

Дисциплина «Информационная безопасность»

Направление - 02.03.03 Математическое обеспечение и администрирование информационных систем

Профиль - Разработка и администрирование информационных систем

Лабораторная работа №3_1

Одноразовый шифроблокнот (one-time pad cipher)

Шифр, который невозможно взломать получил название одноразовый шифроблокнот. Программа для работы с шифром Виженера, которую мы создали при выполнении лабораторной работы № 3, позволяет реализовать этот шифр без внесения в нее каких-либо изменений.

Одноразовый шифроблокнот — это шифр Виженера, который приобретает абсолютную криптографическую стойкость, если ключ удовлетворяет следующим критериям:

- длина ключа совпадает с длиной открытого сообщения;
- символы ключа выбираются абсолютно случайным образом;
- ключ используется всего один раз и больше не применяется ни к каким другим сообщениям.

Придерживаясь этих трех правил, можно сделать зашифрованное сообщение неуязвимым для любых видов криптоанализа. Такой ключ невозможно взломать, даже располагая неограниченными вычислительными ресурсами.

Свое название шифр получил благодаря тому, что обычно его ключи записывали в блокноте. После использования ключа верхний лист блокнота отрывали, чтобы перейти к следующему ключу. Как правило, в блокноте сразу записывали большое количество ключей, помечая ключи конкретными датами, а сам блокнот передавали из рук в руки. Например, если зашифрованное сообщение было получено 31 мая, то следовало пролистать блокнот и найти ключ, соответствующий этой дате.

Причина, по которой нельзя взломать какой-либо шифр, заключается в том, что обычно существует лишь один ключ, применение которого для дешифрования сообщения позволяет получить осмысленный текст.

Поскольку для получения шифр текста с равной вероятностью мог быть использован любой текст, сообщение, зашифрованное с помощью одноразового шифроблокнота, не поддается взлому.

В Python версии 3.6 и выше имеется модуль **secrets**, который в качестве источника случайных чисел использует операционную систему (чаще всего таким источником служат случайные события, например промежуток времени между последовательными нажатиями клавиш). Функция **secrets.randbelow()** возвращает истинно случайное число в диапазоне от 0 до значения, определяемого аргументом (не включая его самого).

```
>>> import secrets
>>> secrets.randbelow(10)
2
>>> secrets.randbelow (10)
0
>>> secrets. randbelow (10)
6
```

Можно также воспользоваться функцией `secrets.choice()`, которая возвращает случайно выбранный элемент из переданной ей строки или списка.

```
>>> import secrets
>>> secrets.choice('ABCDEFGHIJKLMNOPQRSTUVWXYZ')
'R'
>>> secrets.choice(['cat', 'dog', 'mouse'])
'dog'
```

Для создания истинно случайного одноразового шифроблокнота размером 55 символов можно использовать следующий код

```
>>> import secrets
>>> otp = ''
>>> for i in range (55):
...     otp += secrets.choice('ABCDEFGHIJKLMNOPQRSTUVWXYZ')
...
>>> otp
'MVOVAAAYDPELIRNRUZNNQHDNSOUWWNWPJUPIUAIMKFKNHQA
NIIYCHNDHC'
```

Одноразовый шифроблокнот — это методика, позволяющая сделать шифр Виженера неуязвимым для взлома. Для этого необходимо, длина ключа совпадала с размером сообщения, а сам ключ был и случайным и использовался строго один раз. Соблюдение всех трех условий полностью исключает возможность взлома сообщения, зашифрованного с помощью одноразового шифроблокнота. Однако применять такую методику для регулярного шифрования сообщений не очень удобно. Обычно одноразовые шифроблокноты со списком ключей передаются из рук в руки. При этом вы должны быть уверены в том, что список не по; чужие руки!

ЗАДАНИЯ РАБОТЫ

1. Создать проект в среде Visual Studio 2019 с использованием языка программирования Python.
2. Сформировать необходимое окружение языка Python из библиотек, необходимых для выполнения лабораторной работы.
3. Создать два файла-программы в языке python для шифровки и дешифровки с разными алфавитами, во втором модуле применить случайные числа.
4. Сформировать тексты программ, в соответствии с методическими указаниями, для шифровки и дешифровки.
5. Сформировать шифротексты.
6. Выполнить дешифровку шифротекстов.
7. Оформить отчет по работе с указанием описания алгоритма кодирования, описания программ шифровки и дешифровки, описание примеров и указания недостатков и достоинств алгоритма.