电脑病毒入侵实时报警技术及实现

杨德荣　（重庆市财政局信息中心）

摘要：本文利用电脑病毒的共性和 *DOS* 系统功能的特点，提出了一种病毒入侵实时检测方法，并给出了程序实例。克服了以往检测方法的不足。

一、病毒入侵检测方法

目前微机病毒尽管层出不穷，种类繁多，但他们都有一个共性：利用 DOS 内存管理的一些缺陷，以合法或非法的手段欺骗 DOS，DOS 承认它所“盗用”的一块内存，以获得 DOS 管理之外的“安乐窝”，然后再修改中断向量，截获计算机控制权。按宿主的不同，病毒可分为引导型和文件型，下面对它们入侵的检测方法进行分别讨论。

1.引导型病毒

任何引导型病毒进行传染时都要对引导扇区（即 0 面 0 柱 1 扇区）进行写操作，对引导区的写入只能通过调用中断 INT13H 和 INT26H 来实现，故我们可以在上述 ISR（中断服务程序）的入口处插入一段检测程序，当检测到对引导区有写入要求时给出提示，然后再执行原 ISR。本文中程序的提示是响铃直至按下 ESC 键为止。

2.文件型病毒

文件型病毒入侵检测的常见方法有两种，一是 SIZE 法，即检查可执行文件的长度是否发生了变化。DIR-Ⅱ病毒不改变文件的长度，故该法已不完美。二是中断向量表法。即在没有病毒的情况下提取标准中断向量表保存起来，在适当的时候再把当前的中断向理表与已保存的标准中断向量表相比较，看是否发生了变化；通常是修改时钟中断（INT 8H 或 INT 1CH），不断地（每秒 18.2 次）把当前的中断向量表与标准的相比较；这种修改时钟中断的方法，用户使用起来极不方便；因为，一旦运行了合法的需修改中断向量表的程序，就得重新提取标准的中断向量表，甚至重新引导系统。本文中的中断向量表法采取的不是修改时钟中断的方法，克服了上述缺点；它的基本思想是：在每个文件执行开始时，提取标准的中断向量表，在文件执行结束时进行比较，从而可知该文件是否被病毒感染。由于所有的病毒要进行传染都得修改中断向量，故中断向量表法是一种令人满意的方法。

二、DOS 可执行文件的加载及退出与文件型病毒的实时检测

在 DOS 环境下可运行的文件主要有 COM 和 EXE 文件，可执行文件调入内存的方法有两种：一是在 DOS 提示符下，打入文件名，然后回车；二是使用中断 21H 的 4BH 号功能调用来完成。分析 DOS 外部命令执行的过程可知，在 DOS 提示符下打入文件名运行可执行文件的实质也是使用中断 21H 的 4B 号功能，所以，就其本质而言，4BH 号功能是装载一个可执行文件的唯一手段。因此，我们可扩充 INT 21H 的 4B 号功能，使之具有提取并保存中断向量表的功能。

在 DOS 环境下，结束一个进程返回父进程（或 DOS），是通过调用下列中断之一来完成的：INT 20H，INT 21H 的 OH、31H、4CH 子功能，INT 27H。所以，如果分别对这些中断进行扩充，在它们的 ISR 入口处插入比较中断向量表的代码，就能对病毒的入侵起到检测作用。为了有效地编制程序，需要分析一下这几个中断区别；INT 20H 同于 INT 21H 的 OH 号功能，但发出 INT 20H 指令时，代码段寄存器 CS 的内容一定要等于该程序的段地址，该中断的主要功能是以 PSP 中 OA 位移处开始的三个双字域的内容恢复中断向量表中 INT 22H（终止地址）、INT 23H（终止处理）、INT 24H（严重错误处理），然后转到 INT 22H 给出的终止位置；DOS 的 4CH 号功能与 OH 号功能的在于 4CH 号有退出码；INT 21H 的 31H 号功能和 INT 27H 是驻留退出。在执行退出功能前，INT 22H、INT 23H、INT 24H 的中断向量与加载时可能不同，作中断表比较时应跳过这几个中断向量。

源程序 CHECKV.ASM(附后)经过 MASM 编译，用 LINK 连结后，再用 EXE2BIN 转换成 CHECKY.COM。将其放在 AUTOEXEC. BAT 文件中，系统启动

