

Отчёт по лабораторной работе №8

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

Кондрашина Мария Сергеевна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы. Создание программы	6
3	Выводы	9
4	Список литературы	10

List of Figures

2.1	Перевод открытых текстов в шестнадцатеричное представление.	6
2.2	Создание ключа и шифрование текста	7
2.3	Зашифрованные тексты	7
2.4	Определение открытого текста по второму открытому тексту и зашифрованным текстам	8

List of Tables

1 Цель работы

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

2 Выполнение лабораторной работы.

Создание программы

Два текста кодируются одним ключом (однократное гаммирование).

Требуется не зная ключа и не стремясь его определить, прочитав оба текста. Необходимо разработать приложение, позволяющее шифровать и дешифровать тексты P1 и P2 в режиме однократного гаммирования. Приложение должно определить вид шифротекстов C1 и C2 обоих текстов P1 и P2 при известном ключе ; Необходимо определить и выразить аналитически способ, при котором злоумышленник может прочитать оба текста, не зная ключа и не стремясь его определить

1. Перевод открытых текстов в шестнадцатеричное представление. (fig. 2.1)

1. Два текста кодируются одним ключом (однократное гаммирование). Требуется не зная ключа и не стремясь его определить, прочитав оба текста. Необходимо разработать приложение, позволяющее шифровать и дешифровать тексты P1 и P2 в режиме однократного гаммирования. Приложение должно определить вид шифротекстов C1 и C2 обоих текстов P1 и P2 при известном ключе

```
Ввод [19]: t1 = 'НаВашисходящийот1204'
           t2 = 'ВСеверныйфилиалБанка'
           print("Открытые тексты: ", t1, ", ", t2)

Открытые тексты:  НаВашисходящийот1204 ,  ВСеверныйфилиалБанка

Ввод [20]: def to_16(strM):
           str_16 = [(i.encode('cp1251')).hex().upper() for i in strM]
           return str_16

Ввод [21]: t1_16 = to_16(t1)
           t2_16 = to_16(t2)
           print("Открытые тексты в шестнадцатеричном представлении: \n", t1_16, ", \n", t2_16)

Открытые тексты в шестнадцатеричном представлении:
['CD', 'E0', 'C2', 'E0', 'F8', 'E8', 'F1', 'F5', 'EE', 'E4', 'FF', 'F9', 'E8', 'E9', 'EE', 'F2', '31', '32', '30', '34'] ,
['C2', 'D1', 'E5', 'E2', 'E5', 'F0', 'ED', 'FB', 'E9', 'F4', 'E8', 'EB', 'E8', 'E0', 'EB', 'C1', 'E0', 'ED', 'EA', 'E0']
```

Figure 2.1: Перевод открытых текстов в шестнадцатеричное представление.

2. Создание ключа и шифрование текстов (шестнадцатеричное представление (fig. 2.2))

```
Ввод [22]: import random

key = [hex(random.randint(0,255))[2:].upper() for _ in range(len(t1))]
print("Ключ: ", key)

Ключ:  ['51', 'CF', '88', '42', 'E5', '4D', '11', '50', '39', 'D', 'F3', '10', 'F6', '40', 'D5', 'E1', '48', 'CB', '7E', '46']

Ввод [23]: def cipher_text(strM, keyM):
c_text = ['{:02x}'.format(int(i,16) ^ int(k, 16)).upper() for i,k in zip(strM, keyM)]
return c_text

Ввод [30]: t1_cipher_16 = cipher_text(t1_16, key)
t2_cipher_16 = cipher_text(t2_16, key)
print("Зашифрованные тексты в шестнадцатеричном представлении: \n", t1_cipher_16, ", \n", t2_cipher_16)

Зашифрованные тексты в шестнадцатеричном представлении:
['9C', '2F', '4A', 'A2', '1D', 'A5', 'E0', 'A5', 'D7', 'E9', '0C', 'E9', '1E', 'A9', '3B', '13', '79', 'F9', '4E', '72'],
['93', '1E', '6D', 'A0', '00', 'BD', 'FC', 'AB', 'D0', 'F9', '1B', 'FB', '1E', 'A0', '3E', '20', 'A8', '26', '94', 'A6']
```

