

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ**

Федеральное государственное автономное  
образовательное учреждение высшего образования  
«Национальный исследовательский университет ИТМО»

ФАКУЛЬТЕТ БЕЗОПАСНОСТИ ИНФОРМАЦИОННЫХ  
ТЕХНОЛОГИЙ

Основы стеганографии

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1**

«Основы текстовой стеганографии»

Выполнил:  
Студент группы N3351  
Фам Хю Хоанг



Проверил: ассистент ФБИТ,  
Университет ИТМО,  
Давыдов Вадим Валерьевич

Санкт-Петербург

2020

## **Цель работы:**

Целью данной лабораторной работы :

- Применение текстовых стеганографических методов для сокрытия
- Извлечение сообщения из стегоконтейнера
- Анализ исходного текста и стегоконтейнера.

## **Теоретическая часть:**

Стеганография -Способ передачи или хранения информации с учётом сохранения в тайне самого факта такой передачи (хранения). Этот термин ввёл в 1499 году аббат бенедиктинского монастыря Св. Мартина в Шпонгейме Иоганн Тритемий в своём трактате «Стеганография», зашифрованном под магическую книгу. В данной лабораторной работе были рассмотрены следующие 3 алгоритма:

1. Метод замены символов (метод знаков одинакового начертания);
2. Метод с использованием пробелов (метод хвостовых пробелов);
3. Метод с добавлением служебных символов.

Метод знаков одинакового начертания предполагает подмену (бит 1) или отказ от такой подмены (бит 0) русского символа латинским того же начертания. Например, латинская “с” на кириллическую “с”. В моем случае была реализована замена букв с и р .

Метод хвостовых пробелов предполагает дописывание в конце каждой строки файла-контейнера одного пробела, в случае кодирования единичного бита стеганосообщения. Если нужно закодировать нулевой бит, пробел в конце строки не дописывается.

Метод с добавлением служебных символов предполагает добавление служебных символов или подмену одних служебных символов на другие. Кодирование осуществляется таким же образом, как описано выше.

Для того, чтобы зашифровать исходное сообщение, оно преобразуется в двоичный код. Далее двоичная последовательность записывается одним из указанных выше методом. Извлечение информации происходит следующим образом: исходя из выбранного метода текстовый контейнер считывается и записывает полученную последовательность битов. Далее двоичный код преобразуется в зашифрованное сообщение.

## **Практическая часть:**

Исходным контейнером в моей работе является рассказ А. П. Чехова «Агафья» состоящий из 19271 символа в кодировке UTF-8.

Все алгоритмы были написаны на языке Python. Для каждого метода был написан отдельный файл.

Далее следуют три файла methog1.py, methog2.py и methog3.py, отвечающие за первый, второй и третий методы соответственно. Каждый из скриптов принимает ключи и

аргументы. При передаче “1” скрипт будет работать в режиме шифрования. В режиме шифрования нужно один файл становится контейнером(original\_document.txt), а во второй происходит запись(output.txt). Ключ “2” нужен для режима дешифрования.

### Метод замены символов (метод знаков одинакового начертания)

В файле methog1.py происходит посимвольное считывание исходного контейнера и посимвольная запись в файл. Шифрование и извлечение информации происходит описанным выше способом. В моем случае была реализована замена букв с и р .

```
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> py .\methog1.py
1.steganography
2.find secret text
1
please enter secret text
красиво
please open file output.txt!
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> |
```

### File output.txt

Два·капитана¶

ПИСЬМО·ЗА·ГОЛУБЫМ·РАКОМ¶

¶

Помню·просторный·грязный·двор·и·низкие·домики,·обнесённые·забором.·Двор·стоял·у·самой·реки,·и·по·вёснам,·когда·спадала·полая·вода,·он·был·усеян·щепой·и·ракушками,·а·иногда·и·другими,·куда·более·интересными·вещами.·Так,·однажды·мы·нашли·туго·набитую·письмами·сумку,·а·потом·вода·принесла·и·осторожно·положила·на·берег·и·самого·почтальона.·Он·лежал·на·спине,·закинув·руки,·как·будто·заслонясь·от·солнца,·ещё·совсем·молодой,·белокурый,·в·форменной·тужурке·с·блестящими·пуговицами:·должно·быть,·отправляясь·в·свой·последний·рейс,·почтальон·начистил·их·мелом.¶

