The University of Newcastle
School of Electrical Engineering and Computer Science

COMP3260/COMP6360 Data Security

Week 6 Workshop – 4th and 5th April 2019
Solutions

1. For Playfair cipher, estimate the unicity distance, assuming that all keys are equally likely.

Solution:
The unicity distance is defined as U = H(k)/D, where H(k) is the entropy of the key k, and D is the redundancy of the language. For English, we estimate D=3.2
There are 25! possible keys so U = lg 25! / 3.2 ≈ 26.15

2. Decipher the ciphertext AR HM CW CO KI PW, which was enciphered using Playfair cipher with the key shown below.

H A R P S
I C O D B
E F G K L
M N Q T U
V W X Y Z

Solution:
For AR and CO, they are in the same row. When enciphering we shift one to the right, so need to shift one to the left to decipher. So AR becomes HA, and CO becomes IC.

For HM and CW, they are in the same column. When enciphering we shift one position down, so need to shift one up to decipher. HM is VE, and CW is AN.

For KI and PW, they are not in the same row or column, so we look at opposite corners, in the same row when deciphering. So KI becomes ED and PW becomes AY.

Putting all of these together, we get the plaintext HAVE A NICE DAY.

3. Suppose that the keys used with DES consist only of letters A-Z (i.e. capitals only) and are 8 letters long. Give an approximation of the length of time it would take to try all such keys using exhaustive search, assuming each key can be tested in one μsec. Do the same for keys 8 letters or digits long.

Solution:

1 μsec = $1/10^6$ sec.

Number of keys = 26^8 for just capital letters A-Z.

So $26^8 \times 10^{-6}$ sec
≈ $2.08827 \times 10^{11} \times 10^{-6}$
= 208827 sec
≈ 3480 minutes
= 58 hours

Number of keys for upper and lowercase letters, plus digits: 26+26+10=62

So 62^8 keys.

$62^8 \times 10^{-6}$ sec
≈ $2.18340106 \times 10^{14} \times 10^{-6}$
= 218340106 sec
≈ 3639002 min
≈ 60650 hours
≈ 6.9 years. (Approximately 7 years)

4. Let X' denote the bit-by-bit complement of a block X.
a) Show that if $C = DES_K(M)$, then $C' = DES_{K'}(M')$.
b) Explain how this property can be exploited in a chosen -plaintext attack to reduce the search effort by roughly 50%.

Solution:

a) The bit-by-bit complement is equivalent to 1's complement, for example,
A = 0101 B = 1111 C = 1100
A' = 1010 B' = 0000 C' = 0011
The structure of DES is that we have the initial permutation, followed by 16 rounds, then a 32-bit swap followed by the inverse of the initial permutation. Clearly, if a permutation or a swap takes a complemented input, they produce a complemented output. We need to show that that's also the case for each of the 16 rounds.

We focus on a particular round i.

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

We need to show that if the input to the round i is R_{i-1}' , L_{i-1}' and K_i' then the output of round i is L_i' and R_i'; in other words, we need to show that:

$L_i' = R_{i-1}'$
$R_i' = L_{i-1}' \oplus F(R_{i-1}', K_i')$

The first line is straightforward as R_{i-1} and R_{i-1}' are direct complements. We next show that:

$F(R_{i-1}, K_i) = F(R_{i-1}', K_i')$ and $L_{i-1}' \oplus F(R_{i-1}, K_i) = R_i'$

The structure of the $F(R_{i-1}, K_i)$ is such that it is enough to show that $R \oplus K = R' \oplus K'$ in order to show that $F(R_{i-1}, K_i) = F(R_{i-1}', K_i')$

R	K	$R \oplus K$
0	0	0
0	1	1
1	0	1
1	1	0

R′	K′	$R' \oplus K'$
1	1	0
1	0	1
0	1	1
0	0	0

For $L_{i-1}' \oplus F(R_{i-1}, K_i) = R_i'$

L_{i-1}	L_{i-1}'	$F(R_{i-1}, K_i)$	$L_{i-1} \oplus F(R_{i-1}, K_i)$	$L_{i-1}' \oplus F(R_{i-1}, K_i)$
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	0	1

Hence $L_{i-1}' \oplus F(R_{i-1}, K_i) = (L_{i-1} \oplus F(R_{i-1}, K_i))'$
$= (R_i)'$
$= R_i'$

b) Obtain under a chosen-plaintext attack ciphertexts for plaintext X and its complement X′.

