

实验三：进程管理（一）

1. 分别从至少三个虚拟终端登录，以树状形式列出你的系统中当前运行的所有进程及其PID。找出你当前运行进程的所有祖先进程，并说明其各自的作用。分析 Linux 系统中的进程的家族关系。

答：使用 `pstree -p | more` 命令分别在 `tty1~3` 三个虚拟终端下查看进程，截图如下：

tty1(PID 为 3435)

```

|                                     \-{lightdm}(1164)
|                                     |-{lightdm}(1159)
|                                     \-{lightdm}(1160)
|login(3435)---bash(3565)-+more(3668)
|                                     \-pstree(3667)
|mission-control(1960)-+{mission-control}(1968)
|                                     \-{mission-control}\304s\267(1963)
|modem-manager(485)
|polkitd(584)---{polkitd}(586)
|pulseaudio(1417)-+gconf-helper(1427)
|                                     |-{pulseaudio}(1425)
|                                     \-{pulseaudio}(1426)
|rsyslogd(504)-+{rsyslogd}(505)
|                                     |-{rsyslogd}(522)
|                                     \-{rsyslogd}(523)
|rtkit-daemon(1419)-+{rtkit-daemon}(1421)
|                                     \-{rtkit-daemon}(1422)
|system-service-(2029)
|ubuntu-geoip-pr(1786)
|udev(317)-+udev(610)
|                                     \-udev(613)
|udisks-daemon(1485)-+udisks-daemon(1486)
|                                     \-{udisks-daemon}(1487)
|unity-applicati(1818)-+{unity-applicati}(1834)
|                                     \-{unity-applicati}\304s\267(1821)
|unity-files-dae(1817)-+{unity-files-dae}(1820)
|                                     \-{unity-files-dae}(1824)
|unity-lens-vid(1815)---{unity-lens-vid}\304s\267(1860)
|unity-music-dae(1816)---{unity-music-dae}(1819)

```

tty2(PID 为 3415)

```

|                                     |^-{lightdm}(1164)
|                                     |^-{lightdm}(1159)
|                                     |^-{lightdm}(1160)
|-login(3415)---bash(3797)-+-more(3900)
|                                     |^-pstree(3899)
|-login(3435)---bash(3565)---more(3668)
|-mission-control(1960)-+-{mission-control}(1968)
|                                     |^-{mission-control}\004k\267(1963)
|-modem-manager(485)
|-polkitd(584)---{polkitd}(586)
|-pulseaudio(1417)-+-gconf-helper(1427)
|                                     |^-{pulseaudio}(1425)
|                                     |^-{pulseaudio}(1426)
|-rsyslogd(504)-+-{rsyslogd}(505)
|                                     |^-{rsyslogd}(522)
|                                     |^-{rsyslogd}(523)
|-rtkit-daemon(1419)-+-{rtkit-daemon}(1421)
|                                     |^-{rtkit-daemon}(1422)
|-system-service-(2029)
|-ubuntu-geoip-pr(1786)
|-udev(317)-+-udev(610)
|                                     |^-udev(613)
|-udisks-daemon(1485)-+-udisks-daemon(1486)
|                                     |^-{udisks-daemon}(1487)
|-unity-applicati(1818)-+-{unity-applicati}(1834)
|                                     |^-{unity-applicati}\004k\267(1821)
|-unity-files-dae(1817)-+-{unity-files-dae}(1820)
|                                     |^-{unity-files-dae}(1824)
|-unity-lens-vid(1815)---{unity-lens-vid}\004k\267(1860)

```

```
| | ^- {lightdm} (1164)
| | |- {lightdm} (1159)
| | ^- {lightdm} (1160)
|- login(3415)---bash(3797)---more(3900)
|- login(3423)---bash(4030)-+--more(4133)
| | ^-pstree(4132)
|- login(3435)---bash(3565)---more(3668)
|- mission-control(1960)-+-{mission-control}(1968)
| | ^-{mission-control}\024u\267(1963)
|- modem-manager(485)
|- polkitd(584)---{polkitd}(586)
|- pulseaudio(1417)+-gconf-helper(1427)
| | |- {pulseaudio}(1425)
| | ^-{pulseaudio}(1426)
|- rsyslogd(504)+-{rsyslogd}(505)
| | |- {rsyslogd}(522)
| | ^-{rsyslogd}(523)
|- rtkit-daemon(1419)+-rtkit-daemon}(1421)
| | ^-{rtkit-daemon}(1422)
|- system-service-(2029)
|- ubuntu-geoip-pr(1786)
|- udevd(317)+-udevd(610)
| | ^-udev(613)
|- udisks-daemon(1485)+-udisks-daemon(1486)
| | ^-{udisks-daemon}(1487)
|- unity-applicati(1818)+-unity-applicati}(1834)
| | ^-{unity-applicati}\024u\267(1821)
|- unity-files-dae(1817)+-unity-files-dae}(1820)
| | ^-{unity-files-dae}(1824)
```

