Отчёт по лабораторной работе №8 по предмету Информационная безопасность

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

Саттарова Вита Викторовна

Содержание

Список иллюстраций

Список таблиц

1 Цели и задачи работы

Цели:

- Изучить основы криптографии.
- Освоить на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.

Задачи:

- Выполнить все пункты, указанные в методических рекомендациях к лабораторной работе.
- Выполняя задания, познакомиться с основами криптографии.
- Выполняя задания, использовать режим однократного гаммирования для кодирования различных исходных текстов одним ключом.
- Написать отчёт, проанализировав результаты, полученные в ходе выполнения лабораторной работы.

2 Объект и предмет исследования

Объект исследования: основы шифрования.

Предмет исследования: однократное гаммирование.

3 Условные обозначения и термины

Условные обозначения

Отсутствуют

Термины

- шифрование
- однократное гаммирование
- шифротекст
- исходный текст

4 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.

Более подробно о работе см. в [1].

5 Теоретическое введение

5.1 Однократное гаммирование

Предложенная Г. С. Вернамом так называемая «схема однократного использования (гаммирования)» является простой, но надёжной схемой шифрования данных.

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.

В соответствии с теорией криптоанализа, если в методе шифрования используется однократная вероятностная гамма (однократное гаммирование) той же длины, что и подлежащий сокрытию текст, то текст нельзя раскрыть. Даже при раскрытии части последовательности гаммы нельзя получить информацию о всём скрываемом тексте.

Наложение гаммы по сути представляет собой выполнение операции сложения по модулю 2 (XOR) между элементами гаммы и элементами подлежащего сокрытию текста. Такой метод шифрования является симметричным, так как двойное прибавление одной и той же величины по модулю 2 восстанавливает исходное значение, а шифрование и расшифрование выполняется одной и той

же программой.

Если известны ключ и открытый текст, то задача нахождения шифротекста заключается в применении к каждому символу открытого текста и ключа операции XOR. Размерности открытого текста и ключа должны совпадать, и полученный шифротекст будет такой же длины.

Если известны шифротекст и открытый текст, то задача нахождения ключа решается также, а именно, обе части равенства необходимо применить посимвольно операцию XOR к шифротексту и открытому тексту.

Необходимые и достаточные условия абсолютной стойкости шифра:

- полная случайность ключа;
- равенство длин ключа и открытого текста;
- однократное использование ключа.

Шифротексты обеих телеграмм можно получить по формулам режима однократного гаммирования:

- C1 = P1 ⊠ K,
- C2 = P2 ⊠ K.

Предположим, что одна из телеграмм является шаблоном — т.е. имеет текст фиксированный формат, в который вписываются значения полей. Допустим, что злоумышленнику этот формат известен. Тогда он получает достаточно много пар C1 ⊠ C2 (известен вид обеих шифровок). Тогда зная P1 и учитывая, имеем:

- C1 ⊠ C2 ⊠ P1 = P1 ⊠ P2 ⊠ P1 = P2.

Таким образом, злоумышленник получает возможность определить те символы сообщения P2, которые находятся на позициях известного шаблона сообщения P1. В соответствии с логикой сообщения P2, злоумышленник имеет реальный шанс узнать ещё некоторое количество символов сообщения P2. Затем вновь используется с подстановкой вместо P1 полученных на предыдущем шаге новых символов

сообщения P2. И так далее. Действуя подобным образом, злоумышленник даже если не прочитает оба сообщения, то значительно уменьшит пространство их поиска.

Более подробно о работе см. в [1].

6 Техническое оснащение и выбранные методы проведения работы

Техническое оснащение

- Ноутбук
- Python 3
- Jupyter Notebook
- Интернет

Методы проведения работы

- Изучение методической информации
- Выполнение заданий в соответствии с указаниями
- Анализ результатов
- Обобщение проведённой деятельности

7 Выполнение лабораторной работы и полученные результаты

1. Скачала и ознакомилась с методическими указаниями к лабораторной работе (рис. 7.1).

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

8.1. Цель работы

Освоить на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом[1].

8.2. Указание к работе

Исходные данные.
Две телеграммы Центра:
P_1 = НаВашисходящийот1204
P_2 = ВСеверныйфилиалБанка
Ключ Центра длиной 20 байт:
K = 05 0C 17 7F 0E 4E 37 D2 94 10 09 2E 22 57 FF C8 OB B2 70 54
Режим шифрования однократного гаммирования одним ключом двух видов открытого текста реализуется в соответствии со схемой, приведённой на рис. 8.1.

Шифротексты обеих телеграмм можно получить по формулам режима однократного гаммирования:

$$C_1 = P_1 \oplus K, \tag{8.1}$$

Рис. 7.1: Лабораторная работа 8

2. Написала основные функции для приложения с однократным гаммированием (рис. 7.2).

