

Acceso a la línea de comandos

RH124
Capítulo 1

Descripción general

Meta

Iniciar sesión en el sistema Linux y ejecutar comandos simples usando la shell.

Objetivos

- Utilizar la sintaxis de la shell Bash para ingresar comandos en una consola Linux.
- Iniciar aplicaciones en un entorno de escritorio GNOME.
- Utilizar funciones de Bash para ejecutar comandos desde un aviso de shell con menos pulsaciones de tecla.

Secciones

- Acceso a la línea de comandos a través de la consola local (y práctica)
- Acceso a la línea de comandos a través del escritorio (y práctica)
- Ejecución de comandos mediante la shell Bash (y práctica)

Trabajo de laboratorio

- Acceso a la línea de comandos

Acceso a la línea de comandos a través de la consola local

RH124

Acceso a la línea de comandos a través de la consola local

Objetivos

Tras finalizar esta sección, los estudiantes deberían poder iniciar sesión en un sistema Linux en una consola de texto local y ejecutar comandos simples a través de la shell.

La shell bash

Una *línea de comandos* es una interfaz basada en texto que puede utilizarse para introducir instrucciones en un sistema informático. Un programa denominado shell proporciona la línea de comandos de *Linux*. Durante la larga historia de los sistemas tipo UNIX, se han desarrollado muchos intérpretes de comandos. La shell predeterminada para los usuarios en Red Hat Enterprise Linux es **GNU Bourne-Again Shell (bash)**. **Bash** es una versión mejorada de uno de los shells más exitosos que se utiliza en los sistemas tipo UNIX: la **Bourne Shell (sh)**.

Si una shell se utiliza de manera interactiva, muestra una cadena cuando espera un comando del usuario. Esto se denomina *aviso de shell*. Cuando un usuario regular inicia una shell, el aviso predeterminado finaliza con un carácter **\$**.

```
[student@desktopX ~]$
```

El carácter **\$** reemplaza el carácter **#** si la shell se está ejecutando como el superusuario **root**. Con esto, resulta más evidente que se trata de una shell de superusuario, lo que permite evitar accidentes y errores en la cuenta con privilegios.

```
[root@desktopX ~]#
```

El uso de **bash** para ejecutar comandos puede ser eficaz. La shell **bash** proporciona un lenguaje de secuencia de comandos capaz de admitir la automatización de tareas. La shell tiene capacidades adicionales que pueden simplificar operaciones o posibilitar aquellas que son difíciles de realizar con herramientas gráficas.

nota

La shell **bash** es similar en concepto al intérprete de la línea de comandos disponible en versiones recientes de Microsoft Windows **cmd.exe**, pero **bash** posee un lenguaje de secuencia de comando más sofisticado. También es similar a Windows PowerShell en Windows 7 y Windows Server 2008 R2. A los administradores de Mac OS X que utilizan la utilidad **Terminal** de Macintosh les agradará saber que **bash** es la shell predeterminada en Mac OS X.

Consolas virtuales

Los usuarios acceden a la shell **bash** a través de una *terminal*. Un terminal proporciona un teclado para las entradas del usuario y una pantalla para las salidas. En instalaciones basadas en texto, esta puede ser la *consola física* del equipo Linux, el teclado de hardware y la pantalla. El acceso al terminal también puede configurarse a través de puertos en serie.

Otra forma de acceder a una shell es desde una *consola virtual*. La consola física de una máquina con Linux admite múltiples consolas virtuales que funcionan como terminales independientes. Cada consola virtual admite un inicio de sesión independiente.

Importante

En las imágenes virtuales preconfiguradas proporcionadas por Red Hat, se han deshabilitado los avisos de inicio de sesión en las consolas virtuales.

nota

En Red Hat Enterprise Linux 5 y en versiones anteriores, las primeras seis consolas virtuales proporcionaron siempre avisos de inicio de sesión de texto. Cuando se inició el entorno gráfico, se ejecutó en la consola virtual siete (a la que se accede a través **Ctrl+Alt+F7**).

Conceptos básicos de la shell

Los comandos ingresados en el aviso de shell están compuestos por tres partes básicas:

- *Comando* para ejecutar
- *Opciones* para ajustar el comportamiento del comando
- *Argumentos*, que generalmente son destinos del comando

El *comando* es el nombre del programa que se ejecuta. Puede estar seguido de una o más *opciones*, que ajustan el comportamiento del comando o lo que hará. Las opciones generalmente comienzan con uno o dos guiones (**-a** o **--all**, por ejemplo) para que se distingan de los argumentos. Los comandos también pueden estar seguidos de uno o más *argumentos*, que a menudo indican un objetivo en el cual el comando debe funcionar.

