Exercise 2

Module 1 - Introduction to Cryptography and Data Security

Knut Lucas Andersen
Master Applied Information Security
isits AG International School of IT Security

May 15, 2017

Contents

1 DES-history

a) What was/is the name of the authority that called out for the development of DES?

DES was developed by IBM in cooperation with NSA, because the US government asked for a standard. The representative for the US Goverment asking for the standard was NIST (National Institute of Standards and Technology), who at the time was known as NBS (National Bureau of Standards) (Paar and Pelzl, 2010, p. 56).

b) In which year DES was standardised?

DES was first standardised in 1977, but it was only standardised for 10 years (up until 1987) (Paar and Pelzl, 2010, p. 56).

c) Which authority also was said to have influenced the standardisation of DES?

NSA was the authority said to have influenced the DES standard.

d) Which company did submit the cipher?

IBM was the company that submitted the cipher, which was based on the Lucifer cipher. IBM originally named it Lucifer, but after the review from NSA a modified version was returned named DES (Trappe and Washington, 2006, p. 113).

e) Which kind of structure is used in Lucifer?

The structure used in Lucifer is the Feistel system (or Feistel Network (Paar and Pelzl, 2010, p. 58)), named after Horst Feistel who was part of the Lucifer development team (Trappe and Washington, 2006, p. 114).

f) Which key length was supported by Lucifer, originally?

The original key length supported by Lucifer was 128-bits (Alfred J. Menezes, 1997, p. 276-277), (Paar and Pelzl, 2010, p. 56).

2 Basics of block ciphers

a) What meaning does the (balanced) Feistel-network have for the processing of data?

It shortens the processing time by only doing the mathematical operation on half the input. The other half is passed into the next round where it then is processed in the same way.

b) Point out the characteristics of a Feistel-structure related to the encryption and decryption.

The main characteristic is that encryption and decryption are almost the same, and uses the same key for both operations. When decrypting, you are simply reversing the encryption process, where the only difference is that for decryption you need a reversed key schedule.

c) Claude Shannon says there are two types of primitive operations to build up strong encryption algorithms.

The two types of primitive operations named by Shannon is Confusion and Diffusion. The aim of Confusion is to obscure the relationship between the plain - and cipher-text (e.g. substitution). Diffusion focuses on hiding the plain-texts statistical properties, by spreading each plain-text bit over many cipher-text bits (e.g. permutation).

3 Avalanche effect in DES

a) Which S-boxes are influenced by this bit in the first round. Also calculate the input bits of all S-boxes.

Input-String: $\{x_1 = 0, \ldots, x_{19}=1, \ldots, x_{64} = 0\}$
56-bit key: $\{k_1 = 0, \ldots, k_{56} = 0\}$
IP-String: $\{x_1 = 0, \ldots, x_{45}=1, \ldots, x_{64} = 0\}$

DES contains 16 rounds after the initial permutation. Through each round, the bits are split into two; left and right side. The right side is passed directly into the next rounds left side, but the current left side is encrypted by XOR'ing the content with the f-function. The f-function contains 4 steps:

1. Expansion: Using an expansion box (E-Box), going from 32-bits to 48-bits
2. XOR with the 48-bits round key
3. S-box substitution
4. Permutation (to go back from 48-bits to 32-bits)

The right side consists of 32-bits, where the 13^{th} bit is set to 1. Based on the E-box shown in (Paar and Pelzl, 2010, p. 63), the E-box looks like this: E-Box: $\{x_1 = 0, \ldots, x_{18}=1, x_{19} = 0, x_{20}=1, \ldots, x_{48} = 0\}$. XOR'ing the E-box with the round key makes no changes (because $0 \oplus 0 = 0$; $1 \oplus 1 = 0$).

Which S-box(es) are affected by the set bit (marked with 'X') and how do they look like?
Each S-box contains 6-bits, where the 6-bits comes from the result of the E-box $\oplus$ Round-key$_{i=1}$. This means that the only "affected" S-boxes are S3 and S4.

S1 ☐ S2 ☐ S3 ☒ S4 ☒ S5 ☐ S6 ☐ S7 ☐ S8 ☐

S-Box 1: 14 = 1110[1]

0	0	0	0	0	0

S-Box 2: 15 = 1111

0	0	0	0	0	0

S-Box 3: 13 = 1101

0	0	0	0	0	1

S-Box 4: 01 = 0001

0	1	0	0	0	0

S-Box 5: 02 = 0010

0	0	0	0	0	0

S-Box 6: 12 = 1100

0	0	0	0	0	0

S-Box 7: 04 = 0100

0	0	0	0	0	0

S-Box 8: 13 = 1101

0	0	0	0	0	0

S-box output:

1	1	1	0	1	1	1	1
1	1	0	1	0	0	0	1
0	0	1	0	1	1	0	0
0	1	0	0	1	1	0	1

After permutation on the outputted S-boxes[2]:

1	1	0	1	1	1	0	0
1	0	0	1	1	0	0	1
1	1	0	0	1	0	1	0
1	0	1	1	1	1	0	0

b) Calculate the bit string at the end of the first round. (L1 and R1)

$L_1 = R_0$

0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

[1]Based on S-box tables shown in (Paar and Pelzl, 2010, p. 64-65).
[2]Based on permutation table shown in (Paar and Pelzl, 2010, p. 66).

$R_1 = L_1 \oplus f(R_0 \oplus K_0)$

1	1	0	1	1	1	0	0
1	0	0	1	0	0	0	1
1	1	0	0	1	0	1	0
1	0	1	1	1	1	0	0

c) Calculate the output bits for the case that all input bits are zeros! How many bits did change in L1 and R1 compared to exercise b)?

