**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ**

**Федеральное государственное автономное учреждение высшего**

**образования**

**«Омский государственный технический университет»**

Радиотехнический факультет

Кафедра комплексной защиты информации

**Отчет по лабораторной работе №2**

по дисциплине:

«Инструментальное обеспечение компьютерно-технической экспертизы»

на тему:

**«Анализ таблицы файлов»**

Выполнил студент

группы БИТ-191:

Радьков В.П.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

Принял

Старший преподаватель

Трапезников Е. В.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

Омск, 2023

Оглавление

[Подготовка материалов 3](#_Toc132717449)

[Защитный MBR 3](#_Toc132717450)

[Разметка GPT 5](#_Toc132717451)

[Первичный заголовок GPT 5](#_Toc132717452)

[Таблицы разделов диска GPT 7](#_Toc132717453)

[Раздел 9](#_Toc132717454)

[Запись файловой системы 9](#_Toc132717455)

[Записи MFT 11](#_Toc132717456)

[Структура диска 16](#_Toc132717457)

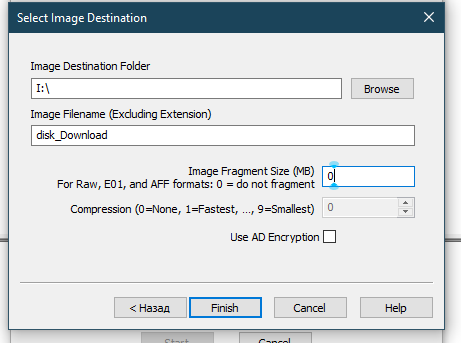
[Программы для просмотра таблице MFT 17](#_Toc132717458)

[Вывод 22](#_Toc132717459)

[Список источников 22](#_Toc132717460)

# Подготовка материалов

С помощью программы FTKIMager был взять дамп диска первых 8 г.



Иллюстранция – получение дампа диска

# Защитный MBR

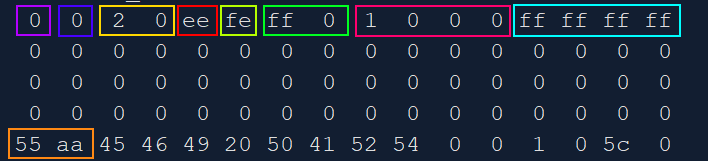
MBR всегда находится в первом секторе жесткого диска. Первая секция длиной в 440 байт содержит исполняемый код, который при загрузке компьютера, BIOS считывает с диска в память и передаёт ему управление. Следующие 4 байта являются идентификатором диска. Следующие 2 байта не используются.

Далее идёт блок таблицы разделов диска 64 байта по 16 байт на каждый раздел диска (не более 4-х разделов). В конце располагается сигнатура (55AA). Важная информация из данных структур является:

* Код типа раздела по смещению 04h поскольку с помощью него можно определить разметку диска.
* Адрес начального сектора раздела. Как правило здесь должен быть указан адрес LBA-1, что соответствует смещению в 512 байт.

Адресация CHS[[1]](#footnote-1) не используется в современных дисках с разметкой GPT. Современные накопители используют Разметку GPT и адресацию LBA[[2]](#footnote-2). Но на первом секторе диска так же располагается MBR-сектор, это было сделано в целях совместимости со старым ПО.

Как видно из иллюстрации 1 в таблице разделов данного сектора находиться единственная запись с типом раздела EE. Раздел должен занимать всё место на жёстком диске и иметь размер 0xFFFFFFFF. Данный раздел помечается как неактивный. При работе компьютера с UEFI, данный MBR-сектор просто игнорируется и никакой код в нём также не выполняется. В секторе MBR и в заголовке GPT используется LBA адресация. Таким образом первая запись MBR располагается по адресу LBA-0.



Иллюстранция 2 – Структура описательного раздела MBR.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | – | Признак активного раздела. 0 – неактивный, 80h – активный. |
|  |  | – | Номер головки диска, с которой начинается раздел. |
|  |  | – | Номер цилиндра с сектора, с которых начинается раздел. |
|  |  | – | Код типа раздела. |
|  |  | – | Номер головки диска, на которой заканчивается раздел. |
|  |  | – | Номер цилиндра и сектора которыми заканчивается раздел. |
|  |  | – | Абсолютный номер начального сектора раздела. Адресация LBA. |
|  |  | – | Число секторов в разделе LBA. |
|  |  | – | Сигнатура конца записи 55AA. |

Если диск имеет разметку GPT то никакой интересной и полезной информации из описательный записей извлечь нельзя.

