Лабораторная работа № 8

Элементы криптографии. Шифрование (кодирование) различных исходных текстов одним ключом

Юрченко Артём Алексеевич

Содержание

Список иллюстраций

1 Цель работы

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

2 Задание

Два текста кодируются одним ключом (однократное гаммирование). Требуется не зная ключа и не стремясь его определить, прочитать оба текста. Необходимо разработать приложение, позволяющее шифровать и дешифровать тексты P1 и P2 в режиме однократного гаммирования. Приложение должно определить вид шифротекстов C1 и C2 обоих текстов P1 и P2 при известном ключе ; Необходимо определить и выразить аналитически способ, при котором злоумышленник может прочитать оба текста, не зная ключа и не стремясь его определить.

3 Теоретическое введение

- Шифрование – это технология кодирования и раскодирования данных.Зашифрованные данные -это результат применения алгоритма для кодирования данных с целью сделать их недоступными для чтения. Данные могут быть раскодированы в исходную форму только путем применения специальный ключа. [1].

- Гаммирование — это наложение (или снятие при расшифровке сообщений) на открытое (или зашифрованное) сообщение так называемой криптографической гаммы. Криптографическая гамма — это последовательность элементов данных, которая вырабатывается с помощью определенного алгоритма. [2].

4 Выполнение лабораторной работы

1. Создаем функцию шифрования (4.1).

```
def encrypt(plaintext, key):
    encrypted_text = ""
    key_length = len(key)
    for i in range(len(plaintext)):
        char = plaintext[i]
        key_char = key[i % key_length]
        encrypted_text += chr((ord(char) + ord(key_char)) % 256)  # Применение шифра Цезаря
    return encrypted_text
```

Рис. 4.1: Функция шифрования

2. Введем данные из условия (4.2).

```
plain_text_1 = "навашисходящийот1204"
plain_text_2 = "ВСеверныйфилиалБанка"
key = "05 0C 17 7F 0E 4E 37 D2 94 10 09 2E 22 57 FF C8 0B B2 70 54"
```

Рис. 4.2: Данные из условия

3. Зашифруем текст с помощью ключа K (4.3).

```
cipher_text_1 = encrypt(plain_text_1, key)
cipher_text_2 = encrypt(plain_text_2, key)
print("Зашифрованный текст (1):", cipher_text_1)
print("Зашифрованный текст (2):", cipher_text_2)
```

Рис. 4.3: Шифрование текста

4. Создадим последовательность, с помощью которой будем расшифровывать текст. Передадим ее в функцию шифрования вместе с зашифрованным текстом (4.4).

```
decrypted_1 = decrypt(cipher_text_1, key)
decrypted_2 = decrypt(cipher_text_2, key)
print("Расшифрованный текст P1:", decrypted_1)
print("Расшифрованный текст P2:", decrypted_2)
```

Рис. 4.4: Расшифровка текста

5. Запустим программу и получим результат (4.5).

```
Зашифрованный текст (1): meR`▯Xr|^k▯ih~^vvRck
Зашифрованный текст (2): BVUbx`n▯Y{~[hu[Eu]mg
Расшифрованный текст P1: =020H8AE>4OI89>B1204
Расшифрованный текст P2: ▯!525@=K9D8;80;▯0=:0
```

Рис. 4.5: Результат

5 Выводы

В рамках данной лабораторной работы было освоено на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.

Список литературы

[1] https://www.kaspersky.ru/resource-center/definitions/encryption

[2] https://xakep.ru/2019/07/18/crypto-xor/