

编号: _____

实验	一	二	三	四	五	六	七	八	总评	教师签名
成绩										

武汉大学国家网络安全学院

课程实验(设计)报告

题 目: _____ 软件安全实验 _____

专业(班): _____ 信息安全 _____

学 号: _____ 2021302181156 _____

姓 名: _____ 赵伯侯 _____

课程名称: _____ 软件安全实验 _____

任课教师: _____ 赵磊 _____

2023 年 10 月 15 日

目 录

实验 1 磁盘格式与数据恢复	1
1.1 实验名称	1
1.2 实验目的	1
1.3 实验步骤及内容	1
1.3.1 第一阶段:	1
1.3.2 第二阶段:	3
1.3.3 第三阶段:	6
1.3.4 第四阶段:	9
1.4 实验关键过程、数据及其分析.....	10
1.4.1 在磁盘分区过程中, 用户提供了哪些信息? 分析分区工具的工作原理。	10
1.4.2 高级格式化与低级格式化的具体原理和区别是什么?	10
1.4.3 查找资料, 对 NTFS 分区的总体结构进行分析, 尝试对 NTFS 下删除的文件进行手工恢复。	11
1.4.4 用数据粉碎工具粉碎指定文件, 分析其数据粉碎原理。	13
1.4.5 通过分区表看到的分区字节数为何与资源管理器中看到的分区字节数有差异?	13
1.4.6 如果删除的文件是长文件名, 如何恢复所有文件名。	13
1.5 实验体会和拓展思考	14

实验 1 磁盘格式与数据恢复

1.1 实验名称

磁盘格式与数据恢复

1.2 实验目的

- 1) 了解磁盘的物理和逻辑结构
- 2) 熟悉 FAT32 文件系统
- 3) 学会使用磁盘编辑软件
- 4) 了解文件删除、格式化的基本原理
- 5) 能够利用工具或者手工恢复被删除的文件

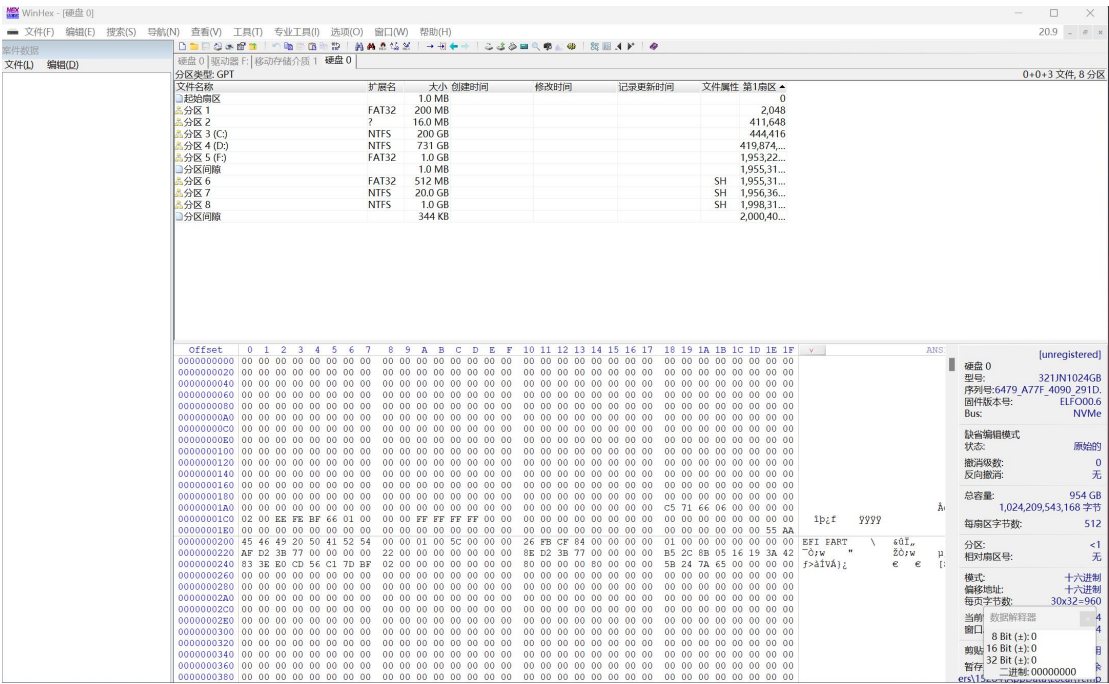
1.3 实验步骤及内容

1.3.1 第一阶段：

- 熟悉 WinHex 的使用。
- 熟悉磁盘工具的使用。
 - 利用 WinHex 查看物理磁盘和逻辑磁盘。
 - 了解 WinHex 中相关工具的用法。

(1) 利用磁盘工具查看本地物理盘

利用 WinHex 打开本地的物理盘的情况如下图所示，存在有 8 个分区和 2 个分区间隔。



使用 WinHex 打开本地的逻辑分区后可以看到在逻辑分区中的众多文件目录。

The screenshot shows the WinHex application window. The top menu bar includes File (F), Edit (E), Search (S), Navigation (N), View (V), Tools (T), Professional Tools (P), Operations (O), Windows (W), and Help (H). The toolbar contains various icons for file operations and viewing. The main window is divided into several panes:

- File List (Left):** Shows a directory tree for drive F: with files like \$RECYCLE.BIN, System Volume Information, 密码学, 操作系统, 零诺, and 计网.
- File Properties (Top Right):** Displays details for the selected file, including size, creation/modification times, and file attributes.
- Hex View (Bottom):** Shows the raw data of the selected file in hexadecimal and ASCII. The ASCII column contains text like "EX MSDOS5.0", "FAT32", "Near: ariai af -fifaifuref-", "Disk error", and "any key to restart".
- Right Panel:** Contains a "Data Interpreter" (数据解释器) window showing bit-level information (8 Bit, 16 Bit, 32 Bit) and a "Disk Information" (磁盘信息) window showing drive F: details, including file system (FAT32), status (original), and disk space (153,432,064 bytes used, 873 MB free).