Input-String: $\{x_1 = 0, \ldots, x_{64} = 0\}$
56-bit key: $\{k_1 = 0, \ldots, k_{56} = 0\}$
IP-String: $\{x_1 = 0, \ldots, x_{64} = 0\}$

S-Box 1: 14 = 1110[3]

0	0	0	0	0	0

S-Box 2: 15 = 1111

0	0	0	0	0	0

S-Box 3: 10 = 1010

0	0	0	0	0	0

S-Box 4: 07 = 0111

0	0	0	0	0	0

S-Box 5: 02 = 0010

0	0	0	0	0	0

S-Box 6: 12 = 1100

0	0	0	0	0	0

S-Box 7: 04 = 0100

0	0	0	0	0	0

S-Box 8: 13 = 1101

0	0	0	0	0	0

S-box output:

1	1	1	0	1	1	1	1
1	0	1	0	0	1	1	1
0	0	1	0	1	1	0	0
0	1	0	0	1	1	0	1

After permutation on the outputted S-boxes[4]:

1	1	0	1	1	0	0	0
1	1	0	1	1	0	0	0
1	1	0	1	1	0	1	1
1	0	1	1	1	1	0	0

[3]Based on S-box tables shown in (Paar and Pelzl, 2010, p. 64-65).
[4]Based on permutation table shown in (Paar and Pelzl, 2010, p. 66).

$L_1 = R_0$

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

$R_1 = L_1 \oplus f(R_0 \oplus K_0)$

1	1	0	1	1	0	0	0
1	1	0	1	1	0	0	0
1	1	0	1	1	0	1	1
1	0	1	1	1	1	0	0

Amount of flipped bits: 7.
1 in L_1 and 6 in R_1.

4 Non-linearity of S-boxes

$S_i(x_1) \oplus S_i(x_2) \neq S_i(x_1 \oplus x_2)$

a) x1 = 000001, x2 = 100000

$S_5(x_1) = S_5(000001) = 14 = 1110$
$S_5(x_2) = S_5(100000) = 04 = 0100$

$S_5(x_1) \oplus S_5(x_2) = 1110 \oplus 0100 = \underline{1010}$

$x_1 \oplus x_2 = 000001 \oplus 100000 = 100001$
$S_5(x_1 \oplus x_2) = S_5(100001) = 11 = \underline{1011}$

$\underline{\underline{S_5(x_1) \oplus S_5(x_2) \neq S_5(x_1 \oplus x_2) \Rightarrow 1010 \neq 1011}}$

b) x1 = 001100, x2 = 111001

$S_5(x_1) = S_5(001100) = 11 = 1011$
$S_5(x_2) = S_5(111001) = 10 = 1010$

$S_5(x_1) \oplus S_5(x_2) = 1011 \oplus 1010 = \underline{0001}$

$x_1 \oplus x_2 = 001100 \oplus 111001 = 110101$
$S_5(x_1 \oplus x_2) = S_5(110101) = 00 = \underline{0000}$

$\underline{\underline{S_5(x_1) \oplus S_5(x_2) \neq S_5(x_1 \oplus x_2) \Rightarrow 0001 \neq 0000}}$

c) x1 = 010011, x2 = 011110

$S_5(x_1) = S_5(010011) = 00 = 0000$
$S_5(x_2) = S_5(011110) = 09 = 1001$

$S_5(x_1) \oplus S_5(x_2) = 0000 \oplus 1001 = \underline{1001}$

$x_1 \oplus x_2 = 010011 \oplus 011110 = 001101$
$S_5(x_1 \oplus x_2) = S_5(001101) = 13 = \underline{1101}$

$S_5(x_1) \oplus S_5(x_2) \neq S_5(x_1 \oplus x_2) \Rightarrow 1001 \neq 1101$

5 Brute-Force Attacke auf den DES

a) How many chips of this kind do we have to run parallel so we can calculate the DES-Key in a single day?

-

b) How much would these chips cost if one chip costs 10 Euro and we calculate 100% overhead for running the chips parallel, the power supply and anything else?

-

c) Why is this design of such a key searching machine only the upper limit of security?

Because:

> "DES has four weak keys and six pairs of semi-weak keys" (Alfred J. Menezes, 1997, p. 257).

What this means is that DES can create identical sub-keys, which then also causes the encryption and decryption function to coincide. Furthermore if an attacker get hold of large amounts of cipher - and plain-text, an analytical attack would be possible.

Another issue, which may not be that relevant for this question, is also the aspect of social engineering. To use a classic comic strip as an example - the visualization is that hackers would spend millions on dollars trying to build the "perfect" machine. When in fact they would just go to a tool shop, buy a $5 hammer, and beat up the person with the right access. Therefore one should also invest just as much into "human security" as hardware/software security.

6 DES bit complement

-

References

Alfred J. Menezes Paul C. van Oorschot, andScott A. Vanstone (1997). *Handbook of applied cryptography*. CRC Press. ISBN: 978-08493-8523-0.

Paar, Christof and Jan Pelzl (2010). *Understanding Cryptography: A textbook for students and practitioners*. Springer. ISBN: 978-3-642-04100-6. DOI: 10.1007/978-3-642-04101-3.

Trappe, Wade and Lawrence C. Washington (2006). *Introduction to cryptography with coding theory*. 2nd ed. Pearson Prentice Hall. ISBN: 0-13-186239-1.