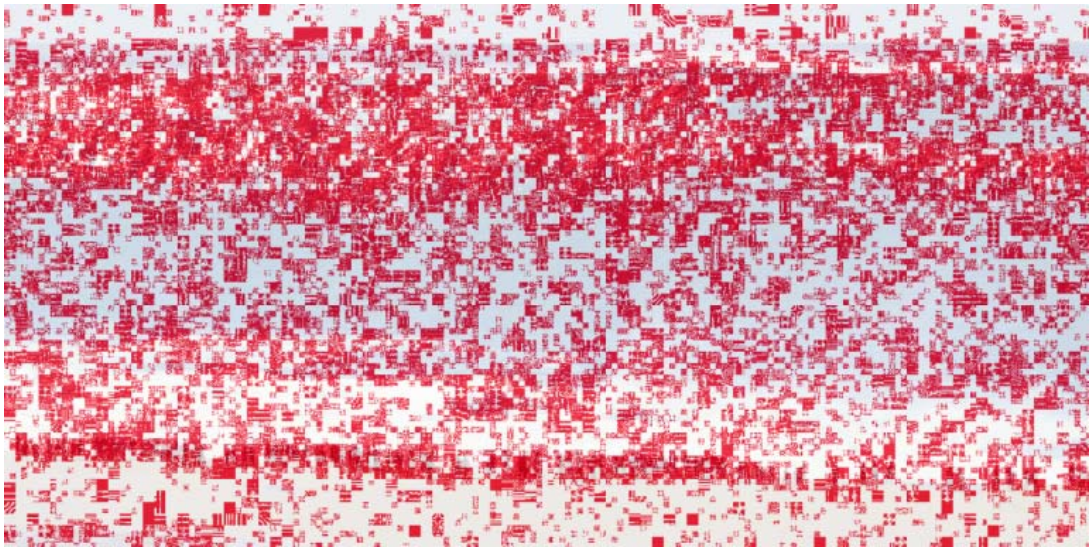
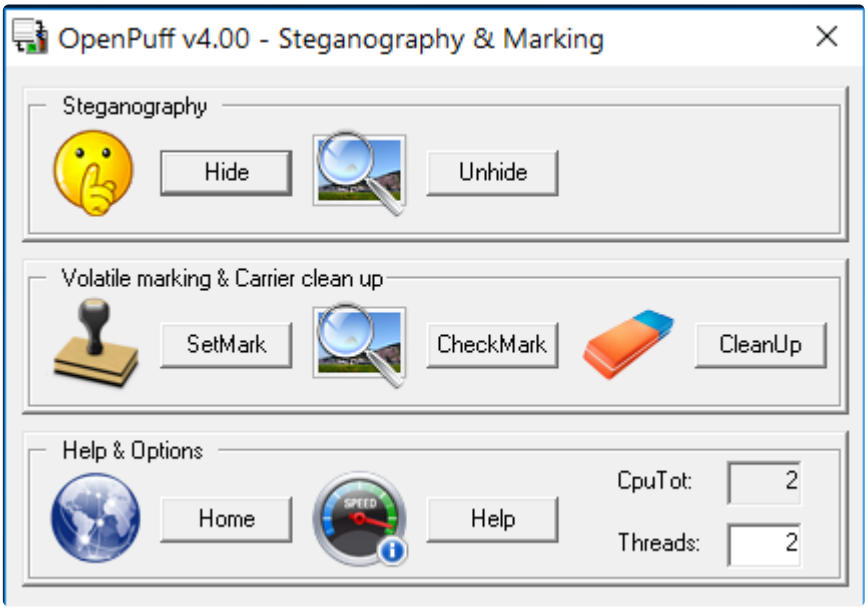
Как спрятать информацию в картинке. Полный курс стеганографии.

Шифрование помогает сохранять данные в секрете, но одновременно привлекает лишнее внимание. Если файл так просто не открыть, значит, в нем наверняка есть что-то ценное. Поэтому бывает важно скрыть само наличие секретной информации. Проще всего это сделать, растворив конфиденциальные данные внутри какого-нибудь безобидного файла. Решается такая задача с помощью стеганографических утилит, которые мы и протестируем.

