МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГБОУ ВО «АЛТАЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт цифровых технологий, электроники и физики

Кафедра вычислительной техники и электроники (ВТиЭ)

Лабораторная работа № 6

**Целостность информации. Простейшие методы стеганографии. Хеширование.**

Выполнил студент 595 гр.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.В. Лаптев

Проверил:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ П.С. Ладыгин

Лабораторная работа защищена

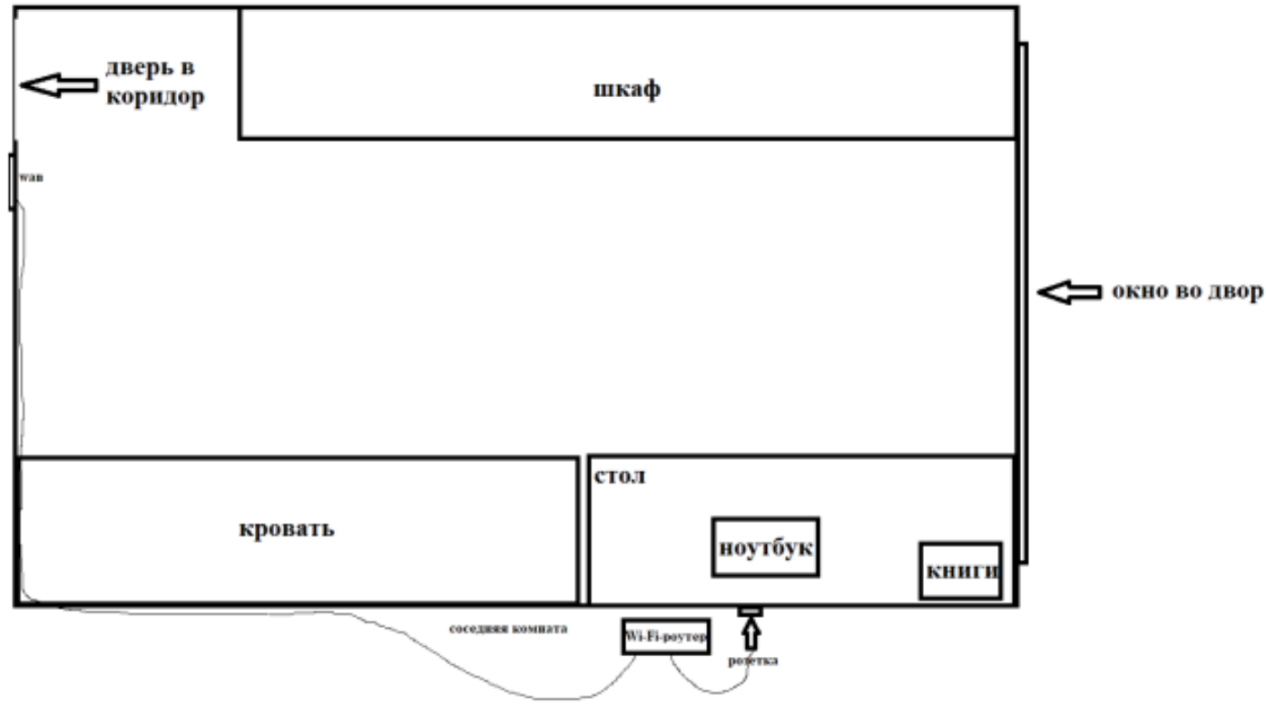
«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2023 г.

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

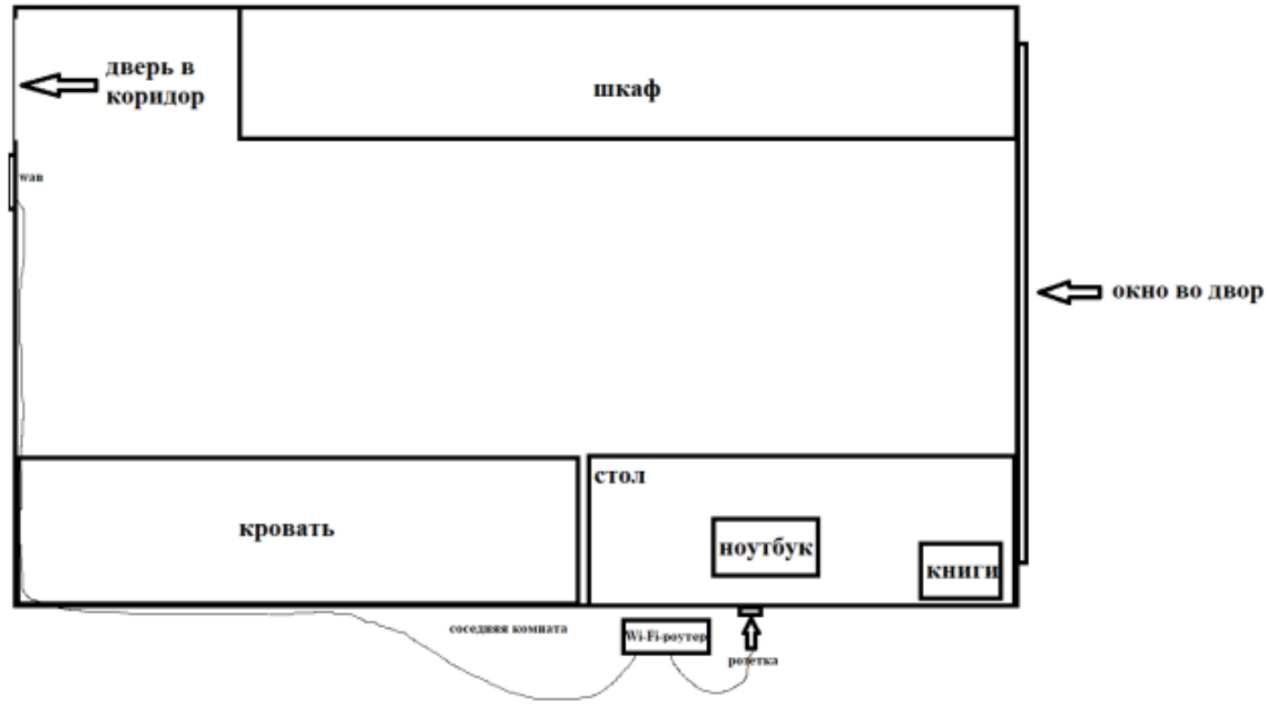
**Цель работы:** рассмотрение методов проверки целостности информации на примере наиболее распространенных методов хэширования.

