[CS304] Introduction to Cryptography and Network Security

Course Instructor: Dr. Dibyendu Roy
Scribed by : Pallikonda Sai Teja
Student ID:202011052

Winter 2022-2023
Lecture (Week 03)

1 One Time Padding OTP

→ OTP provides perfect secrecy under some conditions.

Pr[message | ciphertext] = Pr[message]

2 OTP on one bit encryption

m ∈ {0,1} ⇒ Message
Pr[m=0] = p
Pr[m=1] = 1-p

K ∈ {0,1} ⇒ Key
Pr[k=0] = 1/2
Pr[k=1] = 1/2

2.1 Encryption

c = m ⊕ k
c = 0 ⇒ {m = 0,k = 0} ∪ {m = 1,k = 1}
Pr[c = 0] = Pr[m = 0,k = 0] + Pr[m = 1,k = 1]
= p * 1/2 + (1 - p) * 1/2
$= p + 1 - p/2$
= 1/2
Pr[c = 1] = 1 - Pr[c = 0]
= 1 - 1/2
= 1/2
Pr[M = m /C = c] =? Pr[M = m]

$$\Pr[M = 0 / C = 0] = \frac{Pr[M=0 \cap \ C=0]}{Pr[C=0]}$$
$$= \frac{Pr[C=0/M=0] * Pr[M=0]}{1/2}$$
$$= \frac{1/2 * Pr[M=0]}{1/2}$$
$$= \Pr[M = 0]$$

$\Pr[A/B] = \Pr[A \cap B]/\Pr[B]$
$\Pr[A \cap B] = \Pr[B/A] * \Pr[A]$

Pr[M = 0 /C = 0] = Pr[M = 0]

Thus it provides perfect secrecy

Conditions

1. $M_1 \oplus k = c_1$
   $M_2 \oplus k = c_2$ This will reveal information of messages
   $c_1 \oplus c_2 = (M_1 \oplus k) \oplus (M_2 \oplus k)$
   $= c_1 \oplus c_2 = M_1 \oplus M_2$
   Hence cipher text difference will give us message difference

2. Len(k) < Len(P)
   $c = P \oplus k$
   Example :
   32-bit message P
   16-bit key K
   $P \oplus k = P \oplus 0...0K$ 16-0bits are added to K
   Here first 16-bits of P are same as first 16-bits of ciphertext c

3. Some part of key is repeated
   $P = P_1..P_l..P_n$ P_i is bit at i^{th} position

   $$\begin{array}{l} \quad P = P_1...P_l....P_n \\ \oplus\ k = k_1...k_l k_1..k_n \\ \hline \end{array}$$

   $c = (P_1 \oplus k_1)(P_2 \oplus k_2)..(P_l \oplus k_l)(P_{l+1} \oplus k_1)..(P_n \oplus k_t)$
   $c_1 = (P_1 \oplus k_1)$
   $c_{l+1} = (P_{l+1} \oplus k_1)$
   $c_1 \oplus c_{l+1} = (P_1 \oplus k_1) \oplus (P_{l+1} \oplus k_1)$
   $= P_1 \oplus P_{l+1}$
   Information of the message is revealed.

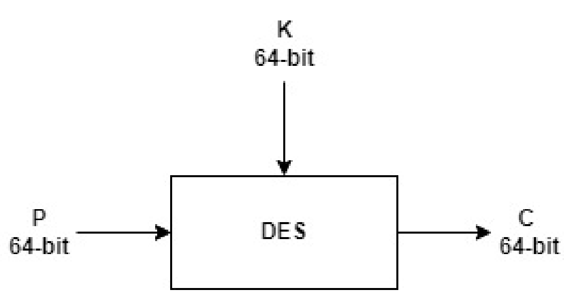
* OTP is not used in real life.

