Отчёт по лабораторной работе №8

Элементы криптографии. Шифрование (кодирование) различных исходных текстов одним ключом

Виктория Михайловна Шутенко

Содержание

# 1 Цель работы

Освоить на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.

# 2 Ход работы

Я выполняла лабораторную работу на языке python. Сначала я подключила библиотеки:

import numpy as np  
import operator as op  
import sys

По условию лабораторной работы, я создала две функции. Также я задала 2 переменные строкового типа , “Я устала и хочу спать.”, “Спокойной ночи, друг!!” и подсчитала длину строк.

p1 = "Я устала и хочу спать."  
p2 = "Спокойной ночи, друг!!"  
print(len(p1))  
print(len(p2))

Первая функция осуществляет перевод в шестнадцатеричную систему, генерирует рандомный ключ с помощью которого будет получаться сообщение в шестнадцатиричной системе и его перевод его в строку.

def encrypt(text1, text2):  
 print("text1: ", text1)  
 newtext1=[]  
 for i in text1:  
 newtext1.append(i.encode("cp1251").hex())  
 print("text1 in 16: ", newtext1)  
 print("text2: ", text2)  
 newtext2=[]  
 for i in text2:  
 newtext2.append(i.encode("cp1251").hex())  
 print("text2 in 16: ", newtext2)  
 r=np.random.randint(0,255, len(text1))  
 key=[hex(i)[2:] for i in r]  
 newkey=[]  
 for i in newkey:  
 key.append(i.encode("cp1251").hex().upper())  
 print("key in 16: ", key)  
 xortext1=[]  
 for i in range(len(newtext1)):  
 xortext1.append("{:02x}".format(int(key[i], 16) ^ int(newtext1[i],16)))  
 print("cypher text1 in 16: ", xortext1)  
 en\_text1=bytearray.fromhex("".join(xortext1)).decode("cp1251")  
 print("cypher text1: ", en\_text1)  
 xortext2=[]  
 for i in range(len(newtext2)):  
 xortext2.append("{:02x}".format(int(key[i],16)^ int(newtext2[i],16)))  
 print("cypher text2 in 16: ", xortext2)  
 en\_text2=bytearray.fromhex("".join(xortext2)).decode("cp1251")  
 print("cypher text2: ", en\_text2)  
 return key, xortext1, en\_text1, xortext2, en\_text2

Выполнила вызов этой функции:

k, t1, et1, t2, et2 = encrypt(p1,p2)

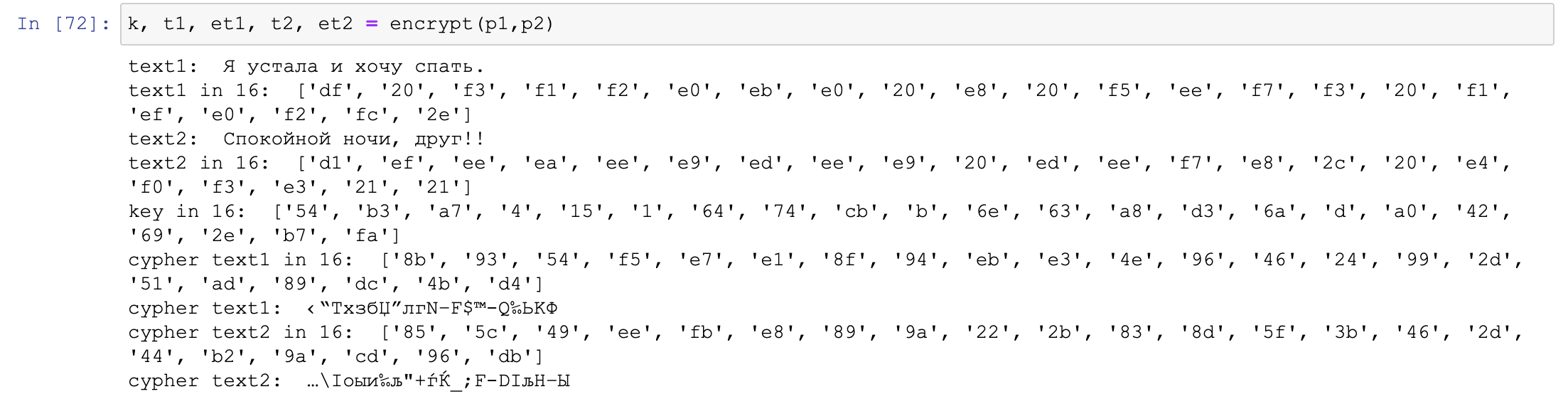


Figure 1: Результат выполнения функции crypt.

