|  |  |
| --- | --- |
|  | **Российский государственный социальный университет** |

**Рубежный контроль № 6.**

**по дисциплине «Информационная безопасность»**

|  |  |
| --- | --- |
| **ФИО студента** | Салов Артём Владимирович |
| **Направление подготовки** | Программная инженерия |
| **Группа** | ПИН-Б-О-Д-2021-1 |

**Москва 2023**

# Анализ OpenPuff.

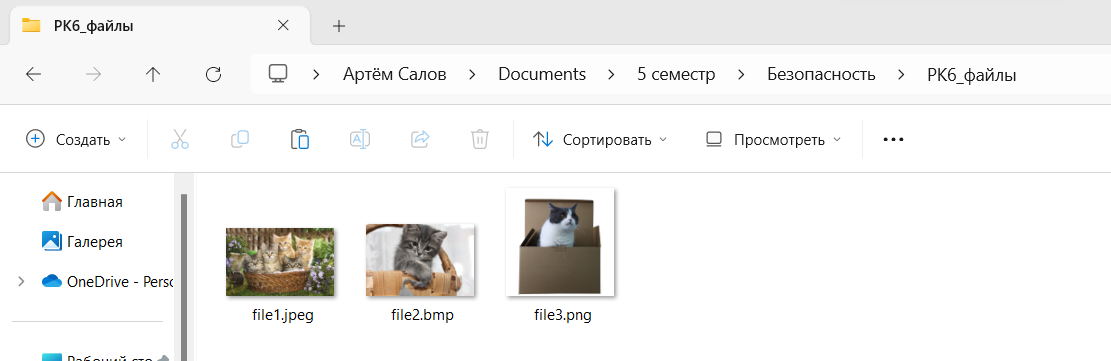


Рисунок 1 – Подготовленные файлы.

В качестве файла, который будет помещается в картинку, будет использован файл второго рубежного контроля – Рисунок 2.

