Лабораторная работа №7

Элементы криптографии. Однократное гаммирование

Легиньких Г.А.

Российский университет дружбы народов, Москва, Россия

Информация

Докладчик

- Легиньких Галина Андреевна
- НФИбд-02-21
- Российский университет дружбы народов
- 1032216447@pfur.ru
- https://github.com/galeginkikh

Выполнение

Цель работы

Освоить на практике применение режима однократного гаммирования.

1. Написала функцию для генерации рандомного ключа(состоит из случайно последовательности символов.

```python
import random

def generate_key(word):
    key = ""
    for _ in range(len(word)):
        key += random.choice("0123456789abcdef") # Шестнадцатеричная систкма
    return key
```

Рис. 1: Ключ

2. Функция шифрования. В основе используется XOR.

```python
def encrypt(plaintext, key):
    chiphertext = ""
    for i in range(len(plaintext)):
        char = plaintext[i]
        key_char = key[i%len(key)]
        encrypted_char = chr(ord(char) ^ ord(key_char)) # XOR операция
        chiphertext += encrypted_char
    return chiphertext
```

Рис. 2: Шифрование

3. Аналогичный принцип для дешифрования.

```
def decript(chiphertext, key):
    decriptedtext = ""
    for i in range(len(chiphertext)):
        char = chiphertext[i]
        key_char = key[i%len(key)]
        decrypted_char = chr(ord(char) ^ ord(key_char)) # XOR операция
        decriptedtext += decrypted_char
    return decriptedtext
```

Рис. 3: Дешифрование

4. Функция нахождения ключей для фрагмента.

```
def find_possible_key(chiphertext, fragment):
    possible_keys = []
    for i in range(len(chiphertext) - len(fragment)+1):
        possible_key = " "
        for j in range(len(fragment)):
            char = chiphertext[i+j]
            fragment_char = fragment[j]
            key_char = chr(ord(char) ^ ord(fragment_char))
            possible_key += key_char
        possible_keys.append(possible_key)
    return possible_keys
```

Рис. 4: Фрагмент

5. Основной кусок кода, где задается строчка и вызов всех функций.

```
text = "С Новым Годом, друзья!"
key = generate_key(text)
encrypted_text = encrypt(text, key)
decrypted_text = decript(encrypted_text, key)
fragment = "С Новым"
possible_keys = find_possible_key(encrypted_text, fragment)
print("Ключ -", key)
print("Зашифрованный текст -", encrypted_text)
print("Дешифрованный текст -", decrypted_text)
print("Вызможные ключи -", possible_keys)
```

Рис. 5: Вызов функций

6. После запуска программы мы получим следующее.

Рис. 6: Вывод

Вывод

Вывод

Освоила на практике применение режима однократного гаммирования.