Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ОТЧЕТ

О ЛАБОРАТОРНОЙ РАБОТЕ

Лабораторная работа №2

по теме:

«Применение стеганографических методов для сокрытия информации»

Исполнитель, студент группы 201-361 \_\_\_\_\_\_\_\_\_\_\_\_ Н.А. Фельдбуш

Москва, 2023

**Постановка задачи**:

Напишите программу для внедрения в файл 28.bmp и извлечения из него хешкода файла leasing.txt, полученного с помощью алгоритма SHA-1. Метод реализации – LSB Replacement. Номера байтов-контейнеров должны содержаться в предварительно сгенерированном ключе. С помощью сторонних приложений оцените объемы получившегося и исходного изображений после сжатия.

**Ход работы:**

Выполним поставленную задачу при помощи средств языка программирования Python. Перед началом следует обратить внимание на две вспомогательных функции (рис.1).

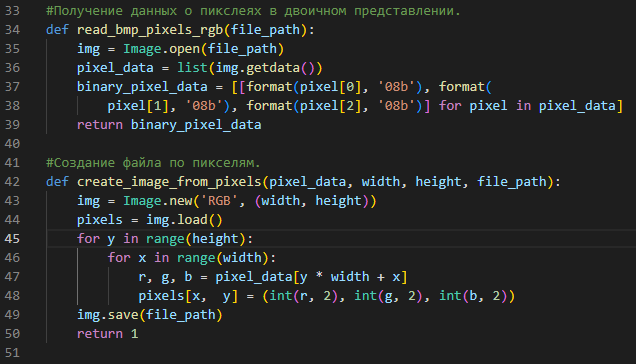


Рисунок 1 – Вспомогательные функции

Функция “read\_bmp\_pixels\_rgb”, отсекает заголовочные данные с помощью библиотеки “Pillow”, откуда вызвана функция “Image”, и передает только значения байтов пикселей. Далее для каждого пикселя происходит перевод из байтов в биты, а после их запись в массив для дальнейшей работы.

Функция “create\_image\_from\_pixels” создает новое изображение по полученным данным. С помощью библиотеки “Pillow” создается rgb файл, в который один за одним добавляются пиксели. Затем файл сохраняется по указанному пути.

Так же стоит обратить внимание на функции “keygen”, которая генерирует ключ, который в дальнейшем будет использоваться для замены битов (рис.2).

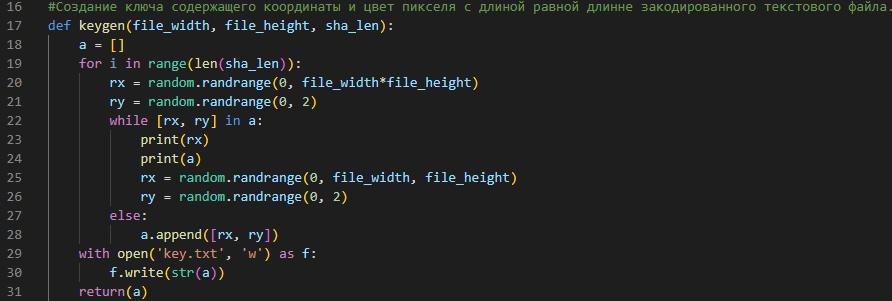


Рисунок 2 – Создание ключа

В цикле for происходит выбор случайного пикселя(rx) и случайного байта цвета(ry). После происходит проверка, если этот байт уже был использован происходит перегенерации значений до тех пор, пока не будет найден уникальный байт. Затем происходит запись ключа в файл.

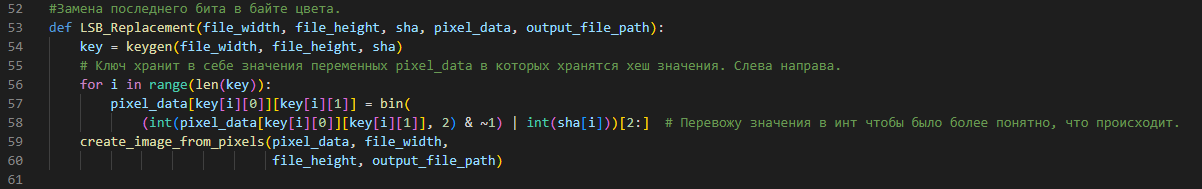
Замена битов происходит с помощью функции “LSB\_Replacement” (рис.3). 

Рисунок 3 – Замена битов

В данной функции происходит получение сгенерированного ключа, который затем используем для получения нужного пикселя. После с помощью операции “& ~1” происходит обнуление последнего бита в последовательности бит пикселя. А затем с помощью операции “|” установка последнего бита в необходимое нам значение

Для получения хэш-кода из изображения необходимо использовать функцию “return\_sha1\_hash” (рис.4)

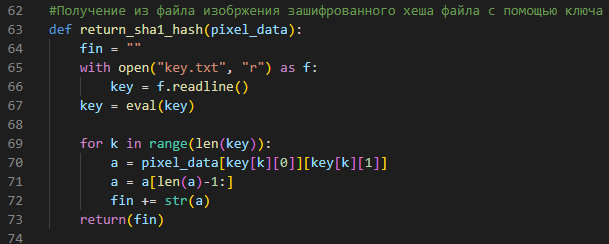


Рисунок 4 – Замена битов

В функции происходит обратное действие: по ключу мы получаем значения, которые записывали в функции “LSB\_Replacement”.

Результатом работы программы является новый файл изображения с закодированным внутри сообщением (рис.5).



Рисунок 5 – Замена битов