

Отчёт по лабораторной работе №8

Дисциплина: Информационная безопасность

Андрианова Марина Георгиевна

Содержание

Цель работы	1
Выполнение лабораторной работы	1
Ответы на контрольные вопросы	3
Выводы	3

Цель работы

Освоить на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.

Выполнение лабораторной работы

Я выполняла лабораторную работу на языке программирования Python, используя функции, реализованные в лабораторной работе №7.

Используя функцию для генерации ключа, генерирую ключ, затем шифрую два разных текста одним и тем же ключом (рис. [-@fig:001]).

```
import random
import string

def generate_key_hex(text):
    key = ''
    for i in range(len(text)):
        key += random.choice(string.ascii_letters + string.digits) #генерация цифры для каждого символа в тексте
    return key

#для шифрования и дешифрования
def en_de_crypt(text, key):
    new_text = ''
    for i in range(len(text)): #проход по каждому символу в тексте
        new_text += chr(ord(text[i]) ^ ord(key[i % len(key)]))
    return new_text

t1 = 'С Новым Годом, друзья!'
key = generate_key_hex(t1)
en_t1 = en_de_crypt(t1, key)
de_t1 = en_de_crypt(en_t1, key)

t2 = "У Слона домов, оного!!"
en_t2 = en_de_crypt(t2, key)
de_t2 = en_de_crypt(en_t2, key)
```

Рисунок 1. Шифрование двух текстов

Расшифровываю оба текста сначала с помощью одного ключа, затем предполагаю, что мне неизвестен ключ, но известен один из текстов и уже расшифровываю второй, зная шифротексты и первый текст (рис. [-@fig:002]).

```
print('Открытый текст: ', t1, "\nКлюч: ", key, '\nШифротекст: ', en_t1, '\nИсходный текст: ', de_t1,)
print('Открытый текст: ', t2, "\nКлюч: ", key, '\nШифротекст: ', en_t2, '\nИсходный текст: ', de_t2,)

r = en_de_crypt(en_t2, en_t1) #C1^C2
print('Расшифровать второй текст, зная первый: ', en_de_crypt(t1, r))
print('Расшифровать первый текст, зная второй: ', en_de_crypt(t2, r))
```

Рисунок 2. Расшифровывание двух текстов

Запускаю написанный код (рис. [-@fig:003]).

```
Открытый текст:  С Новым Годом, друзья!
Ключ:  KN1lDDV6drheIGvD8sHvem
Шифротекст:  ЖпЫЦŨЦХЇŷьkhvkVŴŴaŴкЪL
Исходный текст:  С Новым Годом, друзья!
Открытый текст:  У Слона домов, огого!!
Ключ:  KN1lDDV6drheIGvD8sHvem
Шифротекст:  МпАЕŒŴАŒŒьећфкVŒŒзфшDL
Исходный текст:  У Слона домов, огого!!
Расшифровать второй текст, зная первый:  У Слона домов, огого!!
Расшифровать первый текст, зная второй:  С Новым Годом, друзья!
```

Рисунок 3. Результат работы программы

Листинг программы 1

```
import random
import string

def generate_key_hex(text):
    key = ''
    for i in range(len(text)):
        key += random.choice(string.ascii_letters + string.digits) #генерация
цифры для каждого символа в тексте
    return key

#для шифрования и дешифрования
def en_de_crypt(text, key):
    new_text = ''
    for i in range(len(text)): #проход по каждому символу в тексте
        new_text += chr(ord(text[i]) ^ ord(key[i % len(key)]))
    return new_text

t1 = 'С Новым Годом, друзья!'
key = generate_key_hex(t1)
en_t1 = en_de_crypt(t1, key)
de_t1 = en_de_crypt(en_t1, key)
```

```

t2 = "У Слона домов, оого!!"
en_t2 = en_de_crypt(t2, key)
de_t2 = en_de_crypt(en_t2, key)

print('Открытый текст: ', t1, "\nКлюч: ", key, '\nШифротекст: ', en_t1,
'\nИсходный текст: ', de_t1,)
print('Открытый текст: ', t2, "\nКлюч: ", key, '\nШифротекст: ', en_t2,
'\nИсходный текст: ', de_t2,)

r = en_de_crypt(en_t2, en_t1) #C1^C2
print('Расшифровать второй текст, зная первый: ', en_de_crypt(t1, r))
print('Расшифровать первый текст, зная второй: ', en_de_crypt(t2, r))

```

Ответы на контрольные вопросы

1. Как, зная один из текстов (P_1 или P_2), определить другой, не зная при этом ключа? - Для определения другого текста (P_2) можно просто взять зашифрованные тексты $C_1 \oplus C_2$, далее применить XOR к ним и к известному тексту: $C_1 \oplus C_2 \oplus P_1 = P_2$.
2. Что будет при повторном использовании ключа при шифровании текста? - При повторном использовании ключа мы получим дешифрованный текст.
3. Как реализуется режим шифрования однократного гаммирования одним ключом двух открытых текстов? - Режим шифрования однократного гаммирования одним ключом двух открытых текстов осуществляется путем XOR-ирования каждого бита первого текста с соответствующим битом ключа или второго текста.
4. Перечислите недостатки шифрования одним ключом двух открытых текстов - Недостатки шифрования одним ключом двух открытых текстов включают возможность раскрытия ключа или текстов при известном открытом тексте.
5. Перечислите преимущества шифрования одним ключом двух открытых текстов - Преимущества шифрования одним ключом двух открытых текстов включают использование одного ключа для зашифрования нескольких сообщений без необходимости создания нового ключа и выделения на него памяти.

Выводы

В ходе лабораторной работы я освоила на практике навыки применения режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.