Лабораторная работа №8

Элементы криптографии. Шифрование (кодирование) различных исходных текстов одним ключом

Валиева Найля Разимовна

Содержание

Список таблиц

Список иллюстраций

1 Цель работы

Освоить на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом [1].

2 Задание

1. Написать программу, которая должна определять вид шифротекстов при известных открытых текстах и при известном ключе.
2. Также эта программа должна определить вид одного из текстов, зная вид другого открытого текста и зашифрованный вид обоих текстов (т.е. не нужно использовать ключ при дешифровке).

3 Выполнение лабораторной работы

1. Написала функцию шифрования, которая определяет вид шифротекста при известном ключе и известных открытых текстах “НаВашисходящийот1204” и “ВСеверныйфилиалБанка”. Ниже представлены функция, шифрующая данные (рис - @fig:001), а также работа данной функции (рис - @fig:002).

In [12]:
```
import numpy as np
```

In [17]:
```
def encryption(text1, text2):
    print("Open 1st text", text1)
    text_array1 = []
    for i in text1:
        text_array1.append(i.encode("cp1251").hex())
    print("\nOpen 1st text of 16th format", *text_array1)

    print("Open 2nd text", text2)
    text_array2 = []
    for i in text2:
        text_array2.append(i.encode("cp1251").hex())
    print("\nOpen 2nd text of 16th format", *text_array2)

    key_dec = np.random.randint(0, 255, len(text1))
    key_hex = [hex(i)[2:] for i in key_dec]
    print("\nKey of 16th format", *key_hex)

    crypt_text1 = []
    for i in range(len(text_array1)):
        crypt_text1.append("{:02x}".format(int(text_array1[i], 16) ^ int(key_hex[i], 16)))
    print("\nEncrypted 1st text of 16th format", *crypt_text1)

    crypt_text2 = []
    for i in range(len(text_array1)):
        crypt_text2.append("{:02x}".format(int(text_array2[i], 16) ^ int(key_hex[i], 16)))
    print("\nEncrypted 2nd text of 16th format", *crypt_text2)

    final_text1 = bytearray.fromhex("".join(crypt_text1)).decode("cp1251")
    print("\nEncrypted 1st text", final_text1)

    final_text2 = bytearray.fromhex("".join(crypt_text2)).decode("cp1251")
    print("\nEncrypted 2nd text", final_text2)

    return key_hex, final_text1, final_text2
```

Рис. 3.1: Функция, шифрующая данные

```
In [19]: p1 = "НаВашисходящийот1204"
         p2 = "ВСеверныйфилиалБанка"
         key, res1, res2 = encryption(p1, p2)

Open 1st text НаВашисходящийот1204

Open 1st text of 16th format cd e0 c2 e0 f8 e8 f1 f5 ee e4 ff f9 e8 e9 ee f2 31 32 30 34
Open 2nd text ВСеверныйфилиалБанка

Open 2nd text of 16th format c2 d1 e5 e2 e5 f0 ed fb e9 f4 e8 eb e8 e0 eb c1 e0 ed ea e0

Key of 16th format e8 e7 c3 e 7a b3 fa b4 d5 cd 97 c6 2c 22 fc 2c 61 6a 5 4d

Encrypted 1st text of 16th format 25 07 01 ee 82 5b 0b 41 3b 29 68 3f c4 cb 12 de 50 58 35 79

Encrypted 2nd text of 16th format 2a 36 26 ec 9f 43 17 4f 3c 39 7f 2d c4 c2 17 ed 81 87 ef ad

Encrypted 1st text %□□o,[□A;)h?ДЛ□ЮРХ5y

Encrypted 2nd text *6&мцС□О<9□-ДВ□нѓ‡п
```

Рис. 3.2: Результат работы функции, шифрующей данные

2. Написала функцию дешифровки, которая определяет вид одного из текстов, зная вид другого открытого текста и зашифрованный вид обоих текстов (т.е. не использует ключ). (рис - @fig:003). А также представила результаты работы программы (рис - @fig:004).

```
In [21]: def decryption(cr_text1, cr_text2, op_text1):
             print("Open text", op_text1)
             print("\nEncrypted 1st text", cr_text1)
             print("\nEncrypted 2nd text", cr_text2)

             cr_text_hex1 = []
             for i in cr_text1:
                 cr_text_hex1.append(i.encode("cp1251").hex())
             print("\nOpen text of 16th format", *cr_text_hex1)

             cr_text_hex2 = []
             for i in cr_text2:
                 cr_text_hex2.append(i.encode("cp1251").hex())
             print("\nOpen text of 16th format", *cr_text_hex2)


             op_text_hex1 = []
             for i in op_text1:
                 op_text_hex1.append(i.encode("cp1251").hex())
             print("\nEncrypted text of 16th format", *op_text_hex1)

             cr1_cr2 = []
             op_text_hex2 = []
             for i in range(len(op_text1)):
                 cr1_cr2.append("{:02x}".format(int(cr_text_hex1[i], 16) ^ int(cr_text_hex2[i],16)))
                 op_text_hex2.append("{:02x}".format(int(cr1_cr2[i], 16) ^ int(op_text_hex1[i],16)))

             print("\nOpen 2nd text with 16th format", *op_text_hex2)
             op_text2 = bytearray.fromhex("".join(op_text_hex2)).decode("cp1251")
             return op_text2
```

