Lucerne University of
Applied Sciences and Arts

HOCHSCHULE
LUZERN

Information Technology
FH Zentralschweiz

Secret-key or symmetric cryptography Exercise

Prof. Dr. Josef F. Bürgler

I.BA_MOVK_MM, Semesterweek 03

Write the solution as a Jupyter notebook. This should run when requested in front of the class.
Due-time: The teacher will check the Jupyter notebooks one week after we discussed the topic!

1 DES S-box S_3 F18

The input to the DES S-box S_3 is 110111. What's the output? Use Wikipedia, google, a book or some other source to find the table for S_3.

2 3DES 3DES sind 48 Runden anstatt DES 16 Runden

What is the effective key size of 3DES and why is it not 168 bits?
Effektive Schlüssellänge von 112 Bit aufgrund der meet in the middle attack.

3 Differences between AES and Rijndeal

What are the differences between the AES candidate Rijndeal and AES with respect to block size, key size and number of rounds?

4 AES S-box

If we input the byte 11011101 into the AES S-box, what's the output? Use the table in slides!

5 Other Block ciphers

Compare DES, 3DES and AES with other block ciphers like IDEA, Blowfish, Twofisch, RC5, RC6, Serpent and three more of Your choice. Make a table that shows key size, effective key size, block size, number of rounds, relative velocity of a hard- or software implementation.

6 Modes of operation

You should be able to produce sketches of the 5 modes of operation and You should be able to write down the equations, relating, IVs (if any), plaintext block, key, ciphertext block, encryption and decryption, XOR.
You should also understand the influence of a one-bit error in the ciphertext block.

7 RC4

Use python in Jupyter Notebook to programm RC4. Do some research on RC4 and find out, why it should not be used any more!
Siehe auch Webbrowser: Endgültig Schluss mit RC4 und Der Lange Abschied von RC4.

8 Trivium

Use python in Jupyter Notebook to programm Trivium. This is not an easy task: do it in groups of two!
Use `0x00000000000000000000000000000000` for the key, IV, and plaintext for initial testing.
The expected ciphertext for this should be `0xFBE0BF265859051B517A2E4E239FC97F`.
In the algorithm on slide "*Trivium — Initialization*", the $+$ represents XOR (which in python is "^"), $\cdot$ represents logical AND (which in python is "&"). The key-stream is

$$z_i = t_1 + t_2 + t_3$$

and the ith byte of the ciphertext c_i of the plaintext m_i is

$$c_i = z_i \oplus m_i$$

The following site https://asecuritysite.com/encryption/trivium might be of some help!

9 OTP

Make your own example with one-time pad. Why is it perfectly secure? Make sure, the key is truly random not used more than once and kept secret from adversaries.

Have fun with crypto!