Innopolis University

SYSTEM AND NETWORKING ENGINEERING

Security of Systems and Networks

LABORATORY REPORT 4

Symmetrical Encryption

Student	**ID**
Grebennikov Sergey	47611

Lecturer
Dr. Rasheed Hussain, PhD

November 4, 2017

Contents

1 DES 2

2 AES 3

3 Bonus: RC4 3

4 Bonus: AES 4

5 Conclusion 5

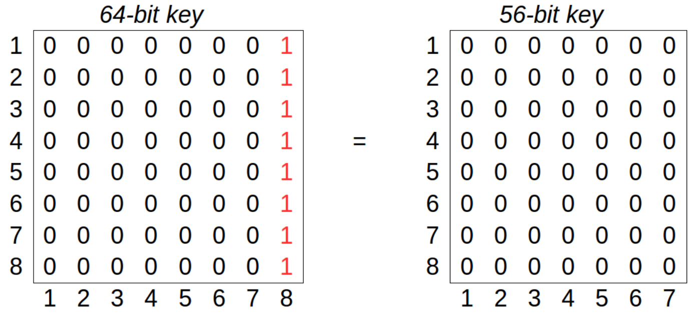
1 DES

The Cryptool 1.x suite contains a simulator for the DES encryption algorithm in the menu ”Individual Procedures/Visualization of Algorithms/DES”. Watch this animation.

1. Next, use the DES simulator at http://lpb.canb.auug.org.au/adfa/src/DEScalc/index.html
Step through the process of encrypting your name with the key 0x0101010101010101 and write the internal state of the device at the 8th round.
Answer:

```
Rnd8     f(R7=67d4de66, SK8=00 00 00 00 00 00 00 00 ) = 79ea2b57
```

2. Inspect the key schedule phase for the given key and explain how the sub keys are generated for each of the 16 steps.
Answer:
Before doing a permutation PC1, the rightmost bits (every 8th bit) are stripped. These bits are for parity checking and have no further influence on the encryption. See Figure 1



64-bit key

1	0	0	0	0	0	0	0	1
2	0	0	0	0	0	0	0	1
3	0	0	0	0	0	0	0	1
4	0	0	0	0	0	0	0	1
5	0	0	0	0	0	0	0	1
6	0	0	0	0	0	0	0	1
7	0	0	0	0	0	0	0	1
8	0	0	0	0	0	0	0	1
	1	2	3	4	5	6	7	8

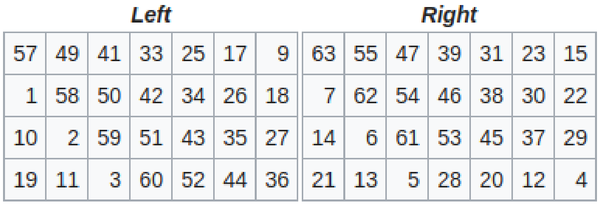
=

56-bit key

1	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0
	1	2	3	4	5	6	7

Figure 1: Transform 64-bit key to 56-bit key

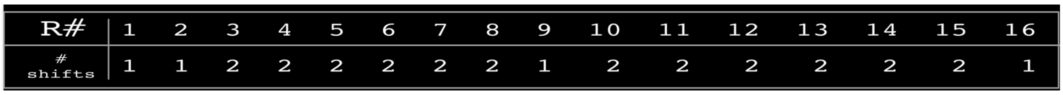
The 56 bits of the key are selected by Permuted Choice 1 (PC-1) matrix. The 56 bits are then divided into two 28-bit halves; each half is thereafter treated separately. (Figure 2)



Left							Right						
57	49	41	33	25	17	9	63	55	47	39	31	23	15
1	58	50	42	34	26	18	7	62	54	46	38	30	22
10	2	59	51	43	35	27	14	6	61	53	45	37	29
19	11	3	60	52	44	36	21	13	5	28	20	12	4

Figure 2: Permuted Choice 1 (PC-1) matrix

In successive rounds, both halves are rotated left by one or two bits specified by table (Figure 3) for each round.

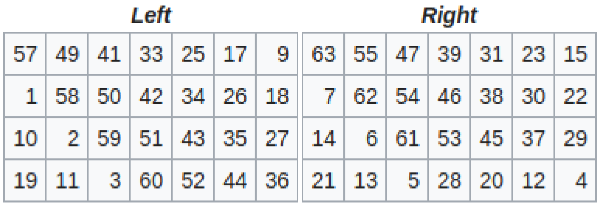


R#	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
# shifts	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Figure 3: Left Circular Shift

Then 48 subkey bits are selected by Permuted Choice 2 (PC-2) matrix - 24 bits from the left half, and 24 from the right. (Figure 4)

As a result, we will get 16 subkeys for each round to proceed to the DES core function. In every round, one subkey is used to perform an encryption on a specified block.



