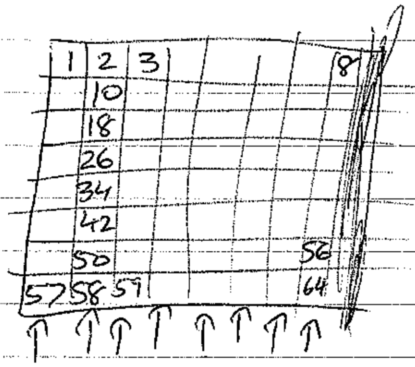
9/26/16

DES TO ENCRYPT

① Apply IP. to Plaintext.

58th bit goes 1st
50th bit goes 2nd
42th bit goes 3rd, etc



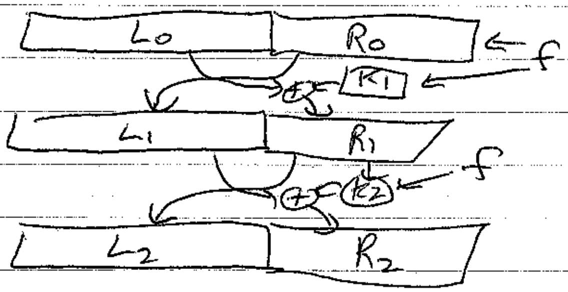
② Run 16 "rounds" of encryption
"feistel rounds"

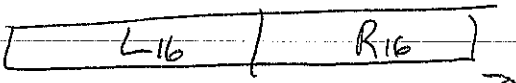
Input L_0R_0, L_0 is left 32 bits
R_0 is right 32 bits

Next = $IP(L_0R_0) = L_0R_0$

```
for (i=1; i<=16; i++) {
    $L_i = R_{i-1}$
    $R_i = L_{i-1} \oplus F(R_{i-1}, K_i)$
              XOR
}
```

key for ~~for~~ round i. 48 bits





③ $C = IP^{-1}(R_{16}L_{16})$ → MAKES DECRYPTION EASIER...

9/26/16

DES

Data Encryption Standard

IBM - Horst Feistel

Private Key // Symmetric

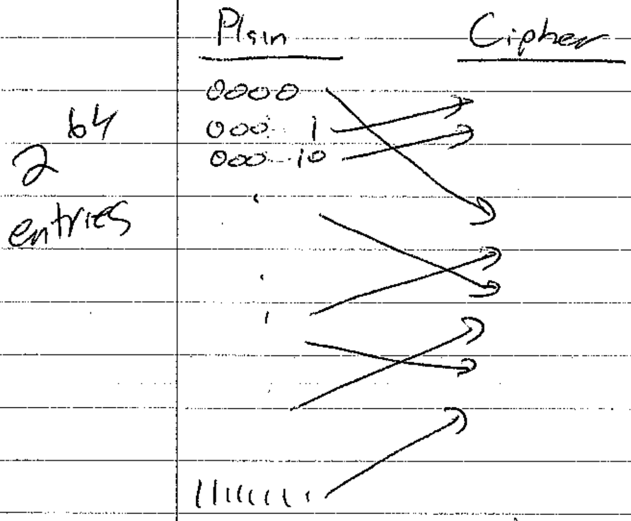
Input: 64 bit blocks
8 bytes

Key: 56 bits

Output: 64 bits

$P = P_1 P_2 P_3 \ldots \quad P_K$, P_i is 64 bits

$C = f(K, P_1) f(K, P_2) f(K, P_3) \ldots$



Some permutation!

Now, we want to focus on $f(K, P)$

After IP, $\overset{\text{orig}}{\text{bit}}$ 58 goes 1st

orig bit 50 goes 2nd

⋮

orig bit 1 goes 40th

IP^{-1}	40	8	48	16	56	24	64	32
	39	7	47	15	55	23	63	31

for speed we hard-code as much as we can beforehand.