¶

Сумку·отобрал·городовой,·а·письма,·так·как·они·размокли·и·уже·никуда·не·годились,·взяла·себе·тётя·Даша.·Но·они·не·совсем·размокли:·сумка·была·новая,·кожаная·и·плотно·запиралась.·Каждый·вечер·тётя·Даша·читала·вслух·по·одному·письму,·иногда·только·мне,·а·иногда·всему·двору.·Это·было·так·интересно,·что·даже·старухи,·ходившие·к·Сковородникову·играть·в·«козла»,·бросали·карты·и·присоединялись·к·нам.·Одно·из·этих·писем·тётя·Даша·читала·чаще·других—так·часто,·что·в·конце·концов·я·выучил·его·наизусть.·С·тех·пор·прошло·много·лет,·но·я·ещё·помню·его·от·первого·до·последнего·слова.¶

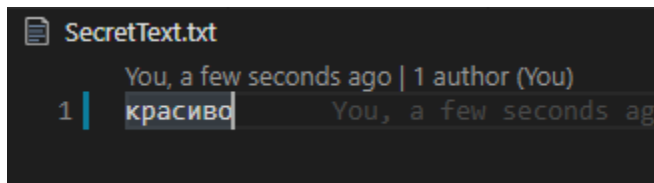
¶

«Глубокоуважаемая·Мария·Васильевна!¶

¶

## File Secret.txt

```
please open file output.txt.  
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> py .\methog1.py  
1.steganography  
2.find secret text  
2  
Please open file SecretText.txt  
  
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> |
```



SecretText.txt  
You, a few seconds ago | 1 author (You)  
1 | красиво You, a few seconds ago

## Метод с использованием пробелов

В файле methog2.py происходит строковое считывание исходного файла и последующая запись с добавлением или добавлением "\t" в конце строки.

```
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> py .\methog3.py  
1.steganography  
2.find secret text  
1  
please enter secret text  
ИТМО  
please open file output.txt!  
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> |
```

## File output.txt

Два капитана → ¶

ПИСЬМО ЗА ГОЛУБЫМ РАКОМ → ¶

→ ¶

Помню просторный грязный двор и низкие домики, обнесённые забором. Двор стоял у самой реки, и по веснам, когда спадала полая вода, он был усеян щепой и ракушками, а иногда и другими, куда более интересными вещами. Так, однажды мы нашли туго набитую письмами сумку, а потом вода принесла и осторожно положила на берег и самого почтальона. Он лежал на спине, закинув руки, как будто заслонясь от солнца, ещё совсем молодой, белокурый, в форменной тужурке с блестящими пуговицами: должно быть, отправляясь в свой последний рейс, почтальон на чистил их мелом. → → ¶

→ → ¶

Сумку отобрал городской, а письма, так как они размокли и уже никуда не годились, взяла себе тётя Даша. Но они не совсем размокли: сумка была новая, кожаная и плотно заперлась. Каждый вечер тётя Даша читала вслух по одному письму, иногда только мне, а иногда всему двору. Это было так интересно, что даже старухи, ходившие к Сковородникову играть в «козла», бросали карты и присоединялись к нам. Одно из этих писем тётя Даша читала чаще других — так часто, что в конце концов я выучил его наизусть. С тех пор прошло много лет, но я ещё помню его от первого до последнего слова. → → ¶

→ → ¶

«Глубокоуважаемая Мария Васильевна! → ¶

→ ¶

```
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> py .\methog2.py
1.steganography
2.find secret text
2
decode:
Please open file SecretText.txt

PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> |
```

## File SecretText.txt



## Метод с добавлением служебных символов

В файле methog3.py также реализовано посимвольное чтение и последующее посимвольная запись с добавлением особенных символов

```
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> py .\methog2.py
1.steganography
2.find secret text
1
please enter secret text
Санкт
please open file output.txt!
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> █
```

### File output.txt

Два°капитана¶

ПИСЬМО.°ЗА°ГОЛУБЫМ°РАКОМ¶

¶

Помню°просторный·грязный°двор·и°низкие°домики,°обнесённые°забором.°Двор°стоял°у°само  
й°реки,°и°по°вёснам,°когда°спала°полая°вода,°он°был°усеян°щепой°и°ракушками,°а°иногда°и°др  
угими,°куда°более°интересными°вещами.°Так,° ° Два°капитана¶

