**1. This problem provides a numerical example of encryption using a one-round version of DES. We start with the same bit pattern for the key and the plaintext, namely,**

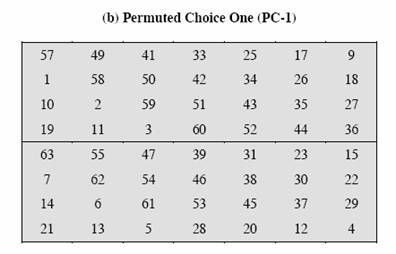
**In hexadecimal notation: 0 1 2 3 4 5 6 7 8 9 A B C D E F**

**In binary notation: 0000 0001 0010 0011 0100 0101 0110 0111**

**1000 1001 1010 1011 0100 1101 1110 1111**

**a. Derive K1, the first-round subkey.**

64-bit key 🡺 PC1 (64 🡪 56) 🡺 shift registers C&D🡺 PC2(56🡪 48)



10110000

11001100

10101010

00001010

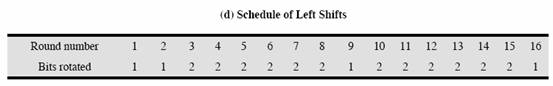
10101100

11001111

00000000

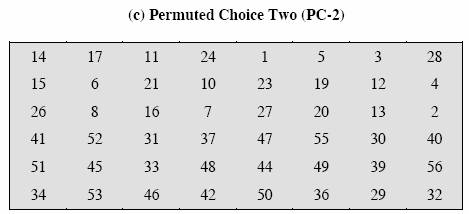
🡺 L = 1011 0000 1100 1100 1010 1010 0000