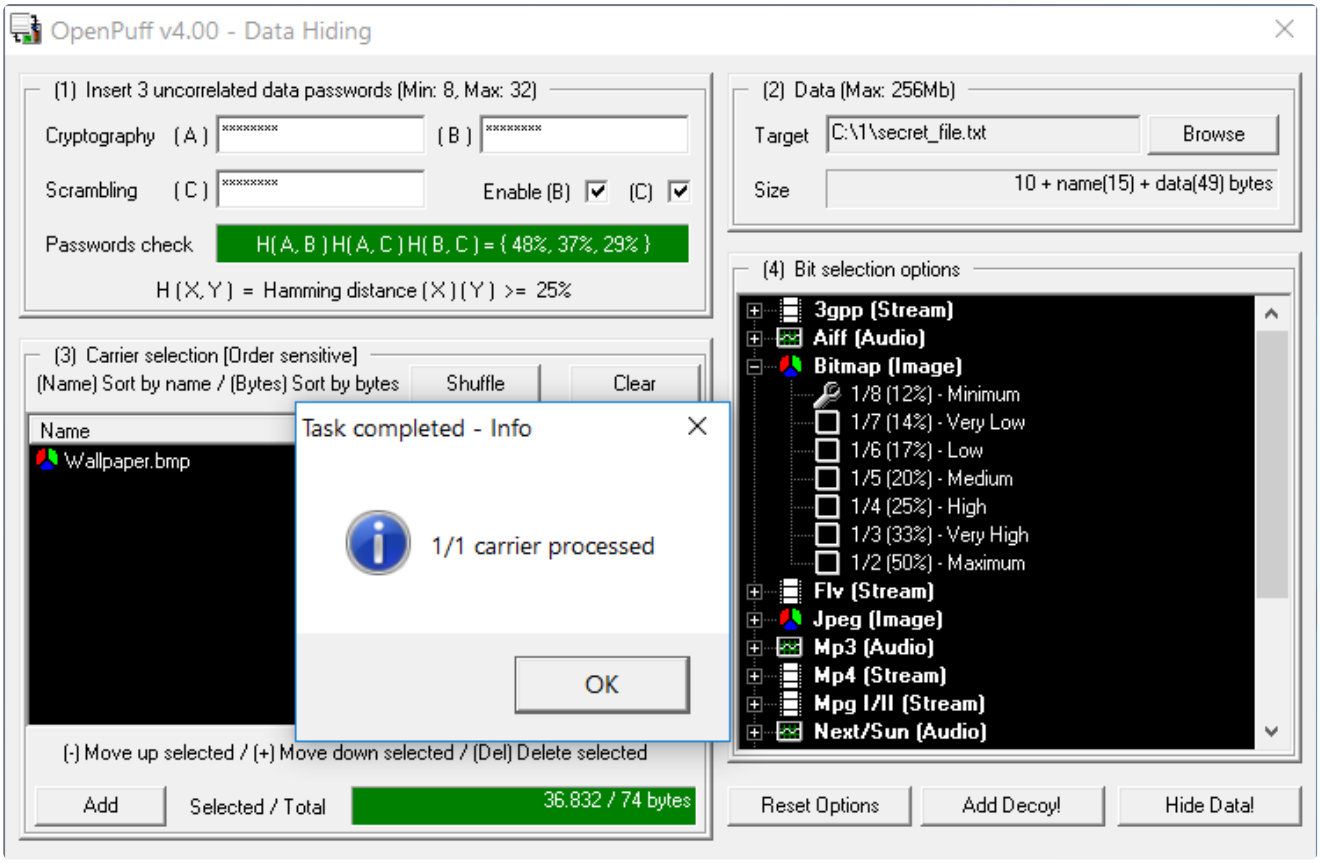


OpenPuff

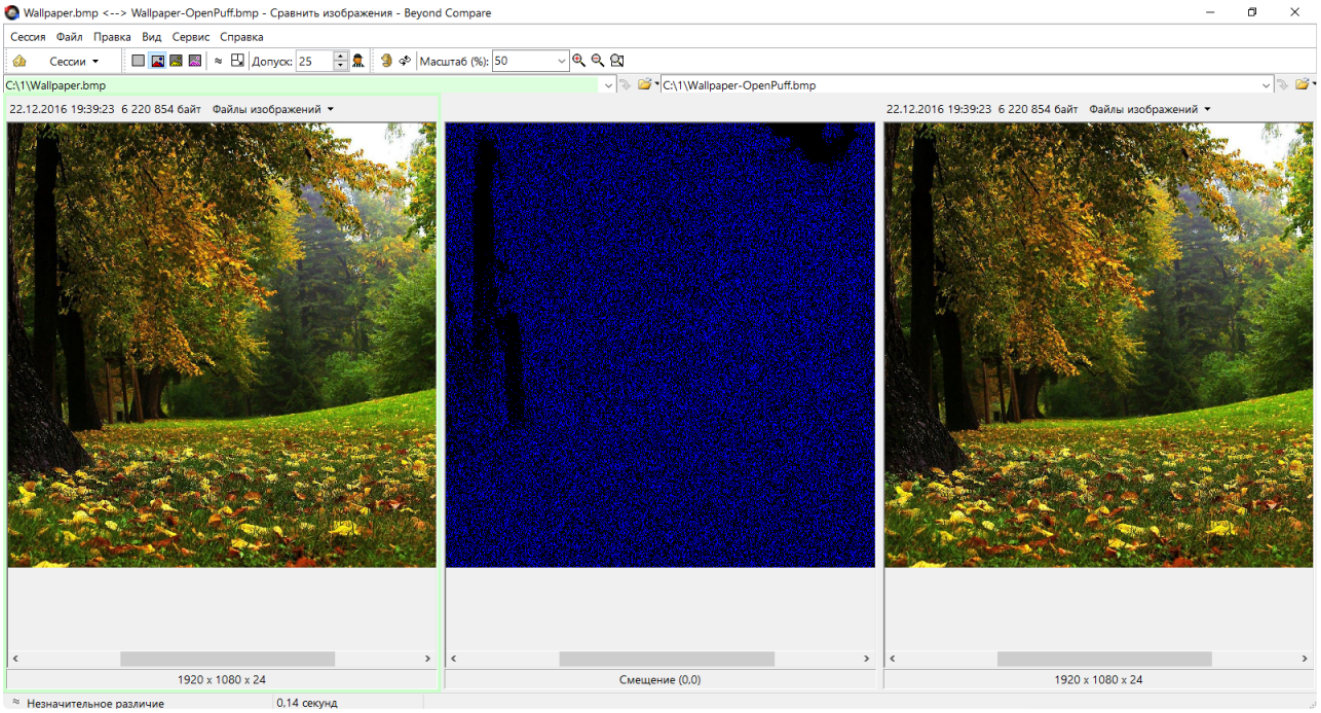
Самая сложная утилита в этом обзоре — OpenPuff. Ее последняя версия (4.00) поддерживает не только сокрытие одних файлов внутри других, но и работу со стегометками произвольного формата. Ей даже можно выделить несколько процессорных ядер, если предстоит большой объем работы.



В отличие от других утилит, поддерживающих парольную защиту скрываемого сообщения, OpenPuff умеет использовать для шифрования криптографически стойкий генератор псевдослучайных чисел (CSPRNG — Cryptographically secure pseudorandom number generator). Если простого пароля недостаточно, то поставь флажки напротив полей B и C, а затем введи в них три разных пароля длиной от 8 до 32 символов. На их основе CSPRNG сгенерирует уникальный ключ, которым и будет зашифровано сообщение.

