Отчёт по лабораторной работе №8

Основы информационной безопасности

Паращенко А.Д.

14 октября 2024

Российский университет дружбы народов, Москва, Россия

Цель работы

Цель работы

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

Выполнение лабораторной работы

Функция для генерации ключа.

```python
def generate_key(text):
    key = ''
    for i in range(len(text)):
        key += random.choice(string.ascii_letters + string.digits)
    return key
```

Рис. 1: Генерация ключа

Функция для (де)шифрования.

```python
def crypt(text, key):
    new_text = ''
    for i in range(len(text)):
        new_text += chr(ord(text[i]) ^ ord(key[i % len(key)]))
    return new_text
```

Рис. 2: Шифрование

Код для вывода результатов и результаты.

```
t1 = 'С Новым Годом, друзья!'
key = generate_key(t1)
encrypt1 = crypt(t1, key)
decrypt1 = crypt(encrypt1, key)

t2 = 'Я люблю инфор, безопа!'
encrypt2 = crypt(t2, key)
decrypt2 = crypt(encrypt2, key)

print('Открытый текст:', t1, "\nКлюч:", key, "\nШифротекст:", encrypt1, "\nИсходный текст
print('\n')
print('Открытый текст:', t2, "\nКлюч:", key, "\nШифротекст:", encrypt2, "\nИсходный текст
print('\n')
```

Рис. 3: Подбор ключа

```
Открытый текст: С Новым Годом, друзья!
Ключ: fR6qkp7o1M1yBg3AKjjj9A
Шифротекст: чгЫяЊлЋОТѲSчѾK‼vЋЩЙЦѶ`
Исходный текст: С Новым Годом, друзья!


Открытый текст: Я люблю инфор, безопа!
Ключ: fR6qkp7o1M1yBg3AKjjj9A
Шифротекст: щгЙпњыѽОЉѰvчЋK‼ѰѾЙєsЉ`
Исходный текст: Я люблю инфор, безопа!
```

Код для расшифровки фразы с помощью второй фразы и результаты.

```
recrypt = crypt(encrypt2, encrypt1)

print('Расшифровка второго текста при помощи первого: ', crypt(t1, recrypt))
print('Расшифровка первого текста при помощи второго: ', crypt(t2, recrypt))
```

Рис. 5: Код расшифровки

```
Расшифровка второго текста при помощи первого:  Я люблю инфор, безопа!
Расшифровка первого текста при помощи второго:  С Новым Годом, друзья!
```

Рис. 6: Результаты расшифровки

Листинг программы.

```python
import random
import string

def generate_key(text):
    key = ''
    for i in range(len(text)):
        key += random.choice(string.ascii_letters + string.digits)
    return key

def crypt(text, key):
    new_text = ''
    for i in range(len(text)):
        new_text += chr(ord(text[i]) ^ ord(key[i % len(key)]))
    return new_text

t1 = 'С Новым Годом, друзья!'
key = generate_key(t1)
encrypt1 = crypt(t1, key)
decrypt1 = crypt(encrypt1, key)

t2 = 'Я люблю инфор, безопа!'
encrypt2 = crypt(t2, key)
decrypt2 = crypt(encrypt2, key)

print('Открытый текст:', t1, "\nКлюч:", key, "\nШифротекст:", encrypt1, "\nИсходный текст:", decrypt1)
print('\n')
print('Открытый текст:', t2, "\nКлюч:", key, "\nШифротекст:", encrypt2, "\nИсходный текст:", decrypt2)
print('\n')

recrypt = crypt(encrypt2, encrypt1)

print('Расшифровка второго текста при помощи первого: ', crypt(t1, recrypt))
print('Расшифровка первого текста при помощи второго: ', crypt(t2, recrypt))
```

Рис. 7: Листинг программы

Вывод

Вывод

В результате выполнения работы мы научились на практике применять режим однократного гаммирования на примере кодирования различных исходных текстов одним ключом.

Список литературы

Список литературы

1) https://esystem.rudn.ru/pluginfile.php/2357159/mod_resource/content/2/008-lab_crypto-key.pdf