Es posible que la lectura de las declaraciones de uso sea una tarea complicada. Se tornan mucho más simples de comprender una vez que el usuario está familiarizado con algunas convenciones básicas:

- Los corchetes, **[]**, comprenden elementos opcionales.
- Todo lo que vaya seguido de . . . representa una lista con longitud arbitraria de elementos de ese tipo.
- Cuando hay múltiples elementos separados por tuberías, **|**, solo *uno* de ellos puede especificarse.
- El texto incluido entre corchetes angulares, **<>**, representa datos variables. Por ejemplo, **<filename>** significa "inserte aquí el nombre de archivo que desee usar". En ocasiones, estas variables simplemente se escriben con mayúsculas (por ejemplo, **FILENAME**).

Tenga en cuenta la primera declaración de uso para el comando **date**:

```
[student@desktopX ~]$ date --help
date [OPTION]... [+FORMAT]
```


nota

La página **man** para un comando tiene una sección SINOPSIS que proporciona información sobre la sintaxis del comando. La página de manual **man-pages(7)** describe cómo interpretar los corchetes, las barras verticales, etc. que los usuarios ven en la SINOPSIS o en un mensaje de uso.

Cuando un usuario termina de usar la shell y desea salir, la sesión puede finalizarse de distintas maneras. El comando **exit** finaliza la sesión de la shell actual. Otra forma de finalizar una sesión es presionando **Ctrl+d**.

Referencias

Páginas del manual **intro(1)**, **bash(1)**, **consola(4)**, **pts(4)** y **man-pages(7)**

*Nota: Algunos detalles de la página de manual de la consola(4) que incluyen **init(8)** e **inittab(5)** son obsoletos.*

Practica: Terminales de acceso a la consola local

Acceso a la línea de comandos con el escritorio

RH124

Acceso a la línea de comandos con el escritorio

Objetivos

Tras finalizar esta sección, los estudiantes deberían poder iniciar sesión en el sistema Linux usando el entorno de escritorio GNOME 3 para ejecutar comandos desde un aviso de shell en un programa de terminal.

Entorno de escritorio de GNOME

El *entorno de escritorio* es la interfaz de usuario gráfica en un sistema Linux. El entorno de escritorio predeterminado en Red Hat Enterprise Linux 7 se proporciona mediante **GNOME 3**. Este proporciona un escritorio integrado para usuarios y una plataforma de desarrollo unificada en la parte superior de una estructura gráfica proporcionada por el **Sistema X Window**.

Entorno de escritorio de GNOME

El *entorno de escritorio* es la interfaz de usuario gráfica en un sistema Linux. El entorno de escritorio predeterminado en Red Hat Enterprise Linux 7 se proporciona mediante **GNOME 3**. Este proporciona un escritorio integrado para usuarios y una plataforma de desarrollo unificada en la parte superior de una estructura gráfica proporcionada por el **Sistema X Window**.

GNOME Shell proporciona las funciones principales de la interfaz de usuario para el entorno de escritorio GNOME. La aplicación **gnome-shell** es muy personalizable. De forma predeterminada, los usuarios de RHEL 7 usan el tema "GNOME Classic" para **gnome-shell**, que es similar al entorno de escritorio GNOME 2. Otra opción disponible es el tema "moderno" GNOME 3 usado por el proyecto de base GNOME. Cualquier tema se puede seleccionar persistentemente en el inicio de sesión al seleccionar el ícono de engranaje junto al botón **Sign In** cuando se ingresa la contraseña del usuario.

Partes de la shell de GNOME

Las diversas partes de la shell de GNOME tienen nombres y propósitos específicos. Estas partes incluyen lo siguiente:

- *barra superior*: La barra que se ejecuta en toda la parte superior de la pantalla. La barra superior proporciona los menús **Applications** y **Places** y controles para el volumen, redes, acceso al calendario, y para seleccionar entre los métodos de entrada del teclado (si hay más de uno configurado). En el menú del nombre del usuario, hay opciones para ajustar los parámetros de la cuenta, bloquear la pantalla, cambiar usuarios, cerrar la sesión del sistema o apagarlo.
- *Menú Applications*: Este menú en la barra superior proporciona una manera de iniciar aplicaciones, categorizadas en submenús. La sección **Activities Overview** también se puede iniciar desde este menú.
- *Menú Places*: Este menú a la derecha del menú **Applications** proporciona un acceso rápido a través de un administrador gráfico de archivos a menús importantes en el directorio de inicio del usuario, a */*, y a exportaciones y archivos compartidos de la red.
- *lista de ventanas*: La barra que se ejecuta en toda la parte inferior de la pantalla. La lista de ventanas proporciona una manera fácil de acceder a todas las ventanas del espacio de trabajo actual, así como de minimizarlas y restaurarlas. En la esquina derecha, hay un indicador que informa al usuario en qué espacio de trabajo está y cuántos están disponibles.