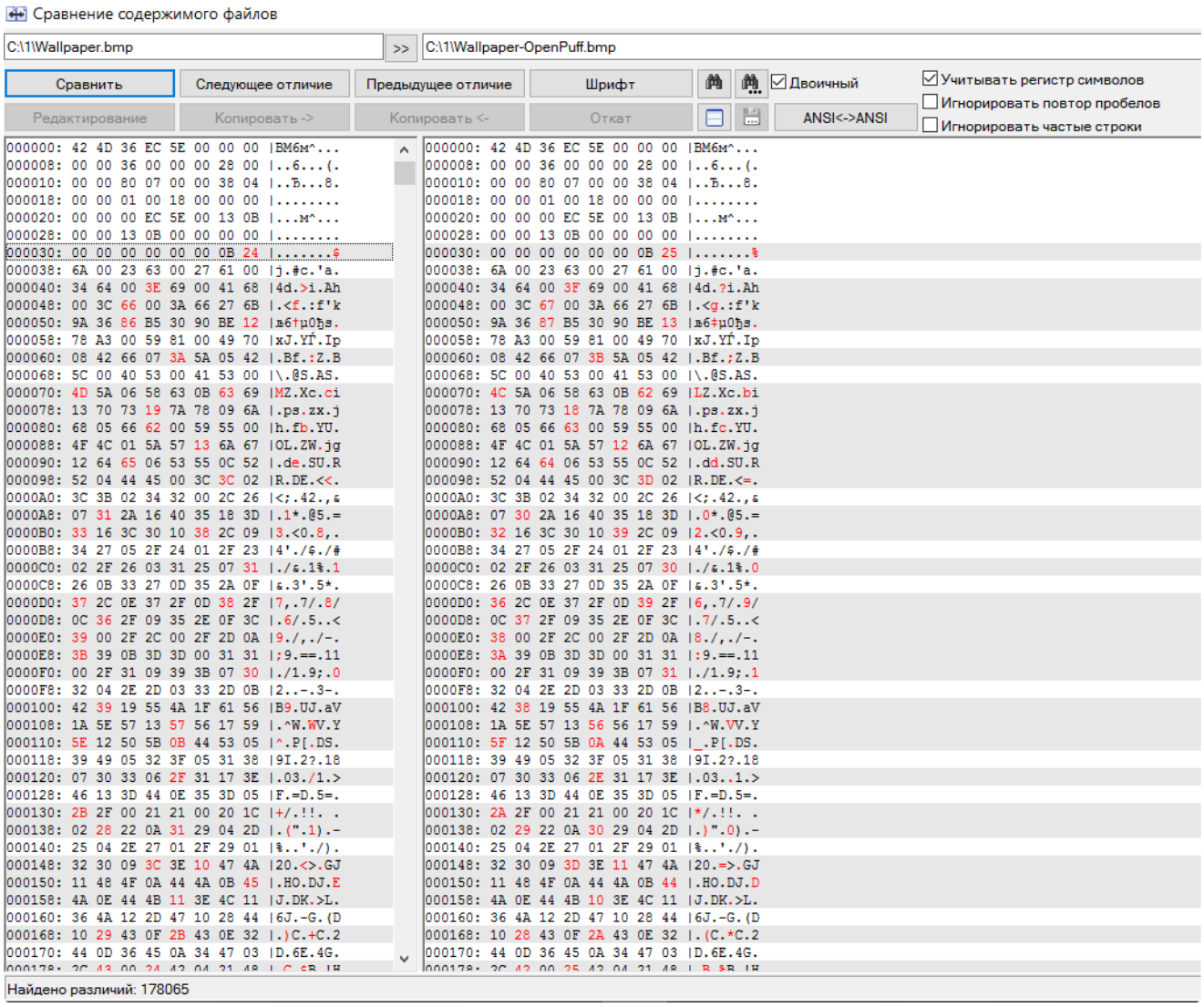


Мелкие файлы можно хранить в картинках и аудиозаписях, а крупные (например, криптоконтейнеры) удобнее прятать в видеозаписях — OpenPuff поддерживает MP4, MPG, VOB и множество других форматов. Максимальный размер скрываемого файла — 256 Мбайт.



Применение CSPRNG на малых файлах сильно увеличивает итоговый размер стегосообщения. Поэтому разница между пустым и заполненным контейнером становится слишком очевидной. Мы снова видим, что измененные пиксели в основном распределяются равномерно, однако на самых светлых и самых темных участках они образуют крупные блоки. Если бы таких блоков не было, результат был бы больше похож на артефакты, получаемые при сжатии при помощи JPEG.

