磁盘文件一体化加密技术

李晓华　(云南省军区自动化站)

一、概 述

在计算机系统的开发研制过程中,磁盘文件加密技术已成为必不可少的部分。目前,国内有关的介绍文章、杂志非常多。例如,通过修改软盘参数表,把软盘格式化成特殊格式来作为密钥盘;利用特殊设备制造出一般驱动器无法生成的特殊格式;磁盘指纹加密;加密卡;掩模加密技术和电磁加密技术等。这些技术对磁盘文件反拷贝起到一定的作用。通常这些加密技术都需制作一个称为 KEY 的加密盘。在执行加密文件时,都要去读 KEY 中的密钥。由于 KEY 盘读写不稳定,且软盘易损坏,因而人们不得不重新设计加密程序,以保护自己所开发软件的安全性。硬盘作为大量存放信息数据的介质,现日益受到软件开发者的重视。由于硬盘参数、型号都不相同,且存储空间较大,因而把软盘上的文件放入硬盘,在硬盘上实施加密已成为广大编程者日益关心的问题。

通常,开发者研制出的产品,需要通过下列步骤:

(1)编制源程序;

(2)调试出一个合格的产品;

(3)设法对产品实施加密;

对前两步,是一般软件编程者自己完成的,而第三步正是本文将要讨论的。为了实现产品的加密,开发者除了向用户提供系统软盘外,还必须把 KEY 盘和安装盘提供给用户,以便用户把加密后的程序安装到硬盘中,供用户使用。这就向开发商提出了更高的要求:如何实施对软盘介质的加密,即制作一个 KEY 盘;如何把软盘上的文件以加密的方式安装到硬盘。对于前者,可以通过制作 KEY 盘的方法完成,使得在 KEY 盘上的文件不能被复制或复制后不能执行。而对于后者,必须制作一个 install 安装程序,以实现硬盘文件的加密。本文试图从这几个方面介绍一体化的加密技术。

二、一体化加密技术的实现

1.设计思想

本文提出的磁盘文件一体化加密技术,主要包括以下三个方面的内容:

(1)KEY 盘的制作:通常商品化的软件是由软盘作为载体提供给用户,这样的软件是不可复制或复制后无效的。

(2)可执行文件的加密:可执行文件的加密,一般采用两种方法实现。但目标只有一个,那就是它不能单独运行。它包括两层意思,其一,把开发出的软件与加密软件作为一体完成,使其加入读取硬盘"指纹"部分,只有当读取"指纹"信息或密钥成功后,程序方能运行;其二,对于执行程序(文件)实施加密预处理,通常是在被加密程序尾部增加一段读取硬盘"指纹"程序,并修改程序入口到增加程序段,只有当该段读取硬盘指纹成功后,方能执行程序。

(3)生成可执行的安装程序:当商品软件制作完成后,用户要使用的最终过程是把软件安装到硬盘上。安装程序具有以下功能:识别是否为 KEY 盘;把产品软件安装到指定的硬盘目录中(若被加密的程序是通过预处理的,则在完成安装后,应对被加密程序实施预处理的逆过程);硬盘指纹的产生。

2.实现方法

前面我们介绍磁盘文件加密的基本设计思想;下面介绍具体的实现方法:

(1)KEY 盘的制作

①软加密:一个 KEY 盘通常由两部分组成:一是在原程序盘制作一个特殊的标志(称为指纹);二是在执行程序中加入判读程序段。实现 KEY 盘的制作,通常采用以下方法:产生密钥磁盘的方法之一就是采用软加密,它主要通过修改软盘扇区上的标识区 ID(其中 ID 标识区共 4 字节:磁道号(C)、磁头号(H)、扇区号(R),扇区长度系数(N))和 ROMBIOS 中的磁盘驱动器参数来实现非 PC 机标准的格式。在 ID 标识区中主要是通过修改扇区指数 N 来改变扇区的大小,N 的取值通常为 0、1、2、3、4、5…参数中的任一个,它分别表示扇区容纳 128、256、512、1024、2048、…个字节的内容。一般标准的格式 N=2,它的扇区大小为 512。一旦 N 值为其它值,则格式化出来的磁盘就不能被标准格式驱动器所读取。只有通过修改相应 ROMBIOS 中的磁盘参数表后,方能读取。这样可在 INSTALL 程序中把 N 值及磁盘参数表修改成与格式化时的参数一致,就可以读取特殊信息的指纹,最后恢复原来值。程序三读取软盘指纹信息,若读取错误则重新启动机器。

