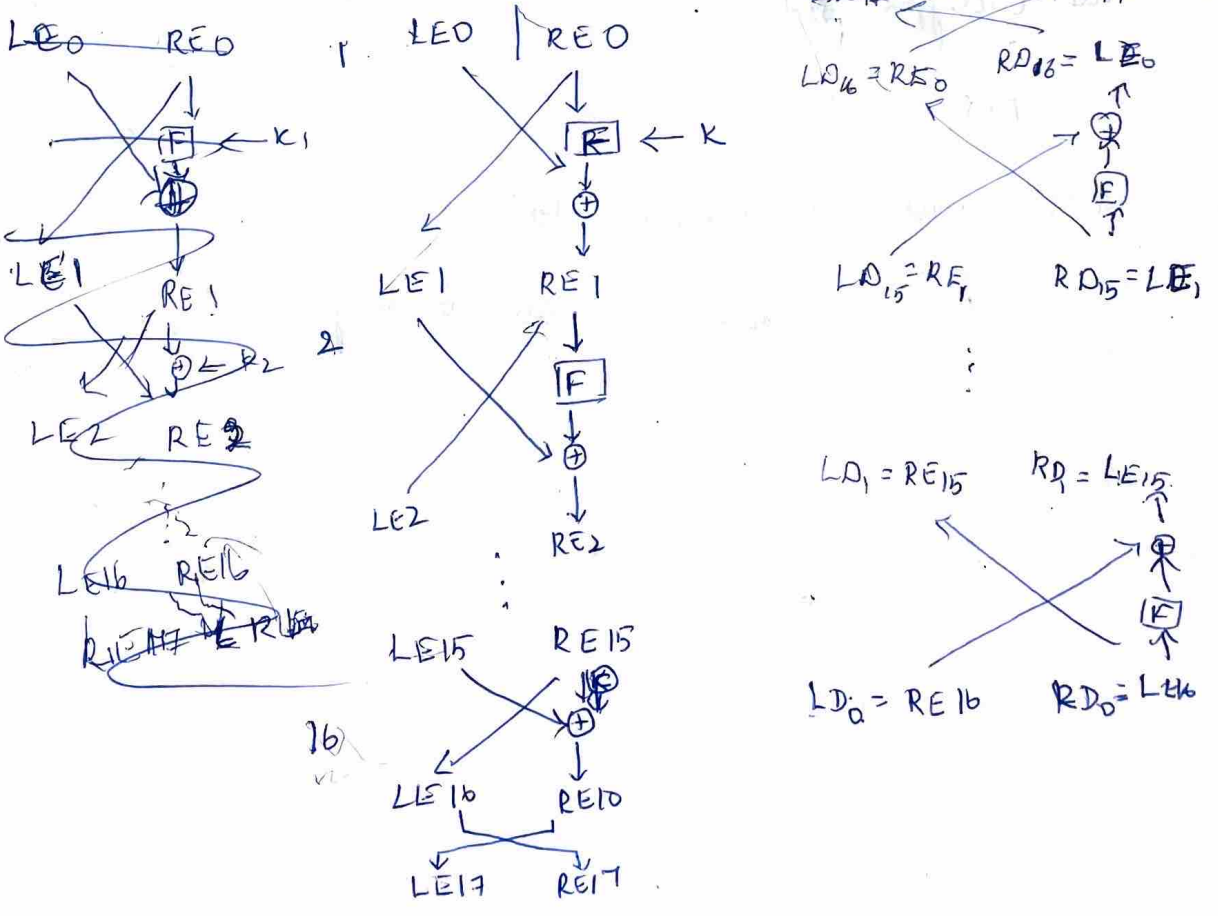
Confusion → Ciphertext & Value of Encryption key - Complex

Feistal Cipher structure

Plain text → 2 ~~bloc~~ halves

LO / RO

n rounds of processing to form ciphertext



Block size → Large block size ↑ Security

New AES - 128 bit

Key size → Large ↑ Security

No. of rounds → multiple rounds

→ 16 rounds

Round fn → R

Fast s/w encryption

Ease of analysis

## Decryption

↳ Use ciphertext as input but subkeys in Reverse Order.

