# CS553: Cryptography

## Assignment 5: Solutions

Rohit Das (11910230)

September 8, 2019

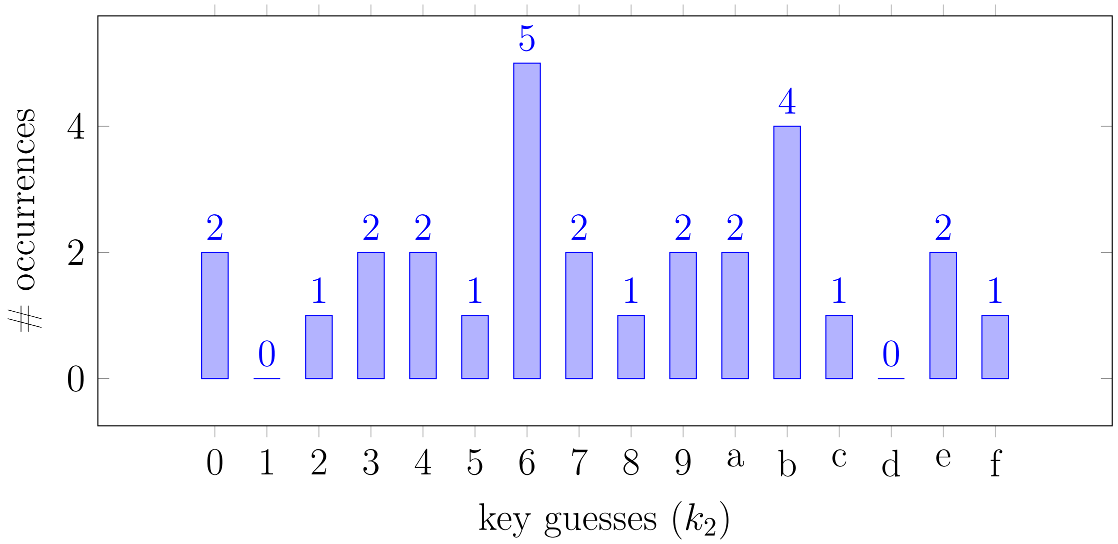
# 1. DC of Sypher002

12-bit key = (4||e||6)

Since we need the characteristic $f \overset{S}{\rightarrow} d$, the difference for the message pairs taken is "f".

Below is the matrix generated for guesses of $k_2$, for which $v_0^{'} \oplus v_1^{'} = d$:

| $k_2$ guess<br>msg pairs | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | a | b | c | d | e | f |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 - f | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 - e | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 - d | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| 3 - c | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 4 - b | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 - a | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 6 - 9 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 7 - 8 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| $\sum_{\textit{over all pairs}}$ | 2 | 0 | 1 | 2 | 2 | 1 | 5 | 2 | 1 | 2 | 2 | 4 | 1 | 0 | 2 | 1 |



From the matrix and SNR graph, it is evident that the signal for $k_2 = 6$ is highest, which is correct given our selection of keys.

## 2. Hamsi vs CryptWizard001

The Hamsi S-box is:

| $x$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | a | b | c | d | e | f |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $S(x)$ | 8 | 6 | 7 | 9 | 3 | c | a | f | d | 1 | e | 4 | 0 | b | 5 | 2 |

The DDT for Hamsi S-box is:

