Fall 2015 CIS 3362 Homework #3 Solutions (Light Coding Group)

1) Let b = 93de8067f5942ca1, a plaintext block (expressed in hexadecimal). What is IP(b)?

Solution:

b= 93de8067f5942ca1

b in binary is:

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	0	0	1	0	0	1	1	1	1	0	1	1	1	1	0	1	0	0	0	0	0	0	0	0	1	1	0	0	1	1	1
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
1	1	1	1	0	1	0	1	1	0	0	1	0	1	0	0	0	0	1	0	1	1	0	0	1	0	1	0	0	0	0	1

b= 1001001111011110100000000110011111110101100101000010110010100001

IP of 64 bit (1~64) can be calculated below table:

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
58	50	42	34	26	18	10	02	60	52	44	36	28	20	12	04	62	54	46	38	30	22	14	06	64	56	48	40	32	24	16	08
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
57	49	41	33	25	17	09	01	59	51	43	35	27	19	11	03	61	53	45	37	29	21	13	05	63	55	47	39	31	23	15	07

So, the IP(b) in binary will be:

IP(b)= 0001 1010 0011 0011 0111 1010 1001 1001 1011 0111 1101 1000 0100 0010 0000 1011

Or in hex:

IP(b)= 1a337a99b7d8420b

2) Using the same block b shown in question #1, calculate $IP^{-1}(b)$.

Solution:

b= 93de8067f5942ca1

b in binary is:

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	0	0	1	0	0	1	1	1	1	0	1	1	1	1	0	1	0	0	0	0	0	0	0	0	1	1	0	0	1	1	1
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
1	1	1	1	0	1	0	1	1	0	0	1	0	1	0	0	0	0	1	0	1	1	0	0	1	0	1	0	0	0	0	1

b= 1001001111011110100000000110011111110101100101000010110010100001

Inverse IP of 64 bit (1~64) can be calculated below table:

01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
40	08	48	16	56	24	64	32	39	07	47	15	55	23	63	31	38	06	46	14	54	22	62	30	37	05	45	13	53	21	61	29
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
36	04	44	12	52	20	60	28	35	03	43	11	51	19	59	27	34	02	42	10	50	18	58	26	33	01	41	09	49	17	57	25

So, the inverse IP of b will be:

IP(b)=1100 0011 0101 0001 1011 1001 0001 1000 1111 0000 1000 1011 1001 0001 1111 0110

And in hex:

IP(b)= c351b918f08b91f6

3) The Feistel function in DES takes in one 32 bit argument (half of the bits in the process of encryption) and a 48 bit argument (the key for that particular round). If R_{i-1} = 94e2d671 (in hex) and K_i = b57c809a743e, calculate $F(R_{i-1}, K_i)$. Your calculation should show the following steps: the use of the expansion matrix E, the XOR with the key, the S-box substitution and the application of the permutation matrix P.

Solution:

R_{i-1}=94e2d671=1001 0100 1110 0010 1101 0110 0111 0001

K_i=b57c809a743e= 1011 0101 0111 1100 1000 0000 1001 1010 0111 0100 0011 1110

Expansion matrix E is as follows:

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

So, after applying expansion matrix to R_{i-1} we will have:

$E(R_{i-1})$= 1100 1010 1001 0111 0000 0101 0110 1010 1100 0011 1010 0011

K_i=1011 0101 0111 1100 1000 0000 1001 1010 0111 0100 0011 1110

R	1	1	0	0	1	0	1	0	1	0	0	1	0	1	1	1	0	0	0	0	0	1	0	1	0	1	1	0	1	0	1	0	1	1	0	0	0	0	1	1	1	0	1	0	0	0	1	1

K	1	0	1	1	0	1	0	1	0	1	1	1	1	1	0	0	1	0	0	0	0	0	0	0	1	0	0	1	1	0	1	0	0	1	1	1	0	1	0	0	0	0	1	1	1	1	1	0
XOR	0	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	0	0	0	0	1	0	1	1	1	1	1	0	0	0	0	1	0	1	1	0	1	1	1	1	0	0	1	1	1	0	1

B= E(R_{i-1}) xor K_i = 011111 111110 101110 000101 111100 001011 011110 011101

Now, after XOR E(R_{i-1}) and K_i, we should apply result (B) as an input to S boxes. The output for each 6 bit is shown below:

B_1=011111 → S_1 → C_1= 8=1000

B_2=111110 → S_2 → C_2= 15=1111

B_3=101110 → S_3 → C_3= 0=0000

B_4=000101 → S_4 → C_4= 11=1011

B_5=111100 → S_5 → C_5= 0=0000

B_6=001011 → S_6 → C_6= 12=1100

B_7=011110 → S_7 → C_7= 1=0001

B_8=011101 → S_8 → C_8= 9=1001

So, out of the S boxes (C) is:

C=1000 1111 0000 1011 0000 1100 0001 1001

Then, The C should be apply to permutation function which is as follows:

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

Finally, the output (f) is as follows:

f= 1101 1010 1100 1000 0100 1000 0101 1000

f(in hex) = dac8 4858

4) Given the following state matrix in AES, show what it would look like after applying the sub bytes step?