R = 1010 1010 1100 1100 1111 0000 0000



C1(L) = 0110 0001 1001 1001 0101 0100 0001

D1(R) = 0101 0101 1001 1001 1110 0000 0001



1-8 0110 0001

9-16 1001 1001

17-24 0101 0100

25-32 0001 0101

33-40 0101 1001

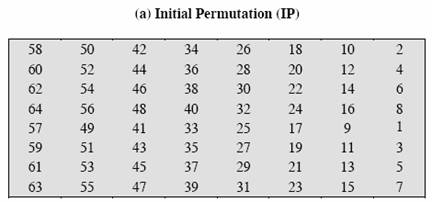
41-48 1110 0000

49-56 0000 0001

K1 = 00000011 00000010 01100111 10010011 00000001 10010101

**b. Derive L0, R0.**

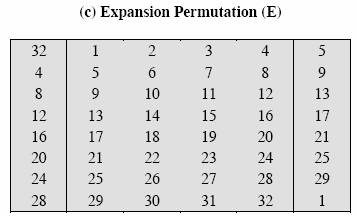
**(Initial permutation)**

****

L0 = 11001100 00000000 11001100 11111111

R0 = 10110000 10101010 11110000 10101010

**c. Expand R0 to get E[R0].**



E[R0] = 010110100001 010101010101 011110100001 010101010101

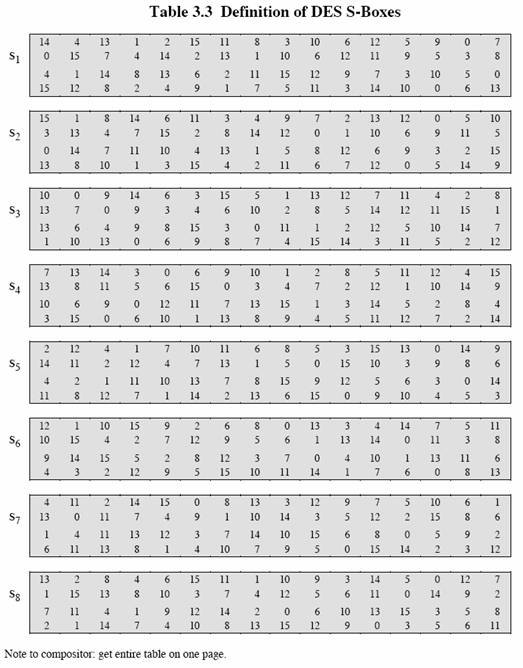
**d. Calculate A = E[R0]⊕K1.**

E[R0] = 010110100001 010101010101 011110100001 010101010101

K1 = 000000110000 001001100111 100100110000 000110010101

010110010001 011100110010 111010010001 010011000000

**e. Group the 48-bit result of (d) into sets of 6 bits and evaluate the corresponding S-box substitutions.**



S1(010110)= 1100

S2(010001)= 1100

S3(011100)=0100

S4(110010)= 0001

S5(111010)= 0011

S6(010001) = 0110

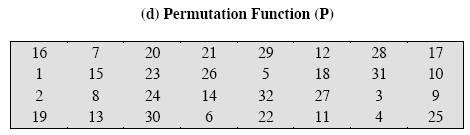
S7(010011) = 0011

S8(000000) = 1101

**f. Concatenate the results of (e) to get a 32-bit result, B.**

B = 1100 1100 0100 0001 0011 0110 0011 1101

**g. Apply the permutation to get P(B).**



P(B) = 10101010 10101001 10001100 10111000

**h. Calculate R1 = P(B)⊕L0**

R1 = P(B)⊕ L0

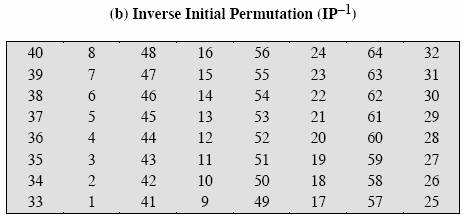
P(B) = 1010 1010 1010 1001 1000 1100 1011 1000

L0 = 1100 1100 0000 0000 1100 1100 1111 1111

0110 0110 1010 1001 0100 0000 0100 0111

R1 =0110 0110 1010 1001 0100 0000 0100 0111

**i. Write down the ciphertext.**



L1 = R0 = 10110000 10101010 11110000 10101010

Ciphertext = IP-1(R1L1) = 0001 0001 0110 0011 0100 0001 0011 0010

1000 1000 1111 1010 0100 1101 1011 1010

**5.6**                **Compare the AES to DES.** **For each of the following elementst of DES, indicate the comparable element in AES or explain why it is not needed in AES.**

**a.**        **XOR of subkey material with the input to the function f function.**

The similar element in AES for XOR of subkey with the input to the function (that passes different stages before XORing ) is the ad d ed round key stage in all the 10 rounds.

**b.**       **XOR of the f function output with left side of the block.**

There is no similar element in AES for XOR the f function output with left half side of the block, this is because AES structure is not a feistel structure. The entire block is processed in parallel (No two halves are using one half to modify the other half ) .

**c.**        **The f function.**

There is no single element that is similar to f function, but the four stages ( Substitution bytes, shift rows, mix columns, added roundly) in each round do the same as f function.

**d.**       **Permutation P.**

The similar element for **P** is the shift rows in each of the 10 rounds.

**e.**        **Swapping of halves of the block.**

No similar element in AES this is because that AES structure not a feistel structure and no need to swap halves since work in parallel (No half needs to modify the other half

**Prob 3: Perform encryption and decryption using the RSA algorithm, as in Figure 9.6 for the following:**

1.        p=3; q=11; e=7; M=5

Answer:

n = p \* q = 3 \* 11 = 33

f(n) = (p-1) \* (q-1) = 2 \* 10 = 20

Now, we need to compute d = e -1 mod f(n) by using backward substitution of GCD algorithm:

According to GCD:

20 = 7 \* 2 + 6

7 = 6 \* 1 + 1

6 = 1 \* 6 + 0

Therefore, we have:

1 = 7 – 6

= 7 – (20 – 7 \* 2)

= 7 – 20 + 7 \* 2

= -20 + 7 \* 3

Hence, we get d = e -1 mod f(n) = e -1 mod 20 = 3 mod 30 = 3

So, the public key is {7, 33} and the private key is {3, 33} , RSA encryption and decryption is following :

2.        p=5; q=11; e=3; M=9

Answer:

n = p \* q = 5 \* 11 = 55

f(n) = (p-1) \* (q-1) = 4 \* 10 = 40

Now, we need to compute d = e -1 mod f(n) by using backward substitution of GCD algorithm:

According to GCD:

40 = 3 \* 13 + 1

13 = 1 \* 13 + 0

Therefore, we have:

1 = 40 – 3 \* 13

Hence, we get d = e -1 mod f(n) = e -1 mod 40 = -13 mod 40 = (27 – 40) mod 40 = 27