| in\out | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | a | b | c | d | e | f |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 16 | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - |
| 1 | - | - | - | - | - | 2 | - | 2 | - | - | 2 | 2 | 2 | - | 4 | 2 |
| 2 | - | - | - | 4 | - | 4 | - | - | - | 4 | - | - | - | - | - | 4 |
| 3 | - | 4 | 2 | - | - | - | 2 | - | - | 2 | - | - | 2 | - | 2 | 2 |
| 4 | - | - | - | - | - | - | 4 | - | - | - | 4 | 4 | - | 4 | - | - |
| 5 | - | 4 | - | 2 | 2 | 2 | 2 | - | 2 | - | - | - | 2 | - | - | - |
| 6 | - | - | 2 | 2 | 2 | 2 | - | - | 2 | 2 | - | - | - | - | 2 | 2 |
| 7 | - | - | - | - | 4 | 2 | - | 2 | - | - | 2 | 2 | 2 | - | - | 2 |
| 8 | - | - | - | 2 | - | 2 | - | 4 | - | 2 | - | - | - | 4 | - | 2 |
| 9 | - | - | - | 2 | - | - | - | 2 | 4 | 2 | 2 | 2 | 2 | - | - | - |
| a | - | - | 2 | - | 2 | - | 4 | - | 2 | - | 4 | - | - | - | 2 | - |
| b | - | 4 | - | - | 2 | - | 2 | - | 2 | 2 | - | - | 2 | - | - | 2 |
| c | - | - | 2 | - | 2 | - | - | - | 2 | - | - | 4 | - | 4 | 2 | - |
| d | - | 4 | 2 | 2 | - | 2 | 2 | - | - | - | - | - | 2 | - | 2 | - |
| e | - | - | 2 | - | 2 | - | - | 4 | 2 | - | - | - | - | 4 | 2 | - |
| f | - | - | 4 | 2 | - | - | - | 2 | - | 2 | 2 | 2 | 2 | - | - | - |

There are 72 places with entry:"2" and 24 with entry:"4".

CryptWizard001's S-box is:

| $x$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | a | b | c | d | e | f |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $S(x)$ | c | 6 | 7 | 9 | 4 | 8 | a | f | 5 | 1 | e | 3 | 0 | b | d | 2 |

The DDT for CryptWizard001's S-box is:

| in\out | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | a | b | c | d | e | f |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 16 | - | - | - | - | - | - | - | - | - | - | - | - | - | - | - |
| 1 | - | - | - | - | 2 | 2 | - | - | - | - | 2 | 2 | 2 | 2 | 2 | 2 |
| 2 | - | - | 2 | - | - | - | - | 2 | - | 2 | - | 4 | - | 2 | 2 | 2 |
| 3 | - | 2 | 4 | - | - | 2 | 4 | - | - | - | - | 2 | - | - | - | 2 |
| 4 | - | 2 | - | 2 | - | 2 | 2 | - | 2 | - | 2 | - | - | 2 | 2 | - |
| 5 | - | 2 | 2 | 2 | 2 | - | - | - | 2 | - | - | - | 2 | - | 4 | - |
| 6 | - | 2 | - | 4 | - | - | 2 | - | 4 | 2 | - | - | - | - | 2 | - |
| 7 | - | - | - | 4 | - | 2 | - | 2 | - | - | - | - | 4 | 2 | - | 2 |
| 8 | - | - | - | 2 | 2 | - | - | 4 | - | 4 | 2 | - | - | 2 | - | - |
| 9 | - | - | 2 | 2 | 2 | - | - | 2 | 4 | - | - | - | - | 2 | - | 2 |
| a | - | - | 4 | - | 2 | 2 | - | - | 2 | 2 | 4 | - | - | - | - | - |
| b | - | 2 | - | - | - | 2 | 4 | - | 2 | - | - | - | 2 | - | - | 4 |
| c | - | 2 | - | - | 2 | - | - | - | - | 2 | 2 | 2 | 4 | 2 | - | - |
| d | - | 2 | - | - | 2 | 4 | 2 | 2 | - | 2 | - | - | - | 2 | - | - |
| e | - | 2 | 2 | - | 2 | - | - | 2 | - | - | 2 | 2 | - | - | 2 | 2 |
| f | - | - | - | - | - | - | 2 | 2 | - | 2 | 2 | 4 | 2 | - | 2 | - |

There are 84 places with entry:"2" and 18 with entry:"4".

A fixed point in an S-Box is a property where S[x] = x. For CryptWizard001's S-Box, there is one fixed point, where S[4] = 4. This can be used to launch Fixed Point Attack on ciphers (Bard G.V. (2009) The Fixed-Point Attack. In: Algebraic Cryptanalysis. Springer, Boston, MA, pp 17-28). Algorithms that generate highly secure and non-linear S-Boxes work to avoid creating fixed points.