3 Data Encryption Standard (DES)

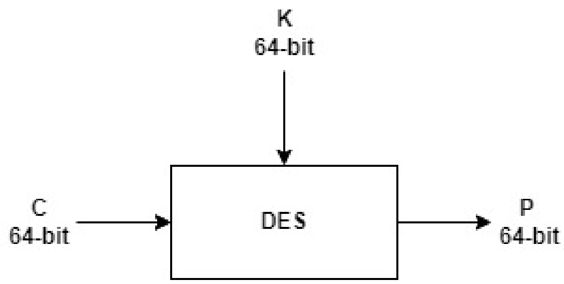
→ It is a block cipher
→ It is designed by IBM

1. Block size = 64 bit
2. Number of rounds = 16
3. Secret key size = 64 bit including 8 parity check bits
4. It is based on Feistel Network

3.1 Encryption

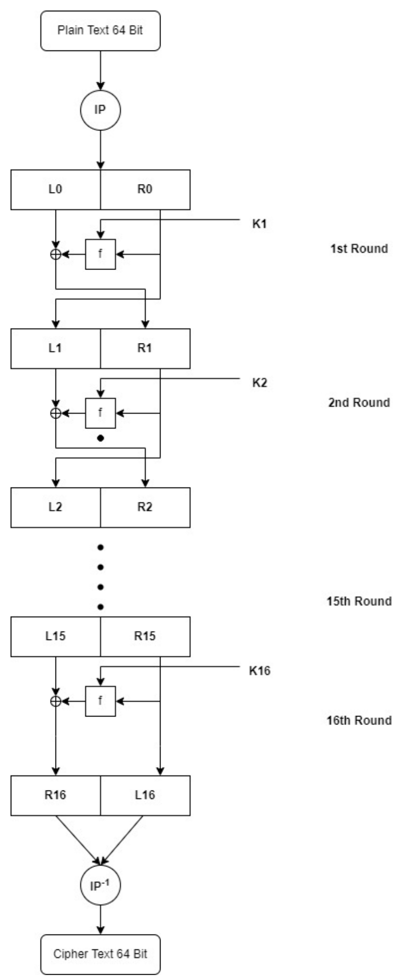


3.2 Decryption



* Secret key is 64 bit with 8 parity check bits.
8^{th},16^{th},....,64^{th} bits are parity check bits.
* In DES, we have 16 round keys k_1,k_2,..,k_{16}
Which are generated using Key scheduling algorithm. Key scheduling algorithm will take the secret key as an input.
$len(k_i) = 48$ bit. $G(k) \rightarrow k_1,k_2,..,k_{16}$

4 Structure of DES



4.1 Encryption

$$f : \{0,1\}^{32} x \{0,1\}^{48} \to \{0,1\}^{32}$$

$L_{i+1} = R_i$
$R_{i+1} = L_i \oplus f(R_i, k_{i+1})$

We have to learn the following

- IP, IP^{-1}.
- What is f (Round function).
- How $k_1, k_2, .., k_{16}$ are generated?

4.2 IP (Initial Permutation)

$$IP : \{0,1\}^{64} \to \{0,1\}^{64}$$

$$\text{IP} : \begin{array}{cccccccc} 58 & 50 & 42 & 34 & 26 & 18 & 10 & 2 \\ 60 & 52 & 44 & 36 & 28 & 20 & 12 & 4 \\ 62 & 54 & 46 & 38 & 30 & 22 & 14 & 6 \\ 64 & 56 & 48 & 40 & 32 & 24 & 16 & 8 \\ 57 & 49 & 41 & 33 & 25 & 17 & 9 & 1 \\ 59 & 51 & 43 & 35 & 27 & 19 & 11 & 3 \\ 61 & 53 & 45 & 37 & 29 & 21 & 13 & 5 \\ 63 & 55 & 47 & 39 & 31 & 23 & 15 & 7 \end{array}$$

$IP(m_1, m_2, .., m_{64}) = (m_{58}, m_{50}, m_{42}, m_{34}, .., m_7)$

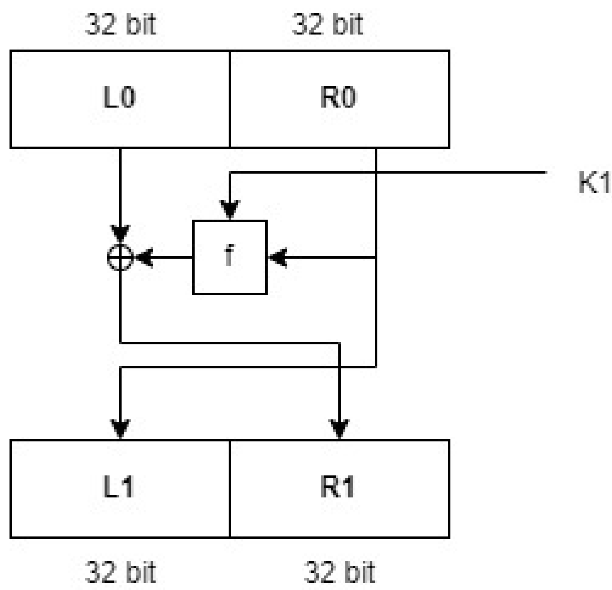
4.3 IP^{-1} (Final Permutation)