So, the public key is {3, 55} and the private key is {27 , 55 }, RSA encryption and decryption is following:

3.        p=7; q=11; e=17; M=8

Answer:

n = p \* q = 7 \* 11 = 77

f(n) = (p-1) \* (q-1) = 6 \* 10 = 60

Now, we need to compute d = e -1 mod f(n) by using backward substitution of GCD algorithm:

According to GCD:

60 = 17 \* 3 + 9

17 = 9 \* 1 + 8

9 = 8 \* 1 + 1

8 = 1 \* 8 + 0

Therefore, we have:

1 = 9 – 8

= 9 – (17 – 9)

= 9 – (17 – (60 – 17 \* 3))

= 60 – 17\*3 – (17 – 60 + 17\*3)

= 60 – 17 \*3 + 60 – 17\*4

= 60\*2 – 17\*7

     Hence, we get d = e -1 mod f(n) = e -1 mod 60 = -7 mod 60 = (53-60) mod 60 = 53

So, the public key is {17 , 77} and the private key is {53, 77 }, RSA encryption and decryption is following:

**prob 3 : Perform encryption and decryption using the RSA algorithm, as in Figure 9.6, for the following**

**(1) p = 3; q = 11, e = 7; M = 5**

**Encryption**

Ciphertext (C) = M e mod n

C = 5^7 mod (p \* q)

C = 5^7 mod (3 \* 11) = 5^7 mod (33)

C = [ ( 5^4 mod 33) \* ( 5 ^ 2 mod 33) \* ( 5 ^ 1 mod 33 ) ] mod 33

C = [ (625 mod 33) \* (25 mod 33) \* (5 mod 33) ] mod 33

C = [ 31 \* 25 \* 5 ] mod 33

C = [ 3875] mod 33

C = 14

**Decryption**

Plaintext (M) = C d mod n

Need to calculate d

e and d are multiplicative inverses mod φ ( n ).

φ ( n ) = (p – 1) (q – 1)

φ ( n ).= (3 -1) (11 – 1) = 2 \* 10 = 20

The multiplicative inverse of

e mod 20 = 7 mod 20 = 3

So, lets calculate C d mod n

14 ^ 3 mod (n) = 14 ^ 3 mod (p \* q) = 14 ^ 3 mod (33)

2744 mod 33 = 5

Therefor e a successful decry p tion gets the original plaintext 5

**(2) p = 5; q = 11, e = 3; M = 9**

**Encryption**

Ciphertext (C) = M e mod n

C = 9 ^ 3 mod (p \* q)

C = 9 ^ 3 mod ( 5 \* 1 1) = 7 29 mod ( 55 )

C = 14

**Decryption**

Plaintext (M) = C d mod n

Need to calculate d

e and d are multiplicative inverses mod φ ( n ).

φ ( n ) = (p – 1) (q – 1)

φ ( n ).= ( 5 -1) (11 – 1) = 4 \* 10 = 4 0

The multiplicative inverse of

e mod 4 0 = 3 mod 4 0 = -1 3

Therefore the positive multiplicative inverse of 3 mod 40 is = 27

So, lets calculate C d mod n

14 ^ 27 mod (n) = 14 ^ 27 mod (p \* q) = 14 ^ 27 mod ( 55 )

[(14 ^ 16 mod 55) \* (14 ^ 8 mod 55) \* (14 ^2 mod 55) \* (14 ^ 1 mod 55) ] mod 55

[(14 ^ 8 mod 55) \* (14 ^ 8 mod 55) \* (14 ^ 8 mod 55) \* (14 ^2 mod 55) \* (14 ^ 1 mod 55)] mod 55

[(14 ^ 4 mod 55) \* (14 ^ 4 mod 55) \* (14 ^ 4 mod 55) \* (14 ^ 4 mod 55) \* (14 ^ 4 mod 55) \* (14 ^ 4 mod 55) \* (14 ^2 mod 55) \* (14 ^ 1 mod 55)] mod 55

[(38416 mod 55) \* (38416 mod 55) \* (38416 mod 55) \* (38416 mod 55) \* (38416 mod 55) \* (38416 mod 55) \* (196 mod 55) \* (14 mod 55)] mod 55

