计算机系统安全管理与维护

齐齐哈尔第一机床厂 计算中心 张会臣

随着现代化大生产的迅速发展，计算机日益广泛地应用于各行各业中，逐步发挥着巨大的作用，成为现代社会进步的一大标志。人们使用计算机，不仅由于它的运算速度快，精度高，文字处理功能强，而且还由于它的可靠性好，这表现在两方面：一是硬件和软件系统的性能好，二是计算机系统的安全管理与维护。后者在计算机系统和应用方面占很大的比重。我厂共有四十多台微型计算机，它们在财务、劳资、生产、销售等部门发挥着重要作用，这在一定程度上与计算机系统安全管理和维护是分不开的，没有它，计算机的使用是没有保障的。下面就以下几个方面谈谈我们的做法，与大家共同探讨。

一、系统口令的设置

为了防止非法用户使用计算机，为了保护自己编写的软件的合法权益，不许别人随便作用这些软件，通常采用给系统或软件加口令的办法来实现。在实践过程中，我们在给系统加口令的时候发现，要想做到系统保密，必须做到以下几点：

1.在输入口令时，要防止用户强行中断，以进入系统。

2.口令在一定时期要经常地改变。

3.口令需给定输入次数，超过者停机或中断文件的执行。

4.口令和存取权限相结合，主要用户和一般用户使用系统的不同部分，这要用口令来区分(主要指软件)。

5.口令要尽可能长一些，并且容易记忆。

具体的做法(以 DOS 系统为例)是：

(1)在操作系统中加入口令。寻找硬磁盘 0 面 0 道 1 扇区中的分区空闲空间嵌入口令程度部分，以达到加密的目的。

(2)修改 COMMAN.COM 文件，在其空间内插入一段口令程度部分。

口令这一级保护，对于专业人员和系统分析人员来说是没有用的，对平等互利遥操作员和外行人员来说还是可行的。

二、计算机病毒的防护

在人们使用计算机获得收益的同时，也受到了计算机病毒的不同程度的干扰和损害，现只简单介绍一下计算机病毒的防护问题(以硬盘为例)。

1.对于引导型病毒，我们采取了下面两种方法之一

(1)我们主要采用保存主引导记录扇区和引导记录扇区内容的办法来实现。在一台机器买来或重新格式化以后，确认无病毒的情况下，采用附录 1 的办法，把文件存入这台机器的软系统盘上(DOS 盘上)，为以后恢复作准备。

(2)使用“158”病毒检测盘中的 BOOTSAFE 进行监视，办法是：把 BOTTSAFE 放入批处理 AUTOEXE.BAT 文件中，当有病毒向主引导区和引导区写入的时候，系统就会发出警告，以便你能选择操作。

2.对于文件型病毒，我们采取了下面两种方法之一

(1)文件长度的核对，文件型病毒在感染文件的同时，一般都使文件的长度增加一定数目的字节，覆盖原文件的地址和寻找文件中的空隙存放病毒体的病毒除外。自己编写一个文件，其功能是对根目录下(全部子目录)的.COM 和.EXE 文件进行文件长度核对。基本方法是把上一次取得并存盘的文件长度和现取得的文件长度做比较并存盘，若结果一致则返回 DOS 系统，否则发出警告，该文件放入批处理 AUTOEXEC.BAT 中。

(2)使用“158 种”病毒检测盘中的 TSAFE 进行监视，办法是：把 TSAFE 放入批处理 AUTOEXEC.BAT，

当有病毒向可执行文件(.COM 和.EXE)写入的时候,系统就会发出警告,以便你能选择操作。

3.完善机房管理制度

除了采用上面的方法以外,我们在管理制度上也加强了防护病毒的措施:

