Защита лабораторной работы №7 Элементы криптографии. Однократное гаммирование

Исаханян Эдуард Тигранович

2022 Sep 21th

RUDN University, Moscow, Russian Federation

Защита лабораторной работы №7

Цель

Цель

Освоить на практике применение режима однократного гаммирования.

Задание

Задание

1. Написать программу, которая должна определить вид шифротекста при известном ключе и известном открытом тексте;
2. Также эта программа должна определить ключ, с помощью которого шифротекст может быть преобразован в некоторый фрагмент текста, представляющий собой один из возможных вариантов прочтения открытого текста.

Функция, шифрующая данные

Ввод [8]:
```
import numpy as np
```

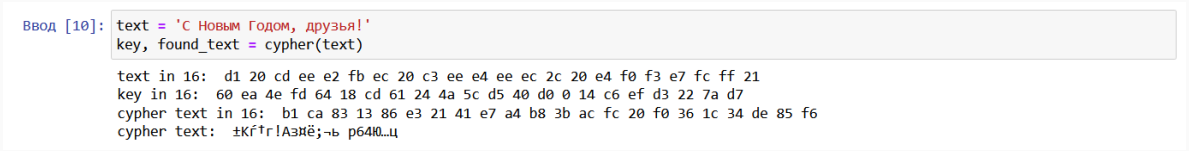
Ввод [9]:
```
def cypher(text):
    array = []
    for i in text:
        array.append(i.encode('cp1251').hex())
    print("text in 16: ", *array)

    key = np.random.randint(0, 255, len(text))
    key_16 = [hex(i)[2:] for i in key]
    print("key in 16: ", *key_16)
    crypt = []
    for i in range(len(array)):
        crypt.append("{:02x}".format(int(array[i], 16) ^ int(key_16[i], 16)))
    print("cypher text in 16: ", *crypt)

    found_text = bytearray.fromhex(''.join(crypt)).decode('cp1251')
    print("cypher text: ", found_text)
    return key_16, found_text
```

Figure 1: Функция, шифрующая данные

Результат работы функции, шифрующей данные



```
Ввод [10]: text = 'С Новым Годом, друзья!'
           key, found_text = cypher(text)

           text in 16:  d1 20 cd ee e2 fb ec 20 c3 ee e4 ee ec 2c 20 e4 f0 f3 e7 fc ff 21
           key in 16:  60 ea 4e fd 64 18 cd 61 24 4a 5c d5 40 d0 0 14 c6 ef d3 22 7a d7
           cypher text in 16:  b1 ca 83 13 86 e3 21 41 e7 a4 b8 3b ac fc 20 f0 36 1c 34 de 85 f6
           cypher text:  ±Кѓ†r!Аз¤ё;¬ь р64Ю…ц
```

Figure 2: Результат работы функции, шифрующей данные

Функция, дешифрующая данные

Ввод [11]:

```
def foundkey(text, found_text):
    print("open text: ", text)
    print("cypher text: ", found_text)
    text_16 = []
    for i in text:
        text_16.append(i.encode('cp1251').hex())
    print("text in 16: ", *text_16)
    found_text_16 = []
    for i in found_text:
        found_text_16.append(i.encode('cp1251').hex())
    print("found text in 16: ", *found_text_16)
    key = [hex(int(i, 16)^int(j, 16))[2:] for (i,j) in zip(text_16, found_text_16)]
    print("key: ", *key)
    return key
```

Figure 3: Функция, дешифрующая данные

Результат работы функции, шифрующей данные

```
Ввод [12]: found_key = foundkey(text, found_text)

open text:  С Новым Годом, друзья!
cypher text:  ±Kґ†г!Аэ¤ё;¬ь р64Ю…ц
text in 16:  d1 20 cd ee e2 fb ec 20 c3 ee e4 ee ec 2c 20 e4 f0 f3 e7 fc ff 21
found text in 16:  b1 ca 83 13 86 e3 21 41 e7 a4 b8 3b ac fc 20 f0 36 1c 34 de 85 f6
key:  60 ea 4e fd 64 18 cd 61 24 4a 5c d5 40 d0 0 14 c6 ef d3 22 7a d7
```

Figure 4: Результат работы функции, шифрующей данные

Сравнение ключей

Ввод [13]:
```
if key == found_key:
    print("key correct")
else:
    print("key incorrect")
```

```
key correct
```

Figure 5: Сравнение ключей

Вывод

Вывод

Освоили на практике применение режима однократного гаммирования.