Let $C_0 = DES_K(X)$ and $C_1 = DES_K(X')$

By enciphering X using key G, we are checking both G and G′:

If $DES_G(X)=C_0$ then K=G; if $DES_G(X)=C_1'$ then K=G′.

5. (Exercise from the Text)
Show that DES decryption is the inverse of DES encryption.

Solution:

Looking at the output from round 16 of the encryption, we can show that the first round of decryption gives the same result.

Encryption side:

$LE_{16} = RE_{15}$
$RE_{16} = LE_{15} \oplus F(RE_{15},K_{16})$

Decryption side:

$LD_1 = RD_0 = LE_{16} = RE_{15}$ Also $LD_0 = RE_{16}$
$RD_1 = LD_0 \oplus F(RD_0,K_{16})$
$= RE_{16} \oplus F(RE_{15},K_{16})$
$= [LE_{15} \oplus F(RE_{15},K_{16})] \oplus F(RE_{15},K_{16})$
$= LE_{15}$
Thus we have $LD_1 = RE_{15}$ and $RD_1 = LE_{15}$

Therefore, the output of the first round of the decryption process is $LE_{15}||RE_{15}$, which is the 32-bit swap of the input to the sixteenth round of the encryption. This correspondence holds all the way through the 16 iterations.

In general terms, for the i^{th} iteration.

$LE_i = RE_{i-1}$
$RE_i = LE_{i-1} \oplus F(RE_{i-1},K_i)$

Rearranging terms.

$RE_{i-1} = LE_i$
$LE_{i-1} = RE_i \oplus F(RE_{i-1},K_i) = RE_i \oplus F(LE_i,K_i)$

6. (Exercise from the Text)
To show that the 32-bit swap after the sixteenth iteration of DES is indeed needed, first consider the following notation.

$A \,||\, B$	= the concatenation of the bit strings A and B
$T_i(R||L)$	= the transformation defined by the i^{th} iteration of the encryption algorithm, for $1 \le i \le 16$
$TD_i(R||L)$	= the transformation defined by the i^{th} iteration of the decryption algorithm, for $1 \le i \le 16$
$T_{17}(R||L)$	= L||R. This transformation occurs after the sixteenth iteration of the encryption algorithm.

a) Show that the composition $TD_1(IP(IP^{-1}(T_{17}(T_{16}(L_{15}||R_{15})))))$ is equivalent to the transformation that interchanges the 32-bit halves, L_{15} and R_{15} . That is, show that

$$TD_1(IP(IP^{-1}(T_{17}(T_{16}(L_{15}||R_{15}))))) = R_{15}||L_{15}$$

b) Now suppose that we did away with the final 32-bit swap in the encryption algorithm. Then we would want the following equality to hold:

$$TD_1(IP(IP^{-1}(T_{16}(L_{15}||R_{15})))) = L_{15}||R_{15}$$

Does it?

Solution:

a) $TD_1(IP(IP^{-1}(T_{17}(T_{16}(L_{15}||R_{15}))))) = R_{15}||L_{15}$

Looking at $T_{16}(L_{15}||R_{15})$

$LE_{16} = RE_{15}$
$RE_{16} = LE_{15} \oplus F(RE_{15},K_{16})$
$LE_{15} = RE_{16} \oplus F(RE_{15},K_{16})$

So $T_{16}(L_{15}||R_{15})$
$= LE_{16} || RE_{16}$
$= RE_{15} || LE_{15} \oplus F(RE_{15},K_{16})$

$T_{17}(RE_{15} || LE_{15} \oplus F(RE_{15},K_{16}))$
$= (LE_{15} \oplus F(RE_{15},K_{16}) ||RE_{15})$

$IP(IP^{-1}(X)) = X$

So we have

$TD_1(LE_{15} \oplus F(RE_{15},K_{16}) ||RE_{15})$

$LD_1 = RD_0 = LE_{16} = RE_{15}$
$RD_1 = LD_0 \oplus F(RD_0 \oplus K_{16})$

$TD_1(LE_{15} \oplus F(RE_{15},K_{16}) ||RE_{15})$
$= RE_{15}||(LE_{15} \oplus F(RE_{15},K_{16})) \oplus F(RE_{15},K_{16})$ (cancel terms)
$= RE_{15}||LE_{15}$

b) $TD_1(IP(IP^{-1}(T_{16}(L_{15}||R_{15})))) = L_{15}||R_{15}$

$T_{16}(L_{15}||R_{15})$
$= LE_{16} || RE_{16}$
$= RE_{15} || LE_{15} \oplus F(RE_{15},K_{16})$

