РЕАЛИЗАЦИЯ МЕТОДА ЦИФРОВОЙ СТЕГАНОГРАФИИ С ПОМОЩЬЮ МОДУЛЕЙ И БИБЛИОТЕК PYTHON

А.М. Ахапкина, С.В. Способ

В век высоких технологий информация представляется наибольшей ценностью. Поэтому не удивительно, что в последнее время создается множество средств для ее защиты. Стеганография – способ передачи или хранения информации с учетом сохранения в тайне самого факта такой передачи. В данном случаи у злоумышленника нет никаких зацепок, где искать закрытые и уже зашифрованные данные, как и, собственно, нету намеков, что уже что-то где-то спрятано.

Существует множество способов и методов стеганографии, каждые из которых преследуют свои цели. В статье будет рассмотрен метод наименее значащих битов (Least Significant Bit, LSB), который считается наиболее популярным для цифровой стеганографии. Цифровая стеганография основывается на ограниченности способностей органов чувств человека и, как следствие, неспособности распознать незначительные вариации звука/цвета. Для простоты понимания рассмотрим

графический контейнер – изображение. В данном формате для описания каждой точки (пикселя) используются 3 байта, обозначающие в какой пропорции необходимо смешивать красный, зеленый и голубой цвета (цветовая схема RGB). Если произвести замену старших бит в этих байтах, цветовые изменения в картинке будут бросаться в глаза. Младшие же биты дают куда более незначительный вклад в изображение. Если использовать по одному младшему биту в каждом цвете для записи скрываемого сообщения, то распознать изменения человеческий глаз будет не способен.

Алгоритм стеганографии можно реализовать с помощью различных языков программирования. В нашей статье будут рассмотрены модули и библиотеки Python. Программу, которая будет записывать и как следсвтие скрывать текст в изображение можно реализовать засчет модуля lsb. Однако, у данного модуля есть большой недостаток – восприятие кириллицы, данный модуль не распознает ее. Поэтому, если необходимо работать как с английским текстом, так и с кириллицей необходимо использовать модуль exifHeader

Однако, в независимости от выбора модуля и метода, отрытое сообщение легко разрушить, сжимая или отображая изображение. При таком подходе не обеспечивается секретность встраивания сообщения: точно известно местоположение информационных битов (каждый крайний с конца бит). Для преодоления второго недостатка можно встраивать сообщение не во все пиксели изображения, а выбирать их при помощи генератора псевдопростых чисел (инициализированного ключом стеганосистемы). Стоит заметить, что пропускная способность при этом уменьшится. Для генерации ключей необходимо воспользоваться библиотеками wheel и steganocryptopy. Генерация ключа происходит методом generate_key(), в параметрах которого необходимо передать путь, куда будет сохранен файл с ключом

Само шифрование происходит методом encrypt(), где параметрами передаются путь до ключа, изображение и путь до файла, в котором будет содержаться сообщение. Затем необходимо вызывать метод save() и передать в нем путь к изображению, в котором будет скрыт текст. Дешифрование происходит методом decrypt() схожим образом.

Естественно у данного метода стеганографии есть недостаток: видимость битых пикселей изображения в случаи скрытия большого количества символов. Однако этот недостаток отлично исправляется высоким разрешением изображения.