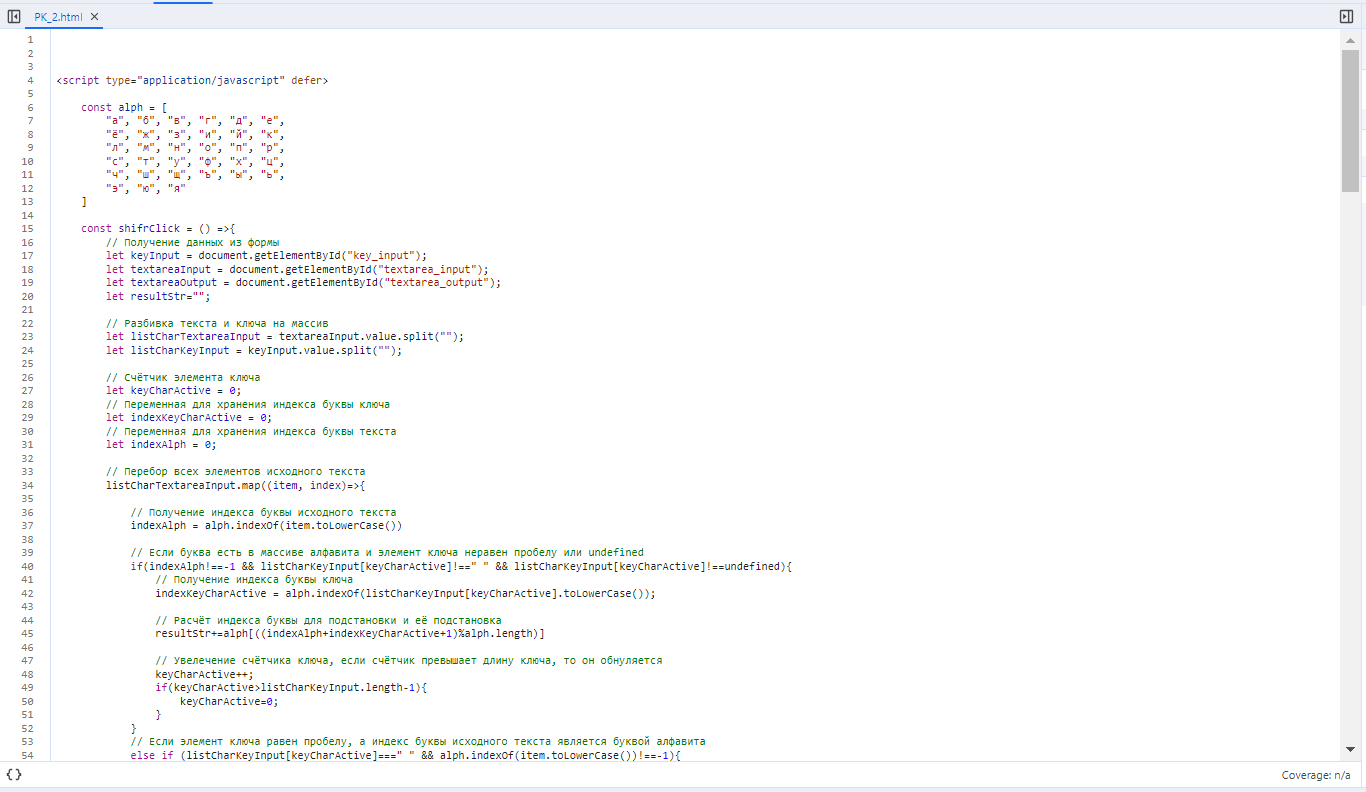


Рисунок 2 – Второй рубежный контроль.

# Процесс шифрования.

В качестве паролей A, B, C были сгенерированы три строки:

1. W\_OWjb4o1\_
2. k8k\_dGk5bv
3. VMVr6SYup-

При шифровании в настройках выставляю минимальный возможный размер для шифрования, пример настроек приведён на рисунке 3.

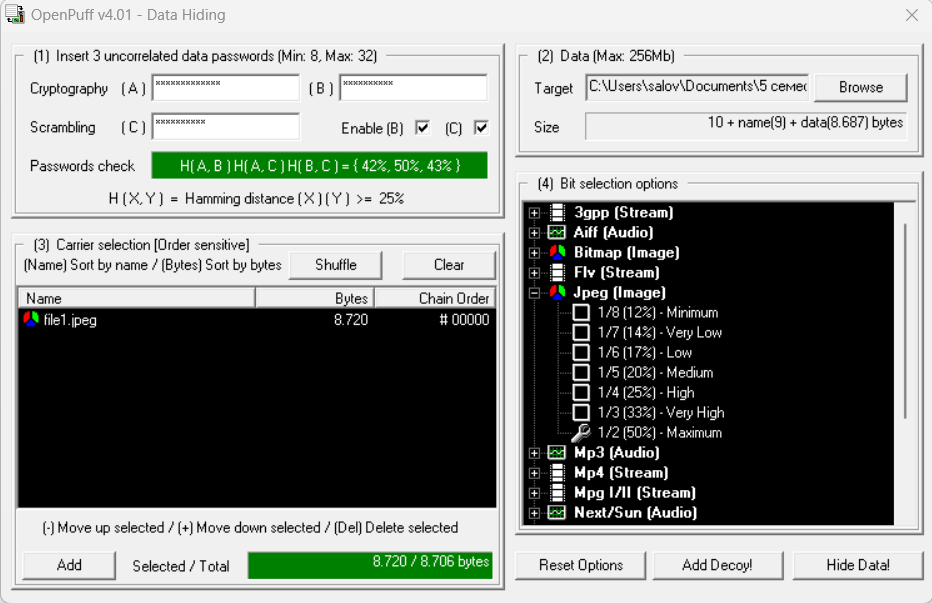


Рисунок 3 – Пример настроек для шифрования.

Таким образов для jpeg потребовалось выставить 50%, а для bmp и png – 12%.

Зашифрованные файлы помечены буквой «i» в конце названия, что видно на рисунке 4.