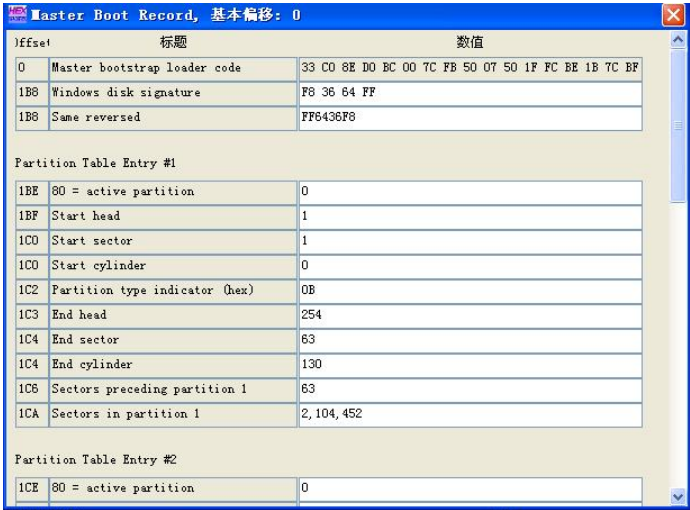
通过使用导航栏中的跳转至偏移量和跳转至扇区可以成功跳转到对应的位置上。

1.3.2 第二阶段:

- 其 分析本地硬盘的主引导扇区
- 其 利用磁盘编辑工具查看 MBR 磁盘分区并分析:
 - 主引导扇区由哪些部分组成?
 - 四个主分区项的内容各代表什么?
 - 分析主扩展分区表的结构。
 - 通过分区项来确定每个本地逻辑盘的位置以及大小，并画出本地硬盘的逻辑结构。
- 其 利用磁盘编辑工具查看 GPT 磁盘分区并分析
 - GPT 分区结构与 MBR 的具体差异有哪些?
 - 主分区头所在扇区包括哪些重要内容，验证这些重要内容的有效性。

(1) 查看主引导扇区的分区表。

将主引导扇区的分区表打开后如下图所示。其中前 1B8H（440）个字节为主引导扇区的引导代码，从第 1BEH（446）个字节开始为主引导扇区的分区表，最后以 55AA 结尾。



四个主分区项的内容如下图所示，

Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ANSI ASCII
0000000288	32	E4	8A	56	00	CD	13	EB	D6	61	F9	C3	49	6E	76	61	2ašv í eÖaüÄInva
0000000304	6C	69	64	20	70	61	72	74	69	74	69	6F	6E	20	74	61	lid partition ta
0000000320	62	6C	65	00	45	72	72	6F	72	20	6C	6F	61	64	69	6E	ble Error loadin
0000000336	67	20	6F	70	65	72	61	74	69	6E	67	20	73	79	73	74	g operating syst
0000000352	65	6D	00	4D	69	73	73	69	6E	67	20	6F	70	65	72	61	em Missing opera
0000000368	74	69	6E	67	20	73	79	73	74	65	6D	00	00	00	00	00	ting system
0000000384	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000400	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000416	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000432	00	00	00	00	00	2C	44	63	F8	36	64	FF	00	00	00	00	,Dcø6dÿ
0000000448	01	00	0B	FE	3F	82	3F	00	00	00	84	1C	20	00	00	00	b???
0000000464	01	83	05	FE	7F	05	C3	1C	20	00	C3	1C	20	00	00	00	f p Ä Ä
0000000480	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000496	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	55	ÄÄ
0000000512	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	U*
0000000528	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000544	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000560	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000576	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000592	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

(2) 分析主引导扇区的内容组成。

其中，每一个分区表项的大小为 16 个字节。在一个分区表项中，首先 00 代表分区的启动盘，010100 代表这个分区的起始扇区为（0 柱面，1 磁头，1 扇区），0B 代表该分区的文件系统为 FAT32，FE3F82 代表磁头号 254；扇区号为 63 柱面号为 255，3F000000 代表该分区前面已有 63 个扇区，这 63 个扇区为系统隐藏扇区。84 1C 20 00 代表表明该分区有 2104452 个扇区。

(3) 分析主拓展分区的分区表

打开一个主拓展分区的分区表如下图所示。

Offset	标题	数值
0	Master bootstrap loader code	33 C0 8E D0 BC 00 7C FB 50 07 50 1F FC BE 1B 7C BF
440	Windows disk signature	22 F4 71 FF
440	Same reversed	FF71F422

Offset	标题	数值
446	80 = active partition	0
447	Start head	0
448	Start sector	1
448	Start cylinder	1
450	Partition type indicator (hex)	0F
451	End head	254
452	End sector	56
452	End cylinder	147
454	Sectors preceding partition 1	14,280
458	Sectors in partition 1	2,099,160