$K_n$ → First round

$K_{n-1}$ → Second round

$$LE_{16} = RE_{16}$$

$$RE_{16} = LE_{15} \oplus F(RE_{15}, K_{16})$$

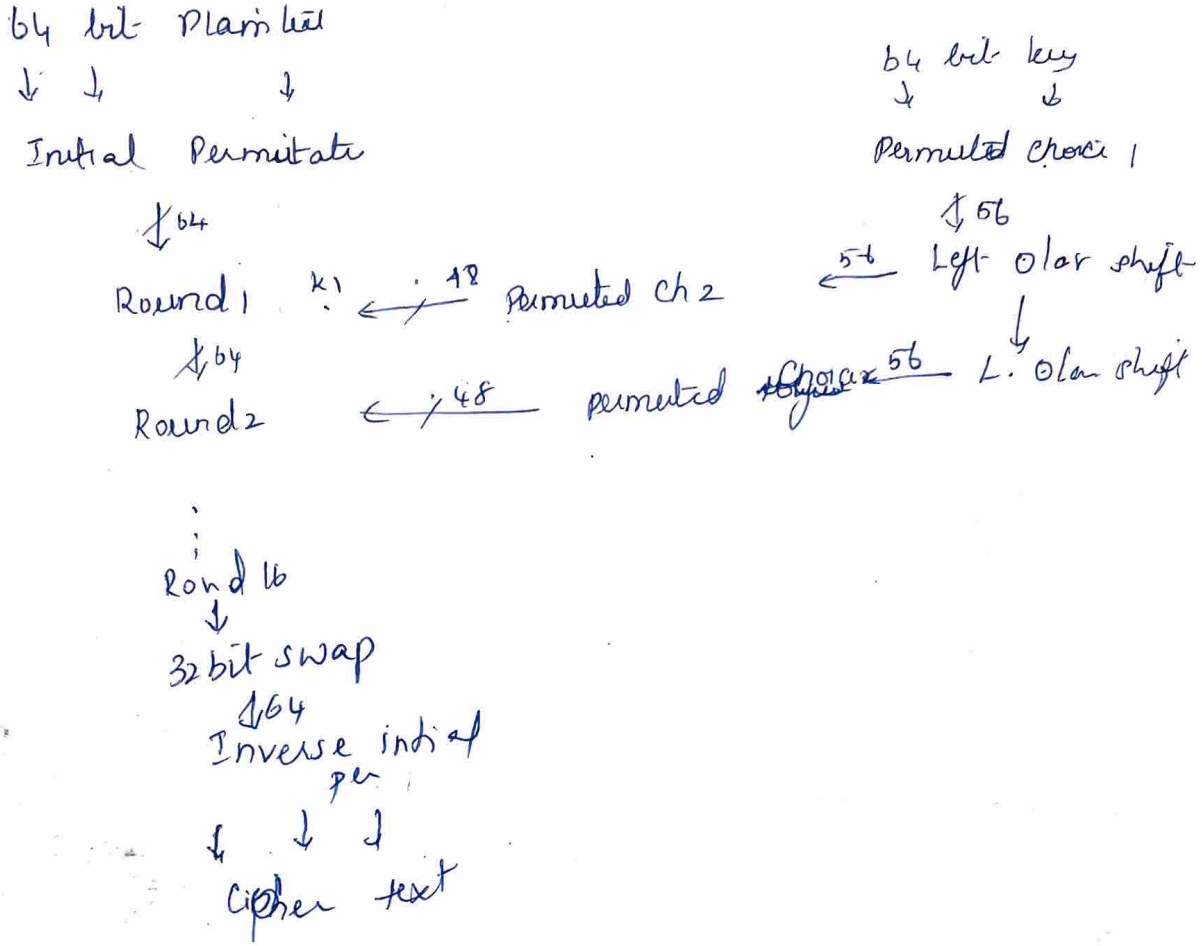
## Data Encryption std

AES - 2001

1977 - National bureau std

64 bit data + 56 bit key

DES Encryption

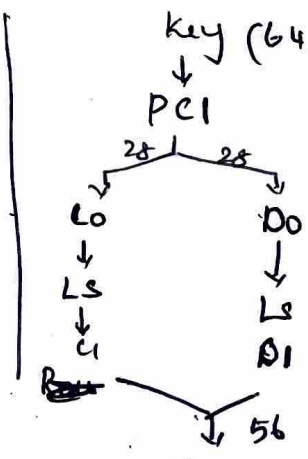



Plaintext → initial permutation
+
permutation (16) rounds + Substitution
Preout
↓
Inverse permutation

→ Key size → 8 parity
8 keys

Each Round { Divide 2 parts
Bit Shuffling
Substitution
Ex-OR

~~Round~~
LCS based on round no.
Round 1, 2, 9, 16
1 Bit shift




8 parity
↳ 8 positions

Random.