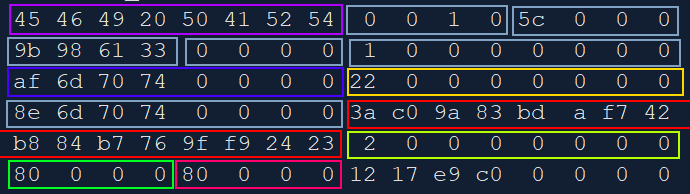
# Разметка GPT

## Первичный заголовок GPT

Как правило заголовок GPT начинается по адресу LBA-1. Заголовок должен начинаться с сигнатуры «EFI PART». Сам заголовок имеет размер 92 байта. Все адреса записаны в формате LBA. Из заголовка GPT можно получить следующую полезную информацию:

* Размер заголовка GPT по смещению 0x0C.
* Адрес первого раздела диска по смещению 0x28.
* Адрес начала таблицы разделов (Должно быть LBA-3) по смещению 0x48.
* Максимальное число разделов, которое может содержать таблица (Должно быть 128) по смещению 0x50.
* Размер записи для раздела (Должно быть 128) по смещению 0x54.
* GUI диска.
* Размер записи GPT и их количество. По умолчанию 128 записей по 128 байт.

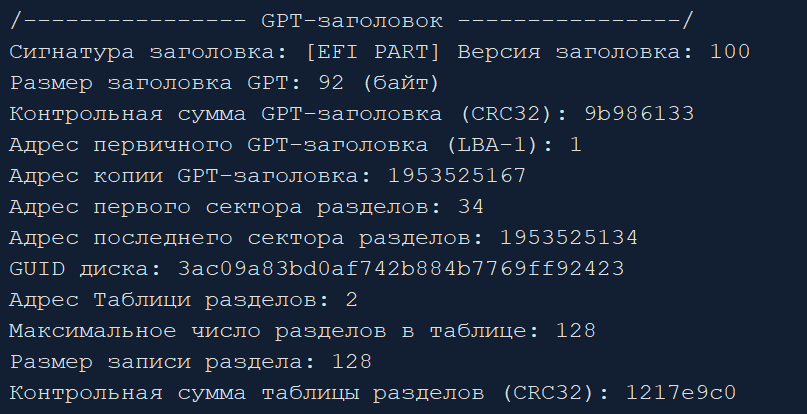
Определив адрес таблицы разделов GPT, можно определить имеющиеся разделы на накопителе и определить смещение до начала каждого из разделов.



Иллюстранция 3 – Структура первичного заголовка GPT

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | – | Сигнатура заголовка. Должна иметь значение «EFI PART» или «45-46-49-20-50-41-52 54». |
|  |
|  |  | – | Адрес сектора, на котором находится копия GPT заголовка. Должен иметь адрес последнего сектора на диске. |
|  |
|  |  | – | Адрес первого раздела диска. |
|  |  | – | GUID диска. |
|  |  | – | Адрес начала таблицы разделов. |
|  |  | – | Максимальное число разделов в таблице. |
|  |  | – | Размер записи раздела. |

Информация, которую удалось извлечь из заголовка GPT.



Иллюстранция 4 – GPT заголовок диска.

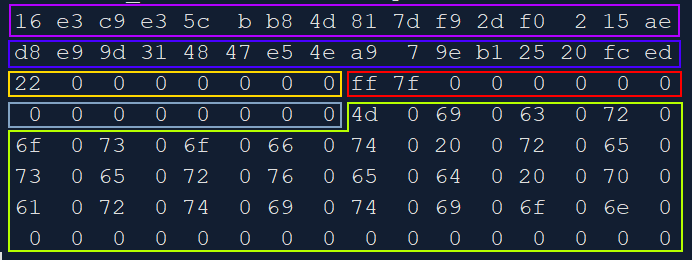
## Таблицы разделов диска GPT

Следующей частью структуры GPT является таблица разделов. В настоящее время операционная система Windows использует следующий формат таблицы разделов — максимум 128 разделов, на каждую запись раздела выделяется по 128 байт, соответственно вся таблица разделов займет 128\*128=16384 байт, или 32 сектора диска.

Из записи разделов диска можно получить следующую информацию:

* Начальный LBA адрес раздела по смещению 0x20.
* Конечный LBA адрес раздела по смещению 0x28.
* Название раздела по смещению 0x38 имеет длину 72 байта в форме Unicode строки.