后一次加载,驻留内存即可。

本程序在 AST386、LX386 及 IBMPC／XT 等机运行通过。

```
;文件名:CHEKCV.ASM
code segment
assume cs:code,ds:code
org 100h
start:jmp init
  old13h   dd  ?;保存原 INT 13H 中断向量
  old26h   dd  ?;保存原 INT 26H 中断向量
  old21h   dd  ?;保存原 INT 21H 中断向量
  old20h   dd  ?;保存原 INT 20H 中断向量
  old27h   dd  ?;保存原 INT 27H 中断向量
intvect db 1024 dup(0);保存中断量;表缓冲
  warn db 'WARNING:the executive'
  db 'file may infected by virus!'  ;提示信息
doscall macro function              ;宏定义调用
  movah, function                   ;DOS 功能
  int 21h
  endm
getput macro intno,oldint,newint    ;宏定义取／置中断向量
  mov al,int no
  doscall 35h
  mov word ptr oldint, bx
  mov word ptr oldint+2,es
  mov dx,offset newint
  doscall 25h
  endm
pushr macro                 ;宏定义保存寄存器
  push ds
  push es
  push ax
  push bx
  push cx
  push dx
  push si
  push di
  push f
  endm
popr macro toold            ;宏定义恢复寄存器并跳转原中断
  popf
  pop di
  pop si
  pop dx
  pop cx
  pop bx
  pop ax
  pop es
  pop ds
  assume ds:nothing
  jmp cs:loold
  endm
new13h proc far       ;新 INT 13H 中断服务程序
  sti
  jmp new
flag db 'checkv'      ;设置安装标志
  new:push r          ;保存寄存器
  cmp ah,03h          ;是磁盘写操作吗?
  jne exit
  cmp dh,0            ; 是写零头吗?
  jmp exit
  cmp ch,oh           ;是写零道吗?
  jne exit
  cmp cl,01h          ;是写一扇区吗?
  jne exit
  call prompt         ; 调用子程序报警
exit:popr old13       ;宏调用恢复寄存器并跳转原中断
new13h endp
new26h proc far       ;新 INT 26H 中断服务程序
  sti
  push r
  cmp dx,0            ; 是写逻辑零扇区吗?
  jne yy
  call prompt         ;调子程序报警
yy:popr old26h
new26h endp           ;写引导区报警子程序
prompt proc near
bb:mov ax,0e07h
  int 10h
  mov ah,1
  int 16h
  jz bb
  mov ah,0
  int 16h
  cmpal, 1bh
  jnz bb
  ret
prompt endp
new21h proc far       ;新 INT 21H 中断服务程序
  sti
  push r              ;是加载程序功能调用吗?
  cmp ah,4bh
  jne yy1
call mvect            ;调子程序提取中断向量表
  jmp yy5             ;是终止进程 OH 功能调用吗?
yy 1:cmp ah,oh
  jne yy2
  jmp yy4
yy2 :cmp ah,04ch      ;是终止进程 4CH 功能调用吗?
```

```
    jne y y3
    jmp yy4              ;是终止进程 31H 功能调用吗?
yy3:cmp ah,31h
    jne yy5
yy 4:
    call cmpp            ;调子程序比较中断向量表
yy5:popr old21h
new21h endp              ;新 INT 20H 中断服务程序
new20h proc far
    sti
    push r
    call cmpp            ;调子程序比较中断向量表
    popr old20h
new20h endp              ;新 INT 27H 中断服务程序
new27h proc far
    sti
    push r4   call cmpp  ;调子程序比较中断向量表

    popr old27h
new27h endp
mvect proc               ;提取中断向量表子程序
    push cs
    pop ds
    leadi,intvect
    xor ax,ax
    mov dx,ax
    mov si,ax
    mov cx,1024
    push cs
    pop es
    cld
    repmovsb
    ret
mvect endp
cmpp proc near           ;比较中断向量表子程序
    push cs
    pop ds
    mov si,offset intvect
    xor ax,ax
    mov es,ax
    mov di,ax
    mov cx,136
    cld
    repz cmpsb
    or cx,cx
    jz next
    jmp aa
next:mov ex,876
    ;跳过 INT 22H、INT23H、INT 24H 三中断向量
    add si,12
    add di,12
    repz cmpsb
    or cx,cx
    jz cmpret
aa:push cs
    pop ds
    mov ah,0eh
    mov si,offset warn
    mov cx,52
    mov bl,07h
loop1:lpdsb
    int 10h
    loop loop1
cmpret:ret
cmpp endp
init:                    ;初始化部分
    push cs
    pop es
    xor ax,ax            ;测试安装标志
    mov di,offset flag
    mov ds,ax
    mov cx,6
    mov bx,13h * 4
    lds si,[bx]
    addsi,4
    cld
    repz cmpsb
    or cs,cx
    jnz install
    push cs
    pop ds
    mov dx,offset msg    ;显示已安装信息
    doscall 09h
    int 20h
install:push cs
    pop ds
    getput 13h,old13h,new13h    ;取／置 INT 13 中断向量
    getput 26h,old26h,new26h    ;取／置 INT 26 中断向量
    getput 20h,old20h,new20h    ;取／置 INT 20 中断向量
    getput 27h,old27h,new27h    ;取／置 INT 21 中断向量
    getput 21h,old21h,new21h    ;取／置 INT 27 中断向量
    call mvect      ;为让本程序通过检测,取一次中断向量表
    push cs
    pop ds
    mov dx,offset init
    int 27h         ;驻留退出
msg db 'The CHECHV hax been'
    db 'loaded¥'
code ends
    end start
```