--More--

从下图可以看出，所有进程的祖先进程为 init，其 PID 始终为 1。

```
init+-NetworkManager--dhclient
|
|l-dnsmasq
|
|~2*[{NetworkManager}]
|2*[{VBoxClient---VBoxClient---{VBoxClient}}]
|VBoxClient---VBoxClient
|VBoxClient---VBoxClient---2*[{VBoxClient}]
|VBoxService---7*[{VBoxService}]
|accounts-daemon---{accounts-daemon}\204z\267
|acpid
|atd
|avahi-daemon---avahi-daemon
|bamfdaemon---2*[{bamfdaemon}]
|bluetoothd
|colord---2*[{colord}]
|console-kit-dae---64*[{console-kit-dae}]
|cron
|cupsd
|2*[{dbus-daemon}]
|dbus-launch
|dconf-service---2*[{dconf-service}]
|gconfd-2
|geoclue-master
|3*[{getty}]
|gnome-keyring-d---5*[{gnome-keyring-d}]
|goa-daemon---{goa-daemon}
|gvfs-afc-volume---{gvfs-afc-volume}
|gvfs-fuse-daemo---3*[{gvfs-fuse-daemo}]
|gvfs-gdu-volume
|gvfs-gphoto2-vo
--More--
```

2. 执行下列命令和操作:

```
$sleep 180
```

 $^{\wedge}Z$

\$jobs

```
$sleep 240&
```

```
$sleep 300&
```

```
$sleep 330&
```

```
$sleep 630&
```

\$jobs

\$fg %4

 $\wedge C$

\$bg %1

```
$kill %5
```

解释命令的作用和执行结果。

答: \$sleep 180 #睡眠 180 秒

^Z #按下 Ctrl+Z, 挂起进程, 作业将暂停执行

`$jobs` #列出作业控制的内容，显示所有 shell 中已启动的作业状态

```
$sleep 240& #以后台方式执行睡眠 240 秒
```

```
$sleep 300& #以后台方式执行睡眠 300 秒
```

```
$sleep 330& #以后台方式执行睡眠 330 秒
```

```
$sleep 630& #以后台方式执行睡眠 630 秒
```

`$jobs` #列出作业控制的内容

\$fg %4 #将 4 号作业切换到前台并继续运行

^C #终止执行当前作业

\$bg %1 #1 号作业保持在后台并继续运行

\$kill %5 #结束 5 号作业

执行过程如下：

```
chy@chy-VirtualBox:/proc/2129$ sleep 180
^Z
[1]+  Stopped                  sleep 180
chy@chy-VirtualBox:/proc/2129$ jobs
[1]+  Stopped                  sleep 180
chy@chy-VirtualBox:/proc/2129$ sleep 240&
[2] 2784
chy@chy-VirtualBox:/proc/2129$ sleep 300&
[3] 2785
chy@chy-VirtualBox:/proc/2129$ sleep 330&
[4] 2787
chy@chy-VirtualBox:/proc/2129$ sleep 630&
[5] 2788
chy@chy-VirtualBox:/proc/2129$ jobs
[1]+  Stopped                  sleep 180
[2]   Running                  sleep 240 &
[3]   Running                  sleep 300 &
[4]   Running                  sleep 330 &
[5]-  Running                  sleep 630 &
chy@chy-VirtualBox:/proc/2129$ fg %4
sleep 330
^C
chy@chy-VirtualBox:/proc/2129$ bg %1
[1]+ sleep 180 &
chy@chy-VirtualBox:/proc/2129$ kill %5
chy@chy-VirtualBox:/proc/2129$
```

3. 利用 /proc 文件系统，选择一个进程，如 -bash，列出并解释 /proc/\$PID/cmdline, /proc/\$PID/stat, /proc/\$PID/status 文件的内容。