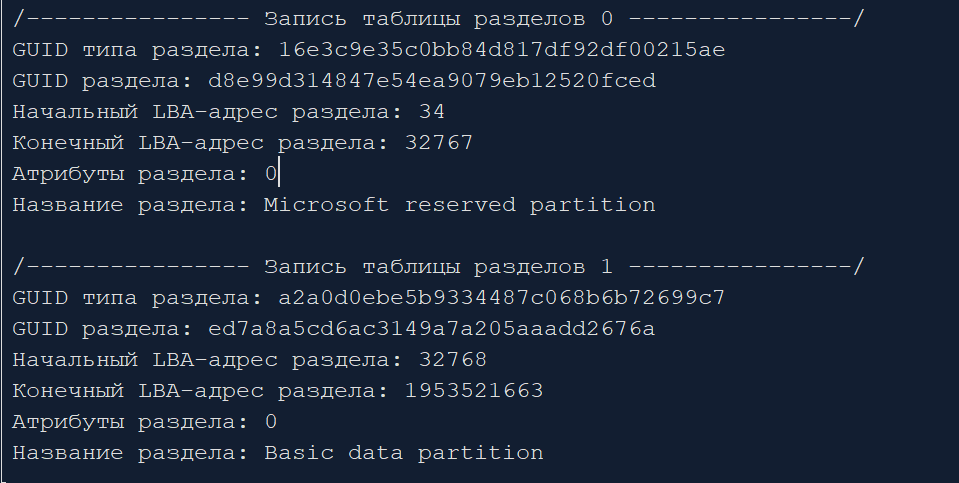
Определив начальный и конечный адрес LBA можно вычислить объём, занимаемый разделом диска. Так же начальный адрес LBA позволит перейти непосредственно к самому разделу диска.



Иллюстранция 5 – Структура записи таблицы разделов.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | – | GUID типа раздела. |
|  |  | – | Уникальный GUID раздела. Генерируется при создании зардела. |
|  |  | – | Начальный LBA адрес раздела. |
|  |  | – | Последний LBA адрес раздела. |
|  |  | – | Название раздела хранится в виде Unicode строки длиной 36 символов. Всего 72 байта. |

На анализируемом диске находятся два тома GPT. Один из которых является резервной областью для использования операционной системой «Microsoft reserved partition - MSR». А другой был создан пользователем и содержит пользовательские файлы «Basic data partition». На данном диске раздел MSR заполнен нулями.



Иллюстранция 6 – Разделы диска.

# Раздел

## Запись файловой системы

Каждый раздел начинается с заголовка файловой системы размером 50 байта. Данный заголовок можно условно поделить на две части: базовый заголовок и расширенный. В базовый заголовок входит общая информация: размер сектора, количество секторов в кластере, количество секторов на дорожке, адрес смещения (кол. секторов) которое нужно прибавить к логическому адресу. Адресация начинается с начала раздела, а не с начала диска. Если раздел размечен как NTFS, то об этом можно узнать из идентификатора OEM. В зависимости от типа разметки будет отличаться расширенная запись заголовка.

|  |  |
| --- | --- |
|  |  |

Иллюстранция 7 – Базовая и расширенная записи раздела Basic data partition «».

В заголовке NTFS можно получить следующую важную информацию:

* Адрес первого кластера MFT.
* Адрес копии первых 4-х записей MFT.
* Размер записи MFT (должно быть 1024).

Адреса записей MFT позволят перейти по смещению к нужной записи. Количество байт, которое нужно прибавить для получения смещения относительно начала диска в нём указано количество секторов, расположенных перед этим разделом.

Копия MFT — это только один кластер, 4 первых записи: $MFT, $MFTMirr, $LogFile, $Volume.

Как видно кластер, в котором находится главная таблица MFT расположен по смещению . Получается.

Что главная запись MFT расположена примерно на 3 Гб тома. Поэтому пришлось брать образ диска в 8 Гб, чтобы захватить главную запись MFT.

## Записи MFT

Каждая MFT запись имеет размер ровно 1 КБ = 1024. Первые 42 байта в заголовке имеют фиксированную структуру. Далее идёт запись массива корректировок размер которого определяется по смещению 0x14 и по идеи может быть разным.

Далее расположены записи атрибутов файла, количество и размер которых может быть разным. Первые 42 байта+массив корректировки хранят заголовок. Заголовок содержит 12 полей. Остальные 982 байта не имеют фиксированной структуры и используются для хранения атрибутов.

В конце записи MFT располагается запись массива корректировок. Она нужна для того, чтобы система смогла обнаружить сбои при записи на накопитель (на иллюстрации массив корректировки помечен зелёным). Запись MFT может быть нарушена, если в процессе записи накопитель был обесточен.

Первые 16 записей называются метафайлами и недоступны системе. Причём первая запись является самой таблицей MFT.

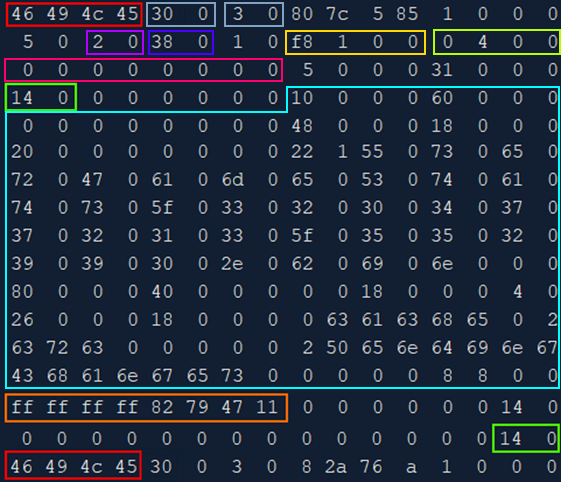
Структурно любой атрибут состоит из атрибутного заголовка и тела атрибута. Заголовок атрибута всегда хранится в файловой записи, расположенной внутри MFT. Атрибуты можно разделить на две большие категории: резидентные и нерезидентные.