ПИСЬМО.°ЗА°ГОЛУБЫМ°РАКОМ¶

¶

Помню°просторный·грязный°двор·и°низкие°домики,°обнесённые°забором.·Двор·  
стоял°у°самой°реки,°и°по·вёснам,°когда·спала°полая·вода,°он°был·  
усеян°щепой°и°ракушками,°а°иногда°и·другими,°куда°более·интересными·вещами.·  
Так,°однажды°мы°нашли°туго°набитую°письмами·сумку,°а·потом°вода·принесла°и·  
осторожно°положила·на°берег°и°самого°почтальона.°Он°лежал°на·спине,·закинув·  
руки,°как°будто°заслонясь°от·солнца,·  
ещё°совсем°молодой,°белокурый,°в°форменной°тужурке°с·  
блестящими°пуговицами:°должно°быть,°отправляясь°в°свой°последний·рейс,°почтальон·  
начистил°их°мелом.¶

¶

Сумку°отобрал·городовой,°а°и°письма,°так°как·они·размокли°и·  
уже°никуда°не°годились,°взяла°себе°тётя°Даша.°Но·они°не°совсем°размокли:°сумка·  
была°новая,°кожаная°и°плотно°запиралась.·Каждый·вечер°тётя°Даша°читала°вслух·по·  
одному°письму,°иногда°только°мне,°а°иногда·всему°двору.°Это°было·так°интересно,°что·даже·  
старухи,°ходившие°к°Сковородникову°играть·в°«козла»,°бросали·карты°и·  
присоединялись°к°нам.·Одно°из°этих·писем°тётя°Даша°читала°чаще·других°—  
так°часто,°что°в°конце°концов°я°выучил°его°наизусть.°С°тех°пор°прошло°много·  
лет,°но°я°ещё°помню°его°от°первого·до°последнего°слова.¶

¶

«Глубокоуважаемая·Мария·Васильевна!¶

```
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> py .\methog2.py
1.steganography
2.find secret text
2
decode:
Please open file SecretText.txt
```

### File SecretText.txt

```
SecretText.txt
You, a few seconds ago | 1 author (You)
1 | Санкт You, a few seconds ag
```

### Проведение экспертной оценки:

Также мною был написан файл оценщик `analy.py`. Этот скрипт сравнивает два файла до вставки сообщения и после – выводит размер каждого файла.

### Метод замены символов

```
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> py .\analy.py
size before steganography:49597
size after steganography: 49547
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> |
```

### Метод с добавлением служебных символов

```
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> py .\analy.py
size before steganography:49597
size after steganography: 49672
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> |
```

### Метод с добавлением служебных символов:

```
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> py .\analy.py
size before steganography:49597
size after steganography: 49672
PS E:\курс 3\ki2\Основы стеганофии\lab\lab1\code> |
```

Можно сделать вывод, что в первых двух случаях встраивание целесообразно, так как текст сам по себе весит внушительное количество байт, из-за чего сообщение скрыто, а вот в третьем случае – нет. Это произошло потому, что изменяются не только необходимые символы, но и все последующие, чтобы обеспечить визуальную незаметность сообщения в стегоконтейнере. К сожалению, размер файла при этом сильно меняется.

Если для просмотра сообщения пользоваться программой, которая не поддерживает отображение всех знаков, то во всех случаях обнаружить сообщение крайне сложно. Если не знать наверняка, что в тексте что-то содержится, то неподготовленный человек ничего не обнаружит. Если же использовать программу, которая позволит отображать все символы, то первый случай сразу же вызовет подозрения, во втором, если приглядываться, тоже можно понять, что что-то не так, а вот в третьем все не так просто. Создается впечатление, что это просто такой стиль письма. Но если иметь исходный текст и стегоконтейнер, полученный с помощью третьего метода, то можно с легкостью сделать вывод, что что-то было встроено, посмотрев на размеры файлов. Этот факт стал для меня решающим при проведении экспертной оценки – данный метод скрытия я не могу назвать успешным.

### **Выводы:**

При выполнении данной лабораторной работы мною были изучены основные методы текстовой стеганографии. Я научился применять их и проводить последующую оценку их применению. По результатам работы были сделаны следующие выводы:

Метод замены символов одинакового начертания легко обнаружим при использовании многофункциональных текстовых редакторов, метод хвостовых пробелов трудно заметить даже при использовании таких редакторов, а реализованный мною метод добавления служебных символов вовсе не целесообразен, хоть и обнаружить его при отсутствии исходного текста довольно сложно.