$$IP : \{0,1\}^{64} \to \{0,1\}^{64}$$

$$\text{IP} : \begin{array}{cccccccc} 40 & 8 & 48 & 16 & 56 & 24 & 64 & 32 \\ 39 & 7 & 47 & 15 & 55 & 23 & 63 & 31 \\ 38 & 6 & 46 & 14 & 54 & 22 & 62 & 30 \\ 37 & 5 & 45 & 13 & 53 & 21 & 61 & 29 \\ 36 & 4 & 44 & 12 & 52 & 20 & 60 & 28 \\ 35 & 3 & 43 & 11 & 51 & 19 & 59 & 27 \\ 34 & 2 & 42 & 10 & 50 & 18 & 58 & 26 \\ 33 & 1 & 41 & 9 & 49 & 17 & 57 & 25 \end{array}$$

$IP(m_1, m_2, .., m_{64}) = (m_{40}, m_8, m_{48}, m_{16}, .., m_{25})$

4.4 Round Function of DES



$$f : \{0,1\}^{32} x \{0,1\}^{48} \rightarrow \{0,1\}^{32}$$

$f(R_i,k_i) = X_i$
where,
R_i is 32 bit
k_i is 48 bit
X_i is 32 bit

$$f(R_i, k_i) = P(S(E(R_i) \oplus k_i))$$

$E : \{0,1\}^{32} \rightarrow \{0,1\}^{48}$ (Expansion Function)
$S : \{0,1\}^{48} \rightarrow \{0,1\}^{32}$ (Substitution Box)
$P : \{0,1\}^{32} \rightarrow \{0,1\}^{32}$ (Permutation Box)

4.5 Expansion Function E

$$E : \begin{array}{cccccc} 32 & 1 & 2 & 3 & 4 & 5 \\ 4 & 5 & 6 & 7 & 8 & 9 \\ 8 & 9 & 10 & 11 & 12 & 13 \\ 12 & 13 & 14 & 15 & 16 & 17 \\ 16 & 17 & 18 & 19 & 20 & 21 \\ 20 & 21 & 22 & 23 & 24 & 25 \\ 24 & 25 & 26 & 27 & 28 & 29 \\ 28 & 29 & 30 & 31 & 32 & 1 \end{array}$$

$E(x_1,x_2,..,x_{32}) = (x_{32},x_1,x_2,x_3,..,x_1)$

4.6 Substitution S

$S : \{0,1\}^{48} \rightarrow \{0,1\}^{32}$

* $X = B_1B_2B_3B_4B_5B_6B_7B_8$
where length od B_i is 6-bit.
* $S_1,S_2,S_3,S_4,S_5,S_6,S_7,S_8$
$S_i : \{0,1\}^6 \rightarrow \{0,1\}^4 \; \forall$ i = 1,2,3,4,5,6,7,8
$S_i(B_i) = C_i$
$S(X) = (S_1(B_1),S_2(B_2),S_3(B_3),S_4(B_4),S_5(B_5),S_6(B_6),S_7(B_7),S_8(B_8))$
* $B_i = b_1b_2b_3b_4b_5b_6$ $\quad b_i \in \{0,1\}$
* $r = (2{*}b_1 + b_6)$ $\quad 0 \leq r \leq 3$
* r is the representation of(b_1b_6)
* c is the representation of$(b_2b_3b_4b_5)$ $\quad 0 \leq c \leq 15$
r $\rightarrow$ row number
c $\rightarrow$ column number
$S_i(B_i) = a_{r,c} \rightarrow$ 4 bit

4.7 Permutation P

$$P : \{0,1\}^{32} \rightarrow \{0,1\}^{32}$$

$$P : \begin{array}{cccc} 16 & 7 & 20 & 21 \\ 29 & 12 & 28 & 17 \\ 1 & 15 & 23 & 26 \\ 5 & 18 & 31 & 10 \\ 2 & 8 & 24 & 14 \\ 32 & 27 & 3 & 9 \\ 19 & 13 & 30 & 6 \\ 22 & 11 & 4 & 25 \end{array}$$

$P(x_1,x_2,..,x_{32}) = (x_{16},x_7,x_{20},x_{21},..,x_{25})$

4.8 We have to understand the key scheduling Algorithm.

- Input : 64 bit key K.
- Output : 16 round key k_i, $1 \leq i \leq 16$

$len(k_i) = 48$ bit.

