Отчет по лабораторной работе №7

Элементы криптографии. Однократное гаммирование

Бармина Ольга Константиновна

2022 Sep 22th

Содержание

Список таблиц

Список иллюстраций

1. Цель работы

Целью данной работы является освоение на практике применение режима однократного гаммирования. [1]

2. Выполнение лабораторной работы

1. Загрузила библиотеки, необходимы для работы со строками и рандомными значениями.

```
import string
import random
```

Рис. 2.1.: рис 1. библиотеки

2. Написала функцию шифрования, которая определяет вид шифротекста при известном ключе и известном открытом тексте. Написала 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.

```
def f1(text):
    return ' '.join(hex(ord(i))[2:] for i in text)

def f2(size):
    return ' '.join(random.choice(string.ascii_letters+string.digits) for _ in range(size))

def f3(text, key):
    return ' '.join(chr(a^b) for a,b in zip(text, key))

def f4(text, enc):
    return ' '.join(chr(a^b) for a,b in zip(text, enc))
```

Рис. 2.2.: рис 2. функции

3. Задала сообщение, создала ключ и его шестнадцатеричное представление.

```
msg = "С Новым Годом, друзья!"
key = f2(len(msg))
hex_key = f1(key)
print(key)
print(hex_key)
```

```
j L O P b S q m u L X c c 1 h M E C W r M b
6a 20 4c 20 4f 20 50 20 62 20 53 20 71 20 6d 20 75 20 4c 20 58 20 63 20 63 20 31 20 68 20 4d 20 45 20 43 20 57 20 72 20 4d 20 6
2
```

Рис. 2.3.: рис 3. задание ключа

4. Зашифровала текст в шестнадцатеричном представлении.

```
enc = f3([ord(i) for i in msg], [ord(i) for i in key])
hex_enc = f1(enc)
hex_enc
```

```
'44b 20 0 20 451 20 41e 20 47d 20 46b 20 46c 20 0 20 471 20 41e 20 467 20 41e 20 44d 20 c 20 4d 20 414 20 435 20 463 20 47b 20
46c 20 417 20 1'
```

```
decr = f3([ord(i) for i in enc], [ord(i) for i in key])
print(decr)
```

```
С Новым Годом, друзья!
```

Рис. 2.4.: рис 4. зашифрованный текст

5. Расшифровала сообщение при помощи ключа.

```
key2 = f4([ord(i) for i in msg], [ord(i) for i in enc])
decr2 = f3([ord(i) for i in enc], [ord(i) for i in key2])
print(decr2)
```

```
С Новым Годом, друзья!
```

Рис. 2.5.: рис 5. расшифрованный текст

3. 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:

7. Получение ключа по окрытому тексту и шифротексту:

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

4. Выводы

В ходе работы мы освоили на практике применение режима однократного гаммирования.

5. Список литературы

1. Методические материалы курса