Если я захочу в будущем поместить стего в текст, то, скорее всего, воспользуюсь методом знаков одинакового начертания, который предполагает замену символа из текста на его аналог из другого языка, которые выглядят точно так же, но имеет другую кодировку

### **Список использованной литературы:**

1. Стеганография & путешествия [Электронный ресурс] – URL: [http://www.nestego.ru/2012/05/blog-post\\_05.html](http://www.nestego.ru/2012/05/blog-post_05.html)
2. Стеганография & путешествия [Электронный ресурс] – URL: [http://www.nestego.ru/2012/05/blog-post\\_03.html](http://www.nestego.ru/2012/05/blog-post_03.html)
3. Основные положения стеганографии [Электронный ресурс] – URL: <http://citforum.ru/internet/securities/stegano.shtml>



## Приложение

Программа 1. Метод замены символов одинакового начертания.

```
def ChangeLetterToBinnary(letter):
    binary=''
    binary = '{0:08b}'.format(ord(letter)-1040)
    return binary

# transform string text to binary
def ChangeToBinary(text):
    binary = ''
    for i in range(len(text)):
        binary = binary + ChangeLetterToBinnary(text[i])
    return binary

# transform binary to string text
def ChangetoString(text):
    result = ''
    for i in range(len(text)//8):
        a = text[i*8:(i+1)*8] # every symbol using a 8bit-encode
        b = int(a,2)
        result += chr(b+1040)
    return result

def steganography(text,data):
    binary = list(ChangeToBinary(text))
    data = list(data)
    index = -1
    for i in range(len(binary)):
        if binary[i] == '0':
            index = data.index('c',index+1) # find index of symbol c in russian
            data[index] = 'c' # change to c in english
        elif binary[i] == '1':
            index = data.index('p',index+1) # find index of symbol p in russian
            data[index] = 'p' # change to p in english
    return ''.join(data)

def findSecretText(data):
    result = ''
    index = -1
    while True:
        index_c = data.find('c',index+1)
```

```

        index_p = data.find('p',index+1)
        if (index_c == -1) and (index_p == -1): break
        if ((index_c < index_p) and (index_c != -1))or(index_p == -1):
            index = index_c
            result += '0' # add bit 0 to result
        elif index_p != -1:
            index = index_p
            result += '1' # add bit 1 to result
    return ChangetoString(result)
# -----main programme-----#

option = int(input('1.steganography \n2.find secret text\n'))
if option==1:
    file_in = open('original_document.txt',mode = 'r',encoding='UTF-8')
    file_out = open('output.txt',mode = 'w',encoding='UTF-8')
    data = file_in.read() # read data from original_document
    text = input('please enter secret text \n') # input the text, that we must en
code
    cipher_text = steganography(text,data)
    file_out.write(cipher_text)
    print('please open file output.txt!')
    file_in.close()
    file_out.close()
elif option==2:
    file_encode = open('output.txt',mode = 'r',encoding='UTF-8')
    data = file_encode.read()
    file_secret = open('SecretText.txt',mode = 'w',encoding='UTF-8')
    text_secret=findSecretText(data)
    file_secret.write(text_secret)
    print('Please open file SecretText.txt\n')
    file_secret.close()
    file_encode.close()

```

Программа 2. Метод хвостовых пробелов.

```

def ChangeLetterToBinnary(letter):
    binary=''
    binary = '{0:08b}'.format(ord(letter)-1040)
    return binary

# transform string text to binary
def ChangeToBinary(text):
    binary = ''