**Задачи:**

1. Примените LSB к одному из изображений из вашей Модели угроз, предварительно сохранив его в удобном формате.
2. Продемонстрируйте в отчёте работоспособность выбранного способа реализации LSB (скриншоты и описание к ним).
3. Используя одну из рассмотренных хэш-функций, показать различие или совпадение хэшей двух изображений.

**

*Рис. 1. Изображение до встраиваемого скрытого текста.*

**

*Рис. 2. Изображение после встраивания скрытого текста.*

Текст, который встраивается в изображение был следующим:

«Hello World! It`s test LSB method.»

**Листинг части программы, отвечающий за иллюстрацию метода LSB:**

def lsb\_coder():

    # Применение метода LSB

    input\_text = "Hello World! It`s test LSB method."

    encode\_text = lsb.hide(Image.open("lab6.png"), input\_text)

    encode\_text.save("outlab6.png")

Для хэширования был выбран алгоритм MD5 и использовалась библиотека hashlib.

MD5: Алгоритм производит хеш со значением в 128 битов. Широко используется для проверки целостности данных. Не подходит для использования в иных областях по причине уязвимости в безопасности MD5.

Подсчитанные хэш-суммы для изображений следующие:

Хэш исходного файла: e28babdb7ee57660238ff6f27a2607b3

Хэш зашифрованного файла: 176d04f15729b307c4c08e8814fe5256

**Листинг кода, который отвечает за хэш-суммы:**

def hash\_check():

# Сравнение хэшей

md5\_result\_input = hashlib.md5(Image.open("lab6.png").tobytes())

print('Хэш исходного файла:\t', md5\_result\_input.hexdigest())

md5\_result\_output = hashlib.md5(Image.open("outlab6.png").tobytes())

print('Хэш зашифрованного файла:\t', md5\_result\_output.hexdigest())

**Вывод:** В ходе лабораторной работы были рассмотрены методы проверки целостности информации на примере наиболее распространенных методов хеширования.

**Ответы на контрольные вопросы:**

1. В какой деятельности могла бы пригодиться стеганография для вас?

О: Создание водяных знаков для защиты собственного контента от использования без упоминания автора; хранение информации, которую желаю скрыть от окружающих; для развлекательных целей.

1. Не используя сети Интернет попробуйте придумать свой способ скрыть сообщение в контейнере. Опишите в 3-5 предложениях.

О: Можно использовать комбинацию LSB и шифра Вижинера. Использовать шифрование в картинке, видео или аудио со сложным ключом, длина и сдвиг которого генерируются случайно. Таким образом можно будет шифровать случайные биты исходного файла, что позволит несколько надежнее защитить информацию.

1. Какая хэш-функция наименее защищена от подбора исходного слова на основе хэша?

О: Из представленных в методичке менее всего защищена md5. Да и в целом алгоритмы md4 и md5 считаются менее защищенными в этом плане.

1. В каких задачах наиболее применим md5?

О: MD5 широко используется для проверки целостности данных. Предназначен для создания контрольных сумм или «отпечатков» сообщения произвольной длины и последующей проверки их подлинности.

**Листинг программы:**

from stegano import lsb

import hashlib

from PIL import Image

def lsb\_coder():

# Применение метода LSB

input\_text = "Hello World! It`s test LSB method."

encode\_text = lsb.hide(Image.open("lab6.png"), input\_text)

encode\_text.save("outlab6.png")

def hash\_check():

# Сравнение хэшей

md5\_result\_input = hashlib.md5(Image.open("lab6.png").tobytes())

print('Хэш исходного файла:\t', md5\_result\_input.hexdigest())

md5\_result\_output = hashlib.md5(Image.open("outlab6.png").tobytes())

print('Хэш зашифрованного файла:\t', md5\_result\_output.hexdigest())

if \_\_name\_\_ == '\_\_main\_\_':

print("1. Применение LSB к изображению.")

print("2. Сравнение хэшей двух изображений.")

variant = int(input("Введите вариант работы программы: "))

while variant < 1 or variant > 2:

variant = int(input("Введите вариант работы программы: "))

if variant == 1:

lsb\_coder()

elif variant == 2:

hash\_check()