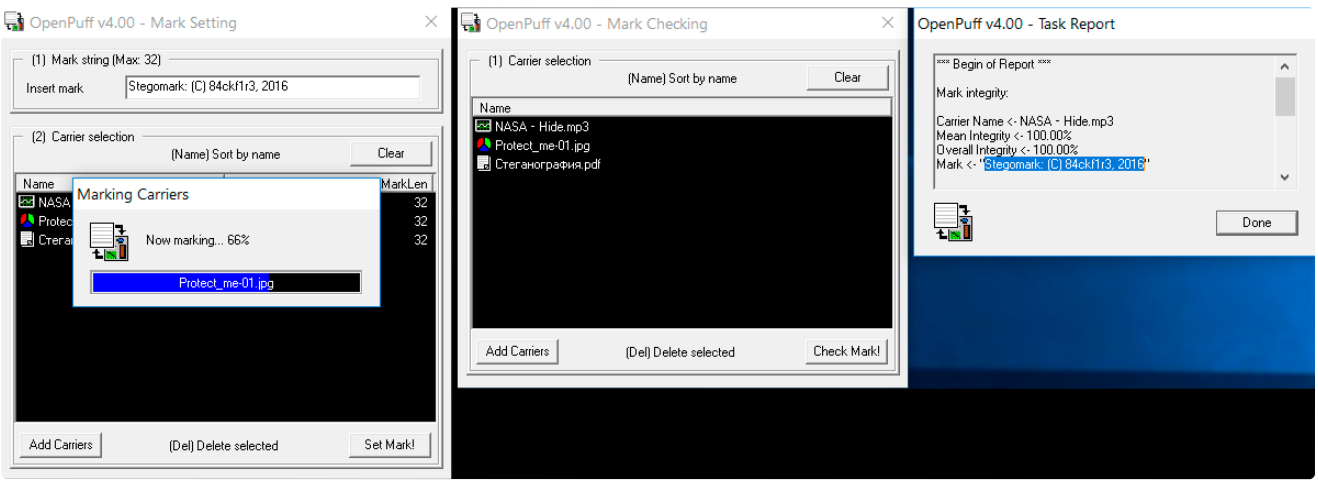
Побайтное сравнение тоже дает очень характерную картину. Несмотря на малый размер скрываемого файла, в контейнере изменены значения у большинства пикселей. Если jHide хватило 330 байт для записи стегосообщения, то OpenPuff использовал для этой же задачи более 170 Кбайт.



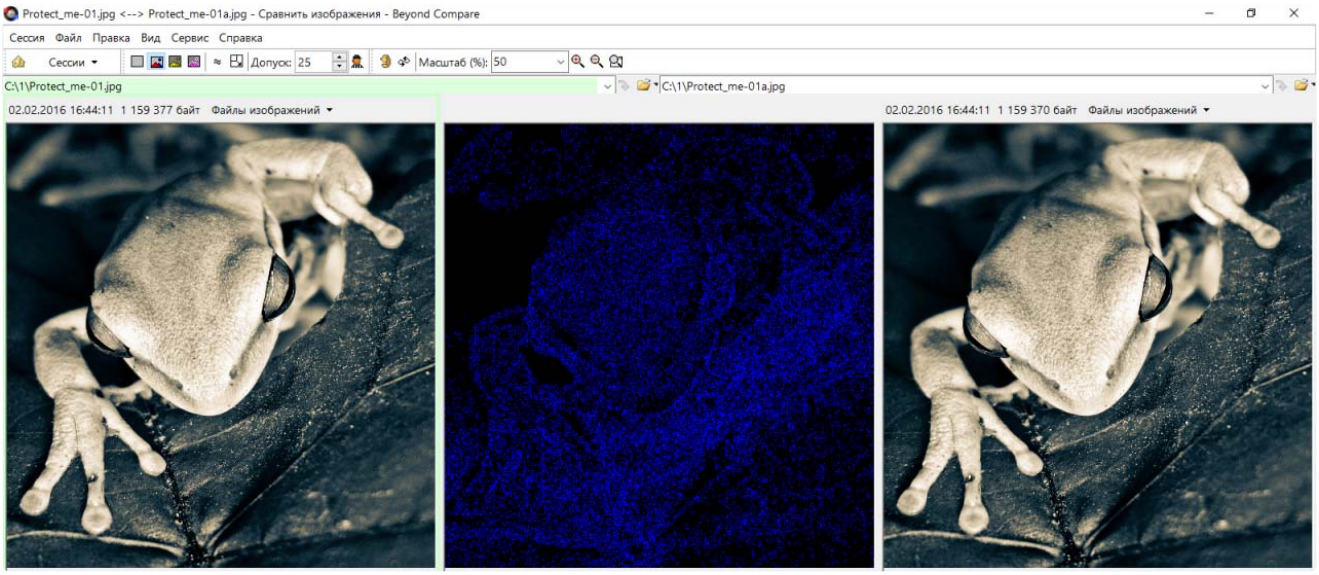
С одной стороны, это плюс: нет прямой корреляции между размером сообщения и числом измененных пикселей. Анализ такого контейнера существенно усложняется. С другой стороны, на создание контейнера приходится затратить дополнительные усилия, что может оттолкнуть неискушенного пользователя.