```

```

    for i in range(len(text)):
        binary = binary + ChangeLetterToBinnary(text[i])
    return binary
# transform binary to string text
def ChangetoString(text):
    result = ''
    for i in range(len(text)//8):
        a = text[i*8:(i+1)*8] # every symbol using a 8bit-encode
        b = int(a,2)
        result += chr(b+1040)
    return result

def steganography(text,data):
    binary = list(ChangeToBinary(text))
    data = list(data)
    index = -1
    for i in range(len(binary)):
        if binary[i] == '0':
            index = data.index('\n',index+1) # find index of endlne
            data[index] = ' \n' # we add 1 spacebar right before endlne
        elif binary[i] == '1':
            index = data.index('\n',index+1) # find index of endlne
            data[index] = '\t\n' # we add 1 tab right before endlne
    return ''.join(data)

#find secret text
def findSecretText(data):
    result = ''
    index = -1
    while True:
        index1 = data.find(' \n',index+1)
        index2 = data.find('\t\n',index+1)
        if (index1 == -1) and (index2 == -1): break
        if ((index1 < index2) and (index1 != -1))or(index2 == -1):
            index = index1
            result += '0' # add bit 0 to result
        elif index2 != -1:
            index = index2
            result += '1' # add bit 1 to result
    return ChangetoString(result)

# -----main programme-----#
option = int(input('1.steganography\n2.find secret text\n'))

```

```

if option==1:
    file_in = open('original_document.txt',mode = 'r',encoding='UTF-8')
    file_out = open('output.txt',mode = 'w',encoding='UTF-8')
    data = file_in.read()
    text = input('please enter secret text \n') # input the text, that we must en
code
    file_out.write( steganography(text,data))
    print('please open file output.txt!')
    file_in.close()
    file_out.close()
elif option==2:
    file_encode = open('output.txt',mode = 'r',encoding='UTF-8')
    data = file_encode.read()
    print('decode:')
    file_secret = open('SecretText.txt',mode = 'w',encoding='UTF-8')
    text_secret=findSecretText(data)
    file_secret.write(text_secret)
    print('Please open file SecretText.txt\n')
    file_secret.close()
    file_encode.close()

```

Программа 3. Добавление служебных символов.

```

def ChangeLetterToBinnary(letter):
    binary=''
    binary = '{0:08b}'.format(ord(letter)-1040)
    return binary

# transform string text to binary
def ChangeToBinary(text):
    binary = ''
    for i in range(len(text)):
        binary = binary + ChangeLetterToBinnary(text[i])
    return binary

# transform binary to string text
def ChangetoString(text):
    result = ''
    for i in range(len(text)//8):
        a = text[i*8:(i+1)*8] # every symbol using a 8bit-encode
        b = int(a,2)
        result += chr(b+1040)
    return result

```

```

def steganography(text,data):
    binary = list(ChangeToBinary(text))
    data = list(data)
    index = -1
    for i in range(len(binary)):
        if binary[i] == '0':
            index = data.index('\n',index+1) # find index of newline
            data[index] = chr(127)+'\n' # we add a special character (that
            # don't show when we open document)
        elif binary[i] == '1':
            index = data.index('\n',index+1) # find index of newline
            data[index] = chr(8196)+'\n' # we add a special character
            # (that don't show when we open document)
    return ''.join(data)

def finSecretText(data):
    result = ''
    index = -1
    while True:
        index1 = data.find( chr(127)+'\n',index+1)
        index2 = data.find(chr(8196)+'\n' ,index+1)
        if (index1 == -1) and (index2 == -1): break
        if ((index1 < index2) and (index1 != -1)) or (index2 == -1):
            index = index1
            result += '0' # add bit 0 to result
        elif index2 != -1:
            index = index2
            result += '1' # add bit 1 to result
    return ChangetoString(result)

# -----main programme-----#
option = int(input('1.steganography\n2.find secret text\n'))
if option==1:
    file_in = open('original_document.txt',mode = 'r',encoding='UTF-8')
    file_out = open('output.txt',mode = 'w',encoding='UTF-8')
    data = file_in.read() # read data from container
    text = input('please enter secret text \n') # input the text, that we must encode
    file_out.write( steganography(text,data))
    print('please open file output.txt!')
    file_in.close()
    file_out.close()
elif option==2:
    file_encode = open('output.txt',mode = 'r',encoding='UTF-8')
    data = file_encode.read()
    print('decode:')

```

```
file_secret = open('SecretText.txt',mode = 'w',encoding='UTF-8')
text_secret=finSecretText(data)
file_secret.write(text_secret)
print('Please open file SecretText.txt\n')
file_secret.close()
file_encode.close()
```

Прорам4: analy.py

```
from os.path import getsize
size_before=getsize('orginal_document.txt')
size_after=getsize('output.txt')
print(f'size before steganography:{size_before}')
print(f'size after steganography: {size_after}')
```