1. Нерезидентные атрибуты хранят своё тело вне MFT водном или нескольких кластерах и соответственно не ограничены по объёму. Сектора перечислены в заголовке атрибута в специальном списке.
2. Резидентные атрибуты хранятся непосредственно в MFT.

Первые 16 байт заголовка атрибута одинаковы следующие байты отличаются в зависимости от категории атрибута.

Форма представления у резидентного и нерезидентного атрибута отличаются. Каждый тип атрибута так же различается формой представления. Для получения информации из каждого типа атрибута нужно знать его структура. Первые четыре байта атрибутного заголовка определяют его тип. Тип атрибута, в свою очередь, определяет формат представления тела атрибута.

Последовательность всех атрибутов завершается последовательностью 8 байт. Всё оставшееся пространство в записи MFT заполняется нулями и не используется.



Иллюстранция 8 – Структура записи MFT (сокращённая)

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | – | Счётчик жёстких ссылок. |
|  |  | – | Смещение начала списка атрибутов. |
|  |  | – | Реальный размер записи MFT. |
|  |  | – | Сигнатура должно быть FILE. |
|  |  | – | Выделенный размер записи MFT. |
|  |  | – | Массив последовательности обновлений «update sequance array». |
|  |  | – | Ссылка на базовую запись MFT. |
|  |  | – | Сигнатура конца тела атрибута ffffffff82794711. |

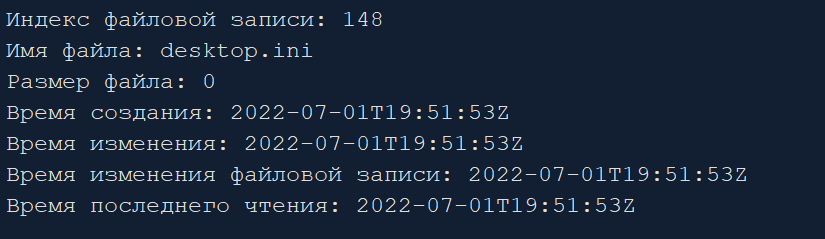
Файловая система NTFS имеет большое число атрибутов, но наиболее информативные из них это: $FileName, $Standart\_Information. Выделенный размер — это область памяти, занимаемая записью MFT, должна быть 1024, а реальный размер — это область, занимаемая заголовком записи MFT и всеми атрибутами. Часть записи MFT остаётся заполненной нулями, если у файлов мало атрибутов.

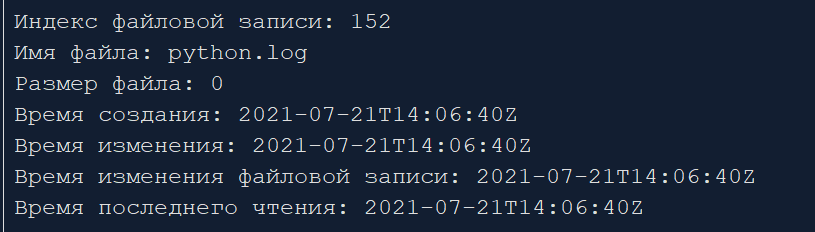
|  |  |
| --- | --- |
| Иллюстранция 9 – Заголовок записи MFT. | Иллюстранция 10 – Содержимое атрибута «$FileName». |

