

7.3 Control de trabajos

Objetivos

Tras finalizar esta sección, los estudiantes deberían poder realizar lo siguiente:

- Explicar los términos "primer plano", "segundo plano" y "terminal de control".
- Utilizar el control de trabajos para administrar múltiples tareas de la línea de comandos.

Control de trabajos

Control de trabajos es una característica de la shell que permite ejecutar y administrar múltiples comandos desde una sola instancia de shell.

Un *trabajo* está asociado con cada tubería ingresada en un shell prompt. Todos los procesos en esa tubería son parte del trabajo y son miembros del mismo *grupo de procesos*. (Si se ingresa solo un comando en un shell prompt, puede considerarse como una "tubería" mínima de un comando. Ese comando sería el único miembro de ese trabajo).

Solo un trabajo puede leer entradas y señales generadas por el teclado desde una ventana de terminal específica por vez. Los procesos que son parte de ese trabajo son procesos *en primer plano* de dicha *terminal de control*.

Un proceso *en segundo plano* de dicha terminal de control es un miembro de cualquier otro trabajo asociado con esa terminal. Los procesos en segundo plano de una terminal no pueden leer entradas ni recibir interrupciones generadas por el teclado desde la terminal, pero pueden escribir en la terminal. Un trabajo en segundo plano puede detenerse (suspenderse) o puede estar ejecutándose. Si un trabajo que se está ejecutando en segundo plano intenta leer desde la terminal, se suspenderá automáticamente.

Cada terminal es su propia *sesión*, y puede tener un proceso en primer plano y procesos en segundo plano independientes. Un trabajo es parte de exactamente una sesión, la que pertenece a su terminal de control.

El comando **ps** mostrará el nombre del dispositivo de la terminal de control de un proceso en la columna TTY. Algunos procesos, como *demonios del sistema*, son iniciados por el sistema y no desde un shell prompt. Estos procesos no tienen una terminal de control, no son miembros de un trabajo y no pueden colocarse en primer plano. El comando **ps** mostrará un signo de interrogación (?) en la columna TTY para estos procesos.

Realización de trabajos en segundo plano

Cualquier comando o tubería puede iniciarse en segundo plano si se anexa el signo ampersand (&) al final de la línea de comandos. La shell **bash** muestra un *número de trabajo* (exclusivo de la sesión) y el identificador de proceso del proceso secundario nuevo. La shell no espera al proceso secundario y vuelve a mostrar el shell prompt.

```
[student@serverX ~]$ sleep 10000 &
[1] 5947
[student@serverX ~]$
```

nota

Cuando se coloca una tubería en segundo plano con un signo ampersand, el identificador de proceso del último comando en la tubería será el de salida. Todos los procesos de la tubería continúan siendo miembros de ese trabajo.

```
[student@serverX ~]$ example_command | sort | mail -s "Sort
output" &
[1] 5998
```

La shell **bash** realiza un seguimiento de trabajos, por sesión, en una tabla que se muestra con el comando **jobs**.

```
[student@serverX ~]$ jobs
[1]+  Running          sleep 10000 &
[student@serverX ~]$
```

Un trabajo en segundo plano se puede colocar en primer plano con el comando **fg** con su ID de trabajo (*%número de trabajo*).

```
[student@serverX ~]$ fg %1
sleep 10000
```

—

En el ejemplo anterior, el comando **sleep** se está ejecutando en primer plano en la terminal de control. La shell se encuentra nuevamente en espera de que exista el proceso secundario.

Para enviar un proceso en primer plano a segundo plano, presione primero la solicitud de *suspensión* generada por el teclado (**Ctrl+z**) en la terminal.

```
sleep 10000
^Z
[1]+  Stopped          sleep 10000
[student@serverX ~]$
```

El trabajo se colocará inmediatamente en segundo plano y se suspenderá.

El comando **ps j** mostrará información relacionada con los trabajos. La PGID es el identificador de proceso del *líder del grupo de procesos*, generalmente el primer proceso en la tubería del trabajo. La SID es el identificador de proceso del *líder de sesión*, que para un trabajo es generalmente la shell interactiva que se está ejecutando en su terminal de control. Dado que el comando **sleep** de ejemplo está suspendido actualmente, su estado de proceso es T.

```
[student@serverX ~]$ ps j
PPID  PID  PGID  SID  TTY    TPGID  STAT  UID   TIME COMMAND
2764  2768  2768  2768 pts/0   6377  Ss   1000   0:00 /bin/bash
2768  5947  5947  2768 pts/0   6377  T    1000   0:00 sleep 10000
2768  6377  6377  2768 pts/0   6377  R+   1000   0:00 ps j
[student@serverX ~]$
```

Para iniciar el proceso suspendido que se está ejecutando en segundo plano, utilice el comando **bg** con la misma ID de trabajo.

```
[student@serverX ~]$ bg %1
[1]+  sleep 10000 &
[student@serverX ~]$
```

La shell emitirá una advertencia al usuario que intente salir de una ventana de terminal (sesión) con trabajos suspendidos. Si el usuario vuelve a intentar salir de inmediato, los trabajos suspendidos se anulan.

Referencias

Es posible encontrar información adicional en el capítulo sobre visualización de procesos de sistemas en la *Guía del administrador del sistema Red Hat Enterprise Linux* para Red Hat Enterprise Linux 7, que se puede encontrar en <https://access.redhat.com/documentation/>

Página de información de **bash** (*Manual de referencia de BASH para GNU*)

- Sección 7: Control de trabajos

Página de información **libc** (*Manual de referencias de la biblioteca GNU C*)

- Sección 24: Manejo de señales
- Sección 26: Procesos

páginas de manual `bash(1)`, `builtins(1)`, `ps(1)`, `sleep(1)`