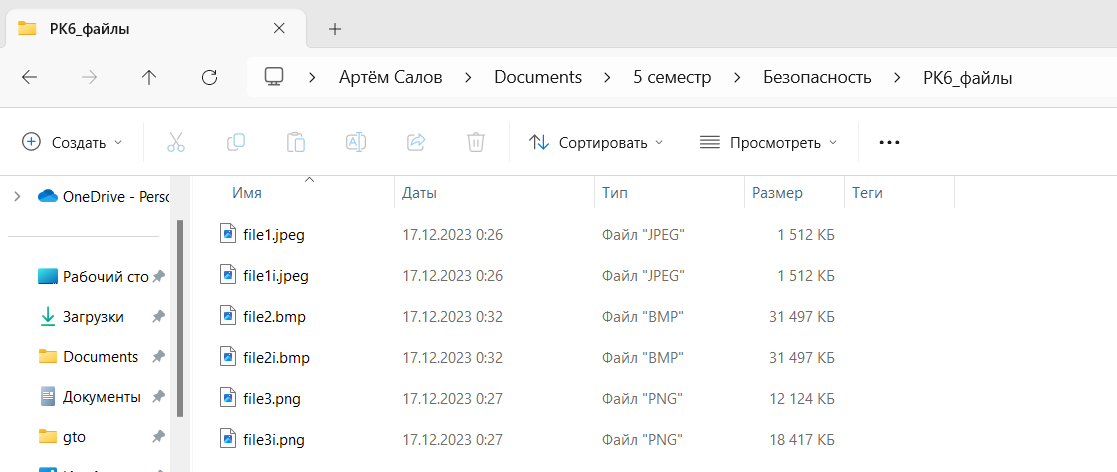


Рисунок 4 – Результат шифрования.

# Провести анализ работы приложения по следующим критериям:

## визуальная незаметность:

Для визуального сравнения был выбран ресурс <https://www.img2go.com/ru/compare-image>. Результаты сравнения приведены на рисунках 5-7.

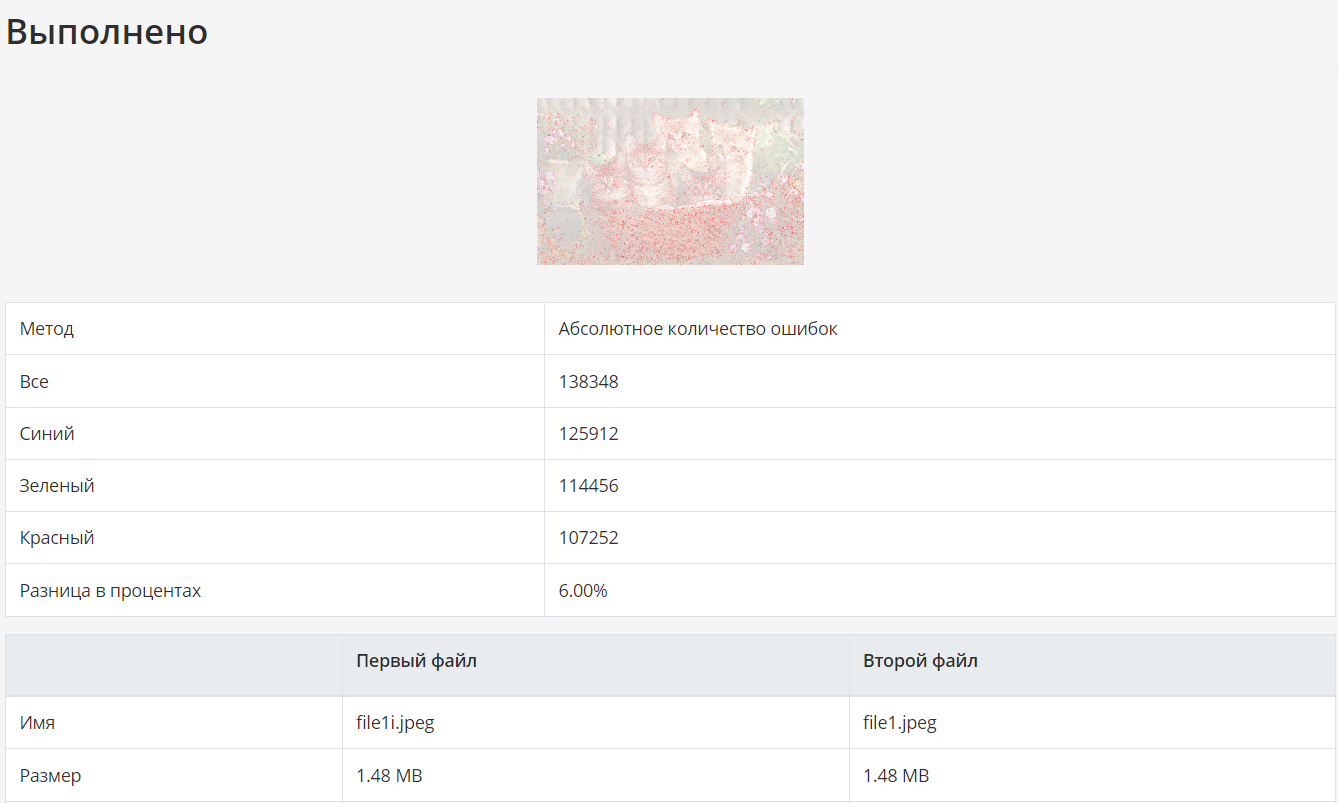


Рисунок 5 - Разница для jpeg файла составила 6%.

