

Отчет по лабораторной работе №7

-

Овениязов Артур

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
4.1	Ответы на вопросы	9
5	Выводы	11
	Список литературы	12

Список иллюстраций

4.1	Код krypto.py	8
4.2	Результат выполнения krypto.py	8

Список таблиц

1 Цель работы

Освоить на практике применение режима однократного гаммирования.

2 Задание

Нужно подобрать ключ, чтобы получить сообщение «С Новым Годом, друзья!». Требуется разработать приложение, позволяющее шифровать и дешифровать данные в режиме однократного гаммирования. Приложение должно:

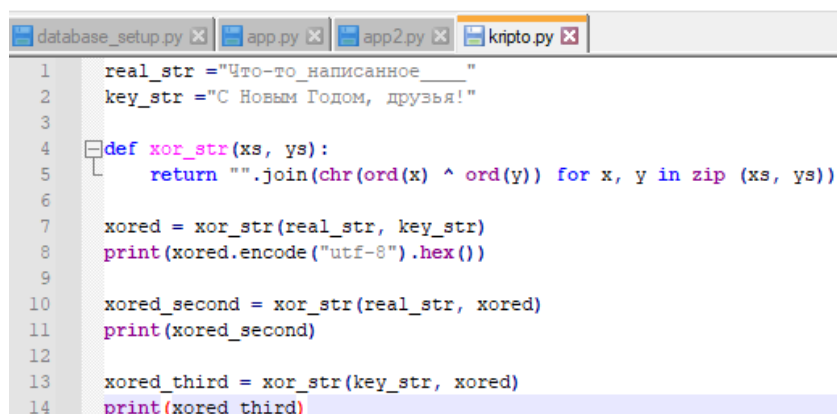
1. Определить вид шифротекста при известном ключе и известном открытом тексте.
2. Определить ключ, с помощью которого шифротекст может быть преобразован в некоторый фрагмент текста, представляющий собой один из возможных вариантов прочтения открытого текста.

3 Теоретическое введение

Гаммирование, или Шифр XOR, — метод симметричного шифрования, заключающийся в «наложении» последовательности, состоящей из случайных чисел, на открытый текст. Последовательность случайных чисел называется гамма-последовательностью и используется для зашифровывания и расшифровывания данных. [1]

4 Выполнение лабораторной работы

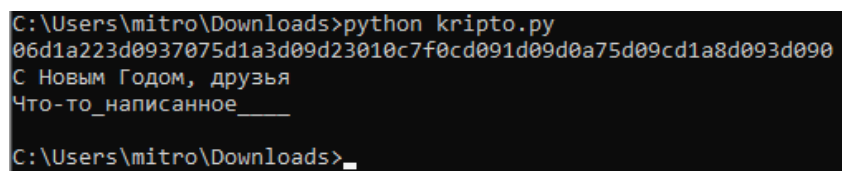
Написал следующий код в файле krypto.py.



```
1 real_str = "Что-то_написанное____"  
2 key_str = "С Новым Годом, друзья!"  
3  
4 def xor_str(xs, ys):  
5     return "".join(chr(ord(x) ^ ord(y)) for x, y in zip (xs, ys))  
6  
7 xored = xor_str(real_str, key_str)  
8 print(xored.encode("utf-8").hex())  
9  
10 xored_second = xor_str(real_str, xored)  
11 print(xored_second)  
12  
13 xored_third = xor_str(key_str, xored)  
14 print(xored_third)
```

Рис. 4.1: Код krypto.py

Полученный результат работы приложения: первая строка соответствует зашифрованной информации, вторая строка – расшифрованному тексту, а третья – ключу.



```
C:\Users\mitro\Downloads>python krypto.py  
06d1a223d0937075d1a3d09d23010c7f0cd091d09d0a75d09cd1a8d093d090  
С Новым Годом, друзья  
Что-то_написанное____  
C:\Users\mitro\Downloads>_
```

Рис. 4.2: Результат выполнения krypto.py

4.1. Ответы на вопросы

1. Поясните смысл однократного гаммирования. Принцип гаммирования представляет собой процедуру наложения, при помощи некой функции G , на входную информационную последовательность гаммы шифра, т.е. псевдослучайной последовательности.
2. Перечислите недостатки однократного гаммирования. Недостатки однократного гаммирования заключается в необходимости иметь огромные объемы данных, которые можно было бы использовать в качестве гаммы.
3. Перечислите преимущества однократного гаммирования. Преимущества однократного гаммирования в том, что не может сказать о дешифровке, верна она или нет из-за равных априорных вероятностей криптоаналитик. Информация о вскрытом участке гаммы не дает информации об остальных ее частях.
4. Почему длина открытого текста должна совпадать с длиной ключа? Так должно быть, потому что мы используем поэлементное перемножение, чтобы размерность шифртекста была равна размерности открытого текста и ключа. Также это ее необходимость заключается в том, чтобы шифрование и расшифрование выполнялось одной и той же программой.
5. Какая операция используется в режиме однократного гаммирования, назовите её особенности? В режиме однократного гаммирования используется операция сложения по модулю 2 (XOR). Двойное прибавление одной и той же величины по модулю 2 восстанавливает исходное значение.
6. Как по открытому тексту и ключу получить шифротекст? Задача нахождения шифротекста при известном ключе и открытом тексте состоит в применении следующего правила к каждому символу открытого текста: $C_i = P_i (+) K_i$.
7. Как по открытому тексту и шифротексту получить ключ? Обе части равенства сложим по модулю 2 с P_i . $C_i (+) P_i = P_i (+) K_i (+) P_i = K_i$, $K_i = C_i (+) P_i$.
8. В чем заключаются необходимые и достаточные условия абсолютной стой-

кости шифра? Необходимые и достаточные условия абсолютной стойкости шифра включают в себя полную случайность ключа, равенство длин ключа и открытого текста, однократное использование ключа.

5 Выводы

В ходе данной лабораторной работы я освоил применение режима однократного гаммирования на практике, разработал приложение, позволяющее шифровать и дешифровать данные в режиме однократного гаммирования.

Список литературы

1. Дискреционное разграничение доступа Linux [Электронный ресурс]. Сайт, 2021. URL: <http://debianinstall.ru/diskreسیونное-razgranichenie-dostupa-linux/>.