Лабораторная работа №7

Элементы криптографии. Однократное гаммирование

Дмитрий Константинович Федотов

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

Освоить на практике применение режима однократного гаммирования.

2 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

3 Выполнение лабораторной работы

1. Написал функцию шифрования, которая определяет вид шифротекста при известном ключе и известном открытом тексте “С Новы Годом, друзья!”. Ниже представлены функция, шифрующая данные (рис - @fig:001), а также работа данной функции (рис - @fig:002).

```
Ввод [20]: def func(text):
               print("Открытый текст: ", text)
               text_ar = []
               for i in text:
                   text_ar.append(i.encode("cp1251").hex())
               print("Текст в шеснадцатеричном исполнении: ", *text_ar)

               key = np.random.randint(0, 255, len(text))
               key_hex = [hex(i)[2:] for i in key]
               print("ключ в шеснацатеричнов виде: ", *key_hex)
               text_crypt = []
               for i in range(len(text_ar)):
                   text_crypt.append("{:02x}".format(int(text_ar[i], 16)^ int(key_hex[i
               print("Зашифрованный текст в шеснадцатеричной: ", *text_crypt)

               text_ret = bytearray.fromhex("".join(text_crypt)).decode("cp1251")
               print("Зашифрованный текст: ", text_ret)
               return key_hex, text_ret
```

Рис. 3.1: Функция, шифрующая данные

```
Ввод [21]: start_text = "С Новым Годом, друзья!"
           crypt_key, text_crypt = func(start_text)

           Открытый текст:  С Новым Годом, друзья!
           Текст в шеснадцатеричном исполнении:  d1 20 cd ee e2 fb ec 20 c3 ee e4 ee e
           c 2c 20 e4 f0 f3 e7 fc ff 21
           ключ в шеснацатеричнов виде:  cb de 74 22 b ad f9 67 98 58 f8 9b dc e2 ad 6
           d 83 40 24 a0 3c b8
           Зашифрованный текст в шеснадцатеричной:  1a fe b9 cc e9 56 15 47 5b b6 1c 7
           5 30 ce 8d 89 73 b3 c3 5c c3 99
           Зашифрованный текст:  ¬ю№МйV╚G[¶Lu0OK‰siГ\Г™
```

Рис. 3.2: Вывод функции

2. Написал функцию дешифровки, которая определяет ключ, с помощью которого шифротекст может быть преобразован в некоторый фрагмент текста, представляющий собой один из возможных вариантов прочтения открытого текста (рис - @fig:003). А также представил результаты работы программы (рис - @fig:004).

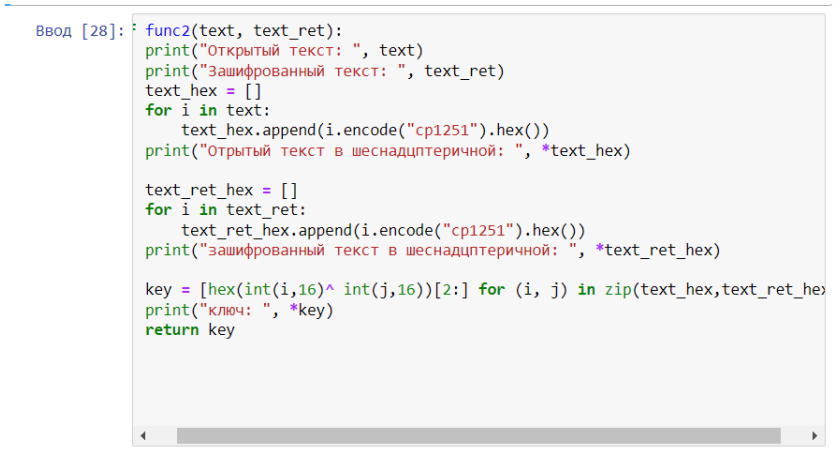

Рис. 3.3: Функция, дешифрующая данные

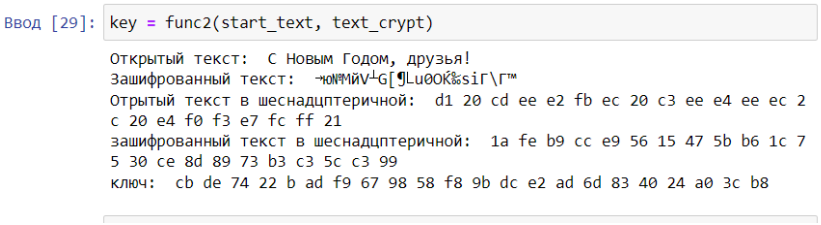

Рис. 3.4: Вывод функции

4 Выводы

Освоил на практике применение режима однократного гаммирования.

5 Ответы на контрольные вопросы

1. Одократное гаммирование - выполнение операции XOR между элементами гаммы и элементами подлежащего сокрытию текста. Если в методе шифрования используется однократная вероятностная гамма (однократное гаммирование) той же длины, что и подлежащий сокрытию текст, то текст нельзя раскрыть. Даже при раскрытии части последовательности гаммы нельзя получить информацию о всём скрываемом тексте.

2. Недостатки однократного гаммирования: Абсолютная стойкость шифра доказана только для случая, когда однократно используемый ключ, длиной, равной длине исходного сообщения, является фрагментом истинно случайной двоичной последовательности с равномерным законом распределения.

3. Преимущества однократного гаммирования: во-первых, такой способ симметричен, т.е. двойное прибавление одной и той же величины по модулю 2 восстанавливает исходное значение; во-вторых, шифрование и расшифрование может быть выполнено одной и той же программой. Наконец, Криптоалгоритм не даёт никакой информации об открытом тексте: при известном зашифрованном сообщении C все различные ключевые последовательности K возможны и равновероятны, а значит, возможны и любые сообщения P.

4. Длина открытого текста должна совпадать с длиной ключа, т.к. если ключ

короче текста, то операция XOR будет применена не ко всем элементам и конец сообщения будет не закодирован, а если ключ будет длиннее, то появится неоднозначность декодирования.

5. Операция XOR используется в режиме однократного гаммирования. Наложение гаммы по сути представляет собой выполнение побитовой операции сложения по модулю 2, т.е. мы должны сложить каждый элемент гаммы с соответствующим элементом ключа. Данная операция является симметричной, так как прибавление одной и той же величины по модулю 2 восстанавливает исходное значение.
6. Получение шифротекста по открытому тексту и ключу: $C_i = P_i \oplus K_i$

7. Получение ключа по окрытому тексту и шифротексту: $K_i = P_i \oplus C_i$

8. Необходимы и достаточные условия абсолютной стойкости шифра: полная случайность ключа; равенство длин ключа и открытого текста; однократное использование ключа.