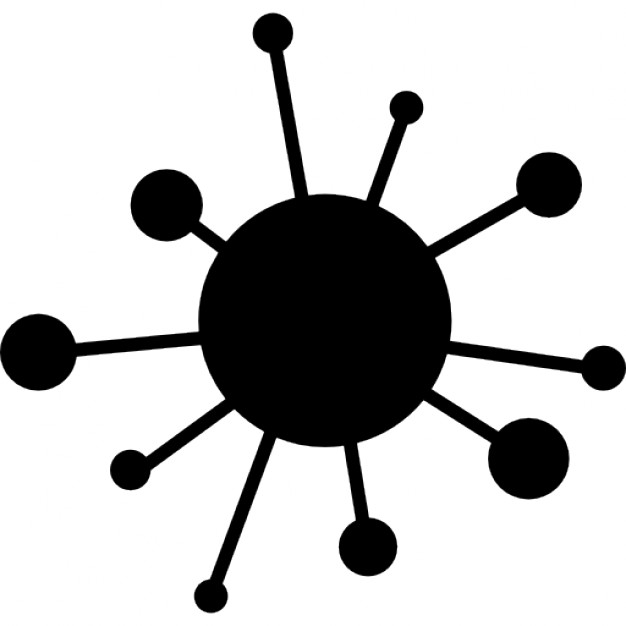
# 计算机病毒学

实验2引导病毒制作过程



学号：0x091395

姓名：uknowho

实验目的

1、 掌握计算机引导型病毒的工作原理。

2、 掌握计算机引导型病毒的设计思路。

3、 掌握计算机引导型病毒的实现步骤。

4、 进一步熟悉工具软件debug或debug32的使用。

实验原理

盗用INT 13中断

假如系统感染了设计的病毒，当0ADD:0100处的用户程序“user code”执行指令int 13h时，在操作系统中断向量区0000:004C处查询到int 13h的入口地址为0ADD:1000，因此，指令指针首先调整为0ADD:1000，开始执行病毒代码“virus code”。

病毒代码“virus code”主体代码执行完后，最后一条执行的指令是jmp far 020A:045D，即跳转到正常int 13h的程序“normal int 13h code”入口，因此，指令指针接着调整为020A:045D。

正常int 13h的程序“normal int 13h code”最后一条指令是“iret”，恢复中断调用产生的现场，指向用户程序“user code”的恢复点，继续往下执行用户程序的其它代码。

实验步骤

1. 点击“开始”按钮，选择“运行”菜单项，输入“cmd”，点击“确定”，进入命令行状态窗口。
2. 输入命令“cd\”回到当前盘的根目录，输入命令“debug”进入debug工作窗口。
3. 输入命令“r”查看程序当前使用的内存段地址，如图1所示，例如：0ADD。
4. 输入命令“d 0:4C L4”查询并记下int 13h的正常入口地址，如图1所示，例如：020A:045D

5、 设计病毒代码“virus code”，显示字符串“Virus code is running …”，在进入正常int 13h代码入口前显示字符串“Enter nomal int 13h code now …”，然后跳转到正常int 13h的代码入口。

示例操作如下：

a 1000

mov ah,09

mov dx,1120

int 21

mov ah,09

mov dx,1140

int 21

jmp far 020A:045D

e 1120 “Virus code is running …” 0d 0a 24

e 1140 “Enter nomal int 13h code now …” 0d 0a 24

6、 在中断向量区修改int 13h的入口地址，指向病毒代码入口0ADD:1000。

示例操作如下：

e 0:4C 00 10 DD 0A

可以使用下面的指令修改并查看修改结果：

-e 0:4c 00 10 DD 0A

-d 0:4c L4

0000:0040 00 10 DD 0A

7、 设计用户代码“user code”，显示字符串“User code is running …”， 然后调用int 13h的00h号功能，复位物理磁盘80h，最后正常退出用户进程的运行。

示例操作如下：

a 100

mov ah,09 //int 21 9号功能，显示字符串

mov dx,0120 //要显示的字符串的位置

int 21

mov ah,00 //int 13 0 号功能，复位磁盘80

mov dl,80 //要复位的磁盘

int 13

mov ax,4c00 //int 21 4c号功能，退出程序（函数exit（0））

int 21

e 0120 “User code is running …” 0d 0a 24

8、 如图7所示，将指令指针修改指向用户代码“user code”入口，使用“p”命令跟踪用户程序运行，在执行13B8:010B处指令“int 13”时，使用命令“t”跟踪程序运行。

