

Цель работы

Освоить на практике применение режима однократного гаммирования.

Выполнение лабораторной работы

Импортирую библиотеки:

```
[1] import numpy as np
import operator as op
import sys
```

Подаю на вход сообщение:

```
[2] s = "С Новым Годом, друзья!"
```

1. Определяю вид шифротекста при известном ключе и известном открытом тексте.

Функция получает на вход строку, переводит ее в шестнадцатеричную систему счисления. Затем в программе случайно генерируется ключ. При помощи ключа получаю зашифрованное сообщение в шестнадцатеричной системе счисления. Затем перевожу это сообщение в строковый вид.

```
[3] # функция шифрования
def encryption(text):
    print("Открытый текст: ", text)
    new_text = []
    for i in text:
        new_text.append(i.encode("cp1251").hex())
    print("\nОткрытый текст в 16-ой системе: ", new_text)
    # генерация ключа
    r = np.random.randint(0, 255, len(text))
    key = [hex(i)[2:] for i in r]
    new_key = []
    for i in key:
        new_key.append(i.encode("cp1251").hex().upper())
    print("\nКлюч в 16-ой системе: ", key)
    # получение зашифрованного сообщения
    xor_text = []
    for i in range(len(new_text)):
        xor_text.append("{:02x}".format(int(key[i], 16) ^ int(new_text[i], 16)))
    print("\nШифротекст в 16-ой системе: ", xor_text)
    # переводу зашифрованное сообщение в строку
    en_text = bytearray.fromhex("".join(xor_text)).decode("cp1251")
    print("\nШифротекст: ", en_text)
    return key, xor_text, en_text
```

Результат работы функции:

```
[4] k, t, et = encryption(s)

Открытый текст:  С Новым Годом, друзья!

Открытый текст в 16-ой системе:  ['d1', '20', 'cd', 'ee', 'e2', 'fb', 'ec', '20', 'c3', 'ee', 'e4', 'ee', 'ec', '2c', '20', 'e4', 'f0', 'f3', 'e7', 'fc', 'ff', '21']

Ключ в 16-ой системе:  ['91', 'ea', 'ba', 'ef', 'e', '43', '94', '46', '44', '6', '9a', '93', '34', '5b', '90', '34', '4d', '9c', '6d', 'b5', 'e5', 'bb']

Шифротекст в 16-ой системе:  ['40', 'ca', '77', '01', 'ec', 'b8', '78', '66', '87', 'e8', '7e', '7d', 'd8', '77', 'b0', 'd0', 'bd', '6f', '8a', '49', '1a', '9a']

Шифротекст:  @Kw@hexf1w~}W^PSoIbъ
```

2. Определяю ключ, с помощью которого шифротекст может быть преобразован в некоторый фрагмент текста, представляющий собой один из возможных вариантов прочтения открытого текста. Функция нахождения ключа получает на вход две строки: открытый текст и

зашифрованный. Затем она преобразует строки в шестнадцатеричный формат и выполняет операцию XOR для нахождения ключа.

```
[5] # функция дешифрования
def find_key(text, en_text):
    print("Открытый текст: ", text)
    print("\nШифротекст: ", en_text)

    new_text = []
    for i in text:
        new_text.append(i.encode("cp1251").hex())
    print("\nОткрытый текст в 16-ой системе: ", new_text)

    tmp_text = []
    for i in en_text:
        tmp_text.append(i.encode("cp1251").hex())
    print("\nШифротекст текст в 16-ой системе: ", tmp_text)

    xor_text = [hex(int(k,16)^int(t,16))[2:] for (k,t) in zip(new_text, tmp_text)]
    print("\nНайденный ключ в 16-ой системе: ", xor_text)
    return xor_text
```

Результат работы функции:

```
key = find_key(s, et)
Открытый текст: С Новым Годом, друзья!
Шифротекст: @Kы@мехfм-}Шы²Р5оbIЪъ

Открытый текст в 16-ой системе: ['d1', '20', 'cd', 'ee', 'e2', 'fb', 'ec', '20', 'c3', 'ee', 'e4', 'ee', 'ec', '2c', '20', 'e4', 'f0', 'f3', 'e7', 'fc', 'ff', '21']
Шифротекст текст в 16-ой системе: ['40', 'ca', '77', '01', 'ec', 'b8', '78', '66', '87', 'e8', '7e', '7d', 'd8', '77', 'b0', 'd0', 'bd', '6f', '8a', '49', '1a', '9a']
Найденный ключ в 16-ой системе: ['91', 'ea', 'ba', 'ef', 'e', '43', '94', '46', '44', '6', '9a', '93', '34', '5b', '90', '34', '4d', '9c', '6d', 'b5', 'e5', 'bb']
```

Проверка:

```
[7] if k == key:
    print("Ключ найден верно")
else:
    print("Ключ наадйен неверно")

Ключ найден верно
```

Ответы на контрольные вопросы

1. Поясните смысл однократного гаммирования.

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.

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.

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 С все различные ключевые последовательности К возможны и равновероятны, а значит, возможны и любые сообщения Р.

4. Почему длина открытого текста должна совпадать с длиной ключа?

Если ключ короче текста, то операция XOR будет применена не ко всем

элементам и конец сообщения будет не закодирован. Если ключ будет длиннее, то появится неоднозначность декодирования.

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

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?

В таком случае задача сводится к правилу:

$C_i = P_i \oplus K_i$, т.е. мы поэлементно получаем символы зашифрованного сообщения, применяя операцию исключающего или к соответствующим элементам ключа и открытого текста.

7. Как по открытому тексту и шифротексту получить ключ?

Подобная задача решается путем применения операции исключающего или к последовательностям символов зашифрованного и открытого сообщений:

$$K_i = P_i \oplus C_i.$$

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

Необходимые и достаточные условия абсолютной стойкости шифра:

- полная случайность ключа;
- равенство длин ключа и открытого текста;
- однократное использование ключа.

Вывод

В ходе данной лабораторной работы я освоила на практике применение режима однократного гаммирования.

Список литературы

- [Кулябов Д. С., Королькова А. В., Геворкян М. Н. Лабораторная работа №7](#)