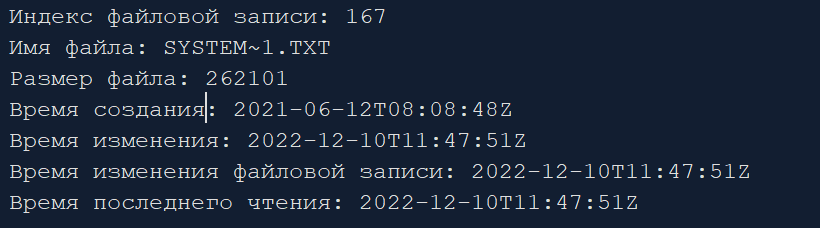
Таблица 1 – Первые 25 записей MFT

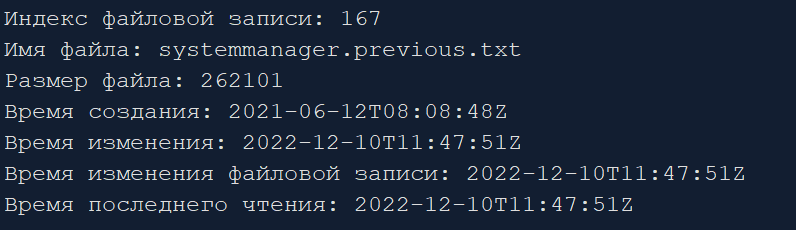
|  |  |  |
| --- | --- | --- |
| **Номер** | **Название** | **Назначение** |
|  | **0-11** | |
| 0 | $MFT | Главная файловая таблица (Master File Table, MFT). |
| 1 | $MFTMirr | Резервная копия первых четырех элементов MFT. |
| 2 | $LogFile | Журнал транзакций (transactional logging file). |
| 3 | $Volume | Серийный номер, время создания, dirty flag (флаг не сброшенного кэша) тома. |
| 4 | $AttrDef | Определение атрибутов. |
| 5 | . | Корневой каталог (root directory) тома. |
| 6 | $Bitmap | Карта свободного/занятого пространства. |
| 7 | $Boot | Загрузочная запись (boot record) тома. |
| 8 | $BadClus | Список плохих кластеров (bad clusters) тома. |
| 9 | $Quota | Информация о квотах (quota information). |
| 10 | $Secure |  |
| 11 | $UpCase | Таблица заглавных символов (uppercase characters) для трансляции имен. |
| 12 | $Extend | Каталоги: $ObjId, $Quota, $Reparse, $UsnJrnl. |
|  | 13-23 – Зарезервировано для записи расширений $MFT | |
| 24 | Файлы поьзователя. | |

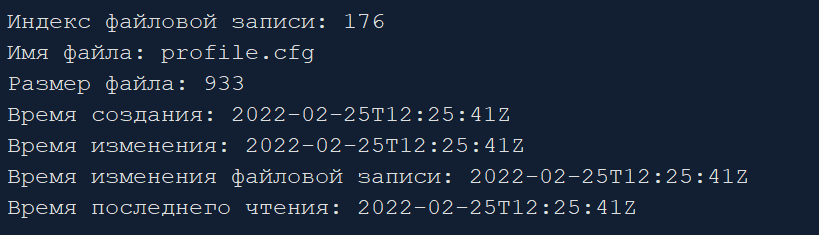
Удалось получить информацию о следующих файлах:

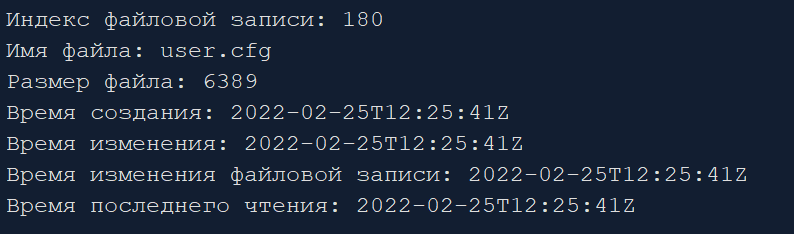


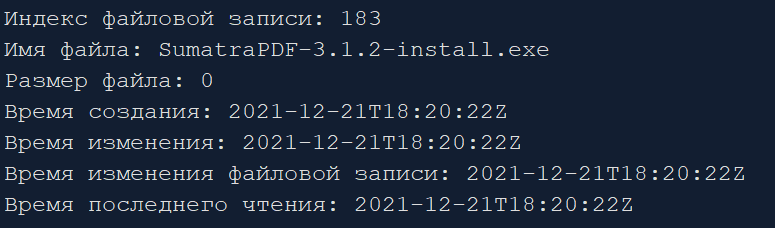


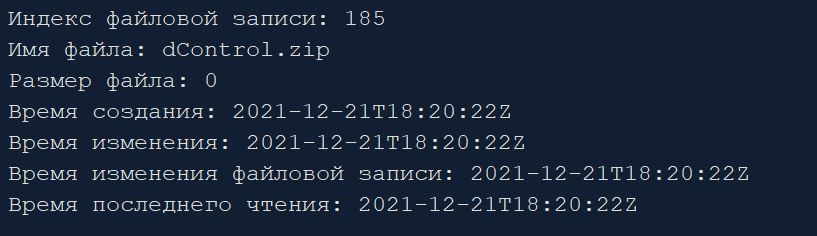


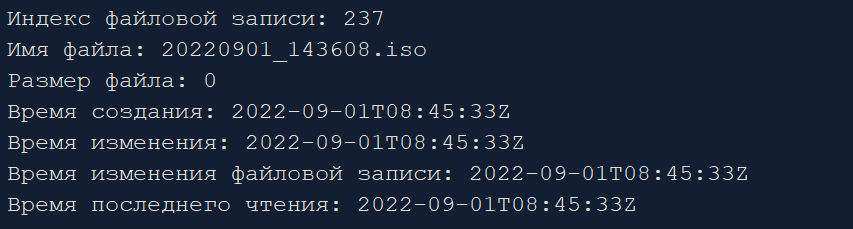






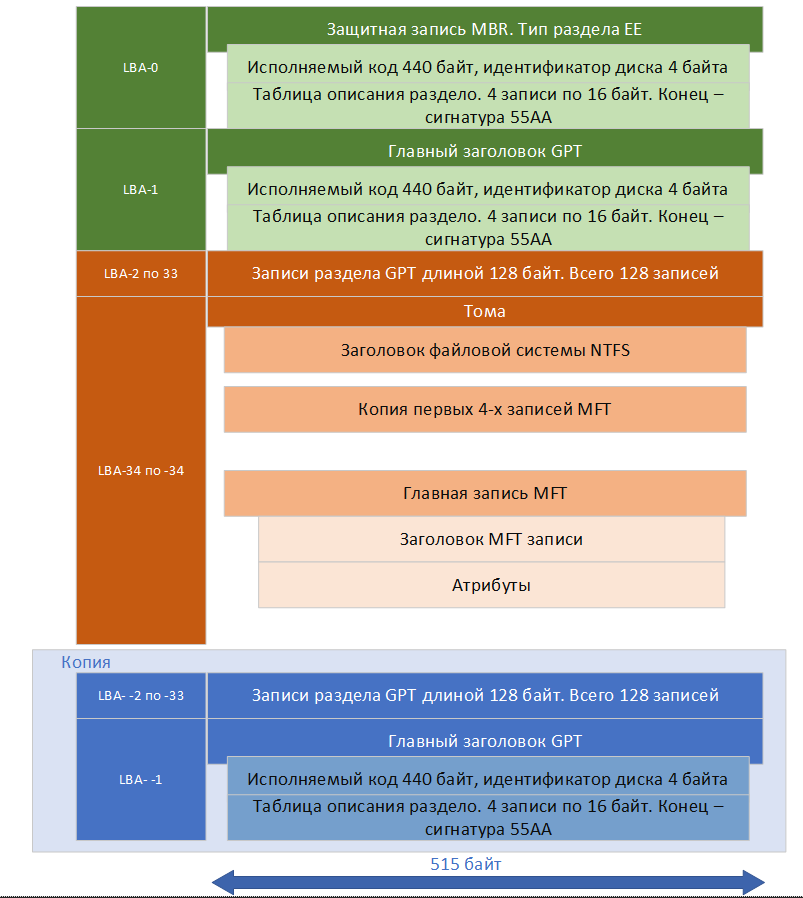






# Структура диска

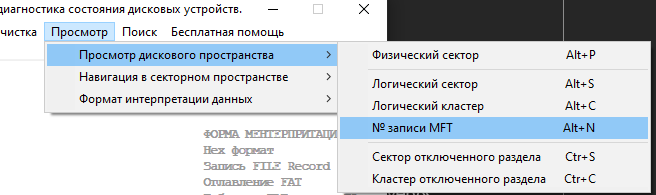
Подводя итог проделанной работе, можно построить структуру жёсткого диска с разметкой GPT и файловой системой NTFS.



Иллюстранция 11 – Структура разметки жёсткого диска

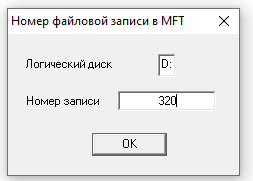
# Программы для просмотра таблице MFT

Таблицу MFT можно просмотреть с помощью программы «Fomsoft Data Recovery». Просмотреть запись таблицы MFT можно комбинацией горячих клавиш Alt + С или вызовом подпункта «№ записи MFT» в меню «Просмотр».

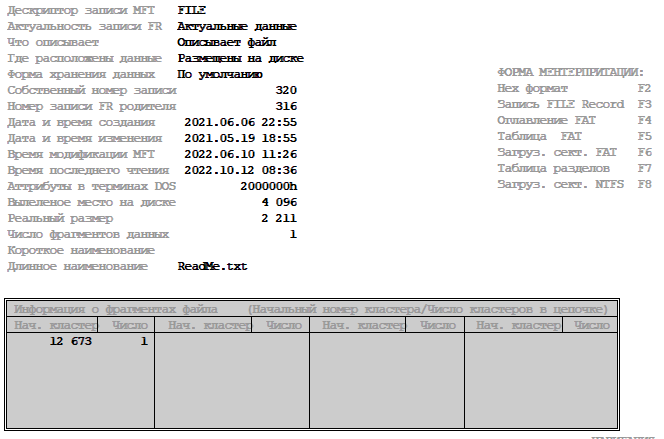


Иллюстранция 12 – Пункт меню «№ записи MFT»

Далее нужно выбрать логический раздел и номер файловой записи MFT.