②硬加密:磁盘采用的硬加密主要是指的激光加密,

它利用激光在磁盘上打孔,制造出无法复制的硬错误。一般激光打孔、方向性能好,用软件不能识别,但造价高,因而不易推广使用。

(2)硬盘指纹的产生:当被加密软件在软盘上制作完成后,它最终必须安装到硬盘上。这些工作由安装程序INSTALL完成。它完成前面介绍的功能。见程序二。

在INSTALL程序中,最关心的是怎样在硬盘上产生指纹信息。在硬盘上产生"指纹"信息不象在软盘上那样容易,它不能格式化出非PC机标准的格式,因而把"指纹"选择到扇区中的什么位置,是硬盘文件加密的关键。本文提出以下方法:

①在硬盘系统信息占用扇区的前部,即隐藏扇区中实现:我们知道,硬盘的存储格式由两大部分组成:系统信息占用扇和正文数据占用扇组成。见图1所示:

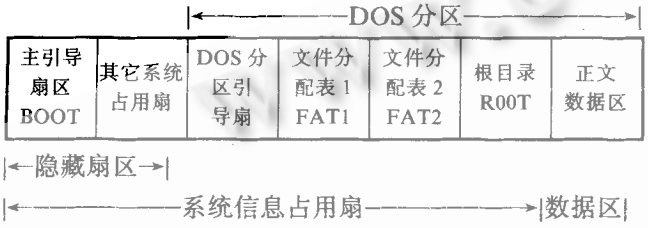

图 1 硬盘存储分配

从图中可以看出,硬盘存储格式由隐藏扇区、DOS分区引导扇、FAT1、FAT2、ROOT、数据区组成。通过用DEBUG分析发现,隐藏扇区通常占0柱0头,其中1扇区为主引导程序所占用,其余扇区为其它系统所用。我们可以把"指纹"信息选择在主引导扇区中的空余区域上安装。若只有一个操作系统则可以把"指纹"信息选择在1扇区以后的扇区上安装。

②在系统信息占用扇区后部,即根目录区的最后。通过对硬盘I/O参数分析,发现在根目录区尾部有很多扇区不会被利用。因而可以利用这些扇区来作为"指纹"信息存放在地方。当然在具体实现时,首先必须确定数据区的起始位置,只有当数据区的起始位置确定好后,根目录靠后的扇区才能被确定。

(3)硬盘指纹的读取:一旦把指纹信息安装后,剩下的工作就是如何去读取"指纹"信息了。读取硬盘指纹信息的程序此处删略。

三、后 记

本文提出磁盘上文件一体化加密的设计方法,对磁盘文件加密、硬盘"指纹"的制作及读取,只是简单的过程。要想实现一个商品化的一体化加密程序,还必须具有较强的反跟踪加密处理和对系统的闭封性完善等功能。有关这方面的文章已有很多,这里不再讨论。