答：使用 `ps tree -p | less` 如下命令查看，得到 bash 进程的 PID 为 5443：

```
chy@chy-VirtualBox: ~
|-getty(941)
|-getty(947)
|-getty(948)
|-getty(950)
|-getty(1256)
|-gnome-keyring-d(1492)-+-{gnome-keyring-d}(1494)
|   |-{gnome-keyring-d}(1495)
|   |-{gnome-keyring-d}(1496)
|   |-{gnome-keyring-d}(1499)
|   |-{gnome-keyring-d}(2005)
|   `--{gnome-keyring-d}(2109)
|-gnome-terminal(5437)-+-bash(5443)-+-less(5650)
|   |   `--pstree(5649)
|   |-gnome-pty-helpe(5442)
|   |-{gnome-terminal}(5438)
|   |-{gnome-terminal}(5439)
|   `--{gnome-terminal}(5449)
|-goa-daemon(2209)---{goa-daemon}(2211)
|-gvfs-afc-volume(1697)---{gvfs-afc-volume}(1698)
|-gvfs-fuse-daemo(1538)-+-{gvfs-fuse-daemo}(1547)
|   |-{gvfs-fuse-daemo}(1548)
|   `--{gvfs-fuse-daemo}(1556)
|-gvfs-gdu-volume(1686)
```

进入/proc/5443 文件夹，查看目录下的文件如下：

```
chy@chy-VirtualBox: /proc/5443
1297 1626 1950 23 5442 969 kpagecount timer_stats
13 1628 1993 234 5443 acpi kpageflags tty
14 1637 1999 235 579 asound latency_stats uptime
1441 1638 2 24 580 buddyinfo loadavg version
1442 1640 2002 25 5908 bus locks version_signature
1455 1641 2016 26 6 cgroups mdstat vmallocinfo
1456 1645 2017 27 64 cmdline meminfo vmstat
1467 1647 2019 28 655 consoles misc zoneinfo
1468 1652 2020 3 656 cpuinfo modules
1476 1686 2021 36 7 crypto mounts
1477 1690 2058 365 712 devices mtrr
1480 1691 2065 368 8 diskstats net
1483 1695 2066 39 811 dma pagetypeinfo
1484 1697 2072 399 842 driver partitions
chy@chy-VirtualBox:/proc$ cd 5443
chy@chy-VirtualBox:/proc/5443$ ls
attr coredump_filter io mounts pagemap stack
autogroup cpuset latency mountstats personality stat
auxv cwd limits net root statm
cgroup environ loginuid ns sched status
clear_refs exe maps oom_adj schedstat syscall
cmdline fd mem oom_score sessionid task
comm fdinfo mountinfo oom_score_adj smaps wchan
chy@chy-VirtualBox:/proc/5443$
```

列出/proc/5443/cmdline 的内容如下：

```

chy@chy-VirtualBox: /proc/5443
13    1628    1993    234    5443    acpi      kpageflags    tty
14    1637    1999    235    579     asound     latency_stats  uptime
1441  1638     2      24     580     buddyinfo   loadavg        version
1442  1640    2002    25     5908     bus        locks          version_signature
1455  1641    2016    26     6       cgroups    mdstat         vmallocinfo
1456  1645    2017    27     64      cmdline    meminfo        vmstat
1467  1647    2019    28     655     consoles   misc           zoneinfo
1468  1652    2020    3      656     cpuinfo     modules
1476  1686    2021    36     7       crypto      mounts
1477  1690    2058    365    712     devices     mtrr
1480  1691    2065    368    8       diskstats   net
1483  1695    2066    39     811     dma         pagetypeinfo
1484  1697    2072    399    842     driver      partitions

chy@chy-VirtualBox:/proc$ cd 5443
chy@chy-VirtualBox:/proc/5443$ ls
attr          coredump_filter  io              mounts          pagemap         stack
autogroup     cpuset           latency         mountstats      personality     stat
auxv          cwd              limits          net             root           statm
cgroup        environ          loginuid        ns              sched           status
clear_refs    exe              maps            oom_adj         schedstat       syscall
cmdline       fd               mem             oom_score        sessionid       task
comm          fdinfo           mountinfo       oom_score_adj    smaps           wchan
chy@chy-VirtualBox:/proc/5443$ cat cmdline
bashchy@chy-VirtualBox:/proc/5443$