Figure 2.2: Создание ключа и шифрование текста

3. Зашифрованные тексты. (fig. 2.3)

```
Ввод [31]: def cipher_text_al(strM):
al_text = [(bytes.fromhex(e1)).decode('cp1251') for e1 in strM]
return al_text

Ввод [33]: t1_cipher_al = cipher_text_al(t1_cipher_16)
t2_cipher_al = cipher_text_al(t2_cipher_16)
print("Зашифрованные тексты: \n", t1_cipher_al, ", \n", t2_cipher_al)

Зашифрованные тексты:
['ъ', '/', 'j', 'ÿ', '\x1d', 'Г', 'а', 'Г', 'Ч', 'й', '\x0c', 'й', '\x1e', 'ø', ';', '\x13', 'у', 'щ', 'N', 'r'],
['«', '\x1e', 'm', '\xa0', '\x00', 'S', 'ь', '«', 'P', 'щ', '\x1b', 'ы', '\x1e', '\xa0', '>', ' ', 'É', '&', '"', '|']
```

Figure 2.3: Зашифрованные тексты

4. Определение открытого текста по второму открытому тексту и зашифрованным текстам (без использования ключа)(fig. ??)

2. Необходимо определить и выразить аналитически способ, при котором злоумышленник может прочитать оба текста, не зная ключа и не стремясь его определить.

```
Ввод [38]: def find_t2 (t1_c16, t2_c16, t1):
            t2_f = ['{:02x}'.format(int(c1,16) ^ int(c2, 16) ^ int(p1, 16))].upper() for c1,c2,p1 in zip(t1_c16, t2_c16, t1)]
            return t2_f

Ввод [43]: t2_f_new = find_t2(t1_cipher_16, t2_cipher_16, t1_16)
            print("Поиск второго текста по известным шифрованным текстам и открытому первому тексту")
            print(t2_f_new)
            print(cipher_text_al(t2_f_new))

            print("\nПоиск первого текста по известным шифрованным текстам и открытому второму тексту")
            t1_f_new = find_t2(t1_cipher_16, t2_cipher_16, t2_16)
            print(t1_f_new)
            print(cipher_text_al(t1_f_new))

Поиск второго текста по известным шифрованным текстам и открытому первому тексту
['C2', 'D1', 'E5', 'E2', 'E5', 'F0', 'ED', 'FB', 'E9', 'F4', 'E8', 'EB', 'E8', 'E0', 'EB', 'C1', 'E0', 'ED', 'EA', 'E0']
['B', 'C', 'e', 'b', 'e', 'p', 'h', 'y', 'й', 'ф', 'и', 'л', 'и', 'a', 'л', 'Б', 'a', 'h', 'к', 'a']

Поиск первого текста по известным шифрованным текстам и открытому второму тексту
['CD', 'E0', 'C2', 'E0', 'F8', 'E8', 'F1', 'F5', 'EE', 'E4', 'FF', 'F9', 'E8', 'E9', 'EE', 'F2', '31', '32', '30', '34']
['H', 'a', 'B', 'a', 'ш', 'и', 'с', 'х', 'о', 'д', 'я', 'щ', 'и', 'й', 'о', 'т', '1', '2', '0', '4']
```

Figure 2.4: Определение открытого текста по второму открытому тексту и зашифрованным текстам

3 Выводы

- Выполнила лабораторную работу №8.
- Освоила на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом

4 Список литературы

1. Методические материалы курса. “Информационная безопасность компьютерных сетей” Кулябов Д. С., Королькова А. В., Геворкян М. Н.