-------- 程序一 --------

```
install.c
insthard.obj
readdisk.obj
readhard.obj
```

-------- 程序二 --------

```
/ * file name:install.c
 1.read disk a:-2 track 0 side 01 sec
 2.call read_disk_key in file readdisk.asm
 3.call inst_hard_key in file insthard.asm
 4.call read_hard_key in file readhard.asm
 5.make PRJ(install.prj)
        install.c
        readdisk.obj
        insthard.obj
        readhard.obj
* /
#include"conio.h"
#include"graphics.h"
extern readdisk();
extern insthard();
extern readhard();
void install();
main()
{
read_disk_key();
printf("\nread DISK key OK!");
getch();
inst_hard_key();
printf("\ninstall HARD DISK key OK!!");
getch();
install();
read_hard_key();
printf("\nread HARD DISK key OK!");
getch();
}
void install()
{
char * ms="Please insert FOX2.0 disk in driver a";
char * ms1="press any key to start to install…";
char * ms2="I am installing FOX program to c:\\FOX";
char * ms3="INSTALLED completed! press any key to quit";
int i;
int gdriver,gmode;
  gdriver=EGA;
  gmode=EGAHI;
  initgraph(gdriver,&gmode,"c:\\tc\\bgi");
setcolor(1);
setbkcolor(2);
cleardevice();
settextjustify(LEFT_TEXT,TOP_TEXT);
settextstyle(TRIPLEX_FONT,HORIZ_DIR,0);
```

```
bar3d(50,15,590,100,10,1);
outtextxy(25,20,"Foxbase 2.0 Install Program");
setcolor(24);
settextstyle(2,HORIZ_DIR,6);
outtextxy(140,60,"Li Xiao Hua Copyright(c) 1995.01.19");
setcolor(5);
setusercharsize(350,textwidth(ms),10,textheight(ms));
bar3d(50,150,590,300,10,1);
settextstyle(3,HORIZ_DIR,0);
outtextxy(55,150,ms);
outtextxy(55,170,ms1);
getch();
system("c:");
mkdir("c:\\fox");
setcolor(6);
setusercharsize(400,textwidth(ms2),20,textheight(ms2));
outtextxy(55,200,ms2);
system("copy a:*.* c:\\fox>NULL");
setcolor(16);
setusercharsize(300,textwidth(ms3),20,textheight(ms3));
outtextxy(55,250,ms3);
getch();
chdir("\\");
system("cls");
closegraph();
return;
}
;———— 程序三 ————
;readdisk.asm
;读取-2 道 01 扇区
_text segment public'code'
dgroup group_data
        assume cs:_text,ds:dgroup,es:dgroup
_text ends
_data segment word public'data'
diskdt db 0dfh,02,25h,01h,04h,1bh
db 0ffh,54h,0f6h,0fh,08h
er_ms db 'Yor use are not INSTALL DISK'
db 0ah,0dh,'￥'
buffer bd 256 dup(0)
save dd 0
drive equ 0(0:a 盘 1:b 盘)
_data ends
_text segment public'code'
        public_read_disk_key
_read_disk_key proc near
sframe struc
baseptr dw ?
retad dw ?
sframe ends
frame equ [bp-baseptr]
        push bp
        mov bp,sp
        sub sp,baseptr
        push ds
        push es
        push di
        push si
        ;
        mov ax,_data
        mov ds,ax
        push ds
        pop es
        ;
        mov ax,351eh
        int 21h
        mov word ptr save,bx
        mov bx,es
        mov word ptr save+2,bx
        lea dx,diskdt
        mov ax,251eh
        int 21h
        ;
        mov bx,offset buffer
        push ds
        pop es
        mov cx,3
loop1:push cx
        mov ax,0201h
        mov dh,0
        mov dl,drive
        mov ch,-2
        mov cl,01h
        int 13h
        pop cx
        dec cx
        jnz loop1
        jc err
        JMP OK
        ;
err: mov dx,offset err_ms
        mov ah,09
        int 21h
        INT 19H
OK:push ds
        pop es
        ;
        mov dx,word ptr save
        mov ds,word ptr save+2
        mov ax,251eh
        int 21h
        ;
        mov ah,00h
        int 13h
        ;
        pop si
        pop di
        pop es
        pop ds
        mov sp,bp
        pop bp
        ret
_read_disk_key endp
_text ends
end;
```