```

发现里面的内容仅有一行 bash(末尾没有换行)，这个文件保存了进程执行时的命令行。

列出/proc/5443/stat 的内容如下：

```

bashchy@chy-VirtualBox:/proc/5443$ cat stat
5443 (bash) S 5437 5443 5443 34817 7102 4202496 2989 35808 0 0 8 3 17 8 20 0 1 0
191567 7778304 1002 4294967295 134512640 135411844 3219454512 3219453288 307767
8116 0 65536 3686404 1266761467 3238387109 0 0 17 0 0 0 0 0

```

部分字段解释如下：

字段	含义
pid	进程 ID.
comm	task_struct 结构体的进程名
state	进程状态，此处为 S
ppid	父进程 ID（父进程是指通过 fork 方式，通过 clone 并非父进程）
pgrp	进程组 ID
session	进程会话组 ID
tty_nr	当前进程的 tty 终点设备号
tpgid	控制进程终端的前台进程号
flags	进程标识位，定义在 include/linux/sched.h 中的 PF_*
minflt	次要缺页中断的次数，即无需从磁盘加载内存页。比如 COW 和匿名页
cminflt	当前进程等待子进程的 minflt
majflt	主要缺页中断的次数，需要从磁盘加载内存页。比如 map 文件
majflt	当前进程等待子进程的 majflt
utime	该进程处于用户态的时间，单位 jiffies
stime	该进程处于内核态的时间，单位 jiffies
cutime	当前进程等待子进程的 utime

cstime	当前进程等待子进程的 utime
priorit	进程优先级
nice	nice 值，取值范围[19, -20]
num_threads	线程个数
itrealvalue	该字段已废弃，恒等于 0
starttime	自系统启动后的进程创建时间，单位 jiffies
vsize	进程的虚拟内存大小，单位为 bytes
rss	进程独占内存+共享库，单位 pages
rsslim	rss 大小上限

列出/proc/2129/status 的内容如下：

```

chy@chy-VirtualBox: /proc/2129
chy@chy-VirtualBox:/proc$ cd 2129
chy@chy-VirtualBox:/proc/2129$ cat status
Name:  bash
State:  S (sleeping)
Tgid:  2129
Pid:   2129
PPid:  2124
TracerPid:  0
Uid:   1000    1000    1000    1000
Gid:   1000    1000    1000    1000
FDSize: 256
Groups: 4 24 27 30 46 109 124 1000 1001
VmPeak:  7548 kB
VmSize:  7548 kB
VmLck:   0 kB
VmPin:   0 kB
VmHWM:   3908 kB
VmRSS:   3908 kB
VmData:  2220 kB
VmStk:   136 kB
VmExe:   880 kB
VmLib:   2136 kB
VmPTE:   28 kB
VmSwap:  0 kB
Threads: 1
SigQ:   0/15992
SigPnd: 0000000000000000
ShdPnd: 0000000000000000
SigBlk: 0000000000010000
SigIgn: 00000000000384004
SigCgt: 000000004b813efb
CapInh: 0000000000000000
CapPrm: 0000000000000000
CapEff: 0000000000000000
CapBnd: ffffffff
Cpus_allowed: 1
Cpus_allowed_list: 0
Mems_allowed: 1
Mems_allowed_list: 0
voluntary_ctxt_switches: 335
nonvoluntary_ctxt_switches: 6
chy@chy-VirtualBox:/proc/2129$ cd 2129

