

Linux操作系统的X86汇编程序设计 (3) PDF转换可能丢失图片或格式, 建议阅读原文

[https://www.100test.com/kao\\_ti2020/144/2021\\_2022\\_Linux\\_E6\\_93\\_8D\\_E4\\_BD\\_c103\\_144989.htm](https://www.100test.com/kao_ti2020/144/2021_2022_Linux_E6_93_8D_E4_BD_c103_144989.htm) I/O 端口编程 但直接访问硬件会怎么样呢? 在 Linux 下你需要一个核心模式的驱动程序来做这些工作... 这意味着你的程序必须分成两个部分, 一个核心模式提供硬件直接操作的功能, 其他的用户模式提供接口. 一个好消息就是你仍然可以在用户模式的程序中使用IN/OUT 来访问端口。要访问端口你的程序必须取得系统的同意. 要做这个, 你必须调用 `ioperm()`. 这个函数只能被有 root 权限的用户使用, 所以你必须用 `setuid()` 使程序到 root 或者直接运行在 root 下. `ioperm()` 的语法是这样: `ioperm( long StartingPort#, long #Ports, BOOL ToggleOn-Off)` `StartingPort#` 指明要访问的第一个端口值(0 是端口 0h, 40h 是端口 40h, 等等), `#Ports` 指明要访问多少个端口(也就是说, `StartingPort# = 30h, #Port = 10`, 可以访问端口30h - 39h), `ToggleOn-Off` 如果是 TRUE(1) 就能够访问, 是 FALSE(0) 就不能访问. 一旦调用了 `ioperm()`, 要求的端口就和平常一样访问. 程序可以调用 `ioperm()` 任意多次, 而不需要在后来调用 `ioperm()`(但下面的例子这样做了), 因为系统会处理这些。

```
.io.asm = BITS 32 GLOBAL szHello GLOBAL main
EXTERN printf EXTERN ioperm SECTION .data szText1 db
Enabling I/O Port Access,0Ah,0Dh,0 szText2 db Disabling I/O Port
Acess,0Ah,0Dh,0 szDone db Done!,0Ah,0Dh,0 szError db Error in
ioperm() call!,0Ah,0Dh,0 szEqual db Output/Input bytes are
equal.,0Ah,0Dh,0 szChange db Output/Input bytes
changed.,0Ah,0Dh,0 SECTION .text main: push dword szText1 call
```

```
printf pop ecx enable_IO: push word 1 . enable mode push dword
04h . four ports push dword 40h . start with port 40 call ioperm .
Must be SUID "root" for this call! add ESP, 10 . cleanup stack
(method 1) cmp eax, 0 . check ioperm() results jne Error .Port
Programming Partmov al, 96 . R/W low byte of Counter2, mode 3
out 43h, al . port 43h = control register WritePort: mov bl, 0EEh .
value to send to speaker timer mov al, bl out 42h, al . port 42h =
speaker timer ReadPort: in al, 42h cmp al, bl . byte should have
changed--this IS a timer :) jne ByteChanged BytesEqual: push dword
szEqual call printf pop ecx jmp disable_IO ByteChanged: push
dword szChange call printf pop ecx .End Port Programming
Partdisable_IO: push dword szText2 call printf pop ecx push word 0
. disable mode push dword 04h . four ports push dword 40h . start
with port 40h call ioperm pop ecx .cleanup stack (method 2) pop
ecx pop cx cmp eax, 0 . check ioperm() results jne Error jmp Exit
Error: push dword szError call printf pop ecx Exit: ret . EOF 100Test
下载频道开通 , 各类考试题目直接下载。 详细请访问
www.100test.com
```