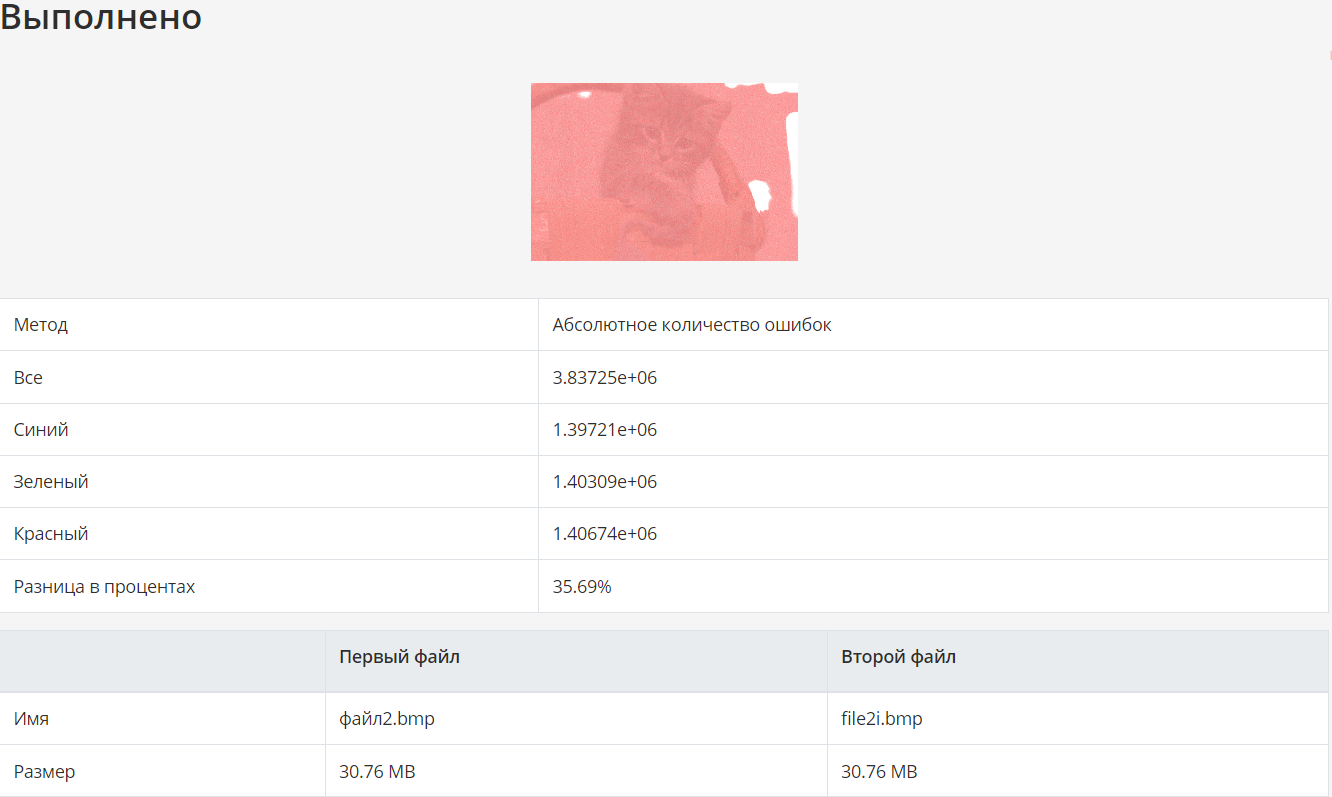


Рисунок 6 - Разница для bmp файла составила 35.69%.

