# Презентация к лабораторной работе 8

Информационная безопасность компьютерных сетей

Еленга Невлора Люглеш.

27 октября 2023

# Докладчик

:::::::::::::: {.columns align=center} ::: {.column width="70%"}

- Еленга Невлора Люглеш
- Студент 4-го курса
- Группа НКНбд-01-20
- Российский университет дружбы народов
- 1032205073
- https://github.com/Newlora501

# Актуальность

Гаммирование – это наложение (снятие) на открытые (зашифрованные) данные криптографической гаммы, то есть последовательности элементов данных, вырабатываемых с помощью некоторого криптографического алгоритма, для получения зашифрованных (открытых) данных.

Шифрование в режиме однократного гаммирования – это один из методов симметричного шифрования, который используется для защиты информации от несанкционированного доступа.

Шифрование в режиме однократного гаммирования – это метод симметричного шифрования, в котором побитово складывается (по модулю 2) открытый текст с ключом-гаммой.

Ключ-гамма – это случайный битовый набор, который используется только для зашифрования одного сообщения и должен быть равен или более длинным, чем сам текст. Ключ-гамма порождается с помощью генератора случайных чисел и не должен быть известен злоумышленнику.

Как это работает?

Для шифрования в режиме однократного гаммирования мы используем операцию побитового XOR (исключающее ИЛИ), которая имеет следующую таблицу истинности:

| a | b | a XOR b |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Рис.1.1

# Цели

- Освоить на практике применение режима одно кратного гаммирования .

- Кодирования различных исходных текстов одним ключом.

- Описание программы

- Запуск программы

# Задача

Необходимо разработать приложение,позволяющее шифровать и дешифровать тексты P1 и P2 в режиме однократного гаммирования.Приложение должно определить вид шифротекстов C1 и C2 обоих текстов P1 и P2 при известном ключе;Необходимо определить и выразить аналитически способ,прикоторомзлоу мышленник может прочитать оба текста,не зная ключа и не стремясь его определить.

# Результаты

- Материалы и методы

*Python

*- Библотека Numpy

- Генерация ключа

Ввод [64]:
```
def gen_key(text):
    rn = np.random.randint(0,255, len(text))
    key = [hex(e)[2:] for e in rn]
    return key
```

Ввод [65]:
```
p1 = "Первый текст телеграммма"
p2 = "Второй текст телеграммма"
print(len(p1),len(p2))
```

```
24 24
```

Рис. 1.1.Генерация ключа

Ввод [66]:
```
def chifrovanie(p1,p2):
    print(f"P1: {p1}")
    print(f"P2: {p2}")

    hex_p1 = []
    hex_p2 = []

    for i in range(len(p1)):
        hex_p1.append(p1[i].encode("cp1251").hex())
        hex_p2.append(p2[i].encode("cp1251").hex())

    print("Hex P1:", hex_p1)
    print("Hex P2:", hex_p2)

    key = gen_key(p1)
    print("Hex key:", key)

    hex_c1 = []
    hex_c2 = []

    for i in range(len(hex_p1)):
        hex_c1.append("{:02x}".format(int(key[i], 16) ^ int(hex_p1[i], 16)))
        hex_c2.append("{:02x}".format(int(key[i], 16) ^ int(hex_p2[i], 16)))

    print("Hex C1: ", hex_c1)
    print("Hex C2: ", hex_c2)

    c1 = bytearray.fromhex("".join(hex_c1)).decode("cp1251")
    c2 = bytearray.fromhex("".join(hex_c2)).decode("cp1251")

    print(f"c1:,{c1}")
    print(f"c2:,{c2}")

    return key, c1, c2
```