Plaintext – 0246bf acee ca 86420

Key – 0f157 1 c 947 d9

Design

No of Rounds

↳ No. of rounds
↳ key schedule
↳ fn f

Design Fn F

---

8 bit → discarded

8, 16, 24, 32, 40, 48, 56, 64 → discarded.

Substitution — Confusion
Diffusion — Transposi

64 bit Plaintext

↓

Initial Perm

↓

Round 1

2

⋮

16

Final Perm

48 Key 56 bit
48 Gen Ci

Round fn

Expansion P box

↓ 48 bit

XOR ⊕

↓

8 boxes

↓ 32 bits

32 P box

6 input

↓

4 o/p

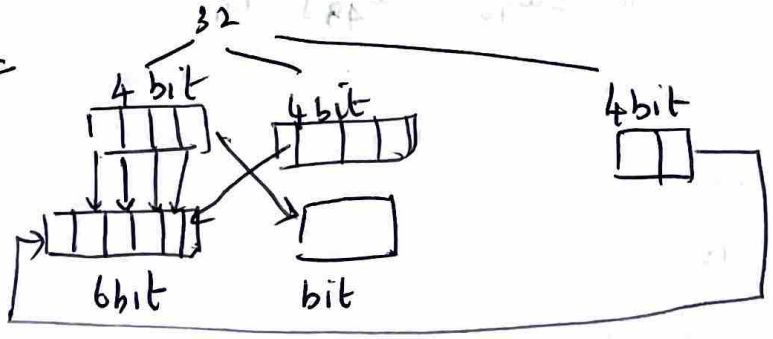
Compression Per

56 bit input

↳ 48 bit

(9, 18, 22, 25, 35, 38, 43, 54 bits)

Expa



AES – Advanced Encryption std.

Plain text – blocks

Block size = 128 bit

Key size = 128 bit

No. of rounds = 10 rounds

words → 32 bit

↓

Processed (4 words) / 16 bytes)

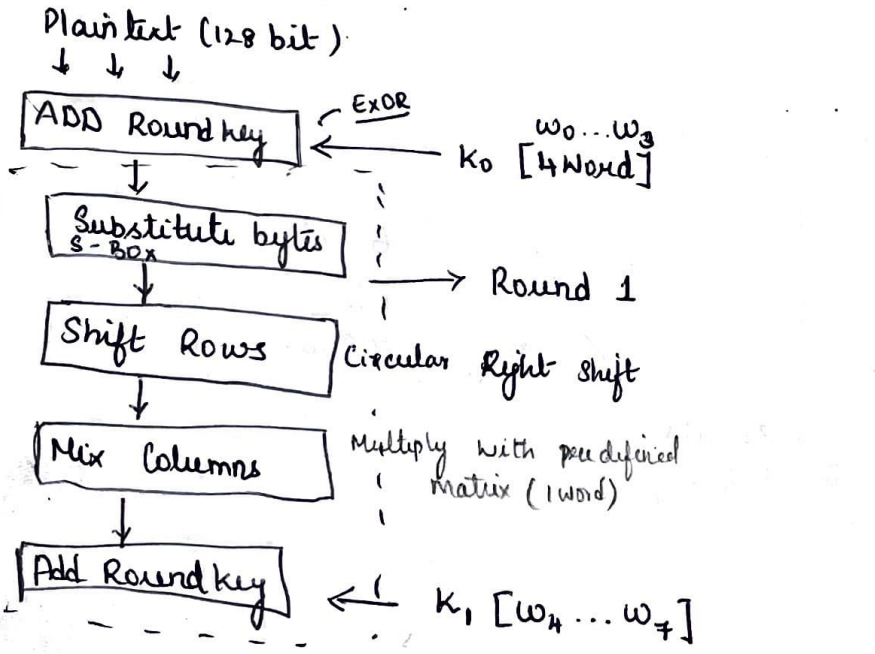
No. of subkeys – 44 sub keys

Each subkey size = 32 bit / 1 word

4 Sub-keys → in each round

Pre round calculation → 4 sub keys (128 bit)

