БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Факультет информационных технологий

и робототехники

Кафедра «Программное обеспечение вычислительной техники и автоматизированных систем»

О Т Ч Ё Т

о лабораторной работе № 3

Дисциплина

«Системное программирование»

Тема

«Файловая система NTFS»

Выполнил: студент гр. 10702217 Храмков Д. С.

Проверил: Разорёнов Н. А.

Минск 2019

***Лабораторная работа № 3***

**ФАЙЛОВАЯ СИСТЕМА NTFS**

**Цель работы:** ознакомление со структурой и расположением системной информации и данных на томе NTFS.

**Изучаемые вопросы**

1. Структура BOOT.
2. Таблица файлов MFT.
3. Записи главной таблицы файлов (FILE RECORD).
4. Метафайлы.
5. Структура файла (небольшого и большого).
6. Аттрибуты файла.
7. Каталоги в NTFS (структура).
8. Изучить алгоритм поиска расположения файлов на диске:
   1. Определение номера начального кластера расположения файла на диске;
   2. Рассчёт номеров кластеров файла на диске;
9. Исследовать изменение элементов системных файлов NTFS при выполнении команд COPY, MOVE, DEL, RENAME.

***Постановка задачи***

Разработать программу, которая выводит на экран:

1. Дамп 1-го сектора BOOT и расшифровывает структуру METADATABOOT.
2. Список атрибутов короткого файла и тело атрибута $DATA. Файл создать любым текстовым редактором в кодировке ANSI, его содержание – две строки: первая – ФИО студента, вторая – дата его рождения.
3. **Структура BOOT.**

BOOT является загрузочной записью тома под NTFS, которая содержит основную информацию о нём (расположение MFT, количество секторов на кластер, всего секторов на томе, код загрузчика, ...).

typedef struct fileBoot {

BYTE dJump[3];

BYTE dSystemId[8];

WORD dBytesPerSector;

BYTE dSectorPerCuster;

BYTE dUnusedA[5];

BYTE dMediaD;

BYTE dUnuseB[2];

WORD dSectorPerTrack;

BYTE dUnuseC[8];

BYTE dUsualy[4];

INT64 dNumberOSectors;

INT64 dLCNofMFT;

INT64 dLCNofMFTMirr;

DWORD ClusterPerMFT;

DWORD ClusterPerIndexes;

BYTE dSerialNumber[8];

BYTE dDataCoder[432];

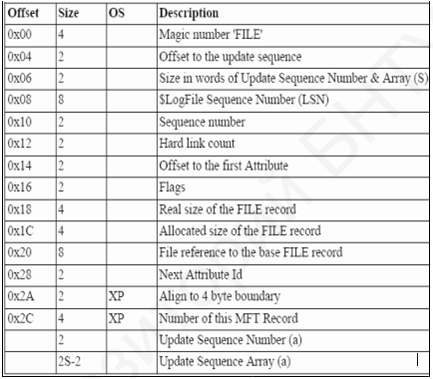
};

1. **Таблица файлов MFT.**

MFT – Master File Table. Каждый файл в томе представлен записью в этой таблице. Первые 16 записей NTFS резервирует для специальной информации, характеризующей этот том. Первая запись таблицы $MFT описывает непосредственно главную файловую таблицу. За ней следует зеркальная запись под именем $MFTMirr, которая содержит зеркальное отражение файла $MFT на случай необходимости восстановления утерянной информации из него. Также существуют такие записи как $Boot (загрузчик), $Bitmap (карта свободного места тома), $Secure (дескрипторы безопасности файловых объектов), $UpCase (таблица, преобразующая символы нижнего регистра в соответствующие символы в верхнем регистре), $Volume (служебная информация), $AttrDef (список стандартных атрибутов файлов на томе), $BadClus (список повреждённых кластеров тома) и $LogFile (журнал файловой системы)

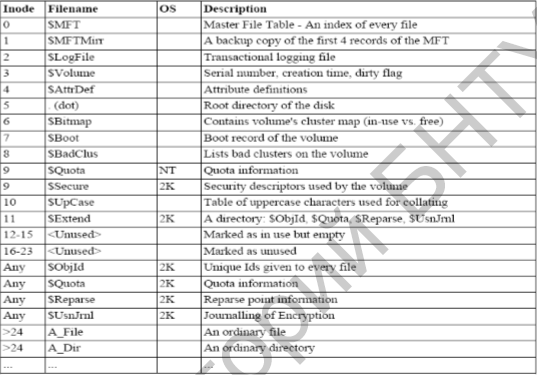
1. **Записи главной таблицы файлов (FILE RECORD).**

В NTFS файл является набором атрибутов, которые представлены в виде потока байтов. Каждый атрибут имеет заголовок, структура которого зависит от того, резидентный атрибут или нет, а также имеет ли он имя. Каждый заголовок атрибута, как и сам атрибут, имеет определённую структуру.