Since CryptWizard001's S-Box creats a fixed point, it is more vulnerable to attacks. Hence, CryptWizard001 is a liar.

## 3. Conforming Message Pairs

The 16-bits keys used for Sypher004 are = (’c253’, ’0466’, ’affe’, ’645e’, ’0509’, ’fff2’).

Python code (Python 3): conform_msg.py

```
# messages taken as input from input.txt
import numpy as np
import sys

orig_stdout = sys.stdout
f = open('output_cnfrm.txt','w')
sys.stdout = f

sbox1 ={0x0: 0x6, 0x1: 0x4, 0x2: 0xc, 0x3: 0x5, 0x4: 0x0,
        0x5: 0x7, 0x6: 0x2, 0x7: 0xe, 0x8: 0x1, 0x9: 0xf,
        0xa: 0x3, 0xb: 0xd, 0xc: 0x8, 0xd: 0xa, 0xe: 0x9,
        0xf: 0xb}

def sbox_4x(msg, bits): # 4-input sbox
    if len(msg) != int(bits,0):# check for message length
        exit("Plaintext size should be of " + bits + " bits")
    subs = []
    for m in msg: # check for invalid literal
        if m not in [format(i,'x') for i in sbox1]:
            exit("Invalid literal")
        subs.append(format(sbox1[int(m,16)],'x'))
    return subs

def pbox(sbox_out,bits):
    if len(sbox_out) != int(bits,0):
        exit("Too small sbox output!!")
    perm = [b for a in sbox_out # changing hex to binary
            for b in list("{0:04b}".format(int(a,16)))]
    # pbox = sbox output in numpy array and transposing
    perm = np.asarray(perm)
    perm = np.reshape(perm,(4,4)).transpose()
    return "".join([format(int("".join(a),2),'x')
                   for a in perm])

def sypher004(msg, bits):
    # uncomment below lines for more verbose output
    keys = ['c253','0466','affe','645e','0509','fff2'] # keys
    if len(msg) != int(bits,0):
        exit("Plaintext size should be of "
             + bits + " bits")
    # print("Round 0:",msg)
    msg = "{:04x}".format(int(msg,16) ^ int(keys[0],16),'x')
    # print("Round 0 (XOR) :","{0:016b}".format(int(msg,16)))

    for i in range(1,len(keys) - 1):
        msg = "".join(sbox_4x(msg,bits))
        # print("Round",str(i),"(Sbox):","{0:016b}"
        # .format(int(msg,16)))
        msg = pbox(msg,bits)
        # print("Round",str(i),"(Perm):","{0:016b}"
        # .format(int(msg,16)))
        msg = "{:04x}".format(int(msg,16)^int(keys[i],16),'x')
        # print("Round",str(i)," (XOR):","{0:016b}"
        # .format(int(msg,16)))
    return msg


def conform_pairs():
    print(" m0 "," m1 "," c0 "," c1 ",)
    print("-"*19)
    count = 0
    with open("./input.txt") as file:
        for line in file:
            m0,m1 = line.strip().split(" ")
            c0 = sypher004(m0,"4")
            c1 = sypher004(m1,"4")
            if int(c0,16)^int(c1,16) == int('8000',16):
                print(m0,m1,c0,c1)
                count += 1
    sys.stdout = orig_stdout
    f.close()
    print("There are",count,"conforming message pairs.")


conform_pairs()
```

Actual output (file generated by code contains conforming message pairs and cipher pairs): output_cnfrm.txt

Number of conforming message pairs = 356

$\therefore$ Probability of differential $(8,0,0,0) \xrightarrow{S} ? \xrightarrow{S} (8,0,0,0) = \dfrac{356}{2^{16}} = 0.00543$.

The conditions for filtering here are to remove message-pairs which activate S-Boxes for nibbles 2,3 and 4, as well as those which give differences that do not match that of differential for S-Box 1.