3E	1E	32	10
67	96	FF	65
A8	6B	AB	9E
5A	06	9C	43

Solution:

We use the S-box for AES in SubBytes step and replace each element of the given matrix (A) with its equivalent in S-box:

a_0=3E → S-box → $S(a_0)$=B2

a_1=67 → S-box → $S(a_1)$=85

a_2=A8 → S-box → $S(a_2)$=C2

a_3=5A → S-box → $S(a_3)$=BE

a_4=1E → S-box → $S(a_4)$=72

a_5=96 → S-box → $S(a_5)$=90

a_6=6B → S-box → $S(a_6)$=7F

a_7=06 → S-box → $S(a_7)$=6F

a_8=32 → S-box → $S(a_8)$=23

a_9=FF → S-box → $S(a_9)$=16

a_{10}=AB → S-box → $S(a_{10})$=62

a_{11}=9C → S-box → $S(a_{11})$=DE

a_{12}=10 → S-box → $S(a_{12})$=C9

a_{13}=65 → S-box → $S(a_{13})$=4D

a_{14}=9E → S-box → $S(a_{14})$=0B

a_{15}=43 → S-box → $S(a_{15})$=1A

So, after SubByte step output matrix is:

B2	72	23	C9
85	90	16	4D
C2	7F	62	0B
BE	6F	DE	1A

5) Now, apply the shift rows operation to your result from question 4, what is the resulting state matrix?

Solution:

In shift rows operation, each row shift to the right cyclically according to the number of rows in matrix. For example, row 1 shift once to the right and row 2 shift twice to the right and etc. Row 0 doesn't shift! So, after applying shift row operation to the result of pervious question the output matrix is:

B2	72	23	C9
90	16	4D	85
62	0B	C2	7F
1A	BE	6F	DE

6) Consider the following Mix Columns step in AES:

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} 79 & 45 & A3 & C4 \\ 28 & 8E & 59 & D9 \\ A5 & C1 & 9C & 21 \\ 96 & F6 & C5 & 18 \end{bmatrix}$$

Calculate the four following entries: (a) row 1, column 2

(b) row 2, column 4

(c) row 3, column 3

(d) row 4, column 1

Solution:

In this problem, for getting each part we need to multiply correspond row of left matrix with correspond column of right matrix. For example in part a (row 1, column 2), we should multiply row 1 of left matrix with column 2 of right matrix. This is not usual multiplication of matrix. The principle of this sort of multiplication was explained in class. For each part the process is as follows:

Part a:

$a=[02\ 03\ 01\ 01] * \begin{bmatrix} 45 \\ 8E \\ C1 \\ F6 \end{bmatrix}$

a_0= 02 * 45=(00000010)*(01000101) ⟹ 1 shift to left a_0= 10001010

a_1= (01+02) * 8E ⟹ $\begin{cases} (01) * (10001110) = 10001110 \\ (10) * (10001110)\ 1 \text{ shift to left } 00011100 \text{ XOR with } 00011011\ 00000111 \end{cases}$

a_1= (10001110) XOR (00000111) =10001001

a_2= 01 * C1=(01)*(11000001)= 11000001

a_3= 01 * F6=(01)*(11110110)= 11110110

a= a_0 XOR a_1 XOR a_2 XOR a_3 = 00110100 = **34**

Part b:

$b=[01\ 02\ 03\ 01] * \begin{bmatrix} C4 \\ D9 \\ 21 \\ 18 \end{bmatrix}$

b_0= 01 * C4=(01)*(11000100)= 11000100

b_1= 02 * D9=(10) $*$ (11011001) 1 shift to left 10110010 XOR with 00011011 b1 = 10101001

b_2= (01+02) * 21 $\Rightarrow$ $\begin{Bmatrix} (01) * (00100001) = 00100001 \\ (10) * (00100001) \text{ 1 shift to left } 01000010 \end{Bmatrix}$

b_2= (00100001) XOR (01000010) =01100011

b_3= 01 * 18 =(01)*(00011000)= 00011000

b= b_0 XOR b_1 XOR b_2 XOR b_3=00010110 =**16**

Part c:

c=[01 01 02 03] $*$ $\begin{bmatrix} A3 \\ 59 \\ 9C \\ C5 \end{bmatrix}$

c_0= 01 * A3=(01)*(10100011)= 10100011

c_1= 01 * 59 =(01)*(01011001)= 01011001

c_2= 02 * 9C=(10) $*$ (10011100)1 shift to left 00111000 XOR with 00011011 c2 = 00100011

c_3= (01+02) * C5 $\Rightarrow$ $\begin{Bmatrix} (01) * (11000101) = 11000101 \\ (10) * (11000101) \text{ 1 shift to left } 10001010 \text{ XOR with } 00011011 \text{ } 10010001 \end{Bmatrix}$

c_3= (11000101) XOR (10010001) =01010100

c= c_0 XOR c_1 XOR c_2 XOR c_3= 10001101= **8D**

part d:

d=[03 01 01 02] $*$ $\begin{bmatrix} 79 \\ 28 \\ A5 \\ 96 \end{bmatrix}$

d_0= (01+02) * 79 $\Rightarrow$ $\begin{Bmatrix} (01) * (01111001) = 01111001 \\ (10) * (01111001) \text{ 1 shift to left } 11110010 \end{Bmatrix}$

d_0= (01111001) XOR (11110010) =10001011

d_1= 01 * 28=(01)*(00101000)= 00101000

d_2= 01 * A5=(01)*(10100101)= 10100101

d_3= 02 * 96=(10) $*$ (10010110) 1 shift to left 00101100 XOR with 00011011 d3 = 00110111

d= d_0 XOR d_1 XOR d_2 XOR d_3= 00110001=**31**