1. **Метафайлы.**

Метафайлами называются файлы, содержащие информацию о других файлах/записях. Первые 16 файлов NTFS являются метафайлами.



1. **Структура файла (небольшого и большого).**

*Небольшой файл.* Если файл имеет небольшой размер, то он может целиком располагаться внутри одной записи MFT, имеющей, например, размер 2 кб. Небольшие файлы NTFS состоят по крайней мере из следующих атрибутов:

* Стандартная информация (STANDART\_INFORMATION);
* Имя файла (FILE\_NAME);
* Данные (DATA);
* Дескриптор безопасности (SECURITY\_DESCRIPTOR);

Из-за того, что файл может иметь переменное количество атрибутов, а также из-за переменного размера атрибутом нельз наверняка утверждать, что файл уместится внутри записи. Однако обычно файлы размером менее 1500 байт помещаются внутри записи MFT размером 2 кб.

*Большой файл.* Если данные файла не помщаются в одну запись MFT, то этот факт отражается в заголовке атрибута DATA, который содержит признак того, что этот атрибут является нерезидентным, т.е. находится в отрезках вне таблицы MFT. В этом случае атрибут DATA содержит адресную информацию каждого отрезка данных.

1. **Атрибуты файла.**

Атрибуты файла являются основой структуры любого файла. Такие элементы, как имя файла, информация защиты и даже данные – это всё атрибуты файла. Каждый атрибут имеет код типа атрибута, по которому он идентифицируется. Если атрибут достаточно велик, то он помещается в отдельном файле.

typedef struct StandartInformation {

FILETIME FileCreated;

FILETIME FileModifider;

FILETIME RecordChanged;

FILETIME LastAccess;

DWORD FilePremissions;

DWORD MaxNumberOfVersion;

DWORD VersionNumber;

DWORD ClassId;

DWORD OwnewId;

DWORD SecurityId;

INT64 QuotaCharged;

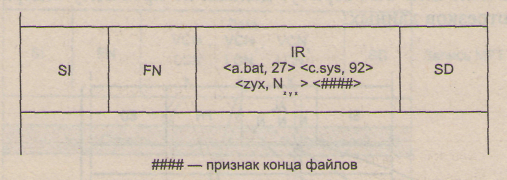
INT64 UpdateSequenceNumber;

};

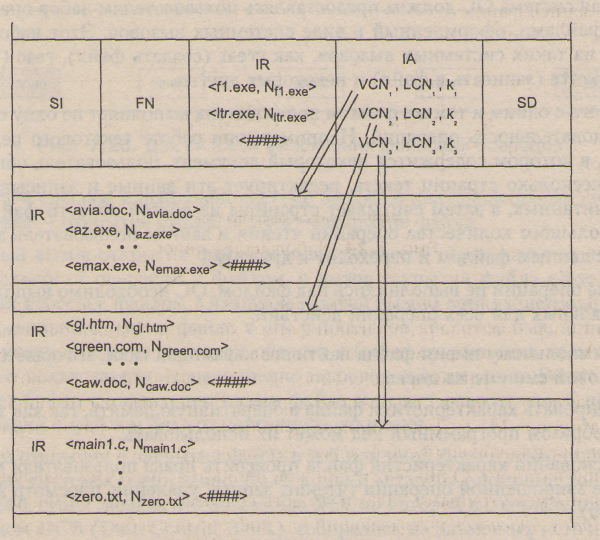
1. **Каталоги в NTFS (структура).**

Каждый каталог NTFS представляет собой один вход в таблицу MFT, который содержит атрибут Index Root. Индекс содержит список файлов, входящих в каталог. Индексы позволяюют сортировать файлы для ускорения поиска, основанного на значении определённого атрибута. Обычно в файловых системах файлы сортируются по имени.

Структура небольших каталогов:



Структура больших каталогов:



1. **Изучить алгоритм поиска расположения файлов на диске:**
   1. **Определение номера начального кластера расположения файла на диске;**

Дл определения номера начального кластера расположения файла на диске нужно в атрибуте $DATA по смещению “Data runs offset” определить начало записи Data run.