- *bandeja de mensajes:* La bandeja de mensajes proporciona una manera de revisar notificaciones enviadas mediante aplicaciones o componentes del sistema a GNOME. Si ocurre una notificación, normalmente esta aparece primero brevemente como una sola línea en la parte inferior de la pantalla, y aparece un indicador persistente en la esquina inferior derecha para informar al usuario la cantidad de notificaciones recibidas recientemente. Se puede abrir la bandeja de mensajes para revisar estas notificaciones al hacer clic en el indicador o al presionar **Super+m**. La tecla Super (a veces, denominada tecla Windows) se encuentra cerca de la esquina inferior izquierda de un teclado de 104/105 teclas de una PC IBM. La bandeja de mensajes se puede cerrar al presionar **Esc** o **Super+m** nuevamente.
- *Descripción general de actividades:* Este es un modo especial que ayuda a organizar ventanas e inicia aplicaciones. La sección Activities Overview se puede iniciar al seleccionar **Applications > Activities Overview**. Las tres áreas principales de Activities Overview son el **guión** a la izquierda de la pantalla, la **descripción general de ventanas** en el centro de la pantalla y el **selector de espacios de trabajo** a la derecha de la pantalla. Para salir de la sección Activities Overview, se puede presionar la tecla **Esc**.
- *guión* Esta es una lista configurable de iconos de las aplicaciones favoritas del usuario, aplicaciones que se están ejecutando actualmente, y un botón de **cuadrícula** que se puede usar para seleccionar aplicaciones de forma arbitraria. Las aplicaciones se pueden iniciar haciendo clic en uno de los iconos o al usar el botón de cuadrícula para buscar una aplicación que se usa con menos frecuencia. Al guión también a veces se lo denomina **dock**.

Espacios de trabajo

Los espacios de trabajo son pantallas de escritorio por separado que tienen diferentes ventanas de aplicaciones. Se pueden utilizar para organizar el entorno de trabajo al agrupar las ventanas abiertas de la aplicación por tarea. Por ejemplo, las ventanas que se están usando para realizar una actividad de mantenimiento del sistema particular (como configurar un nuevo servidor remoto) se pueden agrupar en un espacio de trabajo, mientras que las aplicaciones de correo electrónico y otras aplicaciones de comunicación se pueden agrupar en otro espacio de trabajo.

nota

El uso de la combinación de teclas **Ctrl+Alt+UpArrow** o **Ctrl+Alt+DownArrow** para cambiar los espacios de trabajo no funciona en el entorno de aprendizaje virtual. En cambio, para cambiar de espacio de trabajo, se debe usar el applet del espacio de trabajo en el panel o **Activities Overview**.

Inicio de un terminal

Para obtener un aviso de shell en GNOME, inicie una aplicación de terminal gráfica como **GNOME Terminal**. Esto se puede realizar de varias maneras. A continuación, se detallan los tres métodos más comúnmente usados:

- Seleccione **Applications > Utilities > Terminal**.
- Haga clic derecho en un escritorio vacío o presione la tecla **Menú** y seleccione **Open in Terminal** en el menú de contexto que aparece.
- Desde la sección **Activities Overview**, seleccione **Terminal** desde el **guión** (ya sea desde el área de favoritos o al buscarla con el botón de **cuadrícula** [dentro de la agrupación **Utilities**] o el campo de búsqueda en la parte superior de la **descripción general de ventanas**).

Cuando se abre una ventana de terminal, se muestra un aviso de shell para el usuario que inició el programa de terminal gráfica. El aviso de shell y la barra de títulos de la ventana de terminal indicarán el nombre de usuario actual, el nombre del host y el directorio de trabajo.

Bloqueo de la ventana o cierre de sesión

El bloqueo de la ventana, o el cierre de sesión por completo, se puede realizar desde el menú del nombre del usuario bien a la derecha de la barra superior.

Para bloquear la ventana, seleccione **(User) > Lock** o presione **Ctrl+Alt+L**. La pantalla se bloqueará si la sesión gráfica está inactiva durante unos minutos.

Apagar o reiniciar el sistema

Para apagar el sistema, seleccione **(User) > Power Off** de la barra superior o presione **Ctrl+Alt+Del**. En el diálogo que aparece, el usuario puede elegir entre **Power Off** para apagar, **Restart** para reiniciar la máquina, o **Cancel** para cancelar la operación. Si no elige nada en este cuadro de diálogo, el sistema se apagará automáticamente después de 60 segundos.

Práctica: Entorno de escritorio GNOME 3

Ejecución de comandos con la shell Bash

RH124



Objetivos

Tras finalizar esta sección, los estudiantes deberían poder ahorrar tiempo en la ejecución de comandos a partir de un aviso de shell con los accesos directos de Bash.

Sintaxis básica de comandos

GNU Bourne-Again Shell (**bash**) es un programa que interpreta comandos escritos por el usuario. Cada secuencia escrita en la shell puede tener un máximo de tres partes: el comando, las opciones (que comienzan con un - o --) y los argumentos. Cada palabra escrita en la shell está separada de las otras por espacios. Los comandos son nombres de programas que están instalados en el sistema. Cada comando tiene sus propias opciones y argumentos.

Cuando el usuario esté listo para ejecutar un comando, presiona la tecla **Enter**. Cada comando se escribe en una línea separada y el resultado de cada comando se muestra antes de que la shell muestre un aviso. Si un usuario quiere escribir más de un comando en una sola línea, puede usarse un punto y coma ; como separador de comando. Un punto y coma forma parte de una clase de caracteres denominados *metacaracteres*, que tienen significados especiales para **bash**.

Ejemplos de comandos simples

El comando **date** se usa para mostrar la fecha y hora actuales. Además, puede ser usado por el superusuario para configurar el reloj del sistema. Un argumento que comienza con el signo más (+) especifica una secuencia de formato para el comando de fecha.

```
[student@desktopX ~]$ date
Sat Apr  5 08:13:50 PDT 2014
[student@desktopX ~]$ date +%R
08:13
[student@desktopX ~]$ date +%x
04/05/2014
```

El comando **passwd** cambia la contraseña propia del usuario. La contraseña original de la cuenta debe especificarse antes de que se permita un cambio. De manera predeterminada, **passwd** se configura para solicitar una contraseña más sólida, que esté compuesta por letras minúsculas, letras mayúsculas, números y símbolos, y que se base en una palabra del diccionario. El superusuario puede usar el comando **passwd** para cambiar las contraseñas de otros usuarios.

```
[student@desktopX ~]$ passwd
Changing password for user student.
Changing password for student.
(current) UNIX password: old_password
```

Linux no requiere de extensiones de nombre de archivo para clasificar los archivos por tipo. El comando **file** detecta el comienzo del contenido de un archivo y muestra qué tipo de archivo es. Los archivos que se clasificarán pasan como argumentos para el comando.

```
[student@desktopX ~]$ file /etc/passwd
/etc/passwd: ASCII text
[student@desktopX ~]$ file /bin/passwd
/bin/passwd: setuid ELF 64-bit LSB shared object, x86-64, version 1
(SYSV), dynamically linked (uses shared libs), for GNU/Linux 2.6.32,
BuildID[sha1]=0x91a7160a019b7f5f754264d920e257522c5bce67, stripped
[student@desktopX ~]$ file /home
/home: directory
```

Los comandos **head** y **tail** muestran el comienzo y el final de un archivo, respectivamente. De manera predeterminada, estos comandos muestran 10 líneas, pero ambos tienen la opción **-n** que permite la especificación de una cantidad diferente de líneas. El archivo que se mostrará pasa como un argumento para estos comandos.

```
[student@desktopX ~]$ head /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
[student@desktopX ~]$ tail -n 3 /etc/passwd
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:993:991::/run/gnome-initial-setup:/sbin/nologin
tcpdump:x:72:72::/sbin/nologin
```

El comando **wc** cuenta líneas, palabras y caracteres en un archivo. Puede usar la opción **-l**, **-w** o **-c** para mostrar solo las líneas, las palabras o los caracteres, respectivamente.

```
[student@desktopX ~]$ wc /etc/passwd
 39   70 2005 /etc/passwd
[student@desktopX ~]$ wc -l /etc/passwd ; wc -l /etc/group
 39 /etc/passwd
 63 /etc/group
[student@desktopX ~]$ wc -c /etc/group /etc/hosts
 843 /etc/group
 227 /etc/hosts
1070 total
```