```

部分参数解释如下：

参数	解释
----	----

Name	应用程序或命令的名字
State	任务的状态，运行/睡眠/僵死/
Tgid	线程组号
Pid	任务 ID
PPid	父进程 ID
TracerPid	接收跟踪该进程信息的进程的 ID 号
Uid	Uid
Gid	Gid
FDSize	文件描述符的最大个数，file->fds
Groups	
VmPeak	
VmSize	任务虚拟地址空间的大小
VmLck	任务已经锁住的物理内存的大小。锁住的物理内存不能交换到硬盘
VmPin	
VmHWM	
VmRSS	应用程序正在使用的物理内存的大小，就是用 ps 命令的参数 rss 的值
VmData	程序数据段的大小（所占虚拟内存的大小），存放初始化了的数据；
VmStk	任务在用户态的栈的大小
VmExe(KB)	程序所拥有的可执行虚拟内存的大小，代码段，不包括任务使用的库
VmLib(KB)	被映像到任务的虚拟内存空间的库的大小
VmPTE	该进程的所有页表的大小，单位：kb
Threads	共享使用该信号描述符的任务的个数，在 POSIX 多线程程序应用程序中，线程组中的所有线程使用同一个信号描述符。
SigQ	待处理信号的个数
SigPnd	屏蔽位，存储了该线程的待处理信号
ShdPnd	屏蔽位，存储了该线程组的待处理信号
SigBlk	存放被阻塞的信号
SigIgn	存放被忽略的信号
SigCgt	存放被俘获到的信号
CapInh	Inheritable，能被当前进程执行的程序的继承的能力
CapPrm	Permitted，进程能够使用的能力，可以包含 CapEff 中没有的能力，这些能力是被进程自己临时放弃的，CapEff 是 CapPrm 的一个子集，进程放弃没有必要的能力有利于提高安全性
CapEff	Effective，进程的有效能力
CapBnd	
Cpus_allowed	
Cpus_allowed_list	
Mems_allowed	

Mems_allowed_list	
voluntary_ctxt_switches	进程主动切换的次数
nonvoluntary_ctxt_switches	表示进程被动切换的次数

4. 分别以 root 和普通用户身份运行一个进程，使用 nice、renice 和 top 设置或者改变其优先级，并验证是否成功。

答：经测试，使用 nice 设置静态优先级不论在 root 还是普通身份下都能执行成功，而使用 renice 设置 PID 为 3737 的进程(bash)的 nice 值如下，只在 root 权限(sudo)下执行成功：

```

chy@chy-VirtualBox: ~
chy@chy-VirtualBox:~$ pstree -p | less
chy@chy-VirtualBox:~$ renice 4 3734
renice: failed to set priority for 3734 (process ID): Permission denied
chy@chy-VirtualBox:~$ sudo renice 4 3734
[sudo] password for chy:
3734 (process ID) old priority 5, new priority 4
chy@chy-VirtualBox:~$ top

```

使用普通身份用 top 改变 bash 进程的 nice 值失败，截图如下：

```

chy@chy-VirtualBox: ~
top - 02:12:08 up 16 min, 3 users, load average: 0.58, 0.49, 0.34
Tasks: 150 total, 1 running, 145 sleeping, 4 stopped, 0 zombie
Cpu(s): 2.2%us, 1.0%sy, 0.0%ni, 96.8%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 2061540k total, 561400k used, 1500140k free, 56716k buffers
Swap: 1046524k total, 0k used, 1046524k free, 273860k cached
Renice of PID 3734 to 3 failed: Permission denied

```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1509	chy	9	-11	99652	5424	3944	S	1.5	0.3	0:03.98	pulseaudio
988	root	20	0	99.4m	52m	11m	S	0.9	2.6	0:08.81	Xorg
3043	chy	20	0	74032	15m	10m	S	0.5	0.8	0:01.99	gnome-terminal
404	messageb	20	0	3868	1724	892	S	0.3	0.1	0:01.18	dbus-daemon
1360	chy	20	0	17884	1292	896	S	0.3	0.1	0:02.32	VBoxClient
1037	root	20	0	29120	3356	2728	S	0.1	0.2	0:00.26	console-kit-dae
1367	chy	20	0	4540	2104	644	S	0.1	0.1	0:00.80	dbus-daemon
1381	chy	20	0	149m	15m	11m	S	0.1	0.7	0:00.88	gnome-settings-
1491	chy	20	0	128m	12m	10m	S	0.1	0.6	0:00.84	metacity
1508	chy	20	0	119m	24m	18m	S	0.1	1.2	0:00.74	unity-2d-panel
1512	chy	20	0	219m	51m	30m	S	0.1	2.5	0:01.01	unity-2d-shell
1531	chy	20	0	115m	22m	15m	S	0.1	1.1	0:00.54	nautilus
1777	chy	20	0	73256	15m	9.8m	S	0.1	0.8	0:00.57	unity-panel-ser
1	root	20	0	3672	1988	1296	S	0.0	0.1	0:00.40	init
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.04	ksoftirqd/0
6	root	RT	0	0	0	0	S	0.0	0.0	0:00.00	migration/0