Вторая функция определяет ключ, который будет брать открытый текст и шифровать его в шестнадцатеричную систему.

def decrypt(c1, c2, p1):  
 print("cypher text1: ", c1)  
 newc1=[]  
 for i in c1:  
 newc1.append(i.encode("cp1251").hex())  
 print("cypher text1 in 16: ", newc1)  
 print("cypher text2: ", c2)  
 newc2=[]  
 for i in c2:  
 newc2.append(i.encode("cp1251").hex())  
 print("cypher text2 in 16: ", newc2)  
 print("open text1: ", p1)  
 newp1=[]  
 for i in p1:  
 newp1.append(i.encode("cp1251").hex())  
 print("open text1 in 16: ", newp1)  
 xortmp=[]  
 sp2=[]  
 for i in range(len(p1)):  
 xortmp.append("{:02x}".format(int(newc1[i],16) ^ int(newc2[i], 16)))  
 for i in range(len(p1)):  
 sp2.append("{:02x}".format(int(xortmp[i],16) ^ int(newp1[i], 16)))  
 print("open text2 in 16: ", sp2)  
 p2=bytearray.fromhex("".join(sp2)).decode("cp1251")  
 print("open text2: ", p2)  
 return p1,p2

Выполнила вызов этой функции:

decrypt(et1, et2, p1)

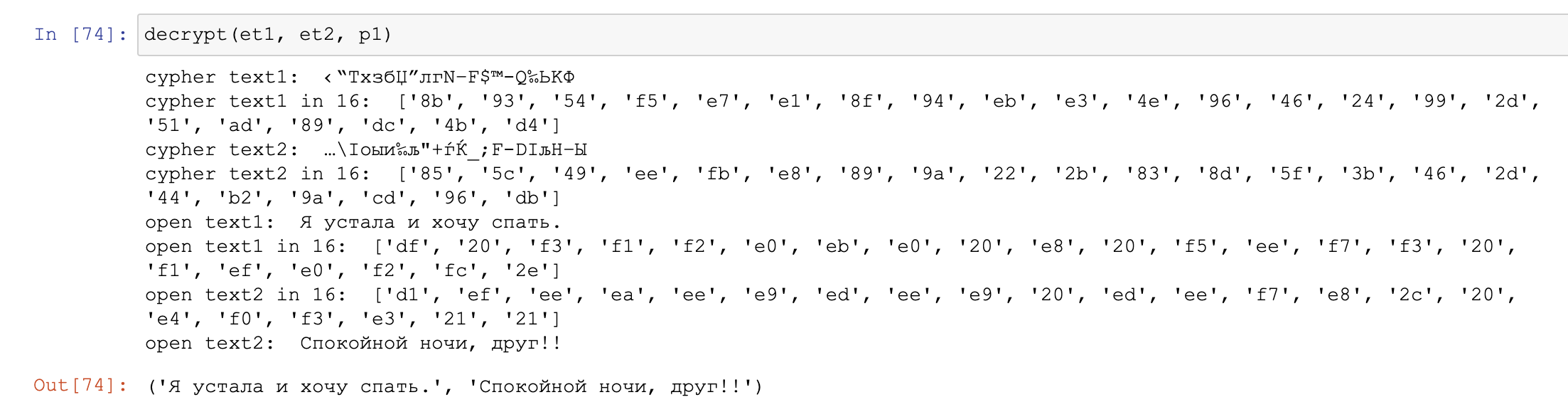


Figure 2: Результат выполнения функции decrypt.