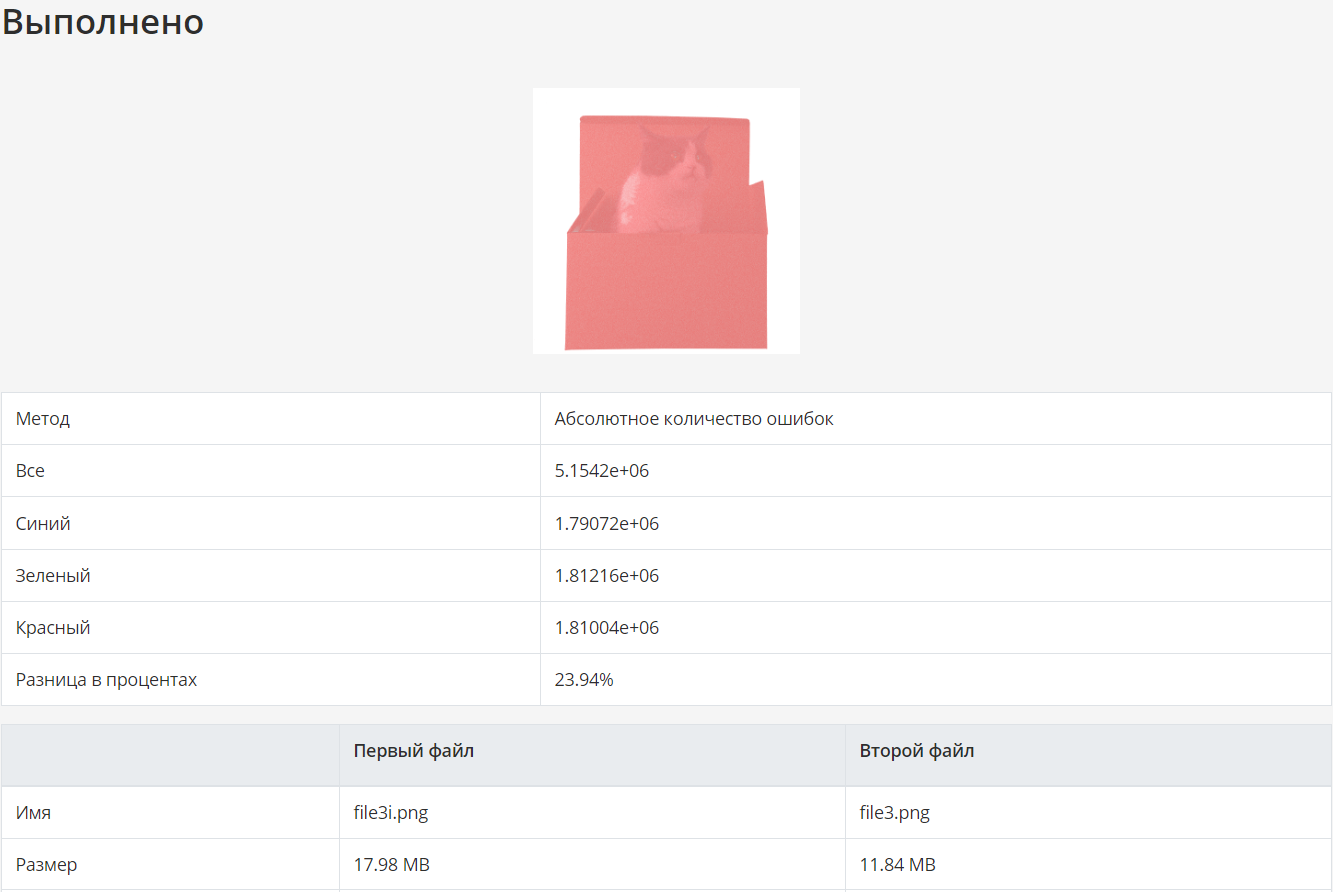


Рисунок 7 - Разница для bmp файла составила 23.94%.

Подводя итоги данного пункта, должно сказать, что искажение в первом файле при прямом сравнении человеком незаметны или очень трудно находимы, что не скажешь про третий и четвёртый файл, у которых при прямом сравнении сразу видны искажения цветов. Тем неимение, ни у одного файла артефакты не обнаружены.

## Эффективность по объему.

Данная программа позволяет настраивать занимаемый объём каждого отдельного типа данных от 12% до 50%.

## Устойчивость к стегоанализу.

В результате работы StegExpose, ни один файл не превысил порог stego, это значит, что скрытых сообщений обнаружено не было. Из чего можно сделать вывод о том, что у данного алгоритма отличная устойчивость к стегоанализу.

