УДК 004.7 Вестник УрФО № 2(20) / 2016, с. 11–14

Куц Д. В., Третьяк Н. В.

ОСОБЕННОСТИ ВОССТАНОВЛЕНИЯ ДАННЫХ В ФАЙЛОВОЙ СИСТЕМЕ FAT 32

В данной статье рассматриваются вопросы восстановления удаленных файлов в файловой системе FAT 32. В силу особенностей файловой системы большинство утилит для восстановления данных не в полной мере справляются со своей задачей. Анализируются основные недостатки утилит и предлагается алгоритм восстановления, который при необходимости позволяет восстанавливать файлы вручную, а также в последующем может лечь в основу программы. Файловая система FAT 32 в течение еще некоторого времени будет сохранять актуальность, и как следствие, необходимо проводить исследования в области поиска оптимальных путей восстановления данных и создавать на их основе программные продукты. Разработанная методика может помочь в первую очередь для восстановления одиночных удаленных файлов.

Ключевые слова: *файловая система FAT 32, восстановление данных, сигнатурный анализ, байт, сектор, кластер, алгоритм.*

Kuts D. V.., Tretiak N. V.

CHARACTERISTICS OF DATA RECOVERY IN FAT 32 FILE SYSTEM

This article describes questions related to recovery of deleted files in FAT 32 file system. Most of utilities for data recovery cannot fully manage this kind of tasks because of characteristics of the file system. The second part of the article is dedicated to the analysis of the most common errors of the utilities and advises an effective algorithm of recovery, which allows to restore files manually if needed, and can lay the basis for a future program as well.

FAT 32 file system will remain actual for some time period, and therefore it's necessary to conduct researches aimed on exploration of the optimal ways of data recovery and create software products based on them. This technique, in the first place, can be of a great help in recovery of single deleted files.

Keywords: *FAT 32 file system, data recovery, signature analysis, bite, cluster, sector, algorithm.*

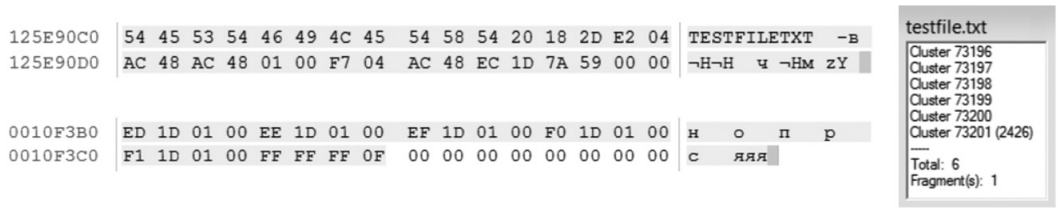

Рис. 1. Файловая запись в каталоге, цепочка кластеров (в таблице FAT и списком)

Файловая система FAT 32 через несколько месяцев отметит свое 20-летие, а семейству FAT в целом уже 40 лет. Однако, несмотря на столь солидный возраст, FAT 32 попрежнему актуальна и используется на многих носителях. Причин этому несколько. Самые основные – высокое быстродействие, простота реализации поддержки на аппаратных устройствах, низкое ресурсопотребление [1]. Однако быстродействие не бывает бесплатным. В этой файловой системе отсутствует журналируемость, поддержка прав доступа. Также жизненно важные структуры дублируются лишь отчасти.

Существует ряд серьезных проблем, связанных с восстановлением данных в этой файловой системе. Большинство проблем связаны с самой концепцией таблицы размещения файлов FAT. Она отслеживает свободное дисковое пространство и отображает размещение содержимого файлов на диске посредством цепочек кластеров [2]. Пример файловой записи и цепочки кластеров приведены на рис. 1.

При удалении файла цепочка кластеров обнуляется, т. е. содержимое всех ячеек таблицы, относящихся к данному файлу, заполняется нулями. После этого в случае, если файл был фрагментирован, можно только угадывать, в каких кластерах находится содержимое файла. С уверенностью можно сказать лишь о первом кластере файла, поскольку его номер хранится не в таблице FAT, а в файловой записи в каталоге и не стирается. Однако во многих случаях и эта уверенность оказывается обманчивой. Дело в том, что адрес первого кластера, в котором хранится содержимое файла, состоит из двух половинок по 2 байта. При удалении файла два старших байта адреса также заполняются нулями [2]. Пример файловой записи до удаления с выделенными старшими и младшими байтами, и файловой записи удаленного файла приведен ниже (см. рис. 2, 3).

В случае, если эти два байта адреса хранили нули, т. е. файл начинался в кластере с номером не более 65535, то адрес первого кластера при удалении не меняется. В этом случае информацию файловой записи можно использовать для восстановления файла, что существенно облегчает задачу. Однако при стандартном размере кластера в 4 Кб это бывает возможным только для файлов, хранившихся в первых 256 МБ памяти носителя. В остальных же случаях информация в файловой записи скорее сбивает с толку программу восстановления данных, нежели помогает ей.

