

Отчёт по лабораторной работе № 8

Дисциплина: Основы информационной безопасности

Перелыгин Сергей Викторович

Содержание

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

Список иллюстраций

4.1	Приложение, реализующее режим однократного гаммирования для двух текстов одним ключом (1)	7
4.2	Приложение, реализующее режим однократного гаммирования для двух текстов одним ключом (2)	8
4.3	Вывод программы	8

1 Цель работы

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

2 Задание

- Сделать отчёт по лабораторной работе в формате Markdown.
- В качестве отчёта предоставить отчёты в 3 форматах: pdf, docx и md.

3 Теоретическое введение

Гаммирование - наложение (снятие) на открытые (зашифрованные) данные последовательности элементов других данных, полученной с помощью некоторого криптографического алгоритма, для получения зашифрованных (открытых) данных.

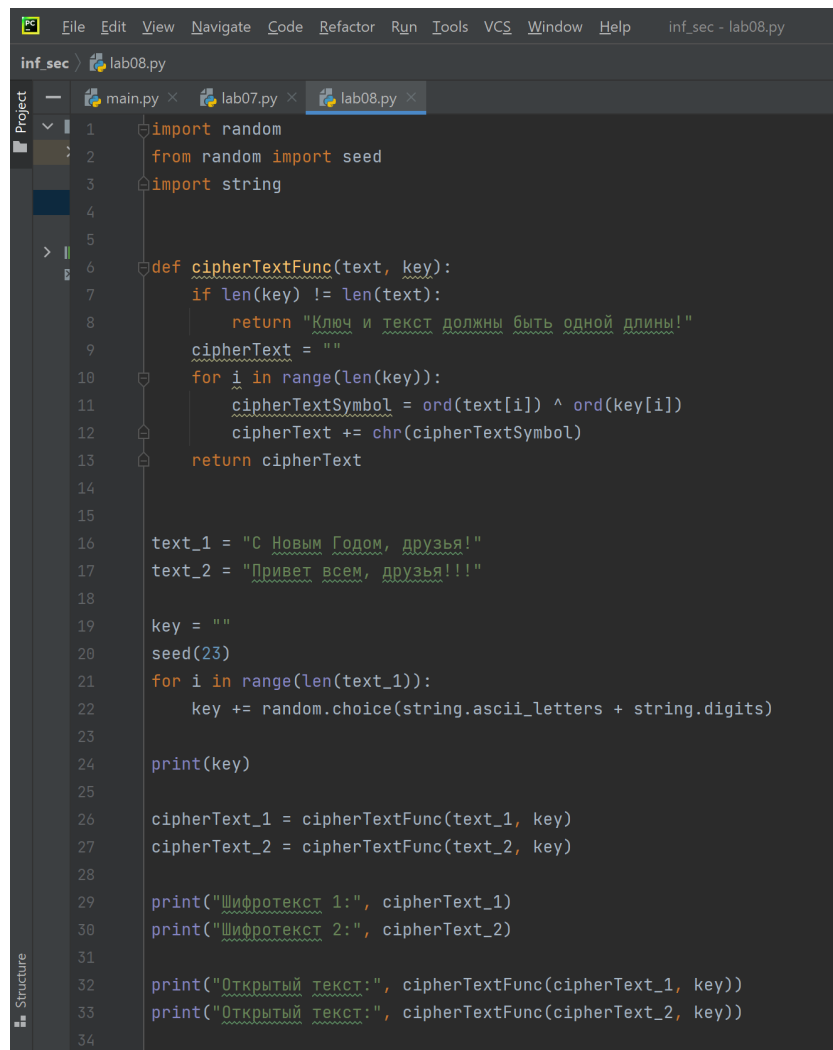
Основная формула, необходимая для реализации однократного гаммирования: $C_i = P_i \text{ XOR } K_i$, где C_i - i -й символ зашифрованного текста, P_i - i -й символ открытого текста, K_i - i -й символ ключа.

В данном случае для двух шифротекстов будет две формулы: $C1 = P1 \text{ xor } K$ и $C2 = P2 \text{ xor } K$, где индексы обозначают первый и второй шифротексты соответственно.

Если нам известны оба шифротекста и один открытый текст, то мы можем найти другой открытый текст, это следует из следующих формул: $C1 \text{ xor } C2 = P1 \text{ xor } K \text{ xor } P2 \text{ xor } K = P1 \text{ xor } P2$, $C1 \text{ xor } C2 \text{ xor } P1 = P1 \text{ xor } P2 \text{ xor } P1 = P2$.