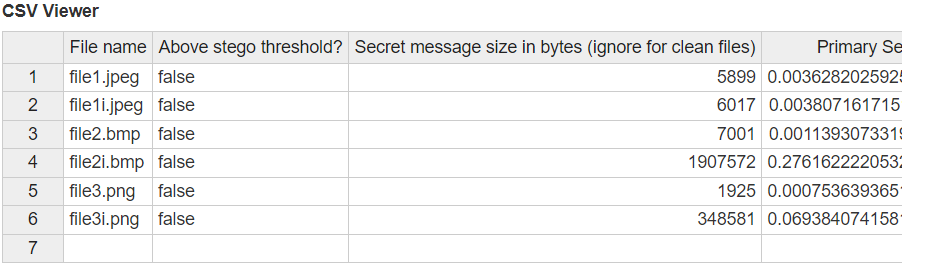


Рисунок 8 – Результаты работы программы StegExpose.

## Устойчивость к сжатию.

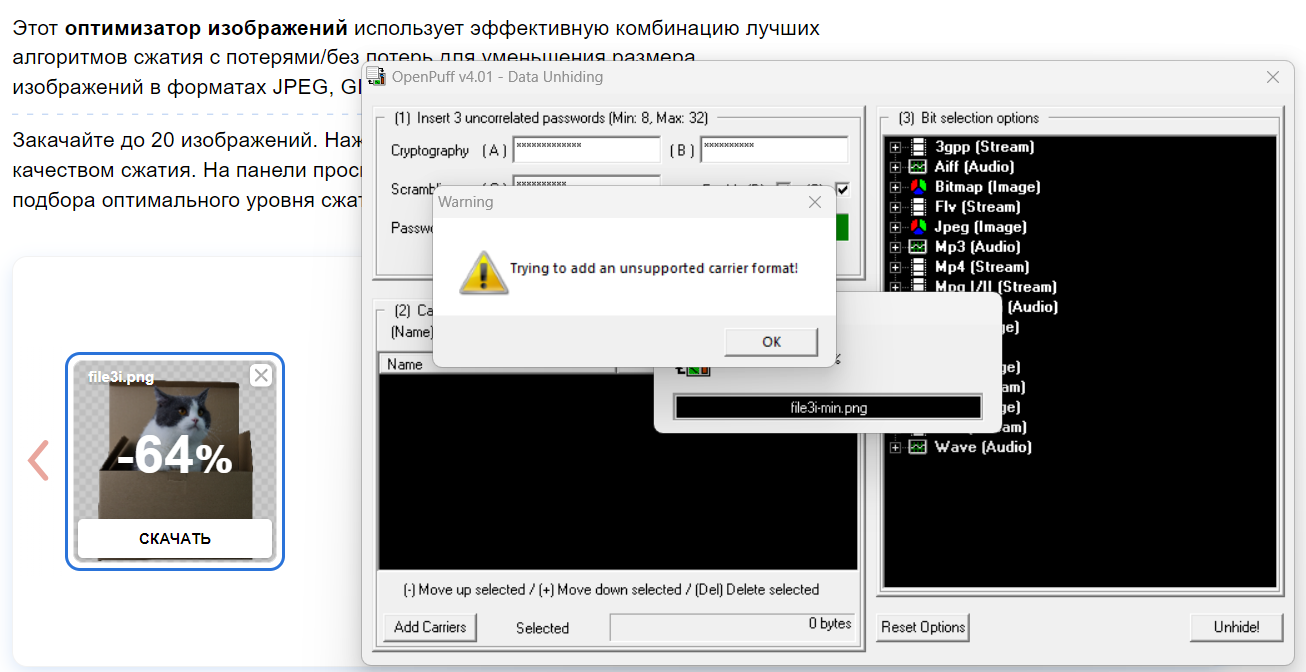


Рисунок 9 – Попытка дешифровать после сжатия.

После сжатия программа выдаёт ошибку о том, что не поддерживает данный тип файлов или «Задача остановлена: поврежден поток».

Итог: устойчивость к сжатию – низкая.

## Устойчивость к изменению данных (инвертировать цвета, изменить насыщенность и оттенок)

При попытке получить данные из изображения поле инвестиции цветов, программа выдала ошибку, что отображено на рисунке 10.