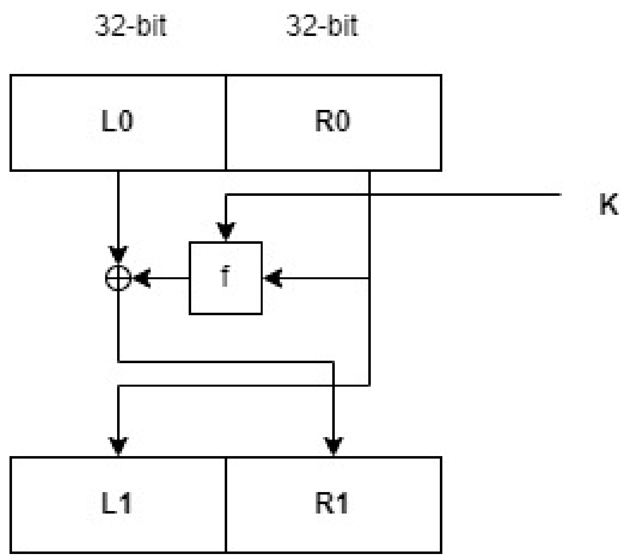
1. v_i, $1 \leq i \leq 16$ where $v_i = 1$
   if $i \in \{1,2,9,16\}$ else $v_i = 2$.
2. Delete the parity check bits. Now key k is 56 bits.
3. T = PC1(k); $PC1 : \{0,1\}^{56} \rightarrow \{0,1\}^{56}$
4. $(C_0,D_0) = T$ where C_0 is of 28 bit,D_0 is of 28 bit
5. for i = 1 to 16
   $C_i = (c_{i-1}$ left circular shift $v_i)$
   $D_i = (D_{i-1}$ left circular shift $v_i)$
   $k_i = PC2(C_i,D_i)$
   $PC2 : \{0,1\}^{56} \rightarrow \{0,1\}^{56}$
6. Round Keys = $\{k_1,k_2,..,k_{16}\}$

PC1 : $\{0,1\}^{56} \to \{0,1\}^{56}$

	57	49	41	33	25	17	9
c_i :	1	58	50	42	34	26	18
	10	2	59	51	43	35	27
	19	11	3	60	52	44	36
	63	55	47	39	31	23	15
d_i :	7	62	54	46	38	30	22
	14	6	61	53	45	37	29
	21	13	5	28	20	12	4

$\mathrm{PC1}(k_1,k_2,..,k_{63}) = (k_{57},x_{44},x_{41},x_{33},..,x_4)$

	14	17	11	24	1	5
	3	28	15	6	21	10
	23	19	12	4	26	8
$PC2_i$:	16	7	27	20	13	2
	41	52	31	37	47	55
	30	40	51	45	33	48
	44	49	39	56	34	53
	46	42	50	36	29	32



$$f(R_i, k_i) = P(S(E(R_i) \oplus k_i))$$

$M = L_0 \| R_0$
$c_1 = L_1 \| R_1$
$\mathrm{FN}(M,K) = c_1$
$\mathrm{FN}(M^c, K^c) = c_2$

X^c = bitwise complement of X

$L_1 = R_0$
$R_1 = L_0 \oplus f(R_0,k)$
$k^c \oplus E(R^c) = (E(R))^c \oplus k^c = E(R) \oplus k$
$P(S(k^c \oplus E(R^c))) = P(S(E(R) \oplus k))$
$L_0 \| R_0 = M$ $\quad$ $L_0{}^c \| R_0{}^c = M^c$
$L_1 = R_0$

$\mathrm{R}_1 = \mathrm{L}_0 \oplus \mathrm{f}(\mathrm{R}_0,\mathrm{k})$
$L_1{}^c = R_0{}^c$
$R_1{}^c = L_0{}^c \oplus f(R_0{}^c, k^c) = L_0{}^c \oplus f(R_0, k) = (L_0 \oplus f(R_0, k))^c = R_1{}^c$
$\mathrm{c}_1 = \mathrm{L}_1 \| \mathrm{R}_1$
$c_2 = L_1{}^c \| R_1{}^c$

$$c_2 = c_1{}^c$$