(1)对外来盘片,均要用公安部或最新版本的检测软件进行检测,以防毒病侵入。

(2)禁止玩游戏。

(3)不在带有硬盘的机器上做软件解密。

(4)不明底细的软件不用。

(5)外来人员使用机器要做好登记。

三、系统备份

计算机磁盘是存储信息的好帮手,它的存储量大,信息保持时间长的优点很容易被人接受。但在一些意外的情况下,可能使信息丢失,如挤压,高温,折叠和磁化等,给人带来很大的损失,为了避免这种现象的发生,一是要妥善保管好这些磁盘;二是要做好系统和为据的备份,这样做的目的是万一在事故发生后,可以用备份盘恢复,免得数据丢失。

(1)定期检查,做好备份。

(2)在编写软件时,要考虑到数据备份和系统维护。

(3)编写一专门软件,用于重要的数据和系统的备份。可编一批处理文件,在关机时执行之。

(4)买来新的系统和软件时,要做备份。

四、系统格式化

我们有时使用 FORMAT.COM 格式化软盘,由于粗心大意,忘记了参数,很可能把硬盘格式化。为了防止下这种现象发生,在装入系统的时候,对 FORMAT.COM 做一些特殊处理,使之只对软盘有效,对硬盘不起作用;或把缺省的驱动器定为 A 盘或 B 盘。如果对硬盘进行格式化时,用软系统盘来做这项工作。

五、异常情况的处理

在计算机系统维护的过程中,很难避免一些异常情况的发生,如计算机病毒,掉电和磁盘损坏等,遇到这种情况发生,道德不要惊慌,经过仔细地研究,一定能找出其原因和处理办法,下面就几个问题作说明(以硬盘为例)

1.系统的恢复

计算机在遭到引导型病毒感染后,引导记录所在的扇区可能被病程序覆盖或引导记录搬迁到其它位置,这样系统启动的时间就比平常启动的时间要长,并破坏信息或系统不能启动,如果参照附录 2 的做法,就可以恢复原系统。

2.系统误格式化的处理

在 DOS2.XYK ,FORMAT.COM 做引导区、FAT 表、根目录区三部分的初始化,对数据区无任何影响,这样只要有根目录区的备份就可以恢复文件和数据,避免大的损失(相同系统下)。DOS3.XY 以上比较复杂,需酌情处理:若 FAT 表的备份或根目录区或数据区的内容不被破坏,则可以恢复,不则就无法恢复了。3.系统文件删除后的恢复

由于误操作,系统文件可能被删除,在这种情况下,请不要往盘中写入信息,使用 PCTOOLS 可以恢复这些文件,进入 PCTOOLS 后,按 F3 键,选择 U 操作,然后按提示操作即可。用 DEBUG.COM 也可做这项工作,其原理如下:

(1)把目录区所在扇区写回原来的位置。

(2)寻找删掉的文件名,此时只可以找到以 E5 开头,其它字符相同的文件名。

(3)把 E5 改成你所希望的字符的 ASCII 值。

(4)把改后的目录区所在扇区写回原来的位置。

4.BACKUP 备份盘损坏后的恢复

我们在给系统做备份时通常使用 BACKUP.COM 文件,这样做的优点是:文件目录的顺序不变,超过软盘容量的大文件也可存下。其缺点是:在某一块盘中的一个文件发生故障时,其它盘中的文件便不可用 RESTORE.COM 来恢复。万一发生故障,在保证文件不被破坏的情况下,可有附录 3 来恢复。其原理为:在每一个文件的开始部分存有 BACKUP 所需的参数及日期,只要把这些 128BYTE 的文件头去掉,就可恢复原文件的本来面目,再把相同的文件合并到一起。

5.DBF 文件损坏后的恢复

企业进行现代化管理和办公室自动化离不开计算机,有很多单位都使用数据库,主要是 dBASEⅢ语言,在某些异常情况下,数据库文件.DBF 可能遭到损坏,造成很大的损失,笔者在实践中编写了一段小程序,可以解决这个问题,参见附录 4.其原理为:在大部分损坏的.DBF 文件中,主要是程序头(结构)遭到破坏或尾部部分数据丢失。只要记住域的个数和域的总长度,就可以把部分数据恢复出来,这些数据可能格式不对,编辑(文本)后便可以使用.