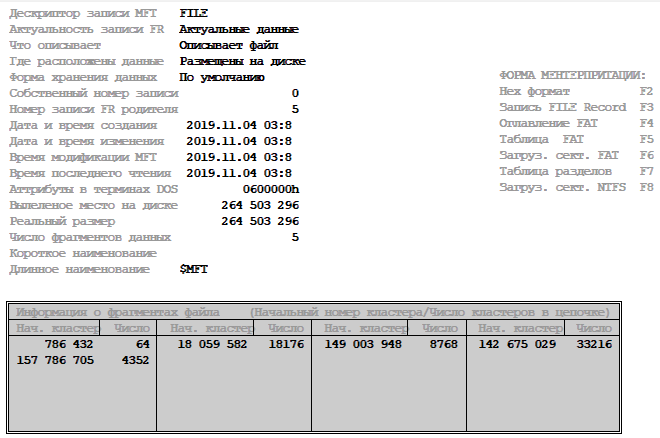


Иллюстранция 13 – Окно выбора фазовой записи



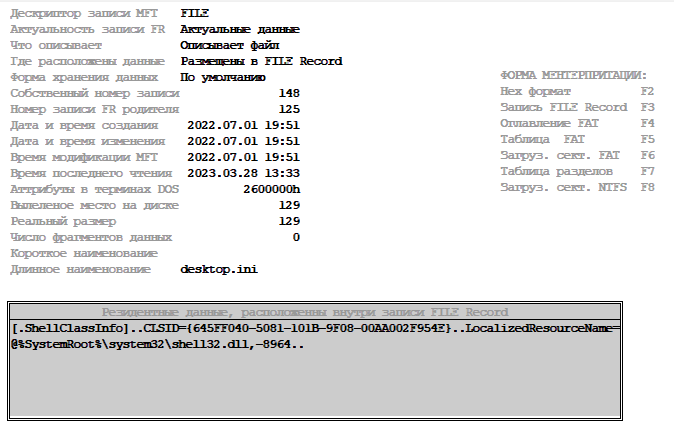
Иллюстранция 14 – Файловая запись 320

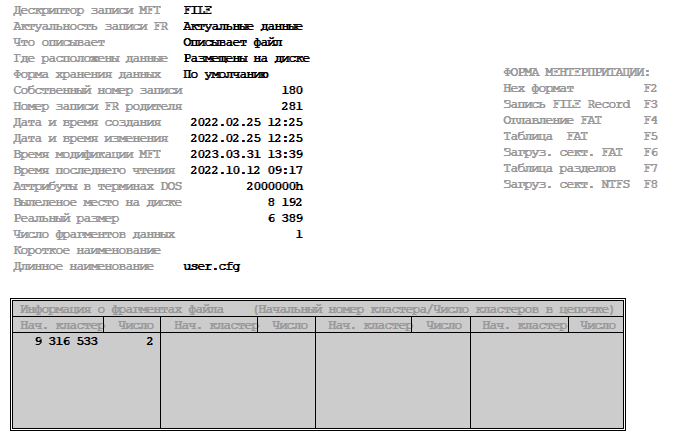
Просмотрим запись $MFT.



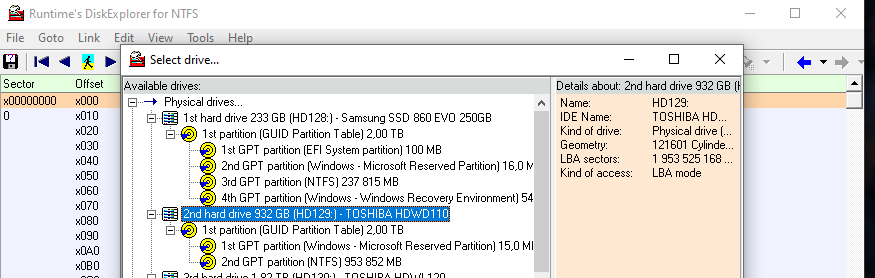
Иллюстранция 15 – Запись MFT

С помощью стрелок можно переключаться по файловым записям.