Ввод [16]:
```
import random as rd
```

Ввод [17]:
```
# строка
message = "С Новым Годом, друзья!"
```

Ввод [18]:
```
# перевод в 16
def hex_string(mes):
    return " ".join([hex(ord(elem))[2:] for elem in mes])
```

Ввод [19]:
```
# ключ
def make_key(mes):
    base = "abcdefghijklmnopqrstuvwxyzабвгдеёжзийклмнопрстуфхцчшщъыьэюя0123456789"
    return "".join([rd.choice(base) for i in range(len(mes))])
```

Ввод [20]:
```
# кодирование
def coding(mes, key):
    return "".join(chr(ord(mes[i]) ^ ord(key[i])) for i in range(len(mes)))
```

Рис. 7.2: Функции

3. Создала два сообщения. Использовала созданные функции, создав ключ для кодирования сообщения, закодировав сообщения и получив шифротексты (рис. 7.3).

Ввод [6]:
```
P1 = "Странное и непонятное сообщение для работы"
P2 = "Просто сообщение для кодирования в работе2"
key = make_key(P1)
hex_key = hex_string(key)
C1 = coding(P1, key)
C2 = coding(P2, key)
```

Ввод [11]:
```
print("P1: ", P1)
print("P2: ", P2)
print("key: ", key)
print("hex_key: ", hex_key)
print("C1: ")
print(C1)
print("C2: ")
print(C2)
```

```
P1:  Странное и непонятное сообщение для работы
P2:  Просто сообщение для кодирования в работе2
key:  0уо8faыфх0эгюоуiчiгшъvвhе7яуедэеюиоес7ро4h
hex_key:  30 79 6f 38 66 430 44b 444 78 30 44d 433 44e 6f 443 69 447 69 72 448 44a 76 432 68 65 37 44
f 443 435 434 44d 435 44e 75 6f 65 441 37 70 6f 34 68
C1:
uqXJж□{è}Ыяv VsiђI□xEzюPE□ÏcёЎУ
C2:
Яйёщ Ф□ж□цЎ|z{ђ{ќѧйщ□ЖькйЎqq□    uz□чOXqIюЭЁZ
```

Рис. 7.3: Применение функций

4. Используя формулы из методических указаний, а также функцию, реали-

зующую однократное кодирование, показала, что для получения одного сообщения без знания ключа необходимо и достаточно знать шифротексты обоих сообщений, закодированных одним ключом, а также одно из исходных сообщений (рис. 7.4). Также показала другие возможности получения некоторой информации, например ключа, зная шифротексты и/или исходные тексты.

```
Ввод [12]: try1 = coding(C1, C2)
           try1

 Out[12]: '>\x02~q\x7f\x03OtO\x06Bt\x00\x02\x06\x08ƹv\x06qEK\x7f\n\x06qw\x07\r\x05\rƹД\tƹѠp\x01\x0f|wσy'

Ввод [13]: coding(try1, P1)

 Out[13]: 'Просто сообщение для кодирования в работе2'

Ввод [14]: coding(try1, P2)

 Out[14]: 'Странное и непонятное сообщение для работы'

Ввод [15]: try2 = coding(C1, P2)
           try2

 Out[15]: '\x0e{\x11I\x19rUaѦ6\\чютха(\x1ftй_ЖэbcFифибpZZ|ÈSб6\x7f\x13СБ'

Ввод [16]: try3 = coding(C1, P1)
           try3

 Out[16]: '0yo8faыфx0эгюoyiчirшъvвhe7яуедэеюиoec7po4h'

Ввод [17]: coding(C2, P2)

 Out[17]: '0yo8faыфx0эгюoyiчirшъvвhe7яуедэеюиoec7po4h'

Ввод [18]: coding(C2, P1)

 Out[18]: '\x0e{\x11I\x19rUaѦ6\\чютха(\x1ftй_ЖэbcFифибpZZ|ÈSб6\x7f\x13СБ'

Ввод [19]: coding(try2, C1)

 Out[19]: 'Просто сообщение для кодирования в работе2'

Ввод [20]: coding(try2, C2)

 Out[20]: 'Странное и непонятное сообщение для работы'

Ввод [21]: coding(try3, C1)

 Out[21]: 'Странное и непонятное сообщение для работы'
```

Рис. 7.4: Применение функций

8 Анализ результатов

Таким образом, были выполнены задания по изучению однократного гаммирования: были созданы и зашифрованы сообщения одним ключом, показано, как можно узнать исходное сообщение, зная оба шифротекста и одно другое исходное сообщение.

9 Заключение и выводы

Таким образом, в ходе выполнения лабораторной работы было сделано следующее:

- Изучены основы криптографии.
- Освоено на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.
- Написан отчёт к лабораторной работе.

10 Список литературы

[1]

1. Информационная безопасность [Электронный ресурс]. Российский университет дружбы народов, 2023. URL: https://esystem.rudn.ru/pluginfile.php/2090286/mod_resource/content/2/008-lab_crypto-key.pdf.