4 Выполнение лабораторной работы

1. Код программы (рис. 4.1 и 4.2) и вывод (рис. 4.3).



```
inf_sec > lab08.py
1 import random
2 from random import seed
3 import string
4
5
6 def cipherTextFunc(text, key):
7     if len(key) != len(text):
8         return "Ключ и текст должны быть одной длины!"
9     cipherText = ""
10    for i in range(len(key)):
11        cipherTextSymbol = ord(text[i]) ^ ord(key[i])
12        cipherText += chr(cipherTextSymbol)
13    return cipherText
14
15
16 text_1 = "С Новым Годом, друзья!"
17 text_2 = "Привет всем, друзья!!!"
18
19 key = ""
20 seed(23)
21 for i in range(len(text_1)):
22     key += random.choice(string.ascii_letters + string.digits)
23
24 print(key)
25
26 cipherText_1 = cipherTextFunc(text_1, key)
27 cipherText_2 = cipherTextFunc(text_2, key)
28
29 print("Шифротекст 1:", cipherText_1)
30 print("Шифротекст 2:", cipherText_2)
31
32 print("Открытый текст:", cipherTextFunc(cipherText_1, key))
33 print("Открытый текст:", cipherTextFunc(cipherText_2, key))
34
```

Рис. 4.1: Приложение, реализующее режим однократного гаммирования для двух текстов одним ключом (1)

```

34
35 cipherText_XOR = cipherTextFunc(cipherText_1, cipherText_2)
36 print("Шифротекст 1 XOR шифротекст 2:", cipherText_XOR)
37
38 print("Открытый текст 1:", cipherTextFunc(cipherText_XOR, text_2))
39 print("Открытый текст 2:", cipherTextFunc(cipherText_XOR, text_1))
40
41 txt1 = text_1[15:21]
42 print("Часть первого открытого текста:", txt1)
43
44 cipherTxt2 = cipherTextFunc(cipherText_1[15:21], cipherText_2[15:21])
45 print("Часть второго открытого текста:", cipherTextFunc(cipherTxt2, txt1))
46
47

```

Рис. 4.2: Приложение, реализующее режим однократного гаммирования для двух текстов одним ключом (2)

```

Run: lab08
D:\pycharm\inf_sec\venv\Scripts\python.exe D:\pycharm\inf_sec\lab08.py
7X8s51fbLtByHwiUmrCaoN
Шифротекст 1: ЖхХэЇоньВуъѸчѸ[Iw36v3Po
Шифротекст 2: ШИссёёФейсёUhyщЖньоК@No
Открытый текст: С Новым Годом, друзья!
Открытый текст: Привет всем, друзья!!!
Шифротекст 1 XOR шифротекст 2: >с%00 MBRBMMсww0Xм30
Открытый текст 1: С Новым Годом, друзья!
Открытый текст 2: Привет всем, друзья!!!
Часть первого открытого текста: друзья
Часть второго открытого текста: узя!!

Process finished with exit code 0

```

Рис. 4.3: Вывод программы

2. Пояснения к программе:

- Lines 1-3: импорт необходимых библиотек
- Lines 6-13: функция, реализующая сложение по модулю два двух строк
- Lines 16-17: открытые/исходные тексты (одинаковой длины)
- Lines 19-24: создание ключа той же длины, что и открытые тексты
- Lines 26-30: получение шифротекстов с помощью функции, созданной ранее, при условии, что известны открытые тексты и ключ
 - Lines 32-33: получение открытых текстов с помощью функции, созданной ранее, при условии, что известны шифротексты и ключ

- Lines 35-36: сложение по модулю два двух шифротекстов с помощью функции, созданной ранее
- Lines 38-39: получение открытых текстов с помощью функции, созданной ранее, при условии, что известны оба шифротекста и один из открытых текстов
- Lines 41-42: получение части первого открытого текста (срез)
- Lines 44-45: получение части второго текста (на тех позициях, на которых расположены символы части первого открытого текста) с помощью функции, созданной ранее, при условии, что известны оба шифротекста и часть первого открытого текста

5 Выводы

Вывод: В ходе выполнения данной лабораторной работы я освоил на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.

6 Библиография

- Медведовский И.Д., Семьянов П.В., Платонов В.В. Атака через Internet. — НПО “Мир и семья-95”, 1997. — URL: <http://bugtraq.ru/library/books/attack1/index.html>
- Медведовский И.Д., Семьянов П.В., Леонов Д.Г. Атака на Internet. — Издательство ДМК, 1999. — URL: <http://bugtraq.ru/library/books/attack/index.html>
- Запечников С. В. и др. Информационная безопасность открытых систем. Том 1. — М.: Горячая линия -Телеком, 2006.
- Введение в информационную безопасность. Типы уязвимостей. (Д.Гамаюнов, МГУ)
- Практические аспекты сетевой безопасности. Вводная лекция. Сетевая безопасность. Стек протоколов TCP/IP. (Д. Гамаюнов, МГУ)
- Практические аспекты сетевой безопасности. Сетевая безопасность. Межсетевые экраны. (В. Иванов, МГУ)
- Практические аспекты сетевой безопасности. Сетевая безопасность. Системы обнаружения и фильтрации компьютерных атак (IDS/IPS). (Д. Гамаюнов, МГУ)
- Практические аспекты сетевой безопасности. Контроль нормального поведения приложений. Security Enhanced Linux (SELinux) (В. Сахаров, МГУ)