Cipher text = 128 bit




$R_2$ $[w_8 \ldots w_{11}]$ – $K_2$

$R_3$ $[w_{13} \ldots w_{15}]$ – $K_3$

$R_{10}$ $[w_{40} \ldots w_{43}]$ – $K_9$

Round 10

↳ Sub. bytes

↓

Shift Rows

↓

Add round key

Rijndael

128 plaintext - Input array -

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

16 × 8 = 128 bit

Intermediate results → State arrays

8 bit

$$\begin{pmatrix} S_{0,0} & S_{0,1} & S_{0,2} & S_{0,3} \\ S_{1,0} & S_{1,1} & S_{1,2} & S_{1,3} \\ S_{2,0} & S_{2,1} & S_{2,2} & S_{2,3} \\ S_{3,0} & S_{3,1} & S_{3,2} & S_{3,3} \end{pmatrix}$$

8 bit

Word

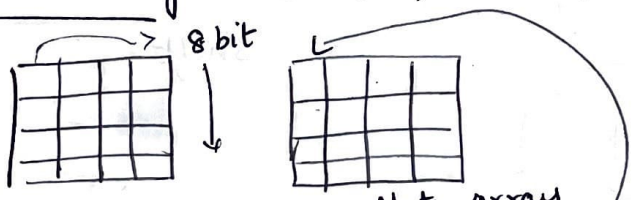
Word

Output array

$$\begin{pmatrix} O_0 & O_4 & O_8 & O_{12} \\ O_1 & O_5 & O_9 & O_{13} \\ O_2 & O_6 & O_{10} & O_{14} \\ O_3 & O_7 & O_{11} & O_{15} \end{pmatrix}$$

Key → $\begin{pmatrix} K_0 & \cdots & K_{12} \\ K_1 & & \vdots \\ K_2 & & \\ K_3 & & K_{15} \end{pmatrix}$ → $W_0 \; W_1 \; W_2 \ldots W_{43}$

Substitution bytes (S-Box)




State array

O/p → 8 bit

First Four → row no → 0 - 15 (F → 15)

Next Four → Colum no. → 0 - 15 F

O/p.

Size ⇒ 16 × 16

eg 0000 0101

0 → 5 Colm

52

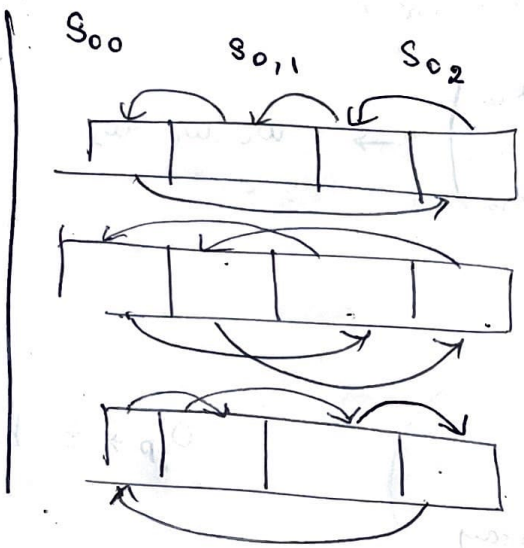
0101 0010 State array

$$\begin{pmatrix} S_{0,0} & S_{0,1} & \cdots & \\ S_{1,0} & S_{1,1} & & \\ S_{2,0} & & & \\ S_{3,0} & & & S_{3,3} \end{pmatrix} \rightarrow \quad S'_{0,0}$$

→ Shift rows.

## Shift Rows

| Row no: | | |
|---|---|---|
| 0 | 0 bits | Olar right shift |
| 1 | 1 bit | |
| 2 | 2 bit | |
| 3 | 3 bits | |




$S_{03}$ → Same.

