Лабораторная работа №8. лементы криптографии. Шифрование (кодирование) различных исходных текстов одним ключом

Выполнила: Лебедева Ольга Андреевна
Преподаватель Кулябов Дмитрий Сергеевич д.ф.-м.н., профессор кафедры прикладной информатики и кибербезопасности

2024

Российский университет дружбы народов, Москва, Россия

Цель работы

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

Теоретическое введение

Гаммирование представляет собой наложение (снятие) на открытые (зашифрованные) данные последовательности элементов других данных, полученной с помощью некоторого криптографического алгоритма, для получения зашифрованных (открытых) данных. Иными словами, наложение гаммы — это сложение её элементов с элементами открытого (закрытого) текста по некоторому фиксированному модулю, значение которого представляет собой известную часть алгоритма шифрования.

Наложение гаммы по сути представляет собой выполнение операции сложения по модулю 2 (XOR) (обозначаемая знаком □) между элементами гаммы и элементами подлежащего сокрытию текста[1].

Задание лабораторной работы

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.

Выполнение лабораторной работы

Код лабораторной работы:

```
import random

def generate_key(word):
    key = ""
    for _ in range(len(word)):
        key += random.choice("qwerty1234567890")
    return key
def en_de_crypt(text, key):
    next_text = ""
    for i in range(len(text)):
        next_text += chr(ord(text[i])^ord(key[i%len(key)]
    return next_text
```

Выполнение лабораторной работы

```
TEXT1 = 'Звёзды мерцали на небе.'
TEXT2 = 'Листья падали на землю'

key = generate_key(TEXT1)
en_TEXT1 = en_de_crypt(TEXT1, key)
de_TEXT1 = en_de_crypt(en_TEXT1, key)
en_TEXT2 = en_de_crypt(TEXT2, key)
de_TEXT2 = en_de_crypt(en_TEXT2, key)
```

Выполнение лабораторной работы

```
print("Key: ", key)
print("Текст1", "\n Зашифрованный текст: ", en_TEXT1, "\n
print("Текст2", "\n Зашифрованный текст: ", en_TEXT2, "\n
```

Выполнение лабораторной работы

В этом коде мы сначала написали функцию для генерации рандомного ключа. После добавили функцию шифрованния и дешифрования, основанную на алгоритме XOR. Задали два предлодения одинаковой длины. Далее, использовали один ключ для шифрования и дешифрования обих предложений. Вывели результаты в консоль.

Выполнение лабораторной работы

Результат работы кода: См. рис. 1

```
Key:  y3rqqrq8w8y2yyq9rq23r92
Текст1
 Зашифрованный текст:  ЁYцхйQЄтпЂтсQЄтQЏIyЌ
 Дешифрованный текст: Звёзды мерцали на небе.
Текст2
 Зашифрованный текст:  ѢЋггннQЇчЌщЉсҮьЉRцЇЏщ
 Дешифрованный текст: Листья падали на землю
```

Рис. 1: Результат работы кода

Аналитическое решение

1. Пусть тексты `P1` и `P2` шифруются с использованием одного ключа `K` при помощи операции XOR: `C1 = P1 ⊕ K C2 = P2 ⊕ K`
2. Если злоумышленник перехватил оба зашифрованных текста `C1` и `C2`, он может воспользоваться тем, что: `C1 ⊕ C2 = (P1 ⊕ K) ⊕ (P2 ⊕ K) = P1 ⊕ P2` Это выражение убирает влияние ключа `K` и возвращает результат `P1 ⊕ P2`, который представляет собой XOR между двумя открытыми текстами.

Аналитическое решение

3. Зная $P1 \oplus P2$, злоумышленник может воспользоваться информацией о возможных шаблонах в текстах, частотных характеристиках языка или общих конструкциях предложений. Например, если один из текстов (например, $P1$) известен или может быть угадан (например, если это стандартный заголовок или часто встречающаяся фраза), то можно вычислить другой текст $P2$: $P2 = (P1 \oplus P2) \oplus P1$
4. В результате, без необходимости восстанавливать ключ K, злоумышленник может восстановить оба текста, используя операцию XOR для двух зашифрованных текстов.

Аналитическое решение

Повторное использование одного ключа для шифрования нескольких сообщений делает систему уязвимой к атаке через анализ XOR зашифрованных текстов. Это является одной из причин, почему однократные блокноты (одноразовые гаммы) должны использоваться только один раз для каждого сообщения.

Заключение

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

Библиографическая справка

[1] Гаммирование: https://www.researchgate.net/profile/Dmitry-Kulyabov/publication/339290917_Informacionnaa_bezopasnost_komputernyh_setej_laboratorny
bezopasnost-komputernyh-setej-laboratornye-raboty.pdf