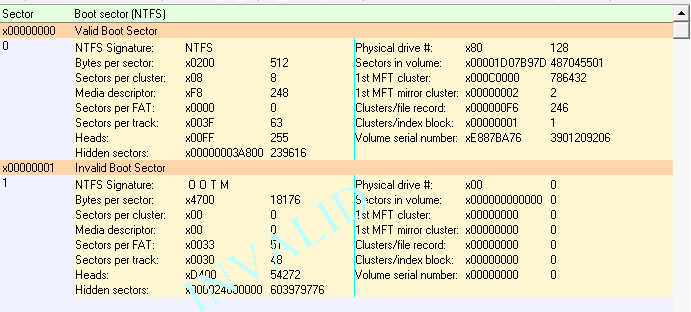




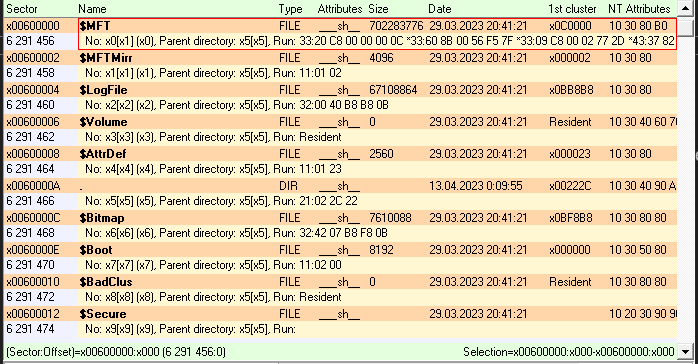
Можно воспользоваться «DiskExplorer». Сначала нужно выбрать накопитель. В разделе file->disk нужно выбрать диск.

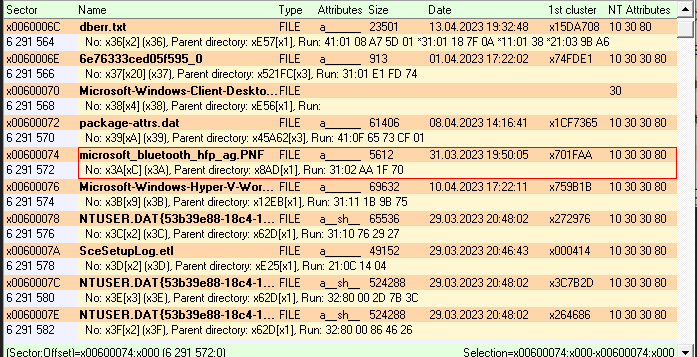


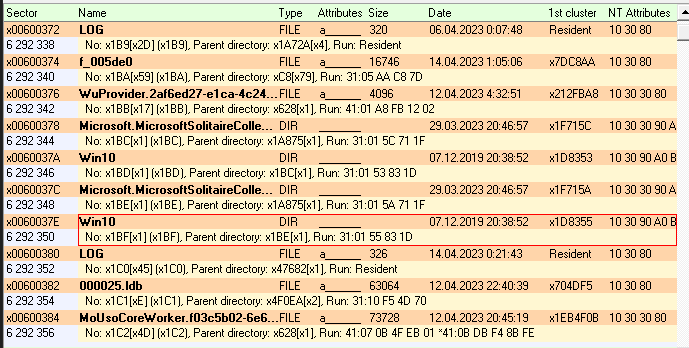
Затем нужно выбрать логический раздел, после чего появится подборная информация о диске.



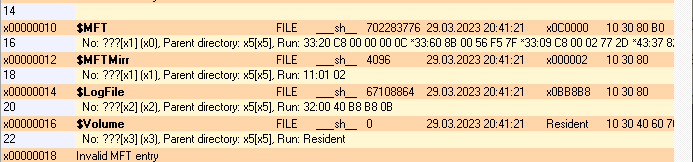
Через пункт манию «GoTo-> MFT» был выполнен переход к MFT таблице.







Копия первых записей MFT.



# Вывод

В результате проделанной работы можно сделать следующий вывод, данные из таблице MFT позволят определить какие файлы были на жёстком диске даже если пользователь их удалил. При обычном удалении файла данные о нём не удаляются. Файловая запись MFT помечается как свадебная и будут перезаписана при первой необходимости. Таким образом экономятся ресурсы системы, поскольку на перезапись или удаление данных требуется время.

# Список источников

1. <https://samag.ru/archive/article/395>
2. <https://konyakov.ru/pubs/books/kris-kaspersky-r_i_p/kris-kaspersky-20.pdf>
3. <https://frolov-lib.ru/datarecovery/articles/ntfs_recovery/index.html>
4. <https://dubeyko.com/development/FileSystems/NTFS/ntfsdoc.pdf>
5. <https://codywu2010.wordpress.com/2010/10/02/bios-parameter-block/>

1. **CHR (Cylinder Head Sector)** – каждый сектор адресуется по трём координатам:

   Cylinder – номер цилиндра (начинается с 0).

   Head – номер головки (начинается с 0).

   Sector – номер сектора (начинается с 1). [↑](#footnote-ref-1)
2. **LBA (Logical Block Address)** – каждый сектор адресуется одним значением. Данный способ адресации не зависит от внутренней физической архитектуре накопителя. Отрицательные значения отсчитываются от конца диска. Единицей измерения является блок в 512 байт. [↑](#footnote-ref-2)