

Procesos Linux ps

Comando ps

El comando ps, se utiliza para enumerar los procesos que se están ejecutando actualmente y sus PID en sistemas Linux y similares a Unix. Al ejecutarlo nos dará al menos dos procesos.

```
root@server02: ~# ps
  PID TTY          TIME CMD
 15530 pts/0        00:00:00 ps
 22082 pts/0        00:00:00 bash
```

Modificadores

Mostrar todos los procesos

Por defecto, `ps` muestra sólo los procesos que se ejecutaron desde su propia terminal (`xterm`, *acceso en modo texto o acceso remoto*). Las opciones `-A` y `-e` hacen que se muestren todos los procesos del sistema y `-x` muestra todos los procesos que pertenecen al usuario que proporciona el comando. La opción `-x` también incrementa la cantidad de información que se muestra sobre cada proceso.

ps -A

```
root@nms: ~# ps -A
  PID TTY          TIME CMD
    1 ?            00:00:34 systemd
    2 ?            00:00:00 kthreadd
    3 ?            00:00:00 rcu_gp
    4 ?            00:00:00 rcu_par_gp
    6 ?            00:00:00 kworker/0:0H-kblockd
    8 ?            00:00:00 mm_percpu_wq
    9 ?            00:01:48 ksoftirqd/0
   10 ?            00:01:23 rcu_sched
   11 ?            00:00:00 rcu_bh
   12 ?            00:00:08 migration/0 .....
```

Mostrar los procesos de un usuario

Puede mostrar los procesos que pertenecen a un usuario concreto con las opciones `-u usuario`, `U usuario`, y `--User usuario`. La variable `usuario` puede ser un nombre de usuario o el ID de usuario

ps -u

```
root@nms: ~# ps -u eduardo
```

Mostrar información adicional.

Las opciones `-f`, `-l`, `j`, `l`, `u` y `v` amplían la información proporcionada en la salida de `ps`. La mayoría de los formatos de `ps` incluye una línea por proceso, pero `ps` puede mostrar tanta información que es imposible adaptarla a una salida con líneas de 80 caracteres. Por este motivo, estas opciones ofrecen varias combinaciones para la salida.

ps -f

```
root@nms: ~# ps -f
UID      PID  PPID  C  STIME TTY          TIME CMD
root      3129  3088  0  08:00 pts/0      00:00:00 -bash
root      4931  3129  0  08:04 pts/0      00:00:00 ps -f
```

ps -l

```
root@nms: ~# ps -l
F S      UID    PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
4 S      0      3129  3088  0  80   0 -   1979 -          pts/0      00:00:00 bash
0 R      0      7803  3129  0  80   0 -   2637 -          pts/0      00:00:00 ps
```

ps j

```
root@nms: ~# ps j
PPID    PID  PGID    SID TTY          TPGID STAT   UID    TIME COMMAND
    1    400    400    400 tty1          400 Ss+      0      0:00 /sbin/agetty -o -p -- \u --noclear tty1
3088    3129  3129  3129 pts/0        7805 Ss       0      0:00 -bash
3129    7805  7805  3129 pts/0        7805 R+       0      0:00 ps j
```

ps l

```
root@nms: ~# ps l
F      UID    PID  PPID  PRI  NI       VSZ   RSS WCHAN  STAT TTY          TIME COMMAND
4        0    400      1   20    0   5612  1512 core_s Ss+  tty1      0:00 /sbin/agetty -o -p -- \u --
4        0   3129  3088   20    0   7916  4532 -      Ss   pts/0      0:00 -bash
0        0   7809  3129   20    0  10548  1272 -      R+   pts/0      0:00 ps l
```

ps u[[editar](#) | [editar código](#)]

```
root@nms: ~# ps u
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	400	0.0	0.0	5612	1512	tty1	Ss+	nov08	0:00	/sbin/agetty -o -p -- \u --nocl
root	3129	0.0	0.0	7916	4532	pts/0	Ss	08:00	0:00	-bash
root	7811	0.0	0.0	10632	3008	pts/0	R+	08:07	0:00	ps u

ps v

```
root@nms: ~# ps v
```

PID	TTY	STAT	TIME	MAJFL	TRS	DRS	RSS	%MEM	COMMAND
400	tty1	Ss+	0:00	2	42	5569	1512	0.0	/sbin/agetty -o -p -- \u --noclear tty1
3129	pts/0	Ss	0:00	0	873	7042	4532	0.0	-bash
7868	pts/0	R+	0:00	0	108	10439	1276	0.0	ps v

Mostrar la jerarquía de procesos

Las opciones `-H`, `-f` y `--forest` agrupan los procesos y emplean sangrías para mostrar la jerarquía de las relaciones entre los procesos. Estas opciones le serán útiles si intenta averiguar los parentescos de un proceso.

Mostrar la salida a todo el ancho

La salida del comando `ps` puede ser superior a las 80 columnas. Normalmente, `ps` trunca su salida para que se adapte a su pantalla o `xterm`. Las opciones `-w` y `W` le indican a `ps` que no haga esto, esto nos puede ser útil si dirigimos la salida a un fichero, como en `ps w > ps.txt`. Después, podemos examinar tranquilamente el fichero de salida en un editor de texto que admita líneas anchas.

Comandos útiles

Mostrar todos los procesos con el ID de proceso y el predecesor

ps -ef

```
root@nms: ~# ps -ef
```

UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	1	0	0	nov08	?	00:00:35	/sbin/init
root	2	0	0	nov08	?	00:00:00	[kthreadd]
root	3	2	0	nov08	?	00:00:00	[rcu_gp]
root	4	2	0	nov08	?	00:00:00	[rcu_par_gp]
root	6	2	0	nov08	?	00:00:00	[kworker/0:0H-kblockd]
root	8	2	0	nov08	?	00:00:00	[mm_percpu_wq]
root	9	2	0	nov08	?	00:01:48	[ksoftirqd/0]

Filtrar

El comando ps con la concatenación del comando grep, permite buscar un proceso

ps -ef | grep texto-a-buscar

```
root@nms: ~# ps -ef | grep php
root      381      1  0 nov08 ?        00:00:19 php-fpm: master process (/etc/php/7.3/fpm/php-fp
www-data 16057    381  0 nov11 ?        00:00:11 php-fpm: pool www
www-data 16292    381  0 nov11 ?        00:00:10 php-fpm: pool www
root      16533   3129  0 08:24 pts/0    00:00:00 grep php
www-data 18178    381  0 nov11 ?        00:00:08 php-fpm: pool www
```

Número de procesos

Con la concatenación del comando wc -l, permite contar los procesos

```
root@nms: ~# ps -ef | wc -l
201
```

Como en cualquier comando linux se permite la concatenación múltiple

```
root@nms: ~# ps -ef | grep php | wc -l
5
```

Revision #1

Created 17 May 2022 18:43:33 by Admin

Updated 23 May 2022 19:26:54 by Admin