Johns Hopkins ■ CS600.424: Network Security ■ Spring 2012

Homework 2 – Solutions

Due Date – March 2 in class

Instructions: Please do not submit handwritten assignment.

1. This problem provides a numerical example of encryption using a one round version of DES. We use the same bit pattern for the key K and the plaintext at the start which is :

0000 0001 0010 0011 0100 0101 0110 0111
1000 1001 1010 1011 1100 1101 1110 1111

(a) Derive the first round key K_1.
First we derive two 28 bit keys from the 64 bit keys as per table 3.4(a). Then perform a left circular shift on two halves separately, then pass the 56 bit result through PC-2 (Table 3.2 (c)).
So the keys before the left shift are:
1 1 1 1 0 0 0   1 0 1 0 1 0 1
0 1 1 0 0 1 1 and   0 1 1 0 0 1 1
0 0 1 0 1 0 1   0 0 1 1 1 1 0
0 1 0 0 0 0 0   0 0 0 0 0 0 0

Upon shifting left one bit for the two keys separetly, the first bit moves to 28th position i.e. becomes the last bit and all other bits are arranged 7 at a time, so we get upon shift the keys as
1 1 1 0 0 0 0   0 1 0 1 0 1 0
1 1 0 0 1 1 0 and   1 1 0 0 1 1 0
0 1 0 1 0 1 0   0 1 1 1 1 0 0
1 0 0 0 0 0 1   0 0 0 0 0 0 1

Now we will number these bits from 1 to 56 i.e. 1 to 28 are in first half and 29 to 56 in the second half and then start placing these numbers according to PC-2 or Table 3.4(c). So the first four bits are 14, 17, 11, 24 = 0 0 0 0, the second 4 bits are 1, 5, 3, 18, = 1 0 1 1, and so on so fourth, so our final result in binary notation is:

0000 1011 0000 0010 0110 0111
1001 1011 0100 1001 1010 0101

And in hex is 0 B 0 2 6 7 9 B 4 9 A 5

(b) Derive L_0 and R_0.

L0, R0 are derived by passing the 64-plaintext through IP (Table 3.2a):

L0 = 1100 1100 0000 0000 1100 1100 1111 1111
R0 = 1111 0000 1010 1010 1111 0000 1010 1010

(c) Expand R_0 to get $E[R_0]$, where E[.] is the expansion function of Table 3.2 of your textbook.

E(R0) = 011110 100001 010101 010101 011110 100001 010101 010101

(d) Calculate $A = E[R0] \oplus K_1$.

A = 011100 010001 011100 110010 111000 010101 110011 110000

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

$$S_1(1110) = S(14) = 0\,(base10) = 0000\ (base\,2)$$

$$S_2(1000) = S(8) = 12\,(base10) = 1100(\,base\,2$$

$$S_3(1110) = S(14) = 2\,(base10) = 0010(\,base\,2)$$

$$S_4(1001) = S(9) = 1(base10) = 0001(base\,2)$$

$$S_5(1110) = S(12) = 6\,(base10) = 0110(\,base\,2)$$

$$S_6(1010) = S(10) = 13(base10) = 1101(\,base\,2)$$

$$S_7(1001) = S(9) = 5(base10) = 0101(base\,2)$$

$$S_8(1000) = S(8) = 0(base10) = 0000(\,base\,2)$$

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

B = 0000 1100 0010 0001 0110 1101 0101 0000

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

Using Table 3.2d, P(B) = 1001 0010 0001 1100 0010 0000 1001 1100

(h) Calculate $R_1 = P(B) \oplus L_0$.

R1 = 0101 1110 0001 1100 1110 1100 0110 0011

(i) Write down the ciphertext

L1 = R0. The ciphertext is the concatenation of L1 and R1.

2 (a) Using RSA, choose p=3, and q=11, and encode the word “dog” by encrypting each letter separately. Apply the decryption algorithm to the encrypted version to recover the original plaintext message. (b) Repeat part (a) but now encrypt “dog” as one message. [Hint – treat each letter as 5 bit number and then concatenate all 15 bits to get the number to encrypt.]

(a) We are given p=3 and q=11. We thus have n=33 and q=11. Choose e = 9 (its a good idea to give a hint that 9 is a good value to choose, since the resulting calculations are less likely to run into numerical stability problems than other choices for e) since 3 and (p-1)*(q-1)=20 have no common factors. Choose d=9 also so that e*d=81 and thus e*d-1=80 is exactly divisible by 20. We can now perform the RSA encryption and decryption using n=33,e=9 , d=9 and

letter	m	m**e	ciphertext = m**e mod 33
d	4	262144	25
o	15	38443359375	3
g	7	40353607	19

ciphertext	c**d	m = c**d mod n	letter
25	38146972265625	4	d
3	19683	15	o
19	322687697779	7	g

(b) We first consider each letter as a 5-bit number: 00100, 01111, 00111. Now we concatenate each letter to get 001000111100111 and encrypt the resulting decimal number m=4583. The concatenated decimal number m (= 4583) is larger than current n (= 33). We need m < n. So we use p = 43, q = 107, n = p*q = 4601, z = (p-1)(q-1) = 4452. e = 61, d = 73

ciphertext = m**e mod 4601

m**e=
2138657760182805780408960215653056718861149986902978873380843880430286459562061 3
9567258407209497648456409561187848752467850332361977771297302589617569184002920 4
8632806197527785447791567255101894492820972508185769802881718983

ciphertext = m**e mod 4601 = 402

c**d=
1283813313619771634195712132539793287643533147482536209328405262793027158861012 3
9205328724963357096749312228022145381501293424137054020458145987149793872321410 1
4703227794586499817945633390592

ciphertext = m**e mod 4601 = 4583

3. Consider RSA with p=5 and q=11.

(a) what are *n* and *z*.

(b) Let *e* be 3, why is this an acceptable choice for *e*.

(c) Find *d* such that *de=1(mod z)* and *d<160*.

(d) Encrypt the message m=8 using the key (n,e). Let *c* denote the corresponding ciphertext. Show all the work. [Hint: To simplify the calculations, use the fact:

[(a mod n) (b mod n)] mod n = (a b) mod n

Solutions: p = 5, q = 11
(a) n = p*q = 55, z = (p-1)(q-1) = 40
(b) e = 3 is less than n and has no common factors with z.
(c) d = 27
(d) m = 8, me = 512, Ciphertext c= me mod n = 17