

LINUX

Para profesionales



**Más de 50 páginas
de consejos y trucos**

Contenido

Acerca de	1
Capítulo1: Introducción a GNU/Linux	2
Sección1.1: Atajos útiles	2
Sección1.2: Comandos de gestión de archivos	3
Sección1.3: Hola Mundo	5
Sección1.4: Utilidades básicas de Linux	5
Sección1.5: Búsqueda de archivos por patrones en el nombre/contenido	6
Sección1.6: Manipulación de archivos	7
Sección1.7: Detalles del archivo/directorio	8
Capítulo2: Detectar el nombre y la versión de la distribución Linux	11
Sección2.1: Detecte en qué distribución basada en Debian está trabajando	11
Sección2.2: Detecte qué distribución basada en systemd está utilizando	11
Sección2.3: Detecte en qué distribución RHEL / CentOS / Fedora está trabajando	12
Sección2.4: Uname - Imprime información sobre el sistema actual	13
Sección2.5: Detectar información básica sobre su distribución	13
Sección2.6: Uso de GNU coreutils	13
Sección2.7: Encuentre el nombre y el número de versión de su sistema operativo Linux (tanto debian como rpm)	14
Capítulo3: Obtener información sobre un kernel Linux en ejecución	15
Sección3.1: Obtener detalles del núcleo Linux	15
Capítulo4: Shell	16
Sección4.1: Cambiar el shell por defecto	16
Sección4.2: Utilidades básicas de Shell	17
Sección4.3: Cree su propio alias de comando	18
Sección4.4: Localizar un archivo en su sistema	18
Capítulo5: Comprobar el espacio en disco	19
Sección5.1: Investigar Directorios para Uso de Disco	19
Sección5.2: Comprobación del espacio en disco	21
Capítulo6: Obtener información del sistema	23
Sección6.1: Estadísticas sobre CPU, Memoria, Red y Disco (operaciones de E/S)	23
Sección6.2: Uso de herramientas como lscpu y lshw	23
Sección6.3: Lista Hardware	24
Sección6.4: Buscar información sobre el modelo/velocidad de la CPU	25
Sección6.5: Seguimiento del proceso y recopilación de información	26
Capítulo7: comando ls	28
Sección7.1: Opciones para el comando ls	28
Sección7.2: comando ls con las opciones más utilizadas	28
Capítulo8: Compresión de archivos con el comando 'tar'	30
Sección8.1: Comprimir una carpeta	30
Sección8.2: Extraer una carpeta de un archivo	30
Sección8.3: Listar el contenido de un archivo	30
Sección8.4: Listar el contenido del archivo	31
Sección8.5: Comprimir y excluir una o varias carpetas	31
Sección8.6: Componentes principales de la banda	31
Capítulo9: Servicios	32
Sección9.1: Lista de servicios en ejecución en Ubuntu	32
Sección9.2: Gestión de servicios Systemd	32
Capítulo10: Gestión de servicios	33
Sección10.1: Diagnosticar un problema con un servicio	33
Sección10.2: Inicio y parada de servicios	33
Sección10.3: Obtener el estado de un servicio	34

Capítulo11: Modificación de usuarios.....	35
<u>Sección11.1: Establecer su propia contraseña</u>	35
<u>Sección11.2: Establecer la contraseña de otro</u>	35
<u>Sección11.3: Añadir un usuario</u>	35
<u>Sección11.4: Eliminar un usuario.....</u>	35
<u>Sección11.5: Eliminar un usuario y su carpeta de inicio.....</u>	35
<u>Sección11.6: Listado de grupos a los que pertenece el usuario actual</u>	35
<u>Sección11.7: Listado de grupos a los que pertenece un usuario.....</u>	35
Capítulo12: Pila LAMP	36
<u>Sección12.1: Instalación de LAMP en Arch Linux</u>	36
<u>Sección12.2: Instalación de LAMP en Ubuntu.....</u>	37
<u>Sección12.3: Instalación de la pila LAMP en CentoOS</u>	38
Capítulo13: comando tee.....	40
<u>Sección13.1: Escribir la salida a stdout, y también a un archivo.....</u>	40
<u>Sección13.2: Escribir la salida desde la mitad de una cadena de tuberías a un archivo y pasarlo de nuevo a la tubería.....</u>	40
<u>Sección13.3: escribir la salida en varios archivos</u>	40
<u>Sección13.4: Indique al comando tee que añada al archivo.....</u>	40
Capítulo14: Secure Shell (SSH)	42
<u>Sección14.1: Conexión a un servidor remoto</u>	42
<u>Sección14.2: Instalación del paquete OpenSSH</u>	42
<u>Sección14.3: Configuración de un servidor SSH para aceptar conexiones</u>	43
<u>Sección14.4: Conexión sin contraseña (utilizando un par de claves)</u>	43
<u>Sección14.5: Generar clave pública y privada.....</u>	43
<u>Sección14.6: Desactivar el servicio ssh.....</u>	43
Capítulo15: SCP	45
<u>Sección15.1: Copia segura.....</u>	45
<u>Sección15.2: Uso básico.....</u>	45
Capítulo16: GnuPG (GPG).....	46
<u>Sección16.1: Exportar su clave pública.....</u>	46
<u>Sección16.2: Crear y utilizar una clave GnuPG rápidamente.....</u>	46
Capítulo17: Configuración de la red.....	47
<u>Sección17.1: Resolución DNS local</u>	47
<u>Sección17.2: Configurar servidores DNS para la resolución de nombres de dominio</u>	47
<u>Sección17.3: Ver y manipular rutas.....</u>	47
<u>Sección17.4: Configurar un nombre de host para otro sistema de la red.....</u>	48
<u>Sección17.5: Detalles de la interfaz</u>	49
<u>Sección17.6: Añadir IP a una interfaz</u>	50
Capítulo18: Comandante de medianoche	52
<u>Sección18.1: Teclas de función del Midnight Commander en modo navegación</u>	52
<u>Sección 18.2: Teclas de función de Midnight Commander en el modo de edición de archivos</u>	52
Capítulo19: Cambiar de raíz (chroot)	54
<u>Sección19.1: Requisitos</u>	54
<u>Sección19.2: Cambiar manualmente la raíz de un directorio</u>	54
<u>Sección19.3: Razones para usar chroot.....</u>	55
Capítulo20: Gestores de paquetes.....	56
<u>Sección20.1: Cómo actualizar paquetes con el gestor de paquetes apt</u>	56
<u>Sección20.2: Cómo instalar un paquete con el gestor de paquetes pacman.....</u>	56
<u>Sección20.3: Cómo actualizar paquetes con el gestor de paquetes pacman</u>	56
<u>Sección20.4: Cómo actualizar paquetes con yum</u>	57
Capítulo21: Compilación del núcleo Linux.....	58
<u>Sección21.1: Compilación del Kernel Linux en Ubuntu</u>	58
Créditos.....	59
También le puede interesar.....	61

Acerca de

Por favor, siéntase libre de compartir este PDF con cualquier persona de forma gratuita, la última versión de este libro se puede descargar desde:

<https://goalkicker.com/LinuxBook>

Este libro de *Notas de Linux® para Profesionales* está compilado de la Documentación de Stack Overflow, el contenido está escrito por la hermosa gente de Stack Overflow. El contenido del texto está liberado bajo Creative Commons BY-SA, ver los créditos al final de este libro quienes contribuyeron a los distintos capítulos. Las imágenes pueden ser copyright de sus respectivos propietarios a menos que se especifique lo contrario.