Offset	标题	数值
462	80 = active partition	0

Offset	标题	数值
00000000432	00 00 00 00 00 2C 44 63 22 F4 71 FF 00 00 00 00	,Dc"ôqy
00000000448	01 01 05 FE 38 93 C8 37 00 00 D8 07 20 00 00 00	p8-E7 0
00000000464	01 94 07 FE 78 26 A0 3F 20 00 D8 07 20 00 00 00	" p x s ? 0
00000000480	41 27 0B FE 78 B9 78 47 40 00 D8 07 20 00 00 00	A' p x ' x G 0 0
00000000496	41 BA 07 FE B8 DF 50 4F 60 00 B0 0F 40 00 55 AA	A° p , B P O ' ° 0 U *

一个拓展分区中可以有多个逻辑驱动器，扩展分区中的每个逻辑驱动器的分区信息都存在一个类似于 MBR 的扩展引导记录中，如上图所示，扩展引导记录包括分区表和结束标志“55 AA”，没有引导代码部分。

(4) 主拓展分区结构。

EBR 中分区表的第一项描述第一个逻辑驱动器，第二项指向下一个逻辑驱动器的 EBR。如果不存在下一个逻辑驱动器，第二项就不需要使用。下图为在磁盘管理器中查看逻辑磁盘状态。绿色框中的部分就是主拓展分区。

磁盘 3	基本	11.99 GB	联机
新加	新加	新加	新加
509	516	1.00	1.00
状态	状态	状态	状态
			8.99 GB
			未指派

(5) 分析主分区项的内容各代表什么？

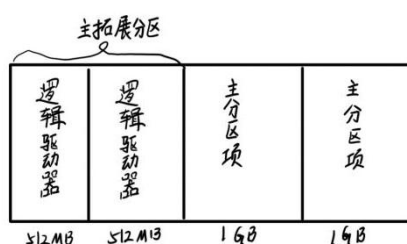
对于第一个分区项（00 00 01 01 05 FE 38 93 C8 37 00 00 D8 07 20 00），开始扇区位置为（0 磁头，1 扇区，1 柱面），总扇区数为 2099160，结束扇区位置为（254 磁头，56 扇区，147 柱面），保留空间为 2,099,160 扇区，即约为 1GB。

对于第二个分区项（00 00 01 94 07 FE 78 26 A0 3F 20 00 D8 07 20 00），开始扇区的位置为（0 扇区，1 磁头，148 柱面），总扇区数为 2,099,160，结束扇区位置为（254 磁头，56 扇区，294 柱面），保留空间大小为 2,099,160 扇区，约为 1GB。

对于第三个分区项（00 00 41 27 0B FE 78 B9 78 47 40 00 D8 07 20 00），开始扇区的位置为（0 扇区，1 磁头，295 柱面），总扇区数为 2,099,160，结束扇区位置为（254 磁头，56 扇区，441 柱面），保留空间大小为 2,099,160 扇区，约为 1GB。

(6) 画出本地硬盘的逻辑结构

本地硬盘的逻辑结构如下图所示：



(7) GPT 分区结构与 MBR 的具体差异有哪些？

1. MBR 中分区表的组织将磁盘的可寻址存储空间限制为 2TB（ $2^{32} \times 512$ 字节）。
2. MBR 只支持 4 个主分区，或者 3 个主分区和 1 个扩展分区的组合。如果要创建更多分区，则需要将其中一个主分区设为“扩展分区”，然后在扩展分区内创建更多逻辑分区。否则，磁盘将转换为动态磁盘。
3. 在 MBR 中，分区表的大小是固定的；在 GPT 分区表头中可自定义分区数量的最大值，也就是说 GPT 分区表的大小不是固定的。

(8) 主分区头所在扇区包括哪些重要内容，验证这些重要内容的有效性。

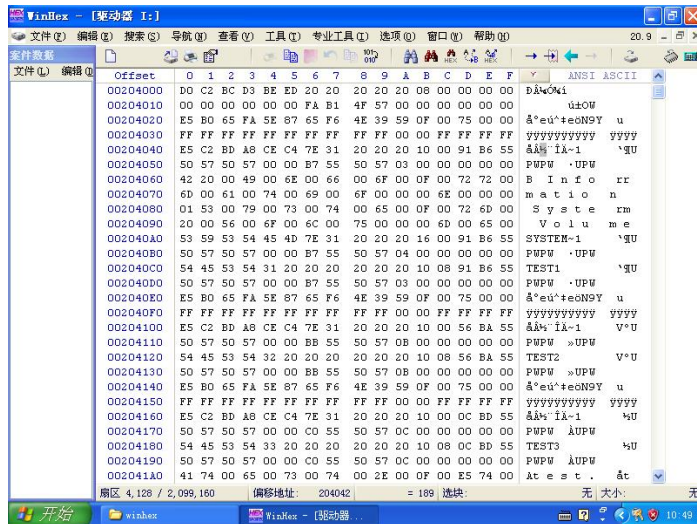
包含主引导扇区代码，bios 在执行自己固有的程序以后就会 jump 到 MBR 中的第一条指令。将系统的控制权交由 mbr 来执行。

1.3.3 第三阶段:

- 熟悉 FAT32 文件格式。
- 用 WinHex 打开某个 FAT32 分区格式的逻辑盘（如 E 盘）。
- 查看该逻辑盘的起始扇区，分析起始扇区中的相关字段（BPB:BIOS Parameter Block）。
- 查看 FAT1 和 FAT2 的内容和大小。
- 查看该逻辑盘的根目录区。
- 查看某个文件的目录项结构和 FAT 链以及具体存储位置。
- 在根目录下建立文本文件 test-学号后 3 位.txt，其中填充 60K 左右的文本字符保存。
- 查看该文件的目录项，对其进行分析，并得到该文件所在位置以及大小。
- 查看首簇位置，并得到簇链表。通过簇链表查看该文件内容。
- 记录首簇位置（1AH-1BH, 14H-15H）和文件大小（1CH-1FH）。

(1) FAT32 逻辑盘 I 的起始扇区相关字段。

用 WinHex 在虚拟机中打开 FAT32 分区格式的逻辑盘 I 如下图所示

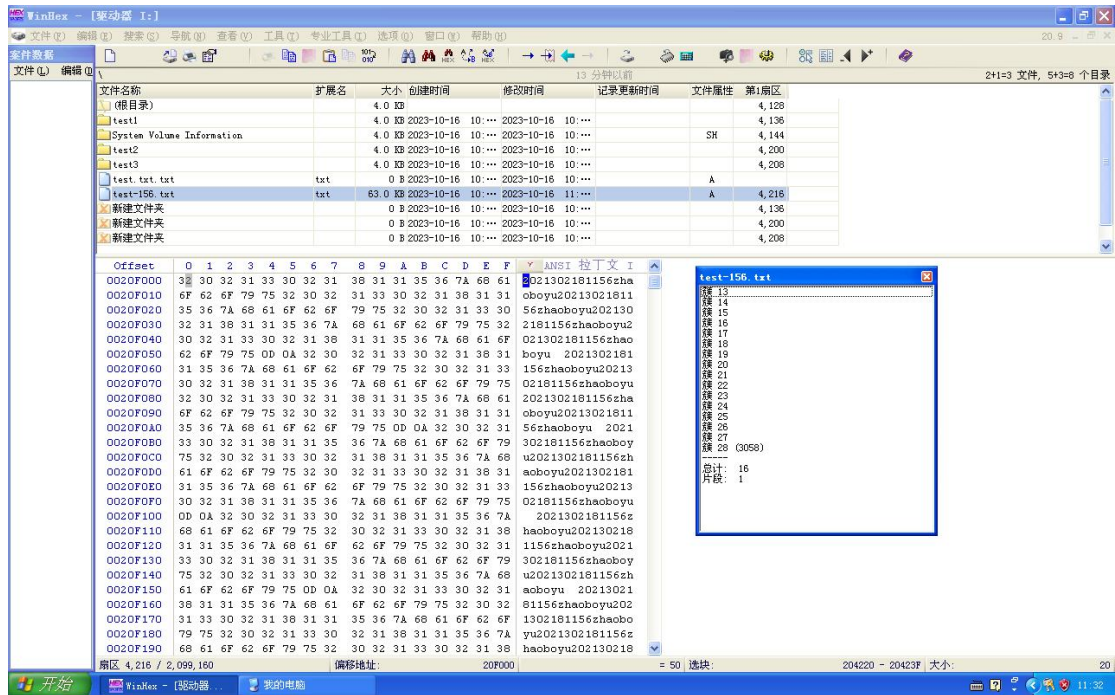


该逻辑盘的起始扇区如下图所示，其中每个扇区的字节数为 512 个字节，每个簇的扇区数为 8。



(4) 查看 test-156 文件的信息。

在 test-156.txt 文件中存有 63KB 的数据后观察其簇的结构, 如下图左下角显示的是该文件的具体存储位置, 右下角显示出的是该文件的簇的格式。



得到的该文件的目录项如下所示。00-07 代表文件的拓展名, 08-0A 代表文件的拓展名, 14H-15H 为起始簇号高 16 位 (0000)。1AH-1BH 为起始簇号低十六位 (0D00)。1CH-1FH 为文件字节长度 (0000FBF2), 为 64498 字节。起始簇号 0000000DH, 为 13 号。

00204210	50 57 50 57 00 00 C7 55	50 57 00 00 00 00 00 00	PUPV CUPV
00204220	54 45 53 54 2D 31 35 36	54 58 54 20 18 37 C2 55	TEST-156TXT 07&U
00204230	50 57 50 57 00 00 AC 59	50 57 0D 00 F2 FB 00 00	PUPV -YFV 0&U
00204240	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	

分析得簇列表如下所示, 该文件共占有 16 个簇, 因此其簇列表有着 16*4=64B 的内容, 其首簇数据为 0000000E=14 指向下一个簇号。最后一个簇的内容为 FFFFFFFF

00004830	FF FF FF 0F 0E 00 00 00	0F 00 00 00 10 00 00 00	yy0000000000000000
00004840	11 00 00 00 12 00 00 00	13 00 00 00 14 00 00 00	00 00 00 00
00004850	15 00 00 00 16 00 00 00	17 00 00 00 18 00 00 00	00 00 00 00
00004860	19 00 00 00 1A 00 00 00	1B 00 00 00 1C 00 00 00	00 00 00 00
00004870	FF FF FF 0F 00 00 00 00	00 00 00 00 00 00 00 00	yy0000000000000000