* 1. **Расчёт номеров кластеров файла на диске;**

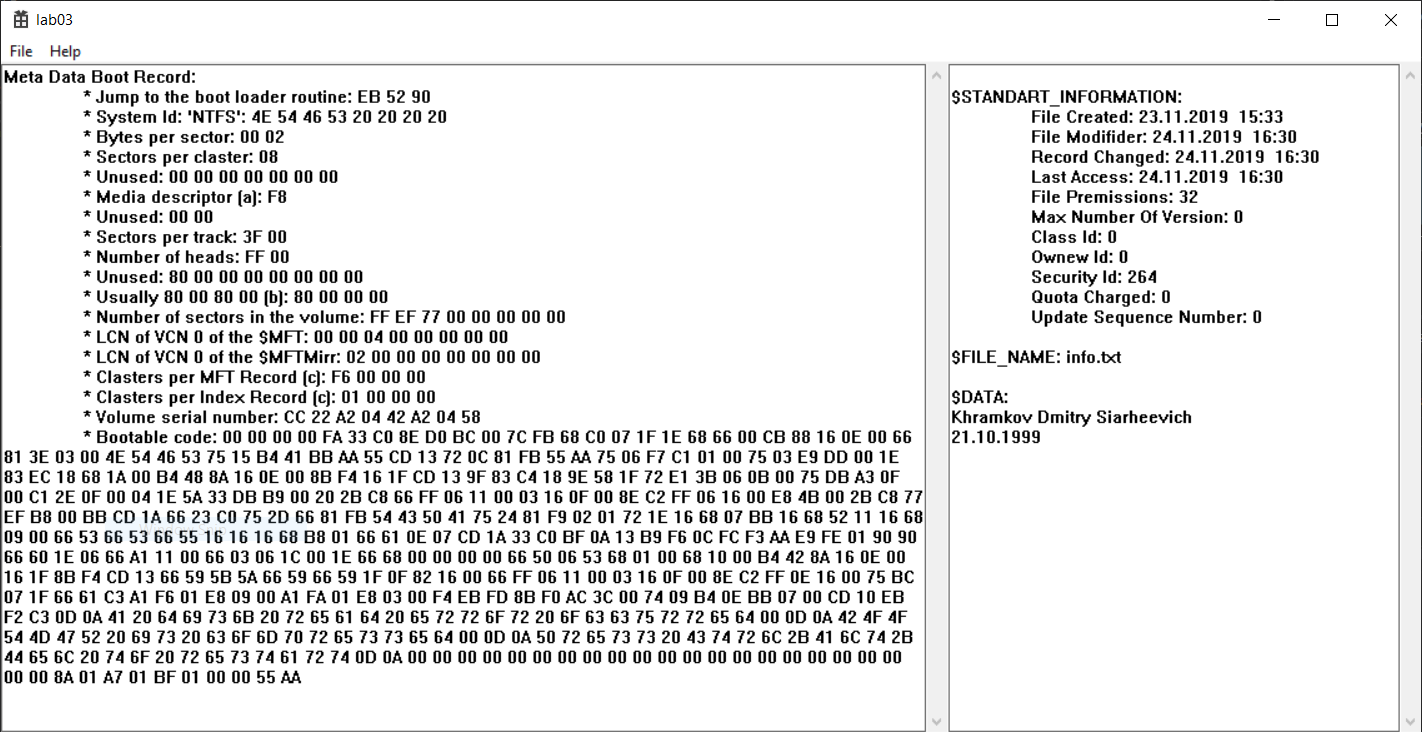
Запись Data run делится на 3 части: размер записи, количество кластеров и номер первого кластера. Для определения размеров записей 2 и 3 нужно разобрать 1 запись: первый бит отвечает за размер второй записи, второй – за размер первой записи.

1. **Исследовать изменение элементов системных файлов NTFS при выполнении команд COPY, MOVE, DEL, RENAME.**

Команды:

* COPY – в MFT была добавлена новая запись без атрибута $OBJECT\_ID;
* MOVE – в атрбут каталога $INDEX\_ROOT была добавлена новая запись;
* DEL – размер таблицы MFT уменьшился, данные остались;
* RENAME – произошли изменения в атрибуте $FILE\_NAME.

**Результаты выполнения:**



**Вывод:**

* 1. Изучена структура BOOT;
  2. Изучена таблица файлов MFT;
  3. Изучено понятие атрибутов файла;
  4. Исследованы последствия выполнения команд COPY, MOVE, DEL, RENAME.