

Отчет по лабораторной работе №7

Информационная безопасность

Байрамгельдыев Довлетмурат

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	10
	Список литературы	11

Список иллюстраций

4.1	Программа, 1	8
4.2	Программа, 2	9
4.3	Результат запуска программы	9

Список таблиц

1 Цель работы

- Освоить на практике применение режима однократного гаммирования.

2 Задание

- Написание программы
- Зашифровка текста по открытому тексту и известному ключу
- Определение ключа по открытому и зашифрованному тексту

3 Теоретическое введение

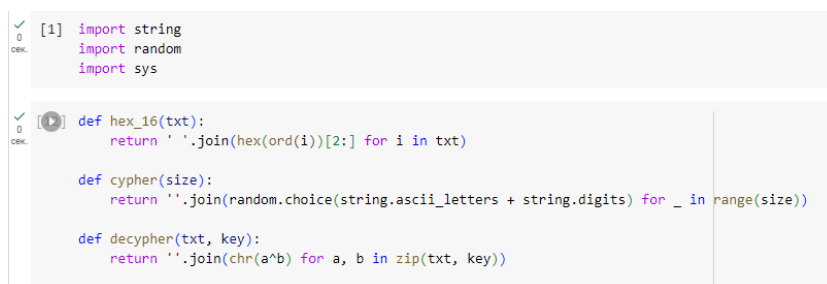
Гаммирование представляет собой наложение (снятие) на открытые (зашифрованные) данные последовательности элементов других данных, полученной с помощью некоторого криптографического алгоритма, для получения зашифрованных (открытых) данных. Иными словами, наложение гаммы — это сложение её элементов с элементами открытого (закрытого) текста по некоторому фиксированному модулю, значение которого представляет собой известную часть алгоритма шифрования.

Наложение гаммы по сути представляет собой выполнение операции сложения по модулю 2 (XOR) между элементами гаммы и элементами подлежащего сокрытию текста. Такой метод шифрования является симметричным, так как двойное прибавление одной и той же величины по модулю 2 восстанавливает исходное значение, а шифрование и расшифрование выполняется одной и той же программой.

Более подробно о см. в [lab-theory?].

4 Выполнение лабораторной работы

Для выполнения лабораторной работы я написал функции для перевода в 16-ричный вид, шифрования и дешифрования, а также импортировал нужные библиотеки (рис. 4.1).



```
[1] import string
import random
import sys

def hex_16(txt):
    return ''.join(hex(ord(i))[2:] for i in txt)

def cypher(size):
    return ''.join(random.choice(string.ascii_letters + string.digits) for _ in range(size))

def decypher(txt, key):
    return ''.join(chr(a^b) for a, b in zip(txt, key))
```

Рис. 4.1: Программа, 1

Далее я задал изначальное открытое сообщение, сгенерировал ключ, закодировал сообщение с помощью этого ключа, а также перевел все в 16-ричную систему. После этого раскодировал закодированное сообщение, чтобы проверить правильность работы программы, и определил используемый ключ по открытому сообщению и закодированному сообщению (рис. 4.2).


```

0 сек.
message = "С новым годом, друзья!"
key = cypher(len(message))
hex_key = hex_16(key)
print("Ключ: ", key)
print("Ключ в 16-ричном виде: ", hex_key)
encr = decypher([ord(i) for i in message], [ord(i) for i in key])
hex_encr = hex_16(encr)
print("Зашифр. сообщение:", hex_encr)
decr = decypher([ord(i) for i in encr], [ord(i) for i in key])
print("Расшифр. сообщение:", decr)

decr1=decypher ([ord(i) for i in encr], [ord(i) for i in key])
print("Ключ: ", key)
print("Вариант прочтения откр. текста: ", decr1)

```

Рис. 4.2: Программа, 2

Полученные сообщения и ключи вывел на экран (рис. 4.3). Сообщение было успешно закодировано и раскодировано, а найденный ключ совпадает с тем, который был сгенерирован для кодирования.

```

Ключ:  rqvTz9i50Kf7dcfEnCdiH
Ключ в 16-ричном виде:  72 71 76 54 7a 39 69 35 30 4b 66 37 64 63 66 45 6e 43 64 69 68 59
Зашифр. сообщение: 453 51 44b 46a 448 472 455 15 403 475 452 409 458 4f 46 471 42e 400 453 425 427 78
Расшифр. сообщение: С новым годом, друзья!
Ключ:  rqvTz9i50Kf7dcfEnCdiH
Вариант прочтения откр. текста:  С новым годом, друзья!

```

Рис. 4.3: Результат запуска программы

5 Выводы

В результате лабораторной работы я получил представление о базовых элементах криптографии и освоил на практике применение режима однократного гаммирования, написав программу, позволяющую зашифровывать и расшифровывать тексты и определять использованные для этого ключи.

Список литературы

1. Элементы криптографии. Однократное гаммирование [Электронный ресурс]. URL: [https://esystem.rudn.ru/pluginfile.php/2090284/mod_resource/content/2/007 lab_crypto-gamma.pdf](https://esystem.rudn.ru/pluginfile.php/2090284/mod_resource/content/2/007_lab_crypto-gamma.pdf).