[26 \* 26 \* 26 \* 26 \* 26 \* 26 \* 31 \* 14] mod 55 = 9

Therefor e a successful decry p tion gets the original plaintext 9

**(3) p = 7; q = 11, e = 17; M = 8**

**Encryption**

Ciphertext (C) = M e mod n

C = 8 ^ 1 7 mod (p \* q)

C = 8 ^ 1 7 mod ( 7 \* 11 ) = 8 ^ 1 7 mod ( 77 )

C = [ ( 8^16 mod 77 ) \* ( 8 ^ 1 mod 77 ) ] mod 77

C = [ (8^8 mod 77) \* (8^8 mod 77) \* (8 ^ 1 mod 77) ] mod 77

C = [(8^4 mod 77) \* (8^ 4 mod 77) \* (8^4 mod 77) \* (8^4 mod 77) (8 ^ 1 mod 77) ] mod 77

C = [4096 mod 77 \* 4096 mod 77 \* 4096 mod 77 \* 4096 mod 77 \* 8] mod 77

C = [15 \* 15 \* 15 \* 15 \* 8] mod 77

C = [ 405000 ] mod 77

C = 57

**Decryption**

Plaintext (M) = C d mod n

Need to calculate d

e and d are multiplicative inverses mod φ ( n ).

φ ( n ) = (p – 1) (q – 1)

φ ( n ).= ( 7 -1) (11 – 1) = 6 \* 10 = 6 0

The multiplicative inverse of

e mod 6 0 = 1 7 mod 6 0 = -7

Therefore the positive multiplicative inverse of 17 mod 60 is = 53

So, lets calculate C d mod n

57 ^ 53 mod (n) = 57 ^ 53 mod (p \* q) = 57 ^ 53 mod ( 77 )

[(57^32 mod 77) \* (57 ^16 mod 77) \* ( 57 ^ 4 mod 77 ) \* ( 57 ^ 1 mod 77 )] mod 77

[(57 ^ 16 mod 77) \* (57 ^ 16 mod 77) \* (57 ^ 16 mod 77) \* (57 \* 4 mod 77) \* (57 ^ 1 mod 77)] mod 77

57 ^ 4 mod 77 = 71

57 ^ 8 mod 77 = [(57 ^ 4 mod 77) \* (57 ^ 4 mod 77)] mod 77

57 ^ 16 mod 77 = [ (57 ^8 mod 77) \* (57 ^ 8 mod 77)] mod 77

Therefore ,

57 ^ 8 mod 77 = [71 \* 71] mod 77 = 36

Therefore,

57 ^ 16 mod 77 = [ 36 \* 36] mod 77 = 64

So, we have

[64 \* 64 \* 64 \* 71 \* 57] mod 77

[ 262144 \* 4047 ] mod 77 = 8

Therefor e a successful decry p ti on gets the original plaintext 8

**(4) p = 11; q = 13, e = 11; M = 7**

**Encryption**

Ciphertext (C) = M e mod n

C = 7 ^ 11 mod (p \* q)

C = 7 ^11 mod ( 11 \* 1 3 ) = 7^11 mod ( 14 3)

C = [ ( 7 ^ 8 mod 14 3) \* ( 7 ^ 2 mod 14 3) \* ( 7 ^ 1 mod 14 3) ] mod 14 3

C = [ (7^4 mod 143) \* (7^4 mod 143) \* (7 ^ 2 mod 143) \* (7 ^ 1 mod 143) ] mod 143

C = [ 1 13 \* 113 \* 49 \* 7 ] mod 14 3

C = [ 4379767 ] mod 14 3

C = 106

**Decryption**

Plaintext (M) = C d mod n

Need to calculate d

e and d are multiplicative inverses mod φ ( n ).

