

## linux tty pty pts 概念 区别

### 基本概念:

#### 1. tty(终端设备的统称):

tty 一词源于 Teletypes, 或者 teletypewriters, 原来指的是电传打字机, 是通过串行线用打印机键盘通过阅读和发送信息的东西, 后来这东西被键盘与显示器取代, 所以现在叫终端比较合适。

终端是一种字符型设备, 它有多种类型, 通常使用 tty 来简称各种类型的终端设备。

#### 2. pty (虚拟终端):

但是如果我们远程 telnet 到主机或使用 xterm 时不也需要一个终端交互么? 是的, 这就是虚拟终端 pty(pseudo-tty)

#### 3. pts/ptmx(pts/ptmx 结合使用, 进而实现 pty):

pts(pseudo-terminal slave)是 pty 的实现方法, 与 ptmx(pseudo-terminal master)配合使用实现 pty。

### Linux 终端:

在 Linux 系统的设备特殊文件目录/dev/下, 终端特殊设备文件一般有以下几种:

#### 1、串行端口终端(/dev/ttySn)

串行端口终端(Serial Port Terminal)是使用计算机串行端口连接的终端设备。计算机把每个串行端口都看作是一个字符设备。有段时间这些串行端口设备通常被称为终端设备, 因为那时它的最大用途就是用来连接终端。这些串行端口所对应的设备名称是/dev/tts/0(或/dev/ttyS0), /dev/tts/1(或/dev/ttyS1)等, 设备号分别是(4,0), (4,1)等, 分别对应于 DOS 系统下的 COM1、COM2 等。若要向一个端口发送数据, 可以在命令行上把标准输出重定向到这些特殊文件名上即可。例如, 在命令行提示符下键入: echo test > /dev/ttyS1 会把单词"test"发送到连接在 ttyS1(COM2)端口的设备上。可接串口来实验。

#### 2、伪终端(/dev/pty/)

伪终端(Pseudo Terminal)是成对的逻辑终端设备(即 master 和 slave 设备, 对 master 的操作会反映到 slave 上)。

例如/dev/ptyp3 和/dev/ttyp3(或者在设备文件系统中分别是/dev/pty/m3 和 /dev/pty/s3)。它们与实际物理设备并不直接相关。如果一个程序把 ptyp3(master 设备)看作是一个串行端口设备, 则它对该端口的读/写操作会反映在该逻辑终端设备对应的另一个 ttyp3(slave 设备)上面。而 ttyp3 则是另一个程序用于读写操作的逻辑设备。

这样, 两个程序就可以通过这种逻辑设备进行互相交流, 而其中一个使用 ttyp3 的程序则认为自己正在与一个串行端口进行通信。这很象是逻辑设备对之间的管道操作。对于 ttyp3(s3), 任何设计成使用一个串行端口设备的程序都可以使用该逻辑设备。但对于使用 ptyp3 的程序, 则需要专门设计来使用 ptyp3(m3)逻辑设备。

例如，如果某人在网上使用 telnet 程序连接到你的计算机上，则 telnet 程序就可能会开始连接到设备 ptyp2(m2)上(一个伪终端端口上)。此时一个 getty 程序就应该运行在对应的 tty2(s2)端口上。当 telnet 从远端获取了一个字符时，该字符就会通过 m2、s2 传递给 getty 程序，而 getty 程序就会通过 s2、m2 和 telnet 程序往网络上返回"login:"字符串信息。这样，登录程序与 telnet 程序就通过“伪终端”进行通信。通过使用适当的软件，就可以把两个甚至多个伪终端设备连接到同一个物理串行端口上。

在使用设备文件系统 (device filesystem)之前，为了得到大量的伪终端设备特殊文件，使用了比较复杂的文件名命名方式。因为只存在 16 个 tty(ttyp0—ttypf) 的设备文件，为了得到更多的逻辑设备对，就使用了象 q、r、s 等字符来代替 p。例如，ttyps8 和 ptys8 就是一个伪终端设备对。不过这种命名方式目前仍然在 RedHat 等 Linux 系统中使用着。

但 Linux 系统上的 Unix98 并不使用上述方法，而使用了“pty master”方式，例如/dev/ptm3。它的对应端则会被自动地创建成/dev/pts/3。这样就可以在需要时提供一个 pty 伪终端。目录 /dev/pts 是一个类型为 devpts 的文件系统，并且可以在被加载文件系统列表中看到。虽然“文件”/dev/pts/3 看上去是设备文件系统中的一项，但其实它完全是一种不同的文件系统。

即: TELNET ---> TTY3(S3: slave) ---> PTYP3(M3: master) ---> GETTY

实验:

1、在 X 下打开一个或 N 个终端窗口

2、#ls /dev/pt\*

3、关闭这个 X 下的终端窗口，再次运行；比较两次输出信息就明白了。

在 RHEL4 环境下: 输出为/dev/ptmx /dev/pts/1 存在一(master)对多(slave)的情况

### 3、控制终端(/dev/tty)

如果当前进程有控制终端(Controlling Terminal)的话，那么/dev/tty 就是当前进程的控制终端的设备特殊文件。可以使用命令“ps -ax”来查看进程与哪个控制终端相连。对于你登录的 shell，/dev/tty 就是你使用的终端，设备号是(5,0)。使用命令“tty”可以查看它具体对应哪个实际终端设备。/dev/tty 有些类似于到实际所使用终端设备的一个联接。

