# Отчёт по лабораторной работе №8

**дисциплина: Информационная безопасность**

Никитаева Александра Семеновна, НПИбд-02-18

# Содержание

# List of Tables

# List of Figures

# 1 Цель работы

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

# 2 Выполнение лабораторной работы

1. Объявила нужные библиотеки. Задала исходные данные – телеграммы Центра.

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

P1 = "GimmeGimmeGimmeA"
P2 = "ManAfterMidnight"
```

Также вывела длины телеграмм, чтобы показать, что они равны (рис. 2.1).

```
print("Длина первой телеграммы {} символов, а второй -- {}".format(len(P1
```

```
Длина первой телеграммы 16 символов, а второй -- 16
```

Figure 2.1: Вывод на экран длин телеграмм

2. Написала функцию, кодирующую исходные тексты в режиме однократного гаммирования.

```
def shifrovka(p1, p2):
    print("Исходная телеграмма 1: ", p1)
    print("Исходная телеграмма 2: ", p2)
```

```
    p1_16 = []
    p2_16 = []
    for i in p1:
        p1_16.append(i.encode("cp1251").hex())
    for i in p2:
        p2_16.append(i.encode("cp1251").hex())
    print("Телеграмма 1 в 16-ой форме: ", p1_16)
    print("Телеграмма 2 в 16-ой форме: ", p2_16)

    K = []
    for i in np.random.randint(0, 255, len(p1)):
        K.append(hex(i)[2:])
    print("Ключ длиной {} байт: {}".format(len(p1), K))

    c1_16 = []
    c2_16 = []
    for i in range(len(p1_16)):
        c1_16.append("{:02x}".format(int(K[i], 16) ^ int(p1_16[i], 16)))
    for i in range(len(p2_16)):
        c2_16.append("{:02x}".format(int(K[i], 16) ^ int(p2_16[i], 16)))
    print("Зашифрованная телеграмма 1 в 16-ой форме: ", c1_16)
    print("Зашифрованная телеграмма 2 в 16-ой форме: ", c2_16)

    c1 = bytearray.fromhex("".join(c1_16)).decode("cp1251")
    c2 = bytearray.fromhex("".join(c2_16)).decode("cp1251")
    print("Зашифрованная телеграмма 1: ", c1)
    print("Зашифрованная телеграмма 2: ", c2)
    return c1, c2
```

3. Закодировала исходные тексты (рис. 2.2).

```
C1, C2 = shifrovka(P1, P2)
```

```
Исходная телеграмма 1:  GimmeGimmeGimmeA
Исходная телеграмма 2:  ManAfterMidnight
Телеграмма 1 в 16-ой форме:  ['47', '69', '6d', '6d', '65', '47', '69', '6d', '6d', '65', '47', '69', '6d', '6d', '65', '41']
Телеграмма 2 в 16-ой форме:  ['4d', '61', '6e', '41', '66', '74', '65', '72', '4d', '69', '64', '6e', '69', '67', '68', '74']
Ключ длиной 16 байт: ['7e', '7', 'e1', '13', 'd', '68', '6e', '89', 'cc', '99', '7', '35', '78', 'e0', 'dc', '27']
Зашифрованная телеграмма 1 в 16-ой форме:  ['39', '6e', '8c', '7e', '68', '2f', '07', 'e4', 'a1', 'fc', '40', '5c', '15', '8d',
'b9', '66']
Зашифрованная телеграмма 2 в 16-ой форме:  ['33', '66', '8f', '52', '6b', '1c', '0b', 'fb', '81', 'f0', '63', '5b', '11', '87',
'b4', '53']
Зашифрованная телеграмма 1:  9nЋ~h/□дЎь@\□ЌNºf
Зашифрованная телеграмма 2:  3fЏRk□□ыґрс[□‡rS
```

Figure 2.2: Шифрование исходных телеграмм

4. Написала функцию, читающую (дешифрующую) оба текста, не зная ключа и не стремясь его определить, но получая на вход шаблон.

