Отчет по лабораторной работе №7

Элементы криптографии. Однократное гаммирование

Исаханян Эдуард Тигранович

2022 Sep 21th

Содержание

List of Tables

List of Figures

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.

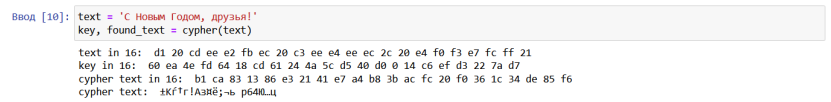
3 Выполнение лабораторной работы

Напишем функцию шифрования, которая определяет вид шифротекста при известном ключе и известном открытом тексте “С Новы Годом, друзья!”. (рис. 3.1)

```
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 3.1: Функция, шифрующая данные

А также посмотрим на работу данной функции. (рис. 3.2)



```
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:  ±КѓГг!АзиЁ;¬ь р64Ю…ц
```

Figure 3.2: Результат работы функции, шифрующей данные

Напишем функцию дешифровки, которая определяет ключ, с помощью которого шифротекст может быть преобразован в некоторый фрагмент текста, представляющий собой о дин из возможных вариантов прочтения открытого текста. (рис. 3.3)

```
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 3.3: Функция, дешифрующая данные

А также посмторим на результаты работы программы. (рис. 3.4)

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

open text:  С Новым Годом, друзья!
cypher text:  ±Кѓ†r!АэЯё;¬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 3.4: Результат работы функции, шифрующей данные

Сравнение ключей, полученных с помощью первой и второй функций. (рис. 3.5)

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

           key correct
```

Figure 3.5: Сравнение ключей

3.1 Контрольные вопросы

1. Поясните смысл однократного гаммирования.

   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. Перечислите недостатки однократного гаммирования.

   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. Перечислите преимущества однократного гаммирования.

   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. Почему длина открытого текста должна совпадать с длиной ключа?

   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. Какая операция используется в режиме однократного гаммирования, назовите её особенности?

   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. Как по открытому тексту и ключу получить шифротекст?

   Получение шифротекста по открытому тексту и ключу: $C_i = P_i \oplus K_i$

7. Как по открытому тексту и шифротексту получить ключ?

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

8. В чем заключаются необходимые и достаточные условия абсолютной стойкости шифра?

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

4 Выводы

Освоили на практике применение режима однократного гаммирования.

Список литературы

1. Методические материалы к лабораторной работе, представленные на сайте “ТУИС РУДН” https://esystem.rudn.ru/

::: {#refs} :::