Shift

## Mix Columns

Predefined matrix

$$\begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} S_{00} \\ S_{10} \\ \cdot \end{bmatrix} = \begin{bmatrix} \\ \\ \\ \end{bmatrix}$$

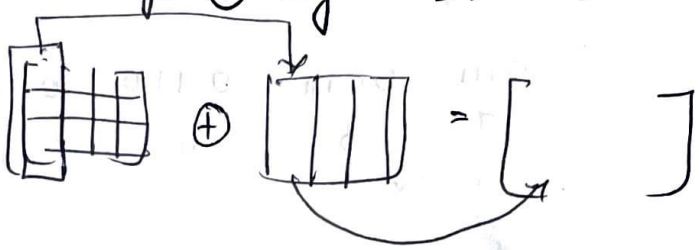
4x1

## Inverse

$$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix}$$

Add round key

Ex-OR (state array) ⊕ key [4 words]



→ AES - Rijndael

block cipher

~~cipher~~key 128, 192, 256

NIST - 1997

DES → Triple DES

(MARS, RC6, Rijndael, Serpent, Twofish)

Features

↳ Sub, Permutation → SP netw

* Single key → multiple rounds

* byte data → (16 bytes)

128 — 10 round

192 — 12 round

256 — 14 rounds

4×4 Matrix → state array

Plaintext 2 1 9 2 ] ⇒ Hex form.

Thats my kung Fu.

⇓

Hex form.

Appln.

Wireless security

Encrypted browser

General file Encryption

## Key Expansion

128 bit → 10 diff keys.

0111 0011 0110 0001

7 3 6 1

73 61 74 69 73 68 63 6a 6g

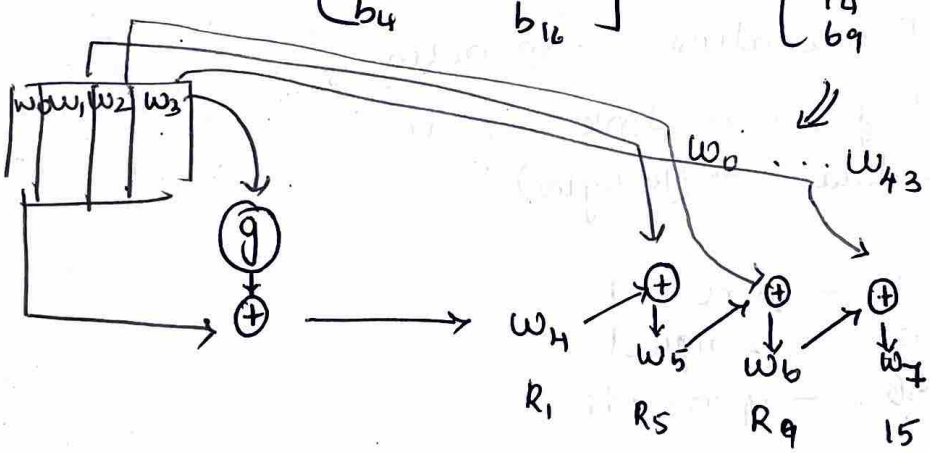
s a t i s h J i _ g

16 bytes

$b_1$ $b_2$ $b_3$ ... $b_{16}$

4x4 matrix

$$\begin{bmatrix} b_1 & & \\ b_2 & \vdots & \\ b_3 & & \\ b_4 & & b_{16} \end{bmatrix} = \begin{bmatrix} 73 & 13 & 69 & 72 \\ 61 & 68 & 73 & 69 \\ 74 & 63 & 73 & 69 \\ 69 & 6a & 62 & 6e \\ & & 6f & 67 \end{bmatrix}$$




$$w_4 = w_0 \oplus g(w_3)$$

72 69 6e 67

Cyclic left shift ⇒ 69 6e 67 72 (Rot word) $(X_1)$

f9 9f 85 40 Sub word $(Y_1)$

byte sub. using Sbox (Aes Sbox)

(XOR) ⇒ Round Const

| $R_1$ | $R_2$ | $R_3$ | | | | | | | $R_{10}$ |
|---|---|---|---|---|---|---|---|---|---|
| 01 | 02 | 04 | 08 | 10 | 20 | 40 | 80 | 1b | 36 |

$Y_1$

$\oplus R_1$

$\Rightarrow g(w_3)$

Compute Rot Word

Sub word

(xor) Round const ⇒ $g(w_3)$

Secret message now

73 65 63 72 65 74 6d 65 73 73 61 67 65 6e 6f 77