Рис. 3.3: Функция, дешифрующая данные

```
In [22]: text2 = decryption(res1, res2, p1)
         print("\nOpen 2nd test", text2)

         Open text НаВашисходящийот1204

         Encrypted 1st text %□□o,[□A;)h?ДЛ□ЮРХ5y

         Encrypted 2nd text *6&мµС□O<9□-ДВ□нЃ‡n

         Open text of 16th format 25 07 01 ee 82 5b 0b 41 3b 29 68 3f c4 cb 12 de 50 58 35 79

         Open text of 16th format 2a 36 26 ec 9f 43 17 4f 3c 39 7f 2d c4 c2 17 ed 81 87 ef ad

         Encrypted text of 16th format cd e0 c2 e0 f8 e8 f1 f5 ee e4 ff f9 e8 e9 ee f2 31 32 30 34

         Open 2nd text with 16th format c2 d1 e5 e2 e5 f0 ed fb e9 f4 e8 eb e8 e0 eb c1 e0 ed ea e0

         Open 2nd test ВСеверныйфилиалБанка

In [23]: text1 = decryption(res1, res2, p2)
         print("\nOpen 1st test", text1)

         Open text ВСеверныйфилиалБанка

         Encrypted 1st text %□□o,[□A;)h?ДЛ□ЮРХ5y

         Encrypted 2nd text *6&мµС□O<9□-ДВ□нЃ‡n

         Open text of 16th format 25 07 01 ee 82 5b 0b 41 3b 29 68 3f c4 cb 12 de 50 58 35 79

         Open text of 16th format 2a 36 26 ec 9f 43 17 4f 3c 39 7f 2d c4 c2 17 ed 81 87 ef ad

         Encrypted text of 16th format c2 d1 e5 e2 e5 f0 ed fb e9 f4 e8 eb e8 e0 eb c1 e0 ed ea e0

         Open 2nd text with 16th format cd e0 c2 e0 f8 e8 f1 f5 ee e4 ff f9 e8 e9 ee f2 31 32 30 34

         Open 1st test НаВашисходящийот1204
```

Рис. 3.4: Результат работы функции, дешифрующей данные

4 Выводы

Освоила на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.

5 Ответы на контрольные вопросы

1. Чтобы определить один из текстов, зная другой, необходимо вопсользоваться следующей формулой: $C_1 \oplus C_2 \oplus + P_1 = P_1 \oplus P_2 \oplus + P_1 = P_2$, где C_1 и C_2 - шифротексты. Т.е. ключ в данной формуле не используется.
2. При повторном использовании ключа при шифровании текста получим исходное сообщение.

3. Режим шифрования однократного гаммирования одним ключом двух открытых текстов реализуется по следующей формуле:

$$C_1 = P_1 \oplus + K$$

$$C_2 = P_2 \oplus + K,$$

где C_i - шифротексты, P_i - открытые тексты, K - единый ключ шифровки

4. Недостатки шифрования одним ключом двух открытых текстов:
   Во-первых, имея на руках одно из сообщений в открытом виде и оба шифротекста, злоумышленник способен расшифровать каждое сообщение, не зная ключа.
   Во-вторых, зная шаблон сообщений, злоумышленник получает возможность определить те символы сообщения P_2, которые находятся на позициях известного шаблона сообщения P_1.

5. Преимущества шифрования одним ключом двух открытых текстов:
   Такой подход помогает упростить процесс шифрования и дешифровки. Также, при отправке сообщений между 2-я компьютерами, удобнее пользоваться одним общим ключом для передаваемых данных

6 Список литературы

1. Кулябов Д. С., Королькова А. В., Геворкян М. Н. Информационная безопасность компьютерных сетей. Лабораторная работа № 8. Элементы криптографии. Шифрование (кодирование) различных исходных текстов одним ключом.