附录 1

1.存主引导记录扇区

```
A>DEBUG
-A100
  MOV AX,0201
  MOV BX,0200
  MOV CX,0001
  MOV DX,0080
  INT 13
  INT 3
-G
-N A:MBOOT.C
-RBX
  0000
-RCX
  0200
-W200
-Q
```

2.DHB 存引导记录扇区

```
A>DEBUG-100 2 0 1
-N A:BOOT.C
-RCX
  0200
-W100
-Q
```

附录 2

1.写主引导记录扇区

```
A>DEBUG
-A100
  MOV AX,0301
  MOV BX,0200
  MOV CX,0001
  MOV DX,0080
  INT 13
  INT 3
-N A:MBOOT.C
-L200
-G
-Q
```

2.写引导记录扇区

```
A>DEBUG
-N A:BOOT.C
-L100
-W100 2 0 1
```

附录 3

本程序用 TURBO PASCAL 5.50 编译

```
Program restore ;
{使用格式:restore a:[b:]c:}
uses dos;
var
  s,ch1,ch2:string[64];
  dirinfor:searchrec;
  f,g:file;
  buf:array[1…12000] of byte;
  cc:integer;
begin
  if paramcount<2 then
                        begin
                        writeln('restore a:c:');
                        exit;
                        end;   ch1=paramstr(1)+'';
  ch2=paramstr(2)+'';
  if not[upcase(ch1[1])in['A','B']]thenexit;
  getdir(byte(upcase(ch2[1]))-byte('a'+1,s);
{读 c:盘缺目录}
  s:=s+'/';
  if(ch<>s)and(ifngth(ch2)<length(s))then ch2:=;
  findfirst(ch1+'*.*',archive,dirinfoe);
  while doserror+0 do
   begin
     assign(f,ch1+dirinfor.name);
     reset(f,1);
     assogm(g,ch2+dirinfor.name);
     {
i-}
     reset(g,1);
     {
i+}
```

```
        if ioresult+0 then seek(g, filesize(g))else rewrie(g, 1);
         else
{去掉头 128 个字节}
        blockread(f, buf, 128, cc);
{复制其它部分}
        repeat
         blockread(f, buf, 12000, cc);
         blockwrite(g, buf, cc);
        until cc = 0;
        close(f);
        close(g);
        findnext(dirinfor);
      end;
end.
```

附录 4

本程序 turbo pascal 3.00 编译

```
program rdata;
label 10, 20, 30, 40;
var
  f: file of char;
  g: text;
  i, j, num, long: integer;
  ch: cjar;
  ss: string[255];
namel, name2: string[20];
begin
cirscr;
for i: = 1 to 10 do writeln;
write('bad database file name < .dbf > : ');
10: readln(namel);
        namel: = namel+'.dbf';
        assign(f, namel);
        assign(f, namel);
        {
i-}
        reset(f);
        {
i+}
        if ioresult < > 0 then
                        begin
                    __ writeln( ' file' , 。 name2, ' not
found.betry');
                        goto 10;
                        end;
    20: write('new file name < output, ist, .... > : ');
        readin(name2);
        assign(g, name2(;
        if(name2 = 'output')or(name2 = 'ist')rhen goto 30;
    {
    i-}
    reset(f);
    {
    i+}
    if ioresult = 0 then
                        begin
                        writein( f' ile' , name2, ' exist,
ovrewrite it(y / n)?');
                        read(kbd, ch);
                        if ( ch = 'y' )or(ch = 'y' then rewrite
(g) else goto 20;
                        end
                else goto 20;
    30: weite('the number of field__name: ');
        readln(num);
        werte('the length of total field: ');
        readln(iong);
        i: = (1+num) * 32+3;
        seek(f, i);
        while not eof(f) do
        bdgin
        read(f, ch);
        ss: = ch;
        for j: = 1 to iong-1 do
        begin
        if eof(f) then goto 40;
        read(f, ch);
        ss: = ss+ch;
        end;
    40: writein(g, ss);
        end;
        close(f);
        close(f);
end.
```