```
def rasshifrovka(c1, c2, p):
    print("Зашифрованная телеграмма 1: ", c1)
    print("Зашифрованная телеграмма 2: ", c2)

    c1_16 = []
    c2_16 = []
    for i in c1:
        c1_16.append(i.encode("cp1251").hex())
    for i in c2:
        c2_16.append(i.encode("cp1251").hex())
    print("Зашифрованная телеграмма 1 в 16-ой форме: ", c1_16)
    print("Зашифрованная телеграмма 2 в 16-ой форме: ", c2_16)

    print("Шаблон: ", p)

    p_16 = []
    for i in p:
        p_16.append(i.encode("cp1251").hex())
    print("Шаблон в 16-ой форме: ", p_16)
```

```
    tmp = []
    pp_16 = []
    for i in range(len(p)):
        tmp.append("{:02x}".format(int(c1_16[i], 16) ^ int(c2_16[i], 16)))
        pp_16.append("{:02x}".format(int(tmp[i], 16) ^ int(p_16[i], 16)))
    print("Вторая телеграмма в 16-ой форме: ", pp_16)

    pp = bytearray.fromhex("".join(pp_16)).decode("cp1251")
    print("Вторая телеграмма: ", pp)
    return pp
```

5. Прочитала оба текста, используя шаблон 1-ой телеграммы (рис. 2.3).

```
P2_dec = rasshifrovka(C1, C2, P1)
```

```
Зашифрованная телеграмма 1:  9nЋ~h/□дЎь@\□ЌNºf
Зашифрованная телеграмма 2:  3fЏRk□□ыѓрс[□‡rS
Зашифрованная телеграмма 1 в 16-ой форме:  ['39', '6e', '8c', '7e', '68', '2f', '07', 'e4', 'a1', 'fc', '40', '5c', '15', '8d',
'b9', '66']
Зашифрованная телеграмма 2 в 16-ой форме:  ['33', '66', '8f', '52', '6b', '1c', '0b', 'fb', '81', 'f0', '63', '5b', '11', '87',
'b4', '53']
Шаблон:  GimmeGimmeGimmeA
Шаблон в 16-ой форме:  ['47', '69', '6d', '6d', '65', '47', '69', '6d', '6d', '65', '47', '69', '6d', '6d', '65', '41']
Вторая телеграмма в 16-ой форме:  ['4d', '61', '6e', '41', '66', '74', '65', '72', '4d', '69', '64', '6e', '69', '67', '68', '7
4']
Вторая телеграмма:  ManAfterMidnight
```

Figure 2.3: Чтение телеграмм по 1-ой

6. Прочитала оба текста, используя шаблон 2-ой телеграммы (рис. 2.4).

```
P1_dec = rasshifrovka(C1, C2, P2)
```

```
Зашифрованная телеграмма 1:  9nЋ~h/□дЎь@\□ЌNºf
Зашифрованная телеграмма 2:  3fЏRk□□ыѓрс[□‡rS
Зашифрованная телеграмма 1 в 16-ой форме:  ['39', '6e', '8c', '7e', '68', '2f', '07', 'e4', 'a1', 'fc', '40', '5c', '15', '8d',
'b9', '66']
Зашифрованная телеграмма 2 в 16-ой форме:  ['33', '66', '8f', '52', '6b', '1c', '0b', 'fb', '81', 'f0', '63', '5b', '11', '87',
'b4', '53']
Шаблон:  ManAfterMidnight
Шаблон в 16-ой форме:  ['4d', '61', '6e', '41', '66', '74', '65', '72', '4d', '69', '64', '6e', '69', '67', '68', '74']
Вторая телеграмма в 16-ой форме:  ['47', '69', '6d', '6d', '65', '47', '69', '6d', '6d', '65', '47', '69', '6d', '6d', '65', '4
1']
Вторая телеграмма:  GimmeGimmeGimmeA
```

Figure 2.4: Чтение телеграмм по 2-ой

# 3 Выводы

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

# 4 Ответы на контрольные вопросы

1. Как, зная один из текстов (P1 или P2), определить другой, не зная при этом ключа?

По формуле (рис. **??**):
Определение другого текста

2. Что будет при повторном использовании ключа при шифровании текста?

Ничего не изменится – получится исходный текст.

3. Как реализуется режим шифрования однократного гаммирования одним ключом двух открытых текстов?

По формуле (рис. **??**):
Реализация однократного гаммирования

4. Перечислите недостатки шифрования одним ключом двух открытых текстов.

Злоумышленнику достаточно знать формат хотя бы одного текста и иметь на руках оба шифротекста, чтобы дешифровать послание.

5. Перечислите преимущества шифрования одним ключом двух открытых текстов.

Режим однократного гаммирования помогает упростить процессы шифровки и дешифровки.

# 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