

# Презентация по лабораторной работе №7

Основы информационной безопасности

Назармамадов У.Д

16 сентября 2025

## Информация

### Докладчик

- Назармамадов Умед Джамshedович
- студент группы НКАбд-03-23
- Российский университет дружбы народов

## Цель

Освоить на практике применение режима однократного гаммирования

## 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

## Выполнение лабораторной работы

Требуется разработать программу, позволяющее шифровать и дешифровать данные в режиме однократного гаммирования. Начнем с создания функции для генерации случайного ключа (рис. [-@fig:001]). обновляю систему и ставлю Python

```
umedn@HUAWEI:~$ sudo apt update && sudo apt install -y python3 python3-pip
[sudo] password for umedn:
Hit:1 http://archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [1415 kB]
Get:5 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [175 kB]
Get:6 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1484 kB]
Get:7 http://archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [299 kB]
```

Требуется разработать программу

*Требуется разработать программу*

создаю рабочую папку для лабораторной.

```
umedn@HUAWEI:~$ mkdir -p ~/labs/lab07 && cd ~/labs/lab07
umedn@HUAWEI:~/labs/lab07$ |
```

создаю рабочую папку

*создаю рабочую папку*

Необходимо определить вид шифротекста при известном ключе и известном открытом тексте. Так как операция исключающего или отменяет сама себя, делаю одну функцию и для шифрования и для дешифрования текста. Нужно определить ключ, с помощью которого шифротекст может быть преобразован в некоторый фрагмент текста, представляющий собой один из возможных вариантов прочтения открытого текста. Для этого создаю функцию для нахождения возможных ключей для фрагмента текста. Проверка работы всех функций. Шифрование и дешифрование происходит верно, как и нахождение ключей, с помощью которых можно расшифровать верно только кусок текста.

Листинг программы:

```
umedn@HUAWEI: ~/labs/lab0 x + v
GNU nano 7.2 otp.py
import sys

def str_to_bytes(s):
    return s.encode('utf-8')

def xor_bytes(data, key):
    return bytes([b ^ k for b, k in zip(data, key)])

if __name__ == "__main__":
    mode = sys.argv[1]
    text = sys.argv[2]
    key = sys.argv[3]

    data = str_to_bytes(text)
    key_b = str_to_bytes(key)

    if len(data) != len(key_b):
        print("Ошибка: длина текста и ключа должна совпадать!")
        sys.exit(1)

    result = xor_bytes(data, key_b)

    if mode == "encrypt":
        print("Шифротекст (hex):", result.hex().upper())
    elif mode == "decrypt":
        print("Расшифровка:", result.decode('utf-8', errors="ignore"))

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line M-E Redo
```

Листинг программы

Листинг программы

## Выводы

В ходе выполнения данной лабораторной работы мной было освоено на практике применение режима однократного гаммирования.

- Менее оптимально представить в виде рисунка, графика, таблицы
- Текст используется, если все предыдущие способы отображения информации не подошли

...