如图7所示，此时，指令指针的入口置于设计的病毒代码入口地址0ADD:1000。

9、 继续使用“p”命令跟踪病毒程序运行过程，执行完0ADD:100E指令，指令指针指向正常int 13h的入口地址020A:045D。

10、对正常的int 13h的程序运行过程暂不研究，使用下面的命令，一次型执行完成，返回用户程序执行指令“int 13”的恢复点0ADD:010D。 示例操作如下：

g 13B8:010D

示例操作如图8所示。

11、继续使用“p”命令，跟踪执行完用户程序，直至程序运行结束，如图7所示。

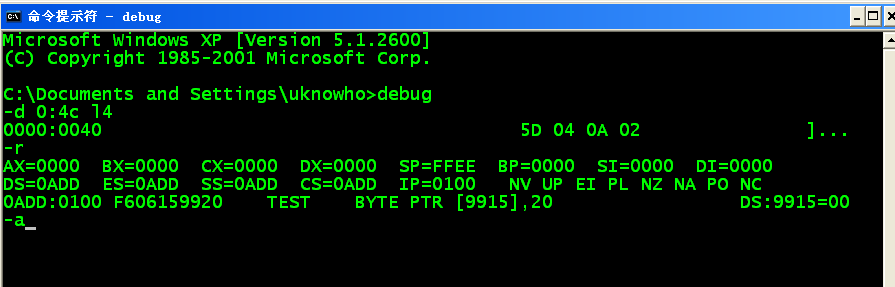


图1 int13位置

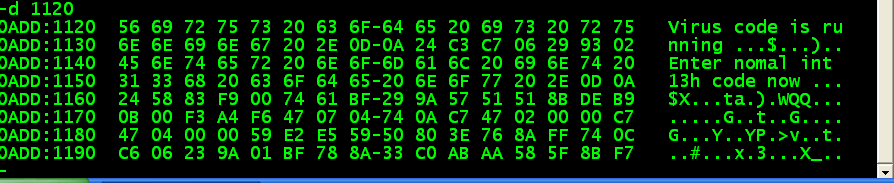


图2 病毒代码

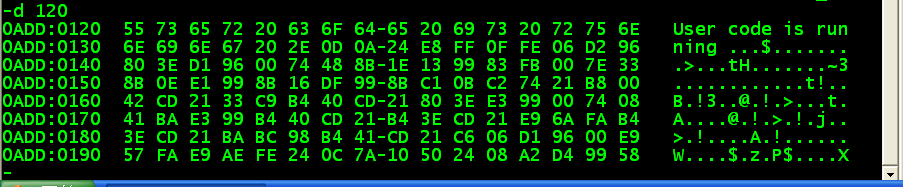


图3 用户代码

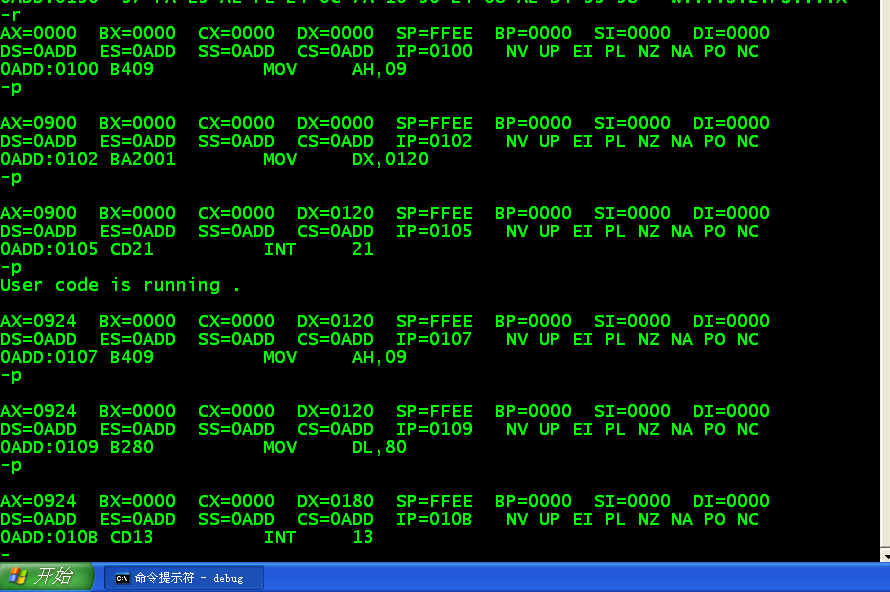


图5 跟踪用户代码

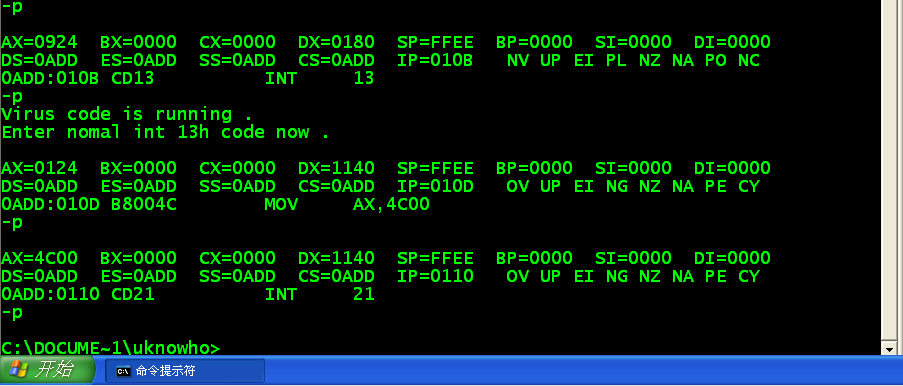


图7用户调用IN13