$IP(IP^{-1}(X)) = X$

So

$TD_1(RE_{15} \,||\, LE_{15} \oplus F(RE_{15},K_{16}))$

$LD_1 = RD_0$
$RD_1 = LD_0 \oplus F(RD_0 \oplus K_{16})$

So $TD_1(RE_{15} \,||\, LE_{15} \oplus F(RE_{15},K_{16}))$
$= LE_{15} \oplus F(RE_{15},K_{16}) \,||\, RE_{15} \oplus F(LE_{15} \oplus F(RE_{15},K_{16}) \oplus K_{16})$
$= RE_{16} \,||\, RE_{15} \oplus F(RE_{16} \oplus K_{16})$
$= RE_{16} \,||\, LE_{16} \oplus F(RE_{16} \oplus K_{16})$

which is like encrypting a further round.

7. What are the sub-keys for the DES key of all 1's and the DES key of all 0's?

Solution:
Since key bits are only ever transposed and not altered. The sub-keys will be all 1's for the key with all 1's and all 0's for the key with all 0 's.

8. The following 4 DES keys are known as "weak keys". Find out why.

0101 0101 0101 0101
1F1F 1F1F 0E0E 0E0E
FEFE FEFE FEFE FEFE
E0E0 E0E0 F1F1 F1F1

Solution:
For DES the key that is used as input for the key generation function is actually the 56 bit key, plus 8 odd parity bits which are used to ensure that the key has not been corrupted. The parity bits are chosen to be either 0 or 1, to ensure that there is an odd number of 1's in the byte. For example, if the original 7 bits of the key are 0001111 we add a 1 so that we have an odd number of 1's, resulting in 00011111 or 1F. If the first 7 bits were 0011001 we add 0, resulting in 00110010.

a) 0101 0101 0101 0101 gives a 64 bit key of
00000001 00000001 00000001 00000001 00000001 00000001 00000001 00000001
where every 8th bit is an odd parity bit, which is ignored in the first permutation. This means that the resulting sub-keys consist of all 0's.

b) 1F1F 1F1F 0E0E 0E0E gives a 64 bit key of

00011111 00011111 00011111 00011111 00001110 00001110 00001110 00001110 results in C0 = 0000000 0000000 0000000 0000000 and D0 = 1111111 1111111 1111111 1111111
This means that all generated sub-keys are identical.

c) FEFE FEFE FEFE FEFE gives a 64 bit key of
11111110 11111110 11111110 11111110 11111110 11111110 11111110 11111110 where every 8th bit is an odd parity bit, which is ignored in the first permutation. This means that the resulting sub-keys would consist of all 1's.

d) E0E0 E0E0 F1F1 F1F1 gives a 64 bit key of
11100000 11100000 11100000 11100000 11110001 11110001 11110001 11110001 results in C0 = 1111111 1111111 1111111 1111111 and D0 = 0000000 0000000 0000000 0000000
This means that all generated sub-keys are identical.

9. In the previous question we have identified some "weak keys" for DES, each of which produces identical sub-keys. Explain how in the case of a weak key in a chosen plaintext attack the Feistel cipher can be broken to discover the plaintext corresponding to the intercepted ciphertext.

Solution: In a Feistel cipher, decryption is exactly the same as encryption except that the sub-keys are used in reverse order. That makes encryption identical to decryption and an intruder who is in a position to perform a chosen plaintext attack can perform the encryption of the ciphertext and thus recover the plaintext.

10. In addition to 4 weak keys identified in Question 2, are there any other weak keys for DES? Prove your answer.

Solution: No. For sub-keys to be identical, we must have $C_0 = C_1 = \ldots = C_{15}$, and similarly $D_0 = D_1 = \ldots = D_{15}$; in every round there is circular left shift by one or two bits; a circular left shift by one bit produces the same number if and only if all bits are the same – either all 0's or all 1's. Thus there are only four different possibilities for C_0 and D_0, and they are precisely the four keys that we've seen.

In addition to "weak" keys, there are also "semi-weak" keys that produce 2 different sub-keys, instead of 16.