Лабораторная работа №7

Однократное гаммирование

Доборщук В.В., НФИбд-01-18

11 декабря 2021

Цель работы

Цель работы

Освоить на практике применение режима однократного гаммирования.

Выполнение лабораторной работы

Выполнение лабораторной работы

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 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

Реализация функционала

Создали дополнительную функцию для генерации случайного ключа:

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

Реализация функционала

```
def crypt_message(open_text, key):
    print(f"Open Text: {open_text}")
    hex_open_text = []
    for ch in open_text:
        hex_open_text.append(ch.encode("cp1251").hex())

    print("Hex Open Text: ", *hex_open_text)

    print("Key: ", *key)
    hex_crypted_text = []
    for i in range(len(hex_open_text)):
        hex_crypted_text.append("{:02x}".format(int(key[i], 16)^
```

Реализация функционала

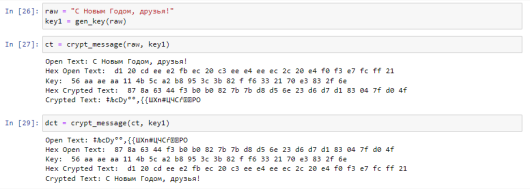
```
def find_key(open_text, crypted_text):
    print(f"Open Text: {open_text}\nCrypted Text: {crypted_text}
    hex_open_text = []
    for ch in open_text:
        hex_open_text.append(ch.encode("cp1251").hex())

    hex_crypted_text = []
    for ch in crypted_text:
        hex_crypted_text.append(ch.encode("cp1251").hex())

    print("Hex Open Text: ", *hex_open_text)
    print("Hex Crypted Text: ", *hex_crypted_text)
```

Проверка шифрования

С помощью реализованного функционала проверяем работоспособность нашего приложения:



```
In [26]: raw = "С Новым Годом, друзья!"
         key1 = gen_key(raw)

In [27]: ct = crypt_message(raw, key1)

         Open Text: С Новым Годом, друзья!
         Hex Open Text:  d1 20 cd ee e2 fb ec 20 c3 ee e4 ee ec 2c 20 e4 f0 f3 e7 fc ff 21
         Key:  56 aa ae aa 11 4b 5c a2 b8 95 3c 3b 82 f f6 33 21 70 e3 83 2f 6e
         Hex Crypted Text:  87 8a 63 44 f3 b0 b0 82 7b 7b d8 d5 6e 23 d6 d7 d1 83 04 7f d0 4f
         Crypted Text: ‡ЂcDу"",{{ШхпЩЧСґШРО

In [29]: dct = crypt_message(ct, key1)

         Open Text: ‡ЂcDу"",{{ШхпЩЧСґШРО
         Hex Open Text:  87 8a 63 44 f3 b0 b0 82 7b 7b d8 d5 6e 23 d6 d7 d1 83 04 7f d0 4f
         Key:  56 aa ae aa 11 4b 5c a2 b8 95 3c 3b 82 f f6 33 21 70 e3 83 2f 6e
         Hex Crypted Text:  d1 20 cd ee e2 fb ec 20 c3 ee e4 ee ec 2c 20 e4 f0 f3 e7 fc ff 21
         Crypted Text: С Новым Годом, друзья!
```

Рис. 1: Проверка шифрования

Все корректно отрабатывает с одним и тем же ключом.

Проверка ключа

В конце проверяем совпадают ли полученный ключ для идентичного и неидентичного текста и начальный ключ.

```
In [32]: key2 = find_key(raw, ct)

         Open Text: С Новым Годом, друзья!
         Crypted Text: ‼AcDy°°,{{ШХл#ЦЧСґШРО
         Hex Open Text:  d1 20 cd ee e2 fb ec 20 c3 ee e4 ee ec 2c 20 e4 f0 f3 e7 fc ff 21
         Hex Crypted Text:  87 8a 63 44 f3 b0 b0 82 7b 7b d8 d5 6e 23 d6 d7 d1 83 04 7f d0 4f
         Key  56 aa ae aa 11 4b 5c a2 b8 95 3c 3b 82 f f6 33 21 70 e3 83 2f 6e

In [34]: key1 == key2

Out[34]: True

In [35]: key3 = find_key("С Новым Годом, друзья?", ct)

         Open Text: С Новым Годом, друзья?
         Crypted Text: ‼AcDy°°,{{ШХл#ЦЧСґШРО
         Hex Open Text:  d1 20 cd ee e2 fb ec 20 c3 ee e4 ee ec 2c 20 e4 f0 f3 e7 fc ff 3f
         Hex Crypted Text:  87 8a 63 44 f3 b0 b0 82 7b 7b d8 d5 6e 23 d6 d7 d1 83 04 7f d0 4f
         Key  56 aa ae aa 11 4b 5c a2 b8 95 3c 3b 82 f f6 33 21 70 e3 83 2f 70

In [36]: key1 == key3

Out[36]: False
```

Рис. 2: Сравнение ключей

Получили, что для идентичного текста ключи совпадают, а для неидентичного, соответственно, не совпадают.

Заключение

Заключение

Мы изучили на практике применение режима однократного гаммирования.