Este es un libro gratuito no oficial creado con fines educativos y no está afiliado a grupo(s) o compañía(s) oficial(es) de Linux® ni a Stack Overflow. Todas las marcas comerciales y marcas registradas son propiedad de sus respectivos propietarios.

No se garantiza que la información presentada en este libro sea correcta ni exacta. Utilícelo bajo su propia responsabilidad.

Envíe sus comentarios y correcciones web@petercv.com

Capítulo 1: Introducción a GNU/Linux

Sección 1.1: Atajos útiles

Uso del terminal

Los ejemplos de este documento asumen que está utilizando un shell compatible con POSIX (como `bash`, `sh`, `zsh`, `ksh`).

Gran parte de la funcionalidad de GNU/Linux se consigue utilizando el terminal. La mayoría de las distribuciones de Linux incluyen emuladores de terminal que permiten a los usuarios interactuar con una shell desde su entorno de escritorio. Un shell es un intérprete de línea de comandos que ejecuta los comandos introducidos por el usuario. `Bash` (Bourne Again SHell) es un intérprete de comandos común por defecto entre muchas distribuciones de Linux y es el intérprete de comandos por defecto de macOS.

Estos atajos funcionarán si estás usando `Bash` con las combinaciones de teclas *de emacs* (configuradas por defecto):

Terminal abierto

- `Ctrl+ Alt+ T` o `Super+ T`

Movimiento del cursor

- `Ctrl+ A` Ir al principio de la línea en la que está escribiendo. Ir al final de la línea en la que está escribiendo.
- `Ctrl+ E` Ir al final de la línea en la que está escribiendo.
- `Ctrl+ X` Desplazarse entre el principio de la línea y la posición actual del cursor.
- `Alt+ F` Avanzar el cursor una palabra en la línea actual. Mover el cursor hacia atrás una palabra en la línea actual.
- `Alt+ B` Avanzar el cursor una palabra en la línea actual. Mover el cursor hacia atrás una palabra en la línea actual.
- `Ctrl+ F` Avanzar el cursor un carácter en la línea actual. Mover el cursor hacia atrás un carácter en la línea actual.
- `Ctrl+ B` Avanzar el cursor un carácter en la línea actual. Mover el cursor hacia atrás un carácter en la línea actual.

Manipulación de textos

- `Ctrl+ U` Corta la línea desde la posición actual hasta el principio de la línea, añadiéndola al portapapeles. Si está al final de la línea, corta toda la línea.
- `Ctrl+ K` Corta la línea desde la posición actual hasta el final de la línea, añadiéndola al portapapeles. Si se encuentra al principio de la línea, córtela entera.
- `Ctrl+ W` Borra la palabra situada antes del cursor, añadiéndola al portapapeles.
- `Ctrl+ Y` Pega lo último del portapapeles que haya cortado recientemente (deshacer el último borrado en el botón posición actual del cursor).
- `Alt+ T` Swap las dos últimas palabras antes del cursor.
- `Alt+ L` Make minúsculas desde el cursor hasta el final de la palabra.
- `Alt+ U` Make mayúsculas desde el cursor hasta el final de la palabra.
- `Alt+ C` Capitaliza hasta el final de la palabra que empieza en el cursor (palabra entera si el cursor está al principio de la palabra).
- `Alt+ D` Delete hasta el final de la palabra que empieza en el cursor (palabra entera si el cursor está al principio de la palabra).
- `Alt+ .` Precede la última palabra escrita en el comando anterior.
- `Ctrl+ T` Intercambia los dos últimos caracteres antes del cursor.

Acceso a la historia

- `Ctrl+ R` Le permite buscar entre los comandos utilizados anteriormente.
- `Ctrl+ G` Abandona el modo de búsqueda en el historial sin ejecutar ningún comando.
- `Ctrl+ J` Le permite copiar el comando coincidente actual en la línea de comandos sin , lo que le permite
-

hacer modificaciones antes de ejecutar el comando.

- **Alt+ R** Revierte cualquier cambio en un comando que hayas sacado del historial, si lo has editado. Muestra el último comando ejecutado, es decir, retrocede en el historial de comandos (Similar a up flecha).
- **Ctrl+ P** Muestra el siguiente comando ejecutado, es decir, avanza por el historial de comandos (similar a flecha hacia abajo).

Control de terminales

Borra la pantalla, de forma similar al clear.

Detiene toda la salida pantalla. Esto es útil cuando se ejecutan comandos con mucha salida larga.

- **Ctrl+ L**
 - **Ctrl+ S**
- Pero esto no detiene el comando en ejecución.
- **Ctrl+ Q** Reanuda la salida a la pantalla después de detenerla con Ctrl+S.
 - **Ctrl+ C** Finaliza el proceso en ejecución y devuelve el prompt.
 - **Ctrl+ D** Cierra la sesión actual del shell, de forma similar al comando exit o logout. En algunos comandos, actúa como señal de fin de archivo para indicar que se ha alcanzado el final de un archivo.
 - **Ctrl+ Z** Suspende (pausa) el proceso que se está ejecutando en primer plano, lo que devuelve el prompt del shell. A continuación, puede utilizar `bg` para que el proceso se ejecute en segundo plano. Para volver a poner ese proceso en primer plano, utilice el comando `fg`. Para ver todos los procesos en segundo plano, utilice el comando `jobs`.
 - **Pest** Autocompletar nombres de archivos y directorios.
 - **Tab** Muestra todas las posibilidades, cuando los caracteres tecleados no coinciden unívocamente con un nombre de archivo o directorio.

Caracteres especiales

- **Ctrl+ H** Igual que Retroceso.
- **Ctrl + J** Igual que Return (históricamente Line Feed).
- **Ctrl + M** Igual que Retorno (históricamente Retorno de carro).
- **Ctrl + I** Igual que Tab.
- **Ctrl + G** Carácter de campana.
- **Ctrl + @** Carácter Nulo.
- **Esc** Tecla muerta equivalente a la **Alt** modificador.

Cerrar Terminal

- **Ctrl+ May ús+ W** Para cerrar la pestaña del terminal.
- **Ctrl+ May ús+ Q** Para cerrar todo el terminal.

Alternativamente, puedes cambiar a las combinaciones de teclas *de vi* en bash usando `set -o vi`. Utiliza `set -o emacs` para volver a la combinación de teclas *de emacs*.

Sección 1.2: Comandos de gestión de archivos

Linux utiliza algunas convenciones para los directorios presentes y padres. Esto puede ser un poco confuso para los principiantes.

Siempre que estés en una terminal en Linux, estarás en lo que se llama el *directorio de trabajo actual*. A menudo su símbolo del sistema mostrará el directorio de trabajo completo, o sólo la última parte de ese. Su prompt podría verse como uno de los siguientes:

```
usuario@host~ /somedir $
usuario@host somedir $
usuario@host $
```

que dice que tu directorio de trabajo actual es `/home/usuario/somedir`.

En Linux `..` representa el directorio padre y `.` representa el directorio actual.

Por lo tanto, si el directorio actual es `/home/usuario/somedir`, entonces `cd ../somedir` no cambiará el directorio de trabajo.