Программы восстановления удаленных файлов с файловых систем FAT в основном используют два алгоритма. Это восстановление на основании данных файловой записи и поиск по сигнатурам некоторых типов файлов в области данных, а также комбинированные алгоритмы на основе первых двух методов. Однако во многих случаях полностью восстановить данные не удается. Наибольшей эффективностью пользуются комбинированные алгоритмы, однако данные, полученные из файловой записи в каталоге, не всегда бывают надежны. В большинстве случаев при удалении файла адрес первого кластера содержит неправильное значение. Т. е. начало файла потеряно. В этих случаях применим только

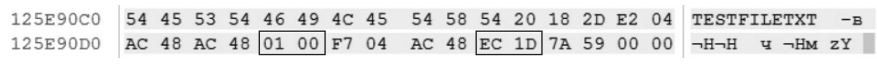

Рис. 2. Рамками выделены 2 старших байта по смещению 0x14 и 2 младших по смещению 0x1A

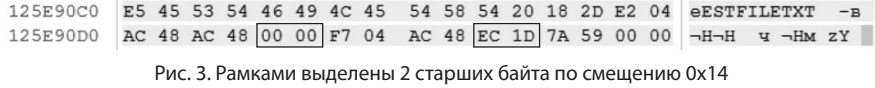

Рис. 3. Рамками выделены 2 старших байта по смещению 0x14
и 2 младших по смещению 0x1A – 2 старших байта обнулились.

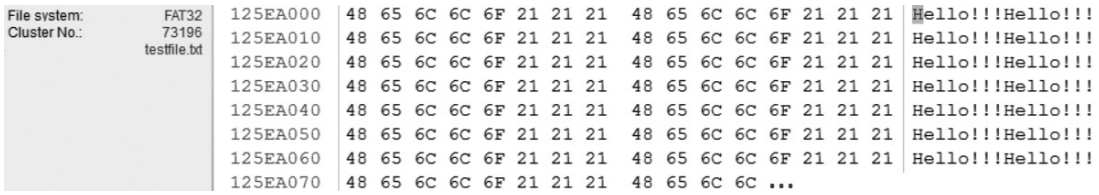

Рис. 4. Так выглядел файл до удаления и последующего восстановления

Рис. 5. Результат попытки восстановления программой

сигнатурный анализ. В некоторых случаях можно обойтись без информации в файловой записи [3]. Например для файлов *.tiff, *.zip, *.avi, *.psd, *.pst, *.jpeg, *.rar и некоторых других мы можем определить размер файла из заголовка. Но мы не можем восстановить название файла, временные отметки, не можем определить точный размер для других типов файлов.

Мы провели тестирование нескольких наиболее популярных утилит для восстановления удаленных файлов с искаженным адресом первого кластера.

Исходя из результатов тестов, мы убедились, что в случае, если при удалении файла обнуляются первые два байта адреса первого кластера, ни одна программа не восстанавливает файл (см. рис. 4, 5). Вместо этого она обращается в кластер области данных, на который указывает измененный адрес в файловой записи, и восстанавливает в файл содержимое этого кластера и последующие кластеры, исходя из размера файла. Т. е. в случае восстановления удаленных файлов по файловой записи программы не проверяют содержимое кластеров и восстанавливают содержимое, не имеющее никакого отношения к исходному. Однако все программы восстановили практически все файлы (нефрагментированные), где в файловых записях мы вручную восстановили адрес первого кластера после удаления.

Для восстановления файлов без использования данных файловой записи программы предлагают сигнатурный анализ, но он имеет ряд недостатков. Во-первых, используя покластерное сканирование логического диска, мы не используем файловую запись и теряем такие важные метаданные, как имя файла, размер файла. У многих типов файлов размер не прописан в заголовке, и определить, где заканчивается данный файл, довольно затруднительно. Кроме того, даже если файл восстановлен корректно, бывает довольно трудно пользователю найти его в огромном количестве восстановленных файлов.

Мы в своей работе предлагаем алгоритм (см. рис. 6) восстановления удаленных файлов, который легко реализовать программно, тем самым значительно повысив успех восстановления файлов, которые начинаются на определённую сигнатуру. Их список довольно обширен. Этот алгоритм будет удобен, если вы пытаетесь восстановить удаленный файл по данным файловой записи, которую вы обнаружили в каталоге. Программу, реализующую этот алгоритм, можно использовать как отдельный инструмент для поиска данных и правки адреса первого кластера в файловой записи или же встроить реализацию алгоритма в уже существующие средства восстановления данных, что нам кажется наиболее эффективным. Также возможно использовать этот алгоритм в «ручном» режиме, используя HEX-редактор.