Left							Right						
57	49	41	33	25	17	9	63	55	47	39	31	23	15
1	58	50	42	34	26	18	7	62	54	46	38	30	22
10	2	59	51	43	35	27	14	6	61	53	45	37	29
19	11	3	60	52	44	36	21	13	5	28	20	12	4

Figure 4: Permuted Choice 2 (PC-2) matrix

3. Comment on the behavior of DES when using the given key.
   Answer:
   As a result each subkey will have the same value and equal to **0** independent of the permutation. Which significantly reduces its cryptographic strength.

2 AES

The Cryptool 1.x suite contains an animation for the AES encryption algorithm. Watch the whole animation of AES using the Rijndael Animation flash video at `http://poincare.matf.bg.ac.rs/~ezivkovm/nastava/rijndael_animacija.swf`

4. Identify the Shannon diffusion element(s).
   Answer:
   Diffusion refers to dissipating the statistical structure of plaintext over the bulk of ciphertext.

5. Also identify the Shannon confusion element(s).
   Answer:
   Confusion refers to making the relationship between the ciphertext and the symmetric key as complex and involved as possible.

3 Bonus: RC4

Follow the instructions at `http://bit.ly/2gWJvqV`, identify the URL and your personal archive accordingly, download it and inspect its contents. There are two files encrypted with the RC4 cipher. One of the files was encrypted using a 40 bit key that when represented in ASCII starts with the character a and contains only lowercase letters while the other uses a 48 bit key that can be written only with digits. Identify the encrypted files and using the brute force tool from `https://gist.github.com/4017169` find the keys and decrypt the file. (Most likely you will have to install the pycrypto and numpy libraries first.)

DECRYPTED MESSAGE:

First file:

```
The Project Gutenberg EBook of On the Origin of Species, by Charles Darwin

This eBook is for the use of anyone anywhere at no cost and with
almost no restrictions whatsoever.  You may copy it, give it away or

...
```

Second file:

```
  T h e  Project Gutenberg EBook of On the Origin of Species, by Charles Darwin

This eBook is for the use of anyone anywhere at no cost and with
almost no restrictions whatsoever.  You may copy it, give it away or

...
```

6. (a) How did you identify the encrypted files?
   Answer:
   I read each file with `less` utility and two of them had a binary format.

(b) What is the effective key strength for each of the keys?
Answer:
Greater than 239 bits, since *entropy* < 7.9 doesn't look like a random stream.

$$\log_2 238.9 \approx 7.9$$

(c) Instrument the code to find out how many decryption attempts you can perform in one second. Where is the most time spent ?

(d) Modify the code to support parallel execution and calculate the speedup.
Answer:

```
__author__ = ’cdumitru’

import sys
from Crypto.Cipher import ARC4
import numpy
import string
import itertools
from multiprocessing import Pool
from time import time
import cProfile

...

def parallel():
    """
    Starts a number of threads that search through the key space
    """
    p = Pool(CPU_COUNT)
    p.map(worker, gen(), chunksize=2)
    p.close()
    p.join()

def serial():
    worker(tuple())


if __name__ == "__main__":
#       serial()
        parallel()
```

(e) If the same message would be encrypted with a key with of length 48 bits but which uses all the printable characters how much time would it take to explore the full key space ?
Answer:
24 years

4 Bonus: AES

7. Modify the code to support AES brute-force in CBC mode. How many keys can you test per second? Julian Assange has released an insurance file encrypted with AES256. Assuming that no disruptive technological breakthrough will take place in the future and the performance of CPUs will double every 18 months, when will it be possible to brute-force the file in reasonable time, i.e. less than 1year, using a single computer?

5 Conclusion

Symmetric-key algorithms are algorithms for cryptography that use the same cryptographic keys for both encryption of plaintext and decryption of ciphertext. The keys may be identical or there may be a simple transformation to go between the two keys. The keys, in practice, represent a shared secret between two or more parties that can be used to maintain a private information link. This requirement that both parties have access to the secret key is one of the main drawbacks of symmetric key encryption, in comparison to public-key encryption (also known as asymmetric key encryption).

Symmetric-key encryption can use either stream ciphers or block ciphers.

- Stream ciphers encrypt the digits (typically bytes) of a message one at a time.
- Block ciphers take a number of bits and encrypt them as a single unit, padding the plaintext so that it is a multiple of the block size. Blocks of 64 bits were commonly used. The Advanced Encryption Standard (AES) algorithm approved by NIST in December 2001, and the GCM block cipher mode of operation use 128-bit blocks.

References

[1] M. Stamp, Information Security: Principles and Practice, Second Edition, 2011, 606 pages.

[2] Data Encryption Standard `https://en.wikipedia.org/wiki/Data_Encryption_Standard`.

[3] Advanced Encryption Standard `https://en.wikipedia.org/wiki/Advanced_Encryption_Standard`.

[4] RC4 `https://en.wikipedia.org/wiki/RC4`.