En la siguiente tabla se enumeran algunos de los comandos de gestión de archivos más utilizados

Navegación por el directorio

Comando	Utilidad
<code>pwd</code>	Obtener la ruta completa del directorio de trabajo actual. <code>cd -</code> Navega al último directorio en el que estabas trabajando. <code>cd ~</code> o simplemente <code>cd</code> Navega al directorio personal del usuario actual.
<code>cd ..</code>	Ir al directorio padre del directorio actual (tenga en cuenta el espacio entre <code>cd</code> y <code>..</code>)

Listado de archivos dentro de un directorio

Comando	Utilidad
<code>ls -l</code>	Lista los archivos y directorios en el directorio actual en formato largo (tabla) (Se recomienda utilizar <code>-l</code> con <code>ls</code> para una mejor legibilidad).
<code>ls -ld dir-name</code>	Listar información sobre el directorio <code>dir-name</code> en lugar de su contenido.
<code>ls -a</code>	Lista todos los archivos, incluidos los ocultos (los nombres de archivo que empiezan por <code>.</code> son archivos ocultos en Linux).
<code>ls -F</code>	Añade un símbolo al final del nombre de un archivo para indicar su tipo (<code>*</code> significa ejecutable, <code>/</code> significa directorio, <code>@</code> significa enlace simbólico, <code>=</code> significa socket, <code> </code> significa tubería con nombre, <code>></code> significa puerta).
<code>ls -lt</code>	Lista los archivos ordenados por la última hora de modificación, mostrando los archivos modificados más recientemente en la parte superior (recuerde que la opción <code>-l</code> proporciona el formato largo, que es más legible).
<code>ls -lh</code>	Lista el tamaño de los archivos en formato legible.
<code>ls -lR</code>	Muestra todos los subdirectorios de forma recursiva.
<code>árbol</code>	Generará una representación en árbol del sistema de archivos a partir del directorio actual.

Creación, copia y eliminación de archivos/directorios

Comando	Utilidad
<code>cp -p origen destino</code>	Copiará el archivo del origen al destino. <code>-p</code> significa preservación. Se conserva los atributos originales del archivo durante la copia, como el propietario, la fecha y hora, el grupo, los permisos, etc.
<code>cp -R directorio_de_origen directorio_de_destino</code>	Copiará el directorio de origen en el destino especificado de forma recursiva.
<code>mv fichero1 fichero2</code>	En Linux no existe el comando renombrar como tal. Por lo tanto <code>mv</code> mueve/renombrará el archivo1 a archivo2.
<code>rm -i nombearchivo</code>	Le pide confirmación antes de cada eliminación de archivos. SI ES UN USUARIO NUEVO EN LA LÍNEA DE COMANDOS DE LINUX, SIEMPRE DEBE UTILIZAR <code>rm -i</code> . Puede especificar varios archivos.
<code>rm -R nombre-dirección</code>	Eliminará el directorio <code>dir-name</code> recursivamente.
<code>rm -rf nombre-dirección</code>	Eliminará el directorio <code>dir</code> de forma recursiva, ignorando los archivos no existentes y se nunca pide nada. ¡TENGA CUIDADO AL USAR ESTE ! Puede especificar varios directorios.
<code>rmdir nombre-dirección</code>	Eliminará el directorio <code>dir-name</code> , si está vacío. Este comando sólo puede eliminar directorios vacíos.
<code>mkdir nombre-dirección</code>	Crea un directorio <code>dir-name</code> .
<code>mkdir -p dir-nombre /dir-nombre</code>	Crea una jerarquía de directorios. Crea directorios padre según sea necesario, si no lo hacen existir. Puede especificar varios directorios.
<code>touch nombearchivo</code>	Crea un archivo <code>filename</code> , si no existe, de lo contrario cambia la marca de tiempo del archivo a la hora actual.

Permisos y grupos de archivos/directorios

Comando	Utilidad
---------	----------

chmod < especificaci ón> nombre de archivo	Cambia los permisos del archivo. Especificaciones= u user, g group, o other,+add permiso, - eliminar, r leer, w escribir,x ejecutar.
chmod -R < especificaci ón> dir-nombre	Cambia los permisos de un directorio de forma recursiva. Para cambiar los permisos de un directorio y de todo lo que contiene, utilice este comando.
chmod go+=r miarchivo	Añadir permiso de lectura para el propietario y el grupo.
chmod a +rwx miarchivo	Permitir a todos los usuarios leer, escribir o ejecutar
miarchivo . chmod go -r miarchivo	Quitar permiso de lectura al grupo y a los demás. chown
propietario1 nombreadarchivo	Cambiar la propiedad de un archivo al usuario owner1.
chgrp grp_propietario nombre_archivo	Cambia la propiedad del grupo primario del archivo nombre_archivo al grupo grp_propietario .
chgrp -R grp_propietario nombre-dirección	Cambiar la propiedad del grupo primario del directorio dir-name al grupo grp_owner de forma recursiva. Para cambiar la propiedad de grupo de un directorio y todo lo que hay dentro de ese directorio, utilice este comando.

Sección 1.3: Hola Mundo

Escriba el siguiente código en su terminal y, a continuación, pulse en **Enter**:

```
echo "Hola Mundo"
```

Esto producirá el siguiente resultado:

```
Hola Mundo
```

Sección 1.4: Utilidades básicas de Linux

Linux tiene un comando para casi cualquier tarea y la mayoría de ellos son intuitivos y fáciles de interpretar.

Obtener ayuda en Linux

Comando	Usabilidad
man < nombre>	Lee la página del manual de <nombre>.
man < section >< name>	Lee la página de manual de <name>, relacionada con la sección dada.
man -k < editor >	Da salida a todo el software cuyas páginas man contengan la palabra clave <editor>.
man -K < palabra clave >	Muestra todas las páginas de manual que <palabra clave>.
apropos < editor >	Salida de todas las aplicaciones cuya descripción de una línea coincide con la palabra <i>editor</i> . Si no recuerda el nombre de la aplicación, utilice este comando.
ayuda	En el shell Bash, esto mostrará la lista de todos los comandos bash disponibles.
help < name>	En el shell Bash, esto mostrará la información sobre el comando bash <name>.
info < name>	Ver toda la información sobre <name>.
dpkg -l	Muestra una lista de todos los paquetes instalados en un sistema basado en Debian.
dpkg -L nombreDelPaquete	Muestra los archivos instalados y los detalles de la ruta de un paquete determinado en Debian.
dpkg -l grep -i < edit >	Devuelve todos los paquetes .deb instalados con <edit> independientemente de los casos.
less /var/lib/dpkg/available	Devuelve descripciones de todos los paquetes disponibles.
qué es vim	Enumera una descripción de una línea de vim.
<nombre-de-l-comando> --help	Muestra información de uso sobre <nombre-de-la-herramienta>. A veces el comando -h también funciona, pero no para todos los comandos.

Identificación de usuarios y quién es quién en el mundo Linux

Comando	Usabilidad
nombre de host	Muestra el nombre de host del sistema.

hostname -f	Muestra el nombre de dominio completo (FQDN) del sistema.
contraseña	Cambia la contraseña del usuario actual.
whoami	Nombre de usuario de los usuarios conectados en el terminal.
quién	Lista de todos los usuarios actualmente conectados como usuario.
w	Muestra el estado actual del sistema, la hora, la duración, la lista de usuarios conectados actualmente al sistema y otra información sobre los usuarios.
último	que ha utilizado el sistema recientemente.
último root	Cuándo fue la última vez que root se conectó como usuario.
últimob	Muestra todos los intentos fallidos de inicio de sesión en el sistema.
chmod	Cambio de permisos - lectura, escritura, ejecución de un archivo o directorio.