1.3.4 第四阶段:

✎ 手工恢复被删除的文件

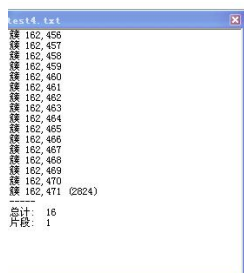
- 删除前面所建立的文件。(del&shift+del)
- 利用 WinHex 在该文件所在盘符查找该.txt 文件的目录项。
- 查看目录项的变化。
- 利用该残余目录项来计算被删除的文件所在的位置。
- 手工恢复该文件(文件名、首簇高位、簇链表修复)。

(1) 恢复前文件信息。

在删除文件之前首先得到该文件的目录表项的内容如下图所示，

00205240	54 45 53 54 34 20 20 20	54 58 54 20 18 39 45 82	TEST4	TXT 09E,
00205250	50 57 50 57 02 00 4E 82	50 57 98 7A 08 FB 00 00	PUPW	N,PW"z00

查看该文件的簇列表如下图所示



000A2A60	99 7A 02 00 9A 7A 02 00	9B 7A 02 00 9C 7A 02 00	9D 7A 02 00 9E 7A 02 00	9F 7A 02 00 A0 7A 02 00	A1 7A 02 00 A2 7A 02 00	A3 7A 02 00 A4 7A 02 00	A5 7A 02 00 A6 7A 02 00	A7 7A 02 00 FF FF FF FF
----------	-------------------------	-------------------------	-------------------------	-------------------------	-------------------------	-------------------------	-------------------------	-------------------------

(2) 恢复后文件信息。

将文件删除之后查看文件的目录项发现簇高位由原来的 0002 被清零，簇列表被全部清零。

000A2A60	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	000A2A70	00 00 00 00 00 00 00 00	000A2A80	00 00 00 00 00 00 00 00	000A2A90	00 00 00 00 00 00 00 00
00205240	55 45 53 54 34 20 20 20	54 58 54 20 18 39 45 82	TEST4	TXT 09E,				
00205250	50 57 50 57 00 00 4E 82	50 57 98 7A 08 FB 00 00	PUPW	N,PW"z00				

(3) 对 test4 进行数据恢复操作

首先计算簇高位，打开 test4 前面和后面的文件发现其簇高位都为 02

002051E0	54 45 53 54 32 20 20 20	54 58 54 20 18 39 45 82	TEST2	TXT 09E,	002052A0	54 45 53 54 35 20 20 20	54 58 54 20 18 39 45 82	TEST5	TXT 09E,
002051F0	50 57 50 57 02 00 4E 82	50 57 88 7A 08 FB 00 00	PUPW	H,PW"z00	002052B0	50 57 50 57 02 00 52 82	50 57 A8 7A 08 FB 00 00	PUPW	R,PW"z00

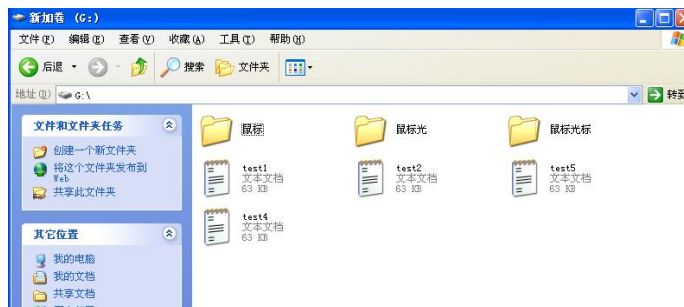
所以推测出该文件的簇高位也应该为 02，将目录表项更改完后的结果应该为

20:5240h:	54 45 53 54 34 20 20 20	54 58 54 20 18 39 45 82	TEST4	TXT 09E,
20:5250h:	50 57 50 57 02 00 4E 82	50 57 98 7A 08 FB 00 00	PUPW	N,PW"z00

接下来修改簇列表，已知簇高位为 02，簇低位为 7A 98，将簇列表修改完成后的结果如下图所示，

A:2A60h:	99 7A 02 00 9A 7A 02 00	9B 7A 02 00 9C 7A 02 00	9D 7A 02 00 9E 7A 02 00	9F 7A 02 00 A0 7A 02 00	A1 7A 02 00 A2 7A 02 00	A3 7A 02 00 A4 7A 02 00	A5 7A 02 00 A6 7A 02 00	A7 7A 02 00 FF FF FF FF
----------	-------------------------	-------------------------	-------------------------	-------------------------	-------------------------	-------------------------	-------------------------	-------------------------

重新打开删除文件 test4 所对应的盘区发现被删除的文件已经被恢复。



1.4 实验关键过程、数据及其分析

1.4.1 在磁盘分区过程中，用户提供了哪些信息？分析分区工具的工作原理。

用户需要提供的信息仅有新分区的文件系统和新分区的大小，工作原理就是首先分区工具读取硬盘上的分区表，获取该磁盘中已知的分区结构和信息，然后根据用户输入的参数计算出新分区的起始位置和大小，并在分区表中创建相应的分区项，同时在磁盘上为新分区分配空间。