Другой режим работы программы — запись и чтение стегометок. Это скрытые строки длиной до 32 символов, которые можно использовать для защиты авторского права. Например, спрятать копирайт в фотографии, музыкальном файле или документе.

Работает эта функция исключительно просто. Пишешь произвольную стегометку в верхней части окна и указываешь ниже файлы, в которые ее надо добавить. Исходные файлы останутся нетронутыми, а их копии с меткой сохранятся в указанном тобой каталоге.

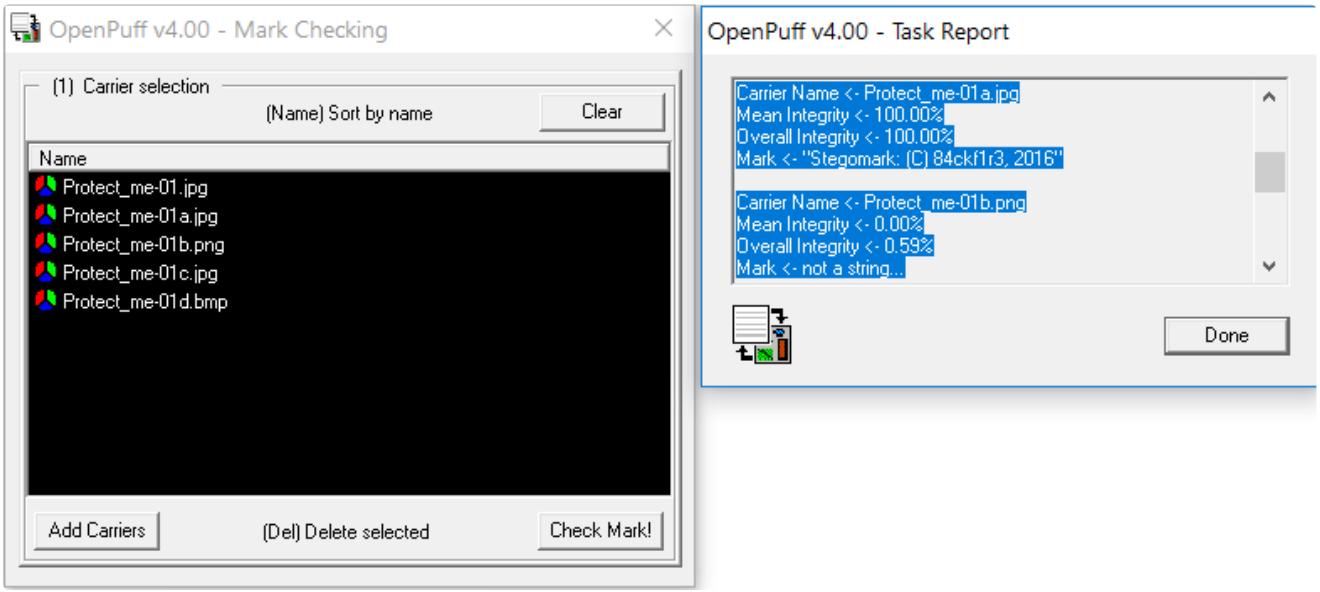


При возникновении любых правовых споров просто запускаешь OpenPuff и показываешь изумленному оппоненту ранее внедренную метку.



Даже лягушка может сказать, кто ее сфотографировал

Сложности возникают в том случае, если файл подвергался модификации. Даже простое конвертирование в другой формат стирает стегометку. Не удается ее считать и в том случае, если файл был снова приведен к исходному формату. Стойкие стегометки существуют, но внедрять их умеют только отдельные программы. Как правило, они привязаны к какому-то конкретному оборудованию (например, модели камеры).



Стегометка испарилась после конвертации файла