1. Проверяем адрес первого кластера в файловой записи. Если в старших байтах нули, например, получается адрес 0x00001DEC, проверяем в таблице FAT свободен ли кластер. Если да, то обращаемся в кластер и проверяем наличие сигнатуры, соответствующей типу восстанавливаемого файла. Если она есть, пытаемся восстановить файл. Просим пользователя проверить, корректно ли восстановлен файл. Если нет, переходим в пункт 2.

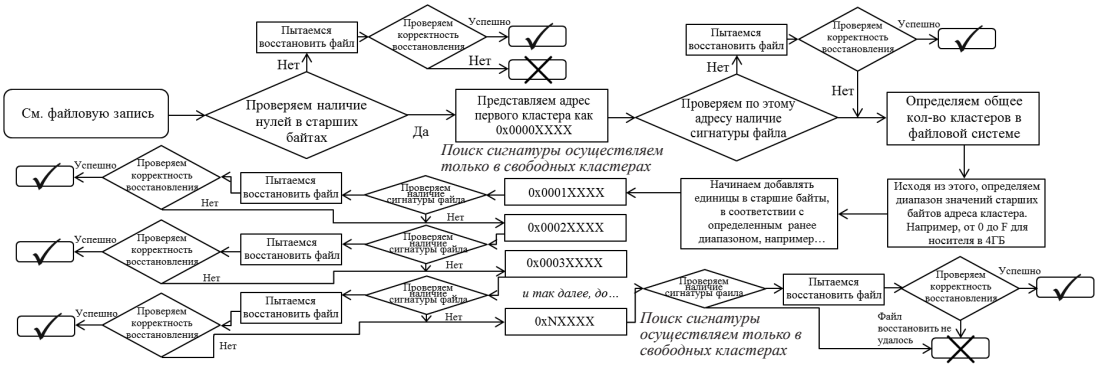

Рис. 8. Алгоритм восстановления удаленных данных

2. Определяем общее количество кластеров в файловой системе. Исходя из этого, определяем максимально возможное значение старших байтов адреса кластера. Например, для флэш-накопителя емкостью 4 ГБ с кластером 4 Кб получается количество кластеров примерно около 1 млн. В шестнадцатиричной системе это число 0xF4240. Т. е. в старших байтах д. б. число от 0 до F. Т. е. всего 16 возможных комбинаций. Для диска в 8 Гб это 32 комбинации, и пр.

3. Осуществляем поиск нужных данных, обращаясь последовательно в кластеры с адресами 0x00001DEC, 0x00002DEC, 0x00003DEC и пр. В случае обнаружения нужной сигнатуры, если кластер свободен, делаем предположение, что обнаруженные данные и есть содержимое файла, и восстанавливаем его на другой диск, затем продолжаем поиск. В результате у нас получается несколько вариантов восстановленного файла, один из которых, весьма вероятно, окажется тем самым.

Для дисков большого объема алгоритм не очень удобен, однако количество возможных вариантов восстановленного файла совсем не обязательно будет большим, т. к. наличие искомой сигнатуры в начале свободного кластера с нужным номером, но с чужими данными – вопрос случайности. Кроме того, мы отсеиваем кластеры, помеченные как занятые в файловой системе.

Примечания

1. Мюллер Скотт. Модернизация и ремонт ПК. 19-е изд. М. : Вильямс, 2011. 1072 с.
2. Кэрриэ Брайан. Криминалистический анализ файловых систем. СПб. : Питер, 2007. 480 с.
3. Леонов В. Восстановление данных. М. : Эксмо, 2009.304 с.

Куц Дмитрий Владимирович, старший преподаватель кафедры теоретических основ радиотехники Института радиоэлектроники и информационных технологий – РтФ, Уральский федеральный университет имени первого Президента России Б. Н. Ельцина. 620002, г. Екатеринбург, ул. Мира, 19. E-mail: d.v.kutc@urfu.ru

Третьяк Наталия Вадимовна, магистрант кафедры управления общественными отношениями Института государственного управления и предпринимательства, Уральский федеральный университет имени первого Президента России Б. Н. Ельцина. 620002, г. Екатеринбург, ул. Мира, 19. E-mail: n.v.tretiak@urfu.ru

Kuts Dmitry Vladimirovich, seniorteacherof the "Basic Theory of Radio Engineering" department, Institute of Radioelectronics and Information Technologies, Ural Federal University named after the first President of Russia B.N.Yeltsin. 620002, Sverdlovsk region, Ekaterinburg, Mira street, 19. E-mail: d.v.kutc@urfu.ru

Tretyak Nataliya Vadimovna, master of the "Public Administration" department, Institute of Public Administration and Entrepreneurship, Ural Federal University named after the first President of Russia B.N.Yeltsin. 620002, Sverdlovsk region, Ekaterinburg, Mira street, 19. E-mail: n.v.tretiak@urfu.ru