Completar con el tabulador

Completar con el tabulador permite al usuario completar comandos o nombres de archivos rápidamente una vez que haya escrito lo suficiente en el aviso como para hacerlo único. Si los caracteres escritos no son únicos, al presionar la tecla **Tab** dos veces, aparecen todos los comandos que comienzan con los caracteres ya escritos.

```
[student@desktopX ~]$ pas<Tab><Tab>
passwd      paste      pasuspender
[student@desktopX ~]$ pass<Tab>
[student@desktopX ~]$ passwd
Changing password for user student.
Changing password for student.
(current) UNIX password:
```

La opción de completar con el tabulador puede usarse para completar nombres de archivo cuando se escriben como argumentos para comandos. Si se presiona la tecla **Tab**, completará el nombre del archivo tanto como pueda. Si se presiona la tecla **Tab** por segunda vez, provoca que la shell enumere todos los archivos que coinciden con el patrón actual. Escriba caracteres adicionales hasta que el nombre sea único; a continuación, use la opción de completar con el tabulador para finalizar la línea de comandos.

```
[student@desktopX ~]$ ls /etc/pas<Tab>
[student@desktopX ~]$ ls /etc/passwd<Tab>
passwd  passwd-
```

Con esta opción, se puede establecer una coincidencia entre argumentos y opciones para muchos comandos. El comando **useradd** es usado por el superusuario, el usuario **root**, para crear otros usuarios en el sistema. Tiene muchas opciones que pueden usarse para controlar el comportamiento del comando. Puede usarse la opción de completar con el tabulador después de una opción parcial para completar la opción sin necesidad de escribir mucho.

```
[root@desktopX ~]# useradd --<Tab><Tab>
--base-dir      --groups        --no-log-init   --shell
--comment       --help          --non-unique    --skel
--create-home   --home-dir      --no-user-group --system
--defaults      --inactive      --password      --uid
--expiredate    --key           --root          --user-group
--gid           --no-create-home --selinux-user
[root@desktopX ~]# useradd --
```

Comando history

El comando **history** muestra una lista de los comandos ejecutados anteriormente que tienen un número de comando como prefijo.

El signo de exclamación, **!**, es un metacaracter utilizado para ampliar comandos anteriores sin tener que escribirlos nuevamente. **!number** se amplía al comando que coincide con el número especificado. **!string** se amplía al comando más reciente que comienza con la cadena especificada.

```
[student@desktopX ~]$ history
```

```
...Output omitted...
```

```
23 clear
24 who
25 pwd
26 ls /etc
27 uptime
28 ls -l
29 date
30 history
```

```
[student@desktopX ~]$ !ls
```

```
ls -l
```

```
total 0
```

```
drwxr-xr-x. 2 student student 6 Mar 29 21:16 Desktop
```

```
...Output omitted...
```

```
[student@desktopX ~]$ !26
```

```
ls /etc
```

abrt	hosts	pulse
adjtime	hosts.allow	purple
aliases	hosts.deny	qemu-ga

```
...Output omitted...
```

Edición de línea de comandos

Cuando se usa en forma interactiva, **bash** tiene una función de edición de línea de comandos. Esto permite al usuario utilizar los comandos del editor de texto para desplazarse y modificar el comando actual que se está escribiendo. El uso de las teclas de flecha para moverse dentro del comando actual y pasar por el historial de comando se presentó anteriormente en esta sesión. En la siguiente tabla, se presentan comandos de edición más contundentes.

Accesos directos para la edición de línea de comandos

Acceso directo	Descripción
Ctrl+a	Ir al inicio de la línea de comandos.
Ctrl+e	Ir al final de la línea de comandos.
Ctrl+u	Borrar desde el cursor hasta el principio de la línea de comandos.
Ctrl+k	Borrar desde el cursor hasta el final de la línea de comandos.
Ctrl+Left Arrow	Ir al inicio de la palabra anterior en la línea de comandos.
Ctrl+Right Arrow	Ir al final de la palabra siguiente en la línea de comandos.
Ctrl+r	Buscar un patrón en la lista de historial de comandos.

Hay muchos otros comandos de edición de línea de comandos disponibles, pero estos son los más prácticos para usuarios principiantes. Los otros comandos están en la página de manual **bash(1)**.

Referencias

Páginas del manual: **bash(1)**, **date(1)**, **file(1)**, **head(1)**, **passwd(1)**, **tail(1)** y **wc(1)**

Práctica: comandos bash y atajos del teclado