然后根据用户提供的文件系统对新分区进行格式化，在新分区的位置创建新的文件系统结构，存储元数据，并准备分区以存储文件和目录。

最后分区工具将对分区表进行更新，将新分区的信息写入分区表中。

1.4.2 高级格式化与低级格式化的具体原理和区别是什么？

原理：

低级格式化是指对物理磁盘介质进行处理，重新建立磁道和扇区的布局。在低级格式化过程中，磁盘驱动器会对磁盘表面进行物理处理，包括将磁道划分为扇区、标定磁道和扇区的位置以及校准磁头等操作。低级格式化通常由磁盘驱动器的制造商在生产过程中完成，用户很少需要手动进行低级格式化。

高级格式化是指在低级格式化之后对磁盘进行逻辑上的处理，包括创建文件系统、分配文件系统结构和元数据等。高级格式化通常由操作系统或格式化工具提供，用户可以在使用新磁盘或重新格式化磁盘时执行高级格式化。

区别：

在原理层面，低级格式化处理物理磁盘介质，重新建立磁道和扇区的布局；而高级格式化处理逻辑上的文件系统结构和元数据。

在作用上，低级格式化主要用于磁盘驱动器的生产阶段，由制造商完成，目的是初始化磁盘；高级格式化用于创建文件系统、分区和准备磁盘以存储数据。

在可控性上，低级格式化通常由磁盘驱动器制造商进行，用户很少需要手动进行；而高级格式化由用户或操作系统控制，并提供了更多的选项和灵活性，例如选择文件系统类型、分区大小等。

1.4.3 查找资料，对 NTFS 分区的总体结构进行分析，尝试对 NTFS 下删除的文件进行手工恢复。

NTFS 分区的总体结构包括以下几个部分。

主引导记录。NTFS 分区通常位于硬盘的主引导记录中。MBR 包含引导代码和分区表，用于引导操作系统和标识磁盘上的分区。

NTFS 引导扇区。NTFS 分区的第一个扇区是 NTFS 引导扇区（NTFS Boot Sector），也称为扇区 0。它包含了引导代码和关键的文件系统参数，如 OEM 标识、扇区大小、分区偏移等。

文件分配表。NTFS 使用了一种称为文件分配表（File Allocation Table）或 MFT（Master File Table）的结构来管理文件和目录的分配和存储。MFT 是 NTFS 中的核心组成部分，记录了文件和目录的属性、大小、权限等信息。

文件记录。NTFS 中的每个文件和目录都在 MFT 中有一个相应的文件记录。文件记录包含了文件的属性、文件名、数据的物理位置等信息。对于较小的文件，文件记录直接存储在 MFT 中；对于较大的文件，文件记录中存储了指向数据存储位置的指针。

数据存储区。NTFS 使用数据存储区来存储文件和目录的实际数据。对于较小的文件，数据存储区可以直接包含文件数据；对于较大的文件，数据存储区中存储了数据的扇区位置。

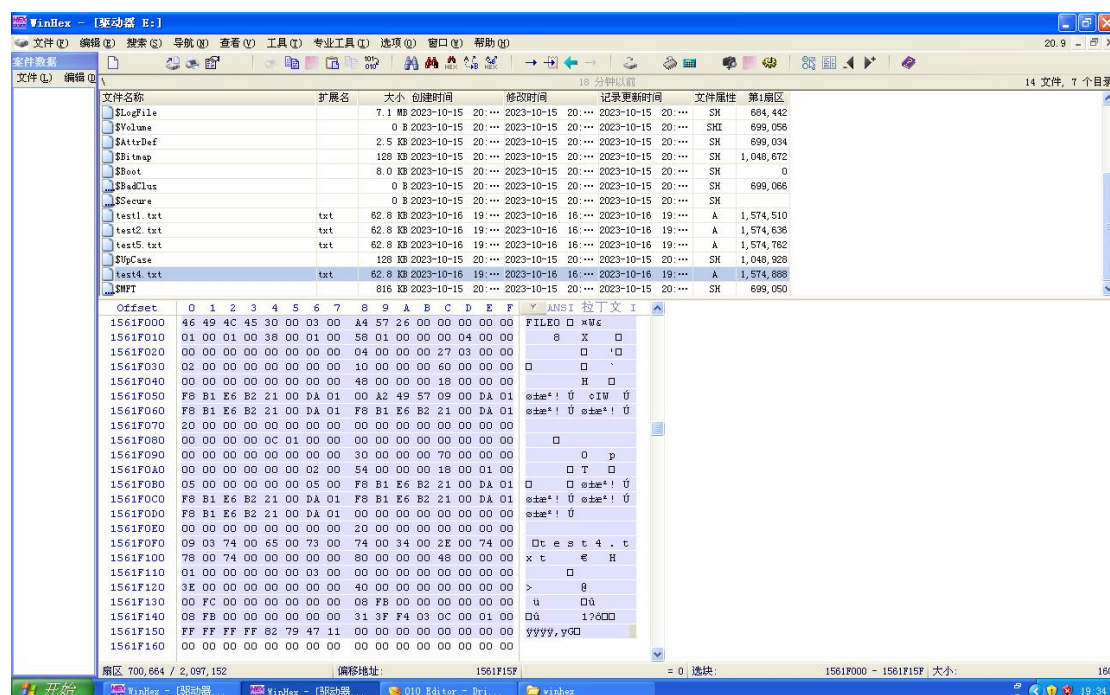
文件属性。NTFS 支持各种文件属性，包括标准属性（如文件大小、创建时间、修改时间等）和扩展属性（如压缩属性、加密属性等）。这些属性存储在文件记录中，并提供了对文件的详细描述和管理。

