Информационная безопасность. Лабораторная работа #7.

Элементы криптографии. Однократное гаммирование

Хохлачева Яна

Содержание

1 Цель работы

- Освоить на практике применение режима однократного гаммирования.

2 Программа шифровки и дешифровки

2.1 Функция шифрования

```
import numpy as np
import operator as op
import sys

sms = "С Новым Годом, друзья!"

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)

    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)
```

```
    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)

    en_text = bytearray.fromhex("".join(xor_text)).decode("cp1251")
    print("\nШифротекст: ", en_text)
    return key,xor_text, en_text

k, t, et = encryption(sms)
```

2.2 Результат выполнения функции шифрования

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: ['58', 'a5', '6f', '1e', 'e0', '2c', 'db', '44', '74', '6b', '88', 'e', '52', 'db', '86', 'b0', '85', '33', 'd1', '44', '7d', '9d']

Шифротекст в 16-ой системе: ['89', '85', 'a2', 'f0', '02', 'd7', '37', '64', 'b7', '85', '6c', 'e0', 'be', 'f7', 'a6', '54', '75', 'c0', '36', 'b8', '82', 'bc']

Шифротекст: ‰…ўpSTXЧ7d·…lasч¦TuA6ë,j

2.3 Функция декодирования

```
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

key = find_key(sms, et)
```

2.4 Результат функции декодирования

Открытый текст: С Новым Годом, друзья!

Шифротекст: ‰…ўpSTXЧ7d·…lasч¦TuA6ё,j

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 ['89', '85', 'a2', 'f0', '02',
 'd7', '37', '64', 'b7', '85', '6c', 'e0', 'be', 'f7', 'a6', '54',

'75', 'c0', '36', 'b8', '82', 'bc']

Найденный ключ в 16-ой системе: ['58', 'a5', '6f', '1e', 'e0', '2c',
'db', '44', '74', '6b', '88', 'e', '52', 'db', '86', 'b0', '85', '33',
'd1', '44', '7d', '9d']

2.5 Результат проверки

```
if k == key:
    print("Ключ найден")
else:
    print("Н")
```

Ключ найден

3 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.

- Такой способ симметричен, т.е. двойное прибавление одной и той же величины по модулю 2 восстанавливает исходное значение.
- шифрование и расшифрование может быть выполнено одной и той же программой.
- 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?

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?

$$C_i = P_i \oplus K_i$$

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.

4 Вывод

- Освоила на практике применение режима однократного гаммирования.