Ввод [67]:
```
key, c1, c2 = chifrovanie(p1,p2)
```

```
P1: Первый текст телеграммма
P2: Второй текст телеграммма
Hex P1: ['cf', 'e5', 'f0', 'e2', 'fb', 'e9', '20', 'f2', 'e5', 'ea', 'f1', 'f
2', '20', 'f2', 'e5', 'eb', 'e5', 'e3', 'f0', 'e0', 'ec', 'ec', 'ec', 'e0']
Hex P2: ['c2', 'f2', 'ee', 'f0', 'ee', 'e9', '20', 'f2', 'e5', 'ea', 'f1', 'f
2', '20', 'f2', 'e5', 'eb', 'e5', 'e3', 'f0', 'e0', 'ec', 'ec', 'ec', 'e0']
Hex key: ['eb', '30', '8f', 'c0', '40', '4b', 'a6', 'a0', '40', '2a', '29', 'f
b', '99', '27', '62', '64', '75', '39', '60', '7a', '45', 'f9', '42', 'e0']
Hex C1:  ['24', 'd5', '7f', '22', 'bb', 'a2', '86', '52', 'a5', 'c0', 'd8', '0
9', 'b9', 'd5', '87', '8f', '90', 'da', '90', '9a', 'a9', '15', 'ae', '00']
Hex C2:  ['29', 'c2', '61', '30', 'ae', 'a2', '86', '52', 'a5', 'c0', 'd8', '0
9', 'b9', 'd5', '87', '8f', '90', 'da', '90', '9a', 'a9', '15', 'ae', '00']
c1:,$X "»ý†RГАШ №X‡ЏЉЂЉљ®⊡®
c2:,)Ва0®ý†RГАШ №X‡ЏЉЂЉљ®⊡®
```

Рис. 1.2.Шифрование Текста

- Использовали C1, C2 , P1 для получения  P2

[68]:
```
def dechifrovanie(c1, c2, p1):
    print(f"C1: {c1}")
    print(f"C2: {c2}")
    print(f"P1: {p1}")

    hex_c1 = []
    hex_c2 = []
    hex_p1 = []


    for i in range(len(p1)):
        hex_c1.append(c1[i].encode("cp1251").hex())
        hex_c2.append(c2[i].encode("cp1251").hex())
        hex_p1.append(p1[i].encode("cp1251").hex())

    print("Hex C1: ", hex_c1)
    print("Hex C2: ", hex_c2)
    print("Hex P1: ", hex_p1)

    hex_p2 = []

    for i in range(len(p1)):
       hex_p2.append("{:02x}".format(int(hex_c1[i], 16) ^ int(hex_c2[i], 16) ^ 

    print("Hex P2: ", hex_p2)
    p2 = bytearray.fromhex("".join(hex_p2)).decode("cp1251")

    print(f"P2: {p2}")

    return p1, p2
```

```
    print(f"P2: {p2}")

    return p1, p2
```

Ввод [69]:
```
p1_decfih, P2_dechif = dechifrovanie(c1, c2, p1)
```

```
C1: $X "»ўтRґАШ №X‡ЏЋЋљ©®
C2: )Ва0®ўтRґАШ №X‡ЏЋЋљ©®
P1: Первый текст телеграммма
Hex C1:  ['24', 'd5', '7f', '22', 'bb', 'a2', '86', '52', 'a5', 'c0', 'd8', '0
9', 'b9', 'd5', '87', '8f', '90', 'da', '90', '9a', 'a9', '15', 'ae', '00']
Hex C2:  ['29', 'c2', '61', '30', 'ae', 'a2', '86', '52', 'a5', 'c0', 'd8', '0
9', 'b9', 'd5', '87', '8f', '90', 'da', '90', '9a', 'a9', '15', 'ae', '00']
Hex P1:  ['cf', 'e5', 'f0', 'e2', 'fb', 'e9', '20', 'f2', 'e5', 'ea', 'f1', 'f
2', '20', 'f2', 'e5', 'eb', 'e5', 'e3', 'f0', 'e0', 'ec', 'ec', 'ec', 'e0']
Hex P2:  ['c2', 'f2', 'ee', 'f0', 'ee', 'e9', '20', 'f2', 'e5', 'ea', 'f1', 'f
2', '20', 'f2', 'e5', 'eb', 'e5', 'e3', 'f0', 'e0', 'ec', 'ec', 'ec', 'e0']
P2: Второй текст телеграммма
```

Ввод [ ]:

Рис.1.3.Получение  P2 через два шифротекста и P1

# Вывод

В ходе выполнения лабораторной работы мы освоили на практике применение режима однократного гаммирования например кодирования различных исходных текстов одним ключом.