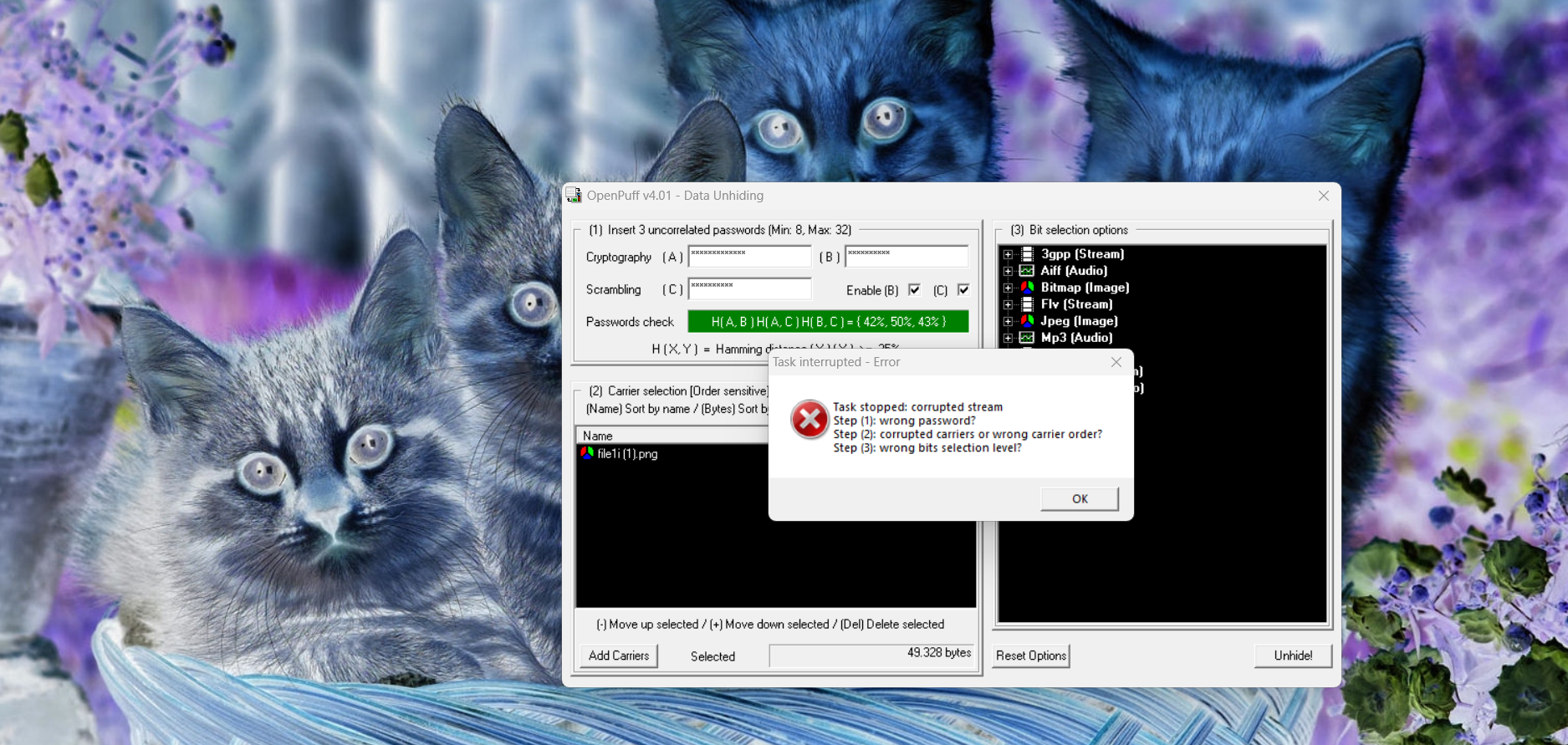


Рисунок 10 – Попытка дешифровать файл из картинки с инвертированными цветами.

Из этого можно сделать вывод, что у этой программы низкая устойчивость к изменению данных.

# Результаты анализа работы алгоритма.

|  |  |
| --- | --- |
|  | OpenPuff |
| Метод | LSB |
| Визуальная незаметность | Средняя |
| Эффективность по объему | От 12% до 50% |
| Устойчивость к стегоанализу | Высокая |
| Устойчивость к сжатию | Низкая |
| Устойчивость к изменению данных | Низкая |
| Область применения | Передача секретных файлов через инструменты, которые не обрабатывают изображения перед передачей |