### 4、控制台终端(/dev/tty0, /dev/console)

在 Linux 系统中，计算机显示器通常被称为控制台终端 (Console)。它仿真了类型为 Linux 的一种终端(TERM=Linux)，并且有一些设备特殊文件与之相关联: tty0、tty1、tty2 等。当你在控制台上登录时，使用的是 tty1。使用 Alt+[F1—F6]组合键时，我们就可以切换到 tty2、tty3 等上面去。tty1—tty6 等称为虚拟终端，而 tty0 则是当前所使用虚拟终端的一个别名，系统所产生的信息会发送到该终端上。因此不管当前正在使用哪个虚拟终端，系统信息都会发送到控制台终端上。你可以登录到不同的虚拟终端上去，因而可以让系统同时有几个不同的会话期存在。只有系统或超级用户 root 可以向 /dev/tty0 进行写操作 即下例:

1、# tty(查看当前 TTY)

/dev/tty1

2、#echo "test tty0" > /dev/tty0

test tty0

## 5 虚拟终端(/dev/pts/n)

在 Xwindows 模式下的伪终端.

## 6 其它类型

Linux 系统中还针对很多不同的字符设备存在有很多其它种类的终端设备特殊文件。例如针对 ISDN 设备的/dev/ttyIn 终端设备等。这里不再赘述。

FAQ: 终端和控制台

吴晋 ([cliffwoo@gmail.com](mailto:cliffwoo@gmail.com))

由于在很多朋友对终端的概念一直不是很清楚，因此写了这个 FAQ，希望能够帮助大家理解这些概念。不妥之处，还请大家来信指出。

**Q:** /dev/console 是什么？

**A:** /dev/console 即控制台，是与操作系统交互的设备，系统将一些信息直接输出到控制台上。目前只有在单用户模式下，才允许用户登录控制台。

**Q:** /dev/tty 是什么？

**A:** tty 设备包括虚拟控制台，串口以及伪终端设备。

**/dev/tty 代表当前 tty 设备**，在当前的终端中输入 `echo "hello" > /dev/tty`，都会直接显示在当前的终端中。

**Q:** /dev/ttyS\*是什么？

**A:** /dev/ttyS\*是串行终端设备。

**Q:** /dev/pty\*是什么？

**A:** /dev/pty\*即伪终端，所谓伪终端是逻辑上的终端设备，多用于模拟终端程序。例如，我们在 X Window 下打开的终端，以及我们在 Windows 使用 telnet 或 ssh 等方式登录 Linux 主机，此时均在使用 pty 设备(准确的说应该 pty 从设备)。

**Q:** /dev/tty0 与/dev/tty1 .../dev/tty63 是什么？它们之间有什么区别？

**A:** /dev/tty0 代表当前虚拟控制台，而/dev/tty1 等代表第一个虚拟控制台，例如当使用 ALT+F2 进行切换时，系统的虚拟控制台为/dev/tty2，当前的控制台则指向/dev/tty2

**Q:** 如何确定当前所在的终端（或控制台）？

**A:** 使用 tty 命令可以确定当前的终端或者控制台。

**Q:** /dev/console 是到/dev/tty0 的符号链接吗？

**A:** 目前的大多数文本中都称/dev/console 是到/dev/tty0 的链接（包括《Linux 内核源代码情景分析》），但是这样说的不确切的。根据内核文档，在 2.1.71 之前，/dev/console 根据不同系统的设定可以链接到/dev/tty0 或者其他 tty\* 上，在 2.1.71 版本之后则完全由内核控制。目前，只有在单用户模式下可以登录/dev/console（可以在单用户模式下输入 tty 命令进行确认）。

**Q:** /dev/tty0 与/dev/fb\*有什么区别?

**A:** 在 Framebuffer 设备没有启用的系统中, 可以使用/dev/tty0 访问显卡。

**Q:** 关于终端和控制台的区别可以参考哪些文本

**A:** 可以参考内核文档中的 Documents/devices.txt 中关于”TERMINAL DEVICES” 的章节。  
另外, 《Linux 内核源代码情景分析》的 8.7 节 以及《Operating Systems : Design and Implementation》中的 3.9 节(第 3 版中为 3.8 节)都对终端设备的概念和历史做了很好的介绍。  
另外在《Modern Operating system》中也有对终端设备的介绍, 由于与《Operating Systems : Design and Implementation》的作者相同, 所以文本内容也大致相同。需要注意的一点是《Operating Systems : Design and Implementation》中将终端设备分为 3 类, 而《Modern Operating system》将终端硬件设备分为 2 类, 差别在于前者将 X Terminal 作为一个类别。

**PS:**

只有 2410 的 2.6 才叫 ttySAC0, 9200 等的还是叫 ttyS0