# Презентация к лабораторной работе 3

Математические основы защиты информации
и информационной безопасности

Еленга Невлора Люглеш.

10 октября 2025

# Докладчик

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

- Еленга Невлора Люглеш
- Студент 4-го курса
- Группа НПИбд-01-25
- Российский университет дружбы народов
- 1032245779
- https://github.com/Newlora501

# Актуальность

Гаммирование – процедура наложения при помощи некоторой функции $F$ на исходный текст гаммы шифра, т.е. псевдослучайной последовательности (ПСП) с выходов генератора G. Псевдослучайная последовательность по своим статистическим свойствам неотличима от случайной последовательности, но является детерминированной, т.е. известен алгоритм ее формирования. Чаще

Обычно в качестве функции F берется операция поразрядного сложения по модулю два или по модулю $N$ ($N$ – число букв алфавита открытого текста).

Простейший генератор псевдослучайной последовательности представить рекуррентным соотношением:

$ Yi = a Yi-1+ b mod(m), i = 1, m,$

можно

где $y - 1$-й член последовательности псевдослучайных чисел, $a, Y, b$ - ключевые

параметры. Такая последовательность состоит из целых чисел от $0$ до т - $1$.

Если элементы $Vi$ и у совпадут, то совпадут и последующие участки: $1+11-/j+1,

Vi+2= 1j+2$. Таким образом, ПСП является периодической. Знание периода гаммы

Саущественно облегчает криптоанализ. Максимальная длина периода равна т. Для ее достижения необходимо удовлетворить следующим условиям:

1. Вит - взаимно простые числа;
2. а - 1 делится на любой простой делитель числа т;
3. а - 1 кратно 4, если т кратно 4.

# Цели

- изучить и реализовать шифрование гаммированием.

# Задача

Необходимо Реализовать алгоритм шифрования гаммированием конечной гаммой.

# Результаты

- Материалы и методы

*Python

*Библотека Numpy

- функция получения алфавита:

```
[34]: import numpy as np
```

```
[46]: def alphavit(choice):
          if choice == 'eng':
              return list(map(chr,range(ord('a'), ord('z')+1)))
          elif choice == 'rus':
              return list(map(chr,range(ord('а'), ord('я')+1)))
          else :
              print('Выбирайте eng или rus')
```

Рис. 1.1

- функция Шифрование

```
[90]: def encrypt_gamma(sms:str, gamma: str):
          alphav = alphavit('eng')
          if sms.lower() not in alphav :
              alphav = alphavit('rus')
          print(alphav)
          mes = len(alphav)
          def encrypt(letters: tuple):
              idx = (letters[0]+1)+(letters[1]+1)%mes
              if idx > mes:
                 idx = idx - mes
              return idx-1
          sms_clear = list(filter (lambda s : s.lower() in alphav, sms))
          gamma_clear = list(filter (lambda s : s.lower() in alphav, gamma))

          sms_ind = list(map(lambda s : alphav.index(s.lower()), sms_clear))
          gamma_ind = list(map(lambda s : alphav.index(s.lower()), gamma_clear))
          for i in range(len(sms_ind)-len(gamma_ind)):
              gamma_ind.append(gamma_ind[i])
          print(f'{sms.upper()} -> {sms_ind}\n{gamma.upper()} -> {gamma_ind}')
          encrypted_ind = list(map(lambda s : encrypt(s), zip(sms_ind, gamma_ind)))
          print(f'encrypted form : {encrypted_ind}\n')
          return ''.join(list(map(lambda s: alphav[s], encrypted_ind))).upper()
```

Рис. 1.2.

# Результаты

- функция для тестирования:

```
[92]: def encrypt_test(sms :str, gamma: str):
          print(f'Криптограмма: {encrypt_gamma(sms, gamma)}')
```

```
[94]: sms = 'ПРИКАЗ'
      gamma = 'гамма'
      encrypt_test(sms, gamma)
```

```
['а', 'б', 'в', 'г', 'д', 'е', 'ж', 'з', 'и', 'й', 'к', 'л', 'м', 'н', 'о', 'п', 'р', 'с', 'т', 'у', 'ф', 'х', 'ц', 'ч', 'ш', 'щ', 'ъ', 'ы', 'ь', 'э',
'ю', 'я']
ПРИКАЗ -> [15, 16, 8, 10, 0, 7]
ГАММА -> [3, 0, 12, 12, 0, 3]
encrypted form : [19, 17, 21, 23, 1, 11]

Криптограмма: УСХЧБЛ
```

```
[96]: sms = 'МЕНЯ ЗОВУТ ЛОРА'
      gamma = 'гамма'
      encrypt_test(sms, gamma)
```

```
['а', 'б', 'в', 'г', 'д', 'е', 'ж', 'з', 'и', 'й', 'к', 'л', 'м', 'н', 'о', 'п', 'р', 'с', 'т', 'у', 'ф', 'х', 'ц', 'ч', 'ш', 'щ', 'ъ', 'ы', 'ь', 'э',
'ю', 'я']
МЕНЯ ЗОВУТ ЛОРА -> [12, 5, 13, 31, 7, 14, 2, 19, 18, 11, 14, 16, 0]
ГАММА -> [3, 0, 12, 12, 0, 3, 0, 12, 12, 0, 3, 0, 12]
encrypted form : [16, 6, 26, 12, 8, 18, 3, 0, 31, 12, 18, 17, 13]

Криптограмма: РЖЪМИТГАЯМТСН
```

- Результаты:

Рис. 1.3

# Вывод

В ходе выполнения данной лабораторной работы изучили и реализовали шифрование гаммированием.