Información relacionada con el proceso

Comando	Usabilidad
arriba	Lista todos los procesos ordenados por su uso actual de recursos del sistema. Muestra una actualización continua de los procesos (por defecto 3 segundos). Utilice la q para salir de top.
ps	Lista los procesos que se están ejecutando en la sesión de shell actual
ps -u root	Lista todos los procesos y comandos que está ejecutando
root ps aux	Lista todos los procesos de todos los usuarios en el sistema actual

Sección 1.5: Búsqueda de archivos por patrones en el nombre/contenido

Una tarea común de alguien que utiliza la línea de comandos (shell) de Linux es buscar archivos/directorios con un determinado nombre o que contengan un determinado texto. Hay 2 comandos con los que debes familiarizarte para lograr esto:

Buscar archivos por nombre

```
find /var/www -name '*.css'
```

Esto imprimirá la ruta/el nombre completo de todos los archivos de `/var/www` que terminen en `.css`. Ejemplo de salida:

```
/var/www/html/text-cursor.css
/var/www/html/style.css
```

Para más información:

hombre encontrar

Buscar archivos que contengan texto

```
grep font /var/www/html/style.css
```

Esto imprimirá todas las líneas que contengan la fuente del patrón en el archivo especificado. Ejemplo de salida:

```
font-weight: bold; font-
family: monospace;
```

Otro ejemplo:

```
grep font /var/www/html/
```

Esto no funciona como esperabas. Lo consigues:

```
grep: /var/www/html/: Es un directorio
```

Necesitas **hacer grep** recursivo para que funcione, usando la opción -R:

```
grep -R fuente /var/www/html/
```

! Echa un vistazo a la salida de este:

```
/var/www/html/admin/index.php: echo '<font color=red><b>Error: no dice</b></font><br/>';  
/var/www/html/admin/index.php: echo '<font color=red><b>Error: try again</b></font><br/>';  
/var/www/html/style.css: font-weight: bold;  
/var/www/html/style.css: font-family: monospace;
```

Tenga en cuenta que cuando **grep** encuentra varios archivos, antepone los nombres de los archivos a las líneas encontradas. Puede utilizar la opción -h para deshacerse de eso, si lo desea.

Para más información:

```
man grep
```

Sección 1.6: Manipulación de archivos

Los archivos y directorios (otro nombre para las carpetas) están en el corazón de Linux, por lo que ser capaz de crearlos, verlos, moverlos y borrarlos desde la línea de comandos es muy importante y bastante potente. Estos comandos de manipulación de archivos te permiten realizar las mismas tareas que realizaría un explorador gráfico de archivos.

Crea un archivo de texto vacío llamado miArchivo :

```
touch miArchivo
```

Cambia el nombre de miArchivo a miPrimerArchivo :

```
mv miArchivo miPrimerArchivo
```

Ver el contenido de un archivo:

```
cat miPrimerArchivo
```

Ver el contenido de un fichero con paginador (un pantallazo cada vez):

```
menos miPrimerArchivo
```

Ver las primeras líneas de un archivo:

```
head miPrimerArchivo
```

Ver las últimas líneas de un archivo:

```
tail miPrimerArchivo
```

Editar un fichero:

```
vi miPrimerArchivo
```

Vea qué archivos hay en su directorio de trabajo actual:

```
ls
```

Crea un directorio vacío llamado `miPrimerDirectorio` :

```
mkdir miPrimerDirectorio
```

Crear directorio multipath: (crea dos directorios, `src` y `myFirstDirectory`)

```
mkdir -p src/miPrimerDirectorio
```

Mueve el archivo al directorio:

```
mv miPrimerArchivo miPrimerDirectorio/
```

También puedes cambiar el nombre del archivo:

```
usuario@ordenador-linux:~$ mv miPrimerArchivo segundoNombreArchivo
```

Cambia el directorio de trabajo actual a `miPrimerDirectorio` :

```
cd miPrimerDirectorio
```

Borrar un fichero:

```
rm miPrimerArchivo
```

Mover al directorio padre (que se representa como `..`):

```
cd ..
```

Borrar un directorio vacío:

```
rmdir miPrimerDirectorio
```

Borrar un directorio no vacío (es decir, que contenga ficheros y/u otros directorios):

```
rm -rf miPrimerDirectorio
```

Tenga en cuenta que al eliminar directorios, que elimine `./` / `no /` que va a borrar todo su sistema de archivos.

Sección 1.7: Detalles del archivo/directorio

El comando `ls` tiene varias opciones que se pueden utilizar juntas para mostrar más información.

Detalles/Derechos

La opción `l` muestra los permisos de los archivos, su tamaño y la fecha de la última modificación. Así, si el directorio raíz contiene un dir llamado

prueba y un archivo `someFile` el comando:

```
usuario@ordenador-linux:~$ ls -l
```

Saldría algo como

```
-rw-r--r-- 1 usuario usuarios70 Jul 22 13:36 someFile.txt
drwxrwxrwx 2 user users 4096 Jul 21 07:18 test
```

Los permisos tienen el formato drwxrwxrwx. El primer carácter representa el tipo de archivo d si es un directorio - en caso contrario. Los tres siguientes rwx son los permisos que el usuario tiene sobre el archivo, los tres siguientes son los permisos que el grupo tiene sobre el archivo, y los tres últimos son los permisos que todos los demás tienen sobre el archivo.

La r de rwx representa si un archivo puede ser leído, la w representa si el archivo puede ser modificado, y la x representa si el archivo puede ser ejecutado. Si no se concede ningún permiso aparecerá un - en lugar de r, w o x.

Así que desde arriba el usuario puede leer y modificar someFile.txt pero el grupo sólo tiene derechos de sólo lectura.

Para cambiar los derechos puedes usar el comando **chmod** *### fileName* Si tienes derechos sudo. r se representa con un valor de 4, w se representa con 2, y x se representa con un 1. Así que si sólo quieres poder modificar el contenido al directorio de **prueba**

```
Propietario rwx= 4+2+1= 7
Grupo r-x= 4+0+1= 5
Otros r-x= 4+0+1= 5
```

Así que todo el comando es

```
chmod 755 prueba
```

Ahora haciendo un **ls -l** mostraría algo como

```
drwxr-xr-x 2 user users 4096 Jul 21 07:20 test
```

Tamaño legible

Utilizada junto con la l, la opción h muestra tamaños de archivo legibles. Ejecutar

```
usuario@ordenador-linux:~$ ls -lh
```

Saldría:

```
total 4166
-rw-r--r-- 1 usuario usuarios70 Jul 22 13:36 someFile.txt
drwxrwxrwx 2 user users 4.0K Jul 21 07:18 test
```

Oculto

Para ver los archivos ocultos utilice la opción a. Por ejemplo

```
usuario@ordenador-linux:~$ ls -a
```

Lista Might

```
.profile
algunArchivo.
txt prueba
```

Tamaño total del directorio

Para ver el tamaño del directorio actual utilice la opción `s` (también se puede utilizar la opción `h` para que el tamaño sea más legible).

```
usuario@ordenador-linux:~$ ls -s
```

Salidas

```
total 4166
algunArchivo.t prue
xt ba
```

Vista recursiva

Supongamos que el directorio de `prueba` tiene un archivo `anotherFile` y quieres verlo desde la carpeta raíz, podrías usar la función `R` que listaría el árbol recursivo.

```
usuario@ordenador-linux:~$ ls -R
```

Salidas

```
./:
algunArchivo.t prue
xt ba
./prueba:
otroArchivo
```

Capítulo 2: Detección del nombre y la versión de la distribución Linux

Sección 2.1: Detectar en qué distribución basada en Debian está trabajando

Basta con ejecutar `lsb_release -a`.

En Debian:

```
$ lsb_release -a
No hay módulos LSB disponibles. ID de
distribuidor: Debian
Descripción:      Pruebas de Debian GNU/Linux (stretch )
Liberación:       pruebas
Nombre en clave:   stretch
```

En Ubuntu:

```
$ lsb_release -a
No hay módulos LSB disponibles.
Distribuidor ID: Ubuntu Descripción:
                  Ubuntu 14.04.4 LTS
Versión:          14.04
Nombre en clave:  trusty
```

En caso de que no tenga `lsb_release` instalado puede que quieras probar a adivinar algo, por ejemplo, hay un archivo `/etc/issue` que a menudo contiene el nombre de la distribución. Por ejemplo, en ubuntu:

```
$ cat /etc/issue
Ubuntu 12.04.5 LTS \n \l
```

No utilice el fichero `/etc/debian_version` porque su contenido no coincide con el nombre de la distribución.

Tenga en cuenta que esto también funcionará en distribuciones que no sean de la familia Debian, como Fedora, RHEL u openSUSE, pero que `lsb_release` puede no estar instalado.

Sección 2.2: Detectar qué distribución basada en systemd está utilizando.

Este método funcionará en versiones modernas de Arch, CentOS, CoreOS, Debian, Fedora, Mageia, openSUSE, Red Hat Enterprise Linux, SUSE Linux Enterprise Server, Ubuntu y otros. Esta amplia aplicabilidad lo hace ideal como primera aproximación, con la posibilidad de recurrir a otros métodos si necesita identificar también sistemas más antiguos.

Mira en `/etc/os-release`. En concreto, fíjese en las variables `NAME`, `VERSION`, `ID`, `VERSION_ID` y `PRETTY_NAME`. En

Fedora, este archivo podría tener el siguiente aspecto:

```
NOMBRE=Fedora
VERSION= "24 (Workstation Edition)" ID=fedora
VERSION_ID=24
PRETTY_NAME= "Fedora 24 (Workstation Edition)" ANSI_COLOR=
"0;34"
```

```
CPE_NAME= "cpe:/o:fedoraproject:fedora:24" HOME_URL=
"https://fedoraproject.org/" BUG_REPORT_URL=
"https://bugzilla.redhat.com/" REDHAT_BUGZILLA_PRODUCT=
"Fedora" REDHAT_BUGZILLA_PRODUCT_VERSION=24
REDHAT_SUPPORT_PRODUCT= "Fedora"
REDHAT_SUPPORT_PRODUCT_VERSION=24
PRIVACY_POLICY_URL=https://fedoraproject.org/wiki/Legal:PrivacyPolicyVARIANT= "Edición
para estaciones de trabajo"
VARIANT_ID=puesto de trabajo
```

En CentOS, este archivo puede tener este aspecto:

```
NOMBRE= "CentOS
Linux" VERSI ÓN= "7
(Core)"
ID= "centos" ID_LIKE=
"rhel fedora" VERSION_ID=
"7"
PRETTY_NAME= "CentOS Linux 7 (Core)"
ANSI_COLOR= "0;31"
CPE_NAME= "cpe:/o:centos:centos:7" HOME_URL=
"https://www.centos.org/" BUG_REPORT_URL=
"https://bugs.centos.org/"

CENTOS_MANTISBT_PROJECT= "CentOS-7"
CENTOS_MANTISBT_PROJECT_VERSION= "7"
REDHAT_SUPPORT_PRODUCT= "centos"
REDHAT_SUPPORT_PRODUCT_VERSION= "7"
```

Este archivo está documentado en el sitio web de freedesktop; en principio, no es específico de systemd - pero existirá en todas las distribuciones basadas en systemd.

Desde el intérprete de comandos bash, se puede acceder al archivo `/etc/os-release` y utilizar directamente las distintas variables, como se indica a continuación:

```
$ ( source /etc/os-release && echo "$PRETTY_NAME" )
Fedora 24 (Edición para estaciones de trabajo )
```

Sección 2.3: Detectar en qué distribución RHEL / CentOS / Fedora está trabajando

Mira el contenido de `/etc/redhat-release`

```
cat /etc/redhat-release
```

Aquí está la salida de una máquina Fedora 24: Fedora release 24 (Veinticuatro)

Como se menciona en la respuesta basada en debian, también puede utilizar el comando `lsb_release -a`, que muestra esto desde una máquina Fedora 24:

```
Versi ón LSB: :core-4.1-amd64:core-4.1-noarch:cxx-4.1-amd64:cxx-4.1-noarch:desktop-4.1-
amd64:desktop-4.1-noarch:languages-4.1-amd64:languages-4.1-noarch:printing-4.1-amd64:printing-4.1- noarch
Distribuidor ID: Fedora
Descripci ón: Fedora release 24 (Veinticuatro)
Release: 24
Nombre en clave: TwentyFour
```


Sección 2.4: Uname - Imprime información sobre el sistema actual

Uname es el nombre abreviado de unix. Sólo tienes que escribir `uname` en la consola para obtener información sobre tu sistema operativo.

`uname` [OPCIÓN]

Si no se especifica ninguna OPCIÓN, `uname` asume la opción `-s`.

`-a` o `--all` - Imprime toda la información, omitiendo `-p` y `-i` si la información es desconocida.

Por ejemplo:

```
> uname -a
```

```
SunOS hope 5.7 Generic_106541-08 sun4m sparcsun4m SUNW,SPARCstation-10
```

Todas las opciones:

<code>-s, --nombre-del-núcleo</code>	Imprime el nombre del núcleo.
<code>-n, --nombredelnodo</code>	Imprime el nombre de host del nodo de red.
<code>-r, --kernel-release</code>	Muestra la versión del núcleo.
<code>-v, --versión-del-núcleo</code>	Muestra la versión del núcleo.
<code>-m, --máquina</code>	Imprime el nombre del hardware de la máquina.
<code>-p, --procesador</code>	Imprime el tipo de procesador, o "desconocido".
<code>-i, --hardware-platform</code>	Imprime la plataforma de hardware, o "desconocida".
<code>-o, --operating-system</code>	Imprime el sistema operativo.
<code>--ayuda</code>	Muestra un mensaje de ayuda y sale.
<code>--versión</code>	Muestra información sobre la versión y sale.

Sección 2.5: Detectar información básica sobre su distribución

basta con ejecutar `uname`

`-a`. En Arch:

```
$ uname -a
```

```
Linux nokia 4.6.4-1-ARCH #1 SMP PREEMPT Lun Jul 11 19:12:32 CEST 2016 x86_64  
código aquí
```

GNU/Linuxenter

Sección 2.6: Uso de GNU coreutils

Así que GNU coreutils debería estar disponible en todos los sistemas basados en Linux (favor, corrígeme si me equivoco).

Si no sabe qué sistema está utilizando, es posible que no pueda saltar directamente a uno de los ejemplos anteriores, por lo que éste puede ser su primer puerto de escala.

```
$ uname -a
```

En mi sistema esto me da lo siguiente...

```
Linux Scibearspace 3.16.0-4-amd64 #1 SMP Debian 3.16.7-ckt25-2+deb8u3 (2016-07-02) x86_64
```

Aquí puede ver lo siguiente :

Scibearspace : el nombre de mi pc

- Scibearspace : el nombre de mi pc
- 3.16.0-4-amd64 : el núcleo y la arquitectura
- SMP Debian 3.16.7-CKT25-2+deb8u3 : me dice que estoy ejecutando debian con el kernel 3.16•

Finalmente la última parte estoy ejecutando debian 8 (actualización 3).

Me gustaría que otros añadieran los resultados de los sistemas RHEL y SuSe.

Sección 2.7: Encuentre el nombre y el número de versión de su sistema operativo linux (tanto debian como rpm)

La mayoría de las distribuciones linux almacenan su información de versión en el archivo `/etc/lsb-release` (debian) o `/etc/redhat-release` (basado en RPM). Usando el comando genérico de abajo debería pasar la mayoría de los derivados de Debian y RPM como Linux Mint y Cent-Os.

Ejemplo en máquina Ubuntu:

```
cat /etc/*release
```

```
DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=14.04
DISTRIB_CODENAME=confianza
DISTRIB_DESCRIPTION="Ubuntu 14.04 LTS"
```

Capítulo 3: Obtener información sobre un núcleo Linux en ejecución

Sección 3.1: Obtener detalles del núcleo Linux

Podemos utilizar el comando `uname` con varias opciones para obtener detalles completos del kernel en ejecución.

```
uname -a
```

```
Linux df1-ws-5084 4.4.0-64-generic #85-Ubuntu SMP Lun Feb 20 11:50:30 UTC 2017 x86_64 x86_64  
x86_64 GNU/Linux
```

Según la página de manual aquí hay algunas

opciones más Uso: `uname [OPTION]...`

Imprime cierta información del sistema. Sin OPCIÓN, igual que `-s`.

```
-a, --all                    imprima toda la información, en el siguiente orden,  
excepto omitir -p y -i si se desconoce:  
-s, --nombre-del-núcleo     imprime el nombre del núcleo  
-n, --nombredelnodo         imprime el nombre de host del nodo de red  
-r, --kernel-release        imprime la versión del núcleo  
-v, --versión-del-núcleo    muestra la versión del núcleo  
-m, --máquina               imprime el nombre del hardware de la máquina  
-p, --procesador            muestra el tipo de procesador (no portátil)  
-i, --hardware-platform     imprime la plataforma de hardware (no portátil)  
-o, --sistema-operativo     imprime el sistema operativo  
--ayuda                    mostrar esta ayuda y salir  
--version                  información de la versión de salida y salir
```

Capítulo 4: Caparazón

El shell ejecuta un programa en respuesta a su prompt. Cuando das un comando, el shell busca el programa y lo ejecuta. Por ejemplo, cuando das el comando `ls`, el shell busca la utilidad/programa llamado `ls`, y luego lo ejecuta en el shell. Los argumentos y las opciones que proporcionas con las utilidades pueden influir en el resultado que obtengas. El shell también se conoce como CLI, o interfaz de línea de comandos.

Sección 4.1: Cambiar el shell por defecto

La mayoría de las distribuciones modernas vienen con BASH (Bourne Again SHell) preinstalado y configurado como shell por defecto. El comando (en realidad un binario ejecutable, un ELF) responsable de cambiar de shell en Linux es `chsh` (change shell).

Primero podemos comprobar qué shells están ya instalados y configurados en nuestra máquina utilizando el comando `chsh -l` que mostrará un resultado similar al siguiente:

```
[usuario@localhost]~$ chsh -l
/bin/sh
/bin/bash
/sbin/nologin
/usr/bin/sh
/usr/bin/bash
/usr/sbin/nologin
/usr/bin/pescado
```

En algunas distribuciones de Linux, `chsh -l` no es válido. En este, la lista de todos los shells disponibles se puede encontrar en el archivo `/etc/shells`. Puede mostrar el contenido del archivo con `cat`:

```
[usuario@localhost]~$ cat /etc/shells
# /etc/shells: login shells válidos
/bin/sh
/bin/bash
/sbin/nologin
/usr/bin/sh
/usr/bin/bash
/usr/sbin/nologin
/usr/bin/pescado
```

Ahora podemos elegir nuestro nuevo shell por defecto, por ejemplo `fish`, y configurarlo usando `chsh -s`,

```
[usuario@localhost]~$ chsh -s /usr/bin/fish Cambio
de shell para el usuario.
Contraseña:
Shell cambiado.
```

Ahora todo lo que queda por hacer es preformar un ciclo logoff-logon, y disfrutar de nuestro nuevo shell por defecto.

Si deseas cambiar el shell por defecto para un usuario diferente, y tienes privilegios administrativos en la máquina, podrás lograrlo usando `chsh` como `root`. Así que asumiendo que queremos cambiar el shell por defecto del usuario `usuario_2` a `fish`, usaremos el mismo comando que antes, pero añadiendo el nombre de usuario del otro usuario, `chsh -s /usr/bin/fish usuario_2`.

Para comprobar cuál es el shell por defecto actual, podemos ver la variable de entorno `$SHELL`, que apunta a la ruta de nuestro shell por defecto, por lo que después de nuestro comando, esperaríamos obtener un resultado similar a este,

```
~ echo $SHELL
/usr/bin/pescado
```

opciones **chsh**:

-s shell

Establece shell como shell de inicio de sesión.

-l , **--list-shells**

Imprime la lista de shells listados en /etc/shells y sale.

-h , **--help**

Imprime un mensaje de uso y sale.

-v , **--version**

Imprimir la información de la versión y salir.

Sección 4.2: Utilidades básicas de Shell

Personalizar el prompt de Shell

El símbolo del sistema por defecto se puede cambiar para que tenga un aspecto diferente y corto. En caso de que el directorio actual sea largo, el símbolo del sistema por defecto se vuelve demasiado grande. El uso de **PS1** es útil en estos casos. Un comando corto y personalizado bonito y elegante. En la siguiente tabla se ha utilizado **PS1** con un número de argumentos para mostrar diferentes formas de prompt de shell. El prompt de comandos por defecto se parece a esto: **user@host ~ \$** en mi caso se parece a esto: **bruce@gotham ~ \$** Se puede cambiar según la siguiente tabla:

Comando	Utilidad
PS1='\w \$ '	~ \$ como nombre de directorio. En este caso el directorio raíz es Root.
PS1='\h \$ '	gotham \$ shell prompt como nombre de host
PS1='\u\$ '	bruce\$ shell prompt como nombre de usuario
PS1='\t \$ '	22:37:31 \$ shell prompt en formato de 24 horas
PS1='@ \$ '	10:37 PM shell prompt en formato de 12 horas
¡PS1='! \$ '	732 mostrará el número de historia del comando en lugar del prompt del shell
PS1='dude \$ '	dude \$ mostrará el prompt del shell a su gusto

Algunos comandos básicos del shell

Comando	Utilidad
Ctrl-k	cortar/matar
Ctrl-y	jalar/pegar
Ctrl-a	llevará el cursor al principio de la línea
Ctrl-e	llevará el cursor al final de la línea
Ctrl-d	borrará el carácter situado después/en el cursor
Ctrl-l	borrará la pantalla/terminal
Ctrl-u	borrará todo lo que haya entre el prompt y el cursor
Ctrl_	deshará lo último escrito en la línea de comandos
Ctrl-c	interrumpirá/detendrá el trabajo/proceso que se esté ejecutando en primer plano

Ctrl-r	búsqueda inversa en el historial
~/.bash_history	almacena los últimos 500 comandos/eventos utilizados en el shell
historial	mostrará el historial de comandos
historial grep <palabra clave>	mostrará todos los comandos del historial que tengan la palabra clave <palabra clave> (útil en casos en los que recuerde parte del comando utilizado en el pasado)

Sección 4.3: Crear su propio alias de comando

Si estás cansado de usar comandos largos en bash puedes crear tu propio alias de comando.

La mejor manera de hacerlo es modificar (o crear si no existe) un archivo llamado `.bash_aliases` en su carpeta home. La sintaxis general es:

```
alias comando_alias= 'comando_actual'
```

donde `comando_actual` es el comando que está renombrando y `alias_comando` es el nuevo nombre que ha dado. Por ejemplo

```
alias install= 'sudo apt-get -y install'
```

asigna el nuevo alias de comando `install` al comando real `sudo apt-get -y install`. Esto significa que cuando usas `install` en una terminal, bash lo interpreta como `sudo apt-get -y install`.

Sección 4.4: Localizar un archivo en su sistema

Usando bash puedes localizar fácilmente un archivo con el comando `locate`. Por ejemplo, digamos que usted está buscando el archivo `mykey.pem`:

```
localizar mykey.pem
```

A veces los archivos tienen nombres extraños, por ejemplo, podrías tener un archivo como `random7897_mykey_0fidw.pem`. Digamos que estás buscando este archivo pero sólo recuerdas las partes `mykey` y `pem`. Podrías combinar el comando `locate` con `grep` usando una tubería como esta:

```
locate pem | grep mykey
```

Lo que mostraría todos los resultados que contengan estas dos piezas.

Tenga en cuenta que no todos los sistemas tienen la utilidad `locate` instalada, y muchos de los que la tienen no la han habilitado. `locate` es rápida y eficiente porque escanea periódicamente su sistema y almacena en caché los nombres y ubicaciones de cada archivo en él, pero si esa recopilación de datos no está habilitada, entonces no puede decirle nada. Puede utilizar `updatedb` para iniciar manualmente el escaneo del sistema de archivos para actualizar la información almacenada en caché sobre los archivos de su sistema de archivos.

Si no dispone de una función de localización, puede recurrir a la utilidad de búsqueda:

```
find / -name miclave.pem -print
```

es más o menos equivalente a `localizar mykey.pem` pero tiene que escanear tu(s) sistema(s) de archivos cada vez que lo ejecutas en busca del archivo en cuestión, en lugar de utilizar datos almacenados en caché. Esto es obviamente más lento y menos eficiente, pero más en tiempo real. La utilidad `find` puede hacer mucho más que encontrar ficheros, pero una descripción completa de sus capacidades está más allá del alcance de este ejemplo.

Capítulo 5: Comprobar el espacio en disco

Sección 5.1: Investigar el uso del disco en los directorios

A veces puede ser necesario averiguar qué directorio consume cuánto espacio en disco, especialmente cuando se utiliza `df -h` y se dio cuenta de que su espacio disponible en disco es bajo.

du:

El comando `du` resume el uso de disco del conjunto de ARCHIVOS, recursivamente para los

directorios. Se utiliza a menudo con la opción `-sh`:

```
-s, --resumir
mostrar sólo un total para cada argumento
-h, --lectura humana
tamaños de impresión en formato legible (por ejemplo, 1K 234M 2G)
```

Para resumir los usos de disco de los archivos en el directorio actual que utilizamos:

```
du -sh *
```

Ejemplo de salida:

```
572K    Documentos
208M    Descargas
4,0K    Música
724K    Imágenes
4,0K    Plantillas
4,0K    públicas
4,0K    Vídeos
```

También podemos incluir archivos ocultos con el uso de:

```
du -sh . [!.] * *
```

Ejemplo de salida:

```
6,3M    .atom
4,0K    .bash_history
4,0K    .bash_logout
8,0K    .bashrc
350M    .caché
195M    .config
12K     .dbus
4,0K    .dmrc
44K     .gconf
60K     .gema
520K    .gimp-2.8
28K     .gnome
4,0K    .ICEautoridad
8,3M    .local
8,0K    .nano
404K    .nv
36K     .pki
```

```

4,0K    .perfil
8,0K    .ssh
0       .sudo_as_admin_successful
4,0K    .Xauthority
4,0K    .xsession-errors 4,0K
        .xsession-errors.old
572K    Documentos
208M    Descargas
4,0K    Música
724K    Imágenes
4,0K    Público
4,0K    Plantillas
4,0K    Vídeos

```

En tercer lugar, puede añadir total a la salida añadiendo la opción `-c`:

```
du -sch . [!.]* *
```

Resultado:

```

.
.
.
4,0K    Plantillas
4,0K    Vídeos
769M    total

```

Lo más importante es utilizar el comando `du` correctamente en el directorio raíz para averiguar qué aplicación/servicio o usuario está consumiendo mucho espacio en disco. Por ejemplo, en el caso de un nivel ridículamente bajo de disponibilidad de espacio en disco para un servidor web y de correo, la razón podría ser un ataque de spam a su servicio de correo y se puede diagnosticar con sólo utilizar el comando `du`.

Investiga el uso del disco en el directorio raíz:

```
sudo du -sch /. [!.]* /*
```

Ejemplo de salida:

```

16K    /.VolumeIcon.icns
24K    /.VolumeIcon.png
13M    /bin
57M    /arranque
4,0K    /cdrom
620K    /dev
13M    /etc
779M    /home
0       /initrd.img
406M    /lib
3,9M    /lib32
4,0K    /lib64
16K    /perdido+encontrado
4,0K    /media
4,0K    /mnt
367M    /opt
du: no se puede acceder a '/proc/18221/task/18221/fd/4': No such file or directory
du: no se puede acceder a '/proc/18221/task/18221/fdinfo/4': No such file or directory du:
no se puede acceder a '/proc/18221/fd/4': No such file or directory

```



```
du: no se puede acceder a '/proc/18221/fdinfo/4': No such file or directory
0      /proc
20K    /root
du: no se puede acceder a '/run/user/1000/gvfs': Permiso denegado
9,4M   /ejecutar
13M    /sbin
4,0K   /srv
0      /sys
72K    /tmp
3,5G   /usr
639M   /var
0      /vmlinuz
5,8G   total
```

Por último, el mejor método consiste en añadir un umbral de tamaño a los directorios para ignorar más pequeños. Este comando sólo mostrará las carpetas con más de 1 GB de tamaño que se encuentran bajo el directorio raíz hasta la rama más lejana de todo el árbol de directorios en su sistema de archivos:

```
sudo du --threshold=1G -ch /. [!.]* /*
```

Ejemplo de salida:

```
1,4G    /usr/lib
1,8G    /usr/compar
3,5G    tir
5,8G    /usr
5,8G    total
```

Sección 5.2: Comprobación del espacio en disco

Es bastante común querer comprobar el estado de las distintas particiones/unidades de disco de tu servidor/ordenador para ver cómo de llenas están. El siguiente comando es el que tendrás que ejecutar:

```
df -h
```

Esto producirá una salida similar a la siguiente:

```
[root@mail~]# df -h
Sistema de archivos      Tamaño Usado Disponible    Uso% Montado en
/dev/mapper/VolGroup-lv_root
                        19G   1.6G   16G     9% /
tmpfs                    245M    0   245M    0% /dev/shm
/dev/sda1                 485M   47M  413M   11% /boot
```

En este ejemplo básico, podemos ver que la partición / sólo tiene un 9% de uso.