EACH ROUND

Calculate a function $F(\underset{\substack{\uparrow \\ \text{Right Half} \\ \text{of previous}}}{\overset{\substack{32 \\ \text{bits}}}{R_{i-1}}}, \underset{\substack{\uparrow \\ \text{Round Key}}}{\overset{\substack{48 \\ \text{bits}}}{K_i}})$

1. $E(R_{i-1})$ - expand 32 bits to 48.

put bit 32 of input 1st

put bit 1 of input 2nd

put bit 2 of input 3rd

⋮

2. Calculate $E(R_{i-1}) \oplus K_i = 48$ bits $(b_1 b_2 b_3 \ldots b_8)$

↓ 6 bit blocks

$$\begin{array}{r} 1001101011 \\ \oplus\ 0010110110 \\ \hline 1011011101 \end{array}$$

bitwise

^ xor in code

9/20/16 ④

3. Use s-boxes to calculate

```
for (i=1; i<=8; i++)
    $c_i = S_i(b_i)$
```

↳ takes 6 bit input
produces 4 bit output

Output = 32 bits

4. $P(c_1 c_2 \ldots c_8)$

9/28/16

$L_i = R_{i-1}$

$R_i = L_{i-1} \underset{XOR}{\oplus} \boxed{F(R_{i-1}, K_i)}$

1. $E(R_{i-1}) \rightarrow$ 48 bits
2. $E(R_{i-1}) \oplus K_i \rightarrow$ 48 bits
3. $\hookrightarrow b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8$ (6 bit blocks

$C_i = S_i(b_i) \quad 1 \le i \le 8$

$\uparrow$ 4 bits $\quad \uparrow$ 6 bits

4. $P(c_1 c_2 .. c_8)$ $\downarrow$ 32 bits

individuals

$d_1 d_2 d_3 d_4 d_5 d_6$ } bits

$b_1 = \underset{d_1 d_2 d_3 d_4 d_5 d_6}{101110}$

row $= d_1 d_6 = 10 = \boxed{2}$

col $= d_2 d_3 d_4 d_5 = 0111 = \boxed{7}$

$S_1(101110) = 11 = \boxed{1011}$

$S_1(111110) = 0 = \boxed{0000}$ row=10 = $\boxed{2}$ col=1111 $\boxed{15}$

$S_1(101010) = 6 = \boxed{0110}$ row = 10 = 2 col = 0101 = 5

$S_1(001110) = 8 = \boxed{1000}$ row = 0 col = 7

Differential Cryptanalysis

→ One goal of NSA requirements was to thwart this type of attack.

Shown you could break DES with 2^{48} pairs of chosen plaintext + corresponding ciphertext.

LATE 90s: DES Challange 9/28/16 (2)(3)

3 months to break a DES key 2^{56}.

$\Longrightarrow$ Distributed Computing, try all keys

$2^{36} \sim 64$ billion

if you can do 10^6 keys / second

$$\frac{6.4\times10^{9}}{10^{6}} = 6.4\times10^{3} = 6{,}400 \text{ seconds}$$

~ 2 hrs

---

if only 10^5 keys/second $\sim \boxed{20 \text{ hrs}}$

DES Key Schedule

9/30/16 ①

Key input 64 bits, 8 of which are odd parity bits.

```
           P
  1001110  1
  0001011  0
```

permuted 56 bits

64 bits w/ parity

1. $K' = PC\text{-}1(K)$

   $K' = \boxed{C_0}\boxed{D_0}$

   28bits 28bits

   $\boxed{57,49,41\ldots 36}$ $\boxed{63,55,\ldots 12,4}$

2. for $(i=1;\ i\le 16;\ i{+}{+})\ \{$

   $C_i = LS_i(C_{i-1}) \cdot D_i = LS_i(D_{i-1})$

   $\boxed{49,41,\ldots\ 36,57}$ $\boxed{55,\ \ldots\ 12,4,63}$

   $K_i = PC\text{-}2(C_i D_i)$

   bit #14 is 10 in round 1
   bit #17 is 51 in round 1

Goal of key Schedule: Use 56 bit input key (w 8 bit parity) to create 16 round keys $K_1, K_2, \ldots K_{16}$ that are each 48 bits

Triple DES

K_1, K_2

$E(E(E(m,K_1),K_2),K_1).$

AES

Advanced Encryption Standard

$\boxed{\text{128 bit key}}$

192 bit key

256 bit key