φ ( n ) = ( p – 1) (q – 1)

φ ( n ).= ( 11 -1) (1 3 – 1) = 10 \* 1 2 = 1 20

The multiplicative inverse of

e mod 1 20 = 11 mod 1 20 = 11

So, lets calculate C d mod n

106 ^ 11 mod (n) = 1 06 ^ 11 mod (p \* q) = 1 06 ^ 11 mod ( 14 3)

[(106 ^ 8 mod 143) \* (106 ^ 2 mod 143) \* (106 ^ 1 mod 143)] mod 143

[(106 ^ 4 mod 143) \* (106 ^ 4 mod 143) \* (106 ^ 2 mod 143) \* (106 ^ 1 mod 143)] mod 143

[3 \* 3 \* 82 \* 106] mod 143

[78228] mod 143 = 7

There for e a successful decry p tion gets the original plaintext 7

**(5) p = 17; q = 31, e = 7; M = 2**

Ciphertext (C) = M e mod n

C = 2 ^ 7 mod (p \* q)

C = 2 ^ 7 mod ( 1 7 \* 3 1 ) = 128 mod ( 527 )

C = 1 28

**Decryption**

Plaintext (M) = C d mod n

Need to calculate d

e and d are multiplicative inverses mod φ ( n ).

φ ( n ) = ( p – 1) ( q – 1)

φ ( n ).= (1 7 -1) ( 3 1 – 1) = 1 6 \* 30 = 48 0

The multiplicative inverse of

e mod 48 0 = 7 mod 480 = -137

Therefore the po sitive multiplicative inverse of 7 mod 480 is = 34 3

So, lets calculate C d mod n

128 ^ 343 mod (n) = 1 28 ^ 343 mod (p \* q) = 128 ^ 343 mod ( 527 )

Binary expansion of 343 = 2^8 + 2^6 + 2 ^ 4 + 2 ^ 2 + 2 ^ 1 + 2^0 = 101010111

So,

[(128 ^ 256 mod 527) \* (128 ^ 64 mod 527) \* (128 ^ 16 mod 527) \* (128 ^ 4 mod 527) \* (128 ^ 2 mod 527) \* (128^1 mod 527)]mod 527

128 ^ 2 mod 527 = 47

Therefore,

128 ^ 4 mod 527 = [ 47 \* 47 ] mod 527 = 101

Therefore,

128 ^ 16 mod 527 = [ 101 \* 101 \* 101 \* 101 ] mod 527 = 35

Therefore,

128^64 mod 527 = [ 35 \* 35 \* 35 \* 35 ] mod 527 = 256

Therefore,

128^256 mod 527 = [ 256 \* 256 \* 256 \* 256 ] mod 527 = 35

So,

[ 35 \* 256 \* 35 \* 101 \* 47 \* 128] mod 527 = [ 35 \* 35 \* 47 \* 10 1 \* 128 \* 256 ] mod 527

[ 1225 \* 4747 \* 32768 ] mod 527 = [ 5815075 \* 32768 ] mod 527

[ 190548377600 ] mod 527 = 2

Therefor e a successful decry p tion gets the original plaintext 2

4.        p=11; q=13; e=11; M=7

Answer:

n = p \* q = 11 \* 13 = 143

f(n) = (p-1) \* (q-1) = 10 \* 12 = 120

Now, we need to compute d = e -1 mod f(n) by using backward substitution of GCD algorithm:

According to GCD:

120 = 11 \* 10 + 10

11 = 10 \* 1 + 1

10 = 1 \* 10 + 0

Therefore, we have:

1 = 11 – 10

= 11 – (120 – 11 \* 10)

= 11 – 120 + 11 \* 10

= -120 + 11 \* 11

Hence, we get d = e -1 mod f(n) = e -1 mod 120 = 11 mod 120 = 11

So, t he public key is {11 , 143 } and the private key is { 11 , 143 }, RSA encryption and decryption is following:

5.        p=17; q=31; e=7; M=2

n = p \* q = 17 \* 31 = 527

f(n) = (p-1) \* (q-1) = 16 \* 30 = 480

Now, we need to compute d = e -1 mod f(n) by using backward substitution of GCD algorithm:

According to GCD:

480 = 7 \* 68 + 4

7 = 4 \* 1 + 3

4 = 3 \* 1 + 1

3 = 1 \* 3 + 0

Therefore, we have:

1 = 4 – 3

= 4 – (7 – 4)

= 4 – (7 – (480 – 7\*68))

= 4 – (7 – 480 + 7\*68)

= 480 – 7\*68 – 7 + 480 – 7\*68

= 480\*2 – 7\*137

Hence, we get d = e -1 mod f(n) = e -1 mod 480 = -137 mod 480 = (343 – 480) mod 480 =343

So, the public key is {7, 527} and the private key is {343, 527}, RSA encryption and decryption is following: