Отчёт по лабораторной работе 7

Хамбалеев Булат Галимович

11 декабря, 2021

Цель работы

Освоить на практике применение режима однократного гаммирования.

Задание

Лабораторная работа подразумевает использование языков программирования для создания программы для щифрования и дещифрования в режиме однократного гаммирования.

Выполнение лабораторной работы

Выполнение лабораторной работы

1. Импортируем библиотеки random,string. Зададим строковую переменную word.

```
[1]: import random
[2]: import string
[3]: word = ("С Новым Годом, друзья!")
```

Figure 1: рис.1. Импорт библиотек.

2. Зададим генератор ключа и напечатаем ключ по этой строке.

Ввод [4]:
```
def key_generator(size=6, chars = string.ascii_letters +string.digits):
    return ''.join(random.choice(chars) for _ in range(size))
def change(s):
    return ":".join("{:02x}".format(ord(c)) for c in s)
```

Ввод [5]:
```
key = key_generator(len(word))
```

Ввод [6]:
```
print(f'Ключ в строчном виде: {key}')
```

```
Ключ в строчном виде: QbWnd43zcct5V4FSa2qXlD
```

Figure 2: рис.2. Генератор ключа.

3. Зададим функции гаммирования, обнаружения ключа и дешифрования.

Ввод [13]:
```
def gammirovanie(word,key):
    word_ascii = [ord(i) for i in word]
    key_ascii = [ord(i) for i in key]
    enc_str = ''.join(chr(s ^ k) for s, k in zip(word_ascii, key_ascii))
    return enc_str
```

Ввод [15]:
```
def true_key_find(word,enc_str):
    sm_ascii = [ord(i) for i in word]
    enc_str_ascii = [ord(i) for i in enc_str]
    true_key = ''.join(chr(s ^ k) for s,k in zip(enc_str_ascii,sm_ascii))
    return true_key
```

Ввод [11]:
```
def unencrypt(enc_str, key):
    enc_str_ascii = [ord(i) for i in enc_str]
    key_ascii = [ord(i) for i in key]
    true_str = ''.join(chr(s ^ k) for s,k in zip(enc_str_ascii, key_ascii))
    return true_str
```

Figure 3: рис.3. Функции.

4. Получим закодированную строку и её шестнадцатеричный вид. Попробуем подобрать ключ и увидим, что он неверный. Используем настоящий ключ и декодированную строку.

```
Ввод [14]: enc_str=gammirovanie(word,key)

Ввод [16]: new_key=key_generator(len(enc_str))
           unencrypted_new_key = unencrypt(enc_str, new_key)
           true_key = true_key_find(word,enc_str)
           unencrypted_true_key = unencrypt(enc_str, true_key)

Ввод [17]: print(f'Закодированная строка: {enc_str}')

           Закодированная строка: ЧВъèiЩZЧйрЋЖfѧCццДУе

Ввод [20]: print(f'В шестнадцатеричной системе:{change(enc_str)}')

           В шестнадцатеричной системе:470:42:44a:450:456:47f:40f:5a:470:45d:440:40b:46a:18:66:467:421:471:446:414:423:65

Ввод [21]: print(f'Подобранный ключ: {new_key}')

           Подобранный ключ: glXQrc7xgVz8tQgBQBypuS

Ввод [22]: print(f'Расшифрованная ключом строка: {unencrypted_new_key}')

           Расшифрованная ключом строка: З.ВЁФМи"ЗЂкгOI№ХЧгпЋі6

Ввод [24]: print(f'Настоящий ключ: {true_key}')

           Настоящий ключ: QbWnd43zcct5V4FSa2qXlD

Ввод [25]: print(f'Декодированная строка: {unencrypted_true_key}')

           Декодированная строка: С Новым Годом, друзья!
```

Figure 4: рис.4. Отказ в доступе.

Спасибо за внимание