

1.1 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.

Acceder a la línea de comando con consolas virtuales

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 intérpretes de comandos más exitoso que se utiliza en los sistemas tipo UNIX: la shell Bourne (**sh**).

Si una shell se utiliza de manera interactiva, muestra una cadena cuando espera un comando del usuario. que recibe el nombre de *shell prompt*. 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 tareas 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. Los administradores de Mac OS X que utilizan la utilidad Terminal de Macintosh estarán encantados al observar que **bash** es la shell predeterminada en Mac OS X.

Consolas virtuales

Los usuarios acceden a la shell **bash** a través de una *terminal*. Una terminal proporciona un teclado para las entradas del usuario y una pantalla para las salidas. En instalaciones basadas en texto, la terminal puede ser la *consola física*, el teclado y la pantalla de la máquina con Linux. El acceso a la terminal puede también 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.

Si el entorno gráfico se encuentra activado, funcionará en la *primera* consola virtual en Red Hat Enterprise Linux 7. Cinco avisos de acceso a sesión de texto adicionales están disponibles en las consolas dos a la seis (o de la consola uno a la cinco si el entorno gráfico está desactivado). Cuando se esté ejecutando un entorno gráfico, presione **Ctrl+Alt** y presione una tecla de función (de **F2** a **F6**) para acceder a un aviso de acceso a sesión de texto en una consola virtual. Presione **Ctrl+Alt+F1** para regresar a la primera consola virtual y al escritorio gráfico.

Importante

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

nota

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

Conceptos básicos de la shell

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

- *Comando* que se ejecuta
- *Opciones* que permiten 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 pueden estar seguidos por uno o más *argumentos*, que generalmente indican un destino sobre el cual el comando debe operar.

Por ejemplo, la línea de comandos **usermod -L morgan** cuenta con un comando (**usermod**), una opción (**-L**) y un argumento (**morgan**). El efecto de este comando es bloquear la contraseña de la cuenta del usuario morgan.

Para usar un comando de manera eficiente, el usuario debe conocer las opciones y los argumentos que acepta, así como el orden en el que espera que se introduzcan (la *sintaxis* del comando). La mayoría de los comandos tienen una opción **--help**. Esto hace que el comando imprima una descripción de su función, es decir, una "declaración de uso" que detalla la sintaxis del comando, su función y una lista de las opciones que acepta.

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". 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]
```

Indica que **date** puede aceptar una lista opcional de opciones (**[OPTION]...**), seguida de una cadena de formato opcional y precedida por el signo "más", **+**, que describe cómo debe mostrarse la fecha actual (**[+FORMAT]**). Puesto que ambas elecciones

son opcionales, **date** funcionará aunque no se hayan especificado las opciones o los argumentos (imprimirá la fecha y hora actuales con su formato predeterminado).

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. para que los usuarios vean 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 de manual **intro(1)**, **bash(1)**, **console(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.*

Next