安全描述符。NTFS 使用安全描述符来管理文件和目录的访问权限。安全描述符包含了访问控制列表（Access Control List, ACL），用于定义允许或拒绝访问文件的用户和组。

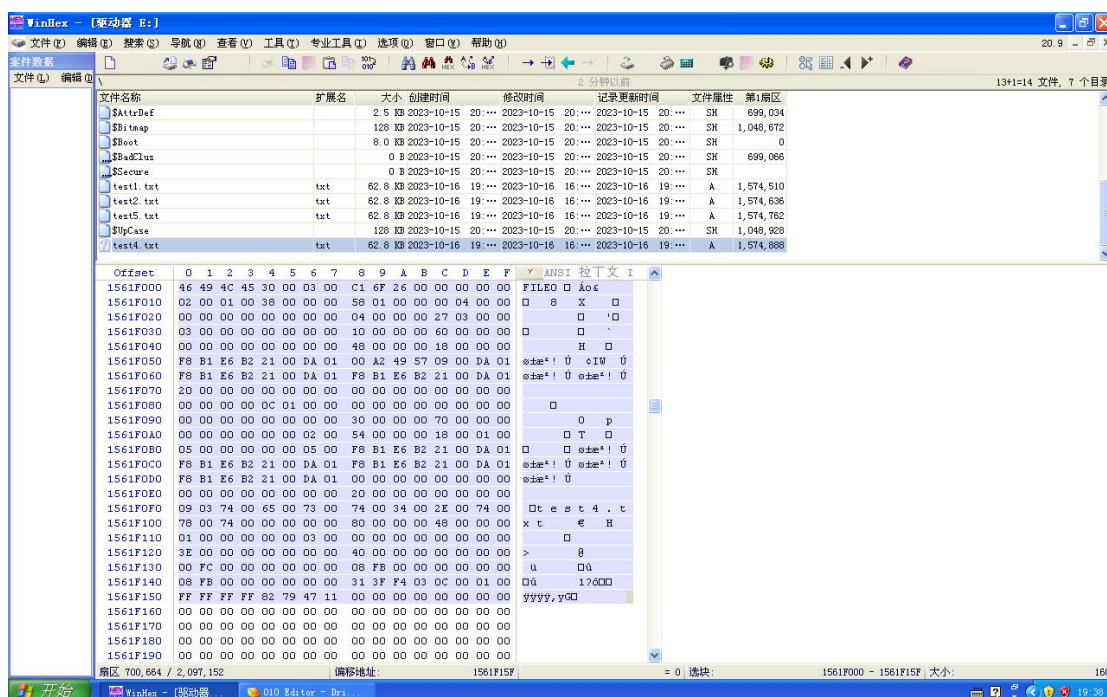
日志文件。NTFS 使用日志文件（Log File）来记录对文件系统的更改操作。日志文件可以用于恢复文件系统的一致性，并提供了对系统崩溃或意外断电的保护。

以下对 NTFS 文件系统中文件进行数据恢复。

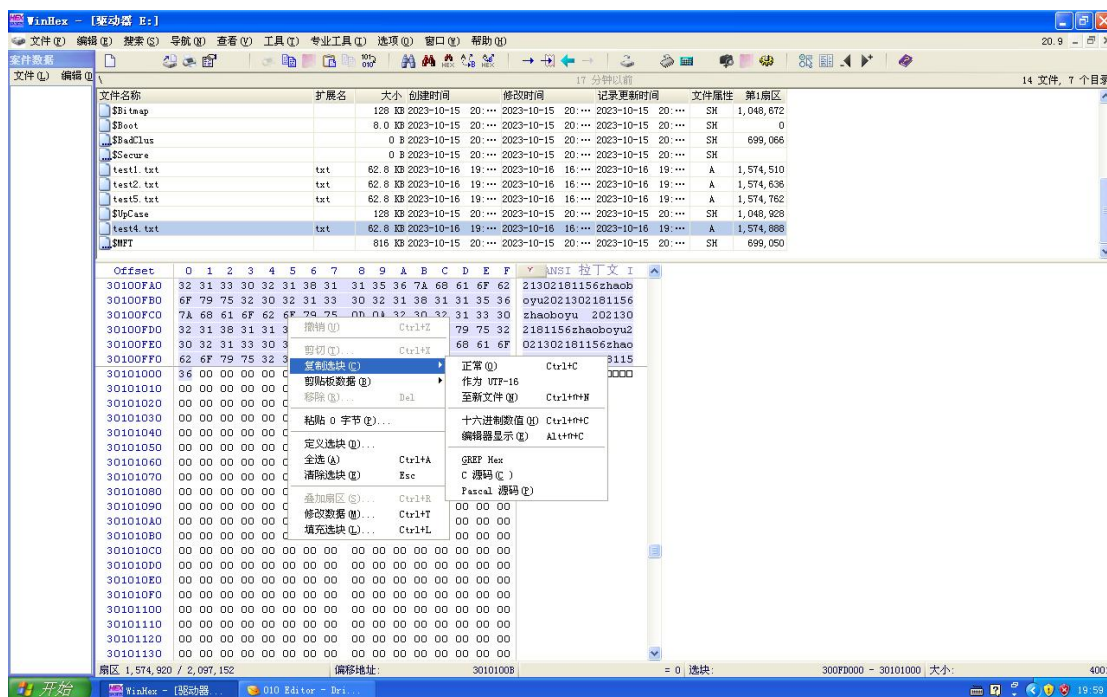
在删除前 test4 的文件目录如下图所示。



将 test4 彻底删除后其目录表如下图所示。

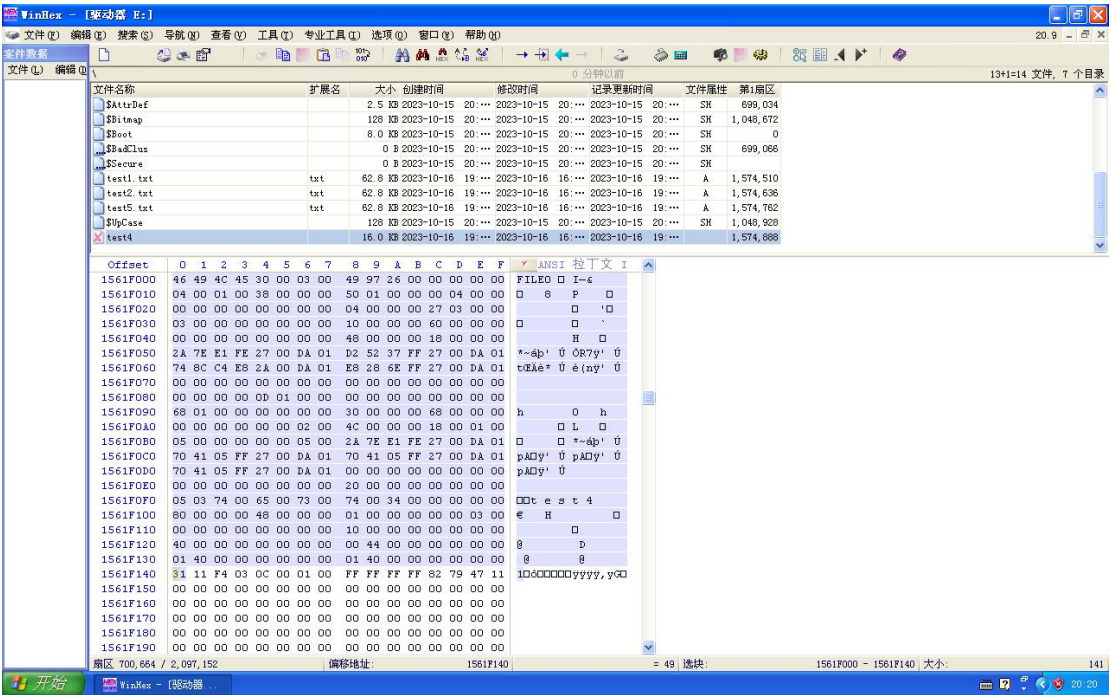


对目录表进行分析，在 80H 中保存有文件的内容，其中 01 代表这是非常驻属性，找到文件的簇流信息为 31 3F F4 03 0C 可得起始簇位置为 0C03F4H=787444,簇的大小为 10H=16，使用 winhex 跳转到起始簇 787444 的位置，并将其选择为块起始位置，尾部位置为 787444+16=787460，将选块复制进新文件即可完成数据恢复。



1.4.4 用数据粉碎工具粉碎指定文件，分析其数据粉碎原理。

使用数据粉碎工具对 test4 进行删除后的结果如下图所示



在 test4 的原文件目录下的所有数据都已经被重写为其他数据，由此推断数据粉碎工具使用一种或多种覆盖算法来重复写入特定的数据模式到要删除的文件或磁盘空间上。这些算法通常使用随机数据、零值或者特定模式的数据进行覆盖。

1.4.5 通过分区表看到的分区字节数为何与资源管理器中看到的分区字节数有差异？

在分区表中，分区字节数通常是通过读取分区表中的元数据来计算得出的。分区表中的元数据包含了有关分区的信息，如起始位置、结束位置等。通过计算这些信息，可以得到分区的字节数。而资源管理器中看到的分区字节数则是通过读取文件系统的信息来计算得出的。文件系统包含了文件和目录的组织结构以及其在磁盘上的分布情况。资源管理器通过读取文件系统的信息，包括文件分配表和目录结构等，计算得出每个分区的字节数。

1.4.6 如果删除的文件是长文件名，如何恢复所有文件名。

如果删除的是长文件名则可以通过查阅文件系统日志来进行数据恢复，某些文件系统会记录文件系统操作的日志，包括文件的创建、修改和删除等操作。通过分析文件系统日志，可以查找已删除文件的相关信息，包括文件名和位置。或者可以借助文件标识符，在某些文件系统中，文件名和文件数据是分开存储的。文件名通常存储在目录中的文件标识符或类似的数据结构中，而文件数据则存储在不同的位置。通过查找文件标识符，可以找到已删除文件的元数据，包括文件名和其他属性。

1.5 实验体会和拓展思考

- 1.在我们平时使用电子设备时的删除功能可能并不是我们理解的删除，更像是将文件的标签进行了删除，而文件的本体还保存在硬盘中。
- 2.市面上所存在的众多文件恢复和文件彻底粉碎产品也是借用了这一原理，文件恢复产品通过恢复文件的标签或者说头目录表对文件本体进行恢复，而文件粉碎产品也是通过读取文件本体数据的各个位置将其覆盖为新的随机数据对文件进行改写。