Para un ejemplo más complejo que también cubre el uso de df para ver varios puntos de montaje, vea a continuación:

```
[root@mail~]# df -h
Sistema de archivos      Talla    Usado Dispon  Utili Montado en
                        ible     zació
                        n
/dev/mapper/VG-root      1.9T    1.7T    89G    95% /
/dev/mapper/VG-var       431G    145G    264G   36% /var
devtmpfs                 7.8G    204K    7.8G    1% /dev
tmpfs                    7.8G    4.0K    7.8G    1% /dev/shm
/dev/md1                 495M    126M    344M   27% /boot
ku.ejemplo.com:9421      2.5T    487G    2.0T   20% /mnt/prueba
tmpfs                    500M    86M    415M   18% /var/ngx_pagespeed_cache
```

En este ejemplo, tenemos una partición / que está llena en un 95% junto con una partición adicional /var que sólo está llena en un 36%.

Tiene un montaje de red externo de 2T que está montado en /mnt/test y un montaje ramdisk/tmpfs de 500M montado en /var/nginx_pagespeed_cache .

Capítulo 6: Obtener información del sistema

Colección de comandos para obtener información relacionada con el sistema.

Sección 6.1: Estadísticas sobre CPU, Memoria, Red y Disco (operaciones de E/S).

Para obtener estadísticas generales sobre los principales componentes de la familia Linux de comandos **stat** son extremadamente útiles

CPU

Para obtener estadísticas relacionadas con los procesadores puede utilizar el comando **mpstat**, pero con algunas opciones proporcionará una mejor visibilidad:

```
$ mpstat 2 10
```

Memoria

Todos conocemos el comando **free** para mostrar la cantidad de RAM (restante) pero para ver todas las estadísticas incluyendo las operaciones de E/S:

```
$ vmstat 2 10
```

Disco

Para obtener información general sobre las operaciones de tu disco en tiempo real puedes utilizar **iostat** .

```
$ iostat -kx 2
```

Red

Para poder ver lo que ocurre con tus servicios de red puedes utilizar **netstat**

```
$ netstat -ntlp # sockets TCP abiertos
$ netstat -nulp # sockets UDP abiertos
$ netstat -nxlp # abrir sockets Unix
```

Pero puedes encontrar monitorización útil para ver el tráfico de la red en tiempo real:

```
$ sudo iftop
```

Opcional

Para generar estadísticas en tiempo real relacionadas con las operaciones de E/S en todos los componentes puede utilizar **dstat** . Esa herramienta que es un reemplazo versátil para **vmstat**, **iostat** y **ifstat**

Sección 6.2: Uso de herramientas como lscpu y lshw

Mediante el uso de herramientas como **lscpu** como **lscpu** es una manera fácil de obtener información de la CPU.

```
$ lscpu
Arquitectura: CPU          x86_64
op-mode(s):                32 bits, 64
Orden de bytes:             bits Little
                             Endian
```

```

CPU(s): 4
Lista de CPU (s) en línea: 0-3
Hilo (s) por núcleo: 1
Núcleo (s) por zócalo: 4
Enchufe(s): 1
Nodo(s) NUMA: 1
ID de proveedor: GenuineIntel
Familia de CPU: 6
Modelo: 23
Pisando: 10
MHz de la CPU: 1998.000
BogoMIPS: 5303.14
Virtualización: VT-x
Caché L1d: 32K
Caché L1i: 32K
Caché L2: 2048K
Nodo NUMA0 CPU(s): 0-3

```

Utilizando la herramienta lshw

```

$ lshw | grep cpu

df1-ws-5084
  descripción: Anchura del
  ordenador: 64 bits
  capacidades: vsyscall32
*-núcleo
  descripción: Identificación
  física de la placa base: 0
*-memoria
  descripción: Memoria del
  sistema id física: 0
  tamaño: 5881MiB
*-cpu
  producto: CPU Intel (R) Pentium (R) G3220 a 3,00GHz
  proveedor: Intel Corp.
  id física: 1 bus
  info: cpu@0
  tamaño: 3GHz
  capacidad: 3GHz
  ancho: 64 bits

```

Sección 6.3: Lista de hardware

Ubuntu:

`lshw` es una pequeña herramienta para extraer información detallada sobre la configuración hardware de la máquina. Puede informar de la configuración exacta de la memoria, la versión del firmware, la configuración de la placa base, la versión y velocidad de la CPU, la configuración de la caché, la velocidad del bus, etc.

```

$ sudo lshw | less (o more)
$ sudo lshw -html > myhardware.html
$ sudo lshw -xml > myhardware.xml

```

Para mostrar información PCI

```
$ lspci -tv
```

Para ver la información del USB

```
$ lsusb -tv
```

Para visualizar la información de la BIOS

```
$ dmidecode -q | menos
```

Para ver información específica sobre el disco (disco sda en el ejemplo) puede utilizar:

```
$ hdparm -i /dev/sda
```

Algunas utilidades/comandos adicionales ayudarán a recopilar información adicional:

```
$ smartctl -A /dev/sda | grep Power_On_Hours # ¿Cuánto tiempo ha estado encendido este disco (sistema) en total?
$ hdparm -tT /dev/sda # Realiza una prueba de velocidad de lectura en el disco sda
$ badblocks -s /dev/sda # Comprueba si hay bloques ilegibles en el disco sda
```

Sección 6.4: Buscar información sobre el modelo/velocidad de la CPU

Ubuntu:

```
$ cat /proc/cpuinfo
```

Muestra de salida:

```
processor      : 0
vendor_id     : GenuineIntel
cpu family    : 6
model         : 15
name          : CPU Intel(R) Core(TM)2 Quad          Q6600 a 2,40 GHz
stepping      : 11
cpu MHz       : 1596.000
cache size    : 4096 KB
id física     : 0
hermanos id   : 4
núcleo        : 0
núcleos cpu   : 4
apicid        : 0

apicid inicial : 0
fpu            : sí
fpu_exception : yes
nivel cpuid    : 10

wp            Sí
bande         : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts
aesni mmx fxsr sse sse2 ss ht tm pbe syscall nx lm constant_tsc arch_perfmon pebs bts rep_good pn1 dtes64
monitor ds_cpl vmx est tm2 ssse3 cx16 xtpr pdcm lahf_lm tpr_shadow vnmi flexpriority bogomips : 4800.18
clflush tamaño : 64
cache_alignment : 64
dirección tamaños : 36 bits física, 48 bits virtual
gestión de energía:
....
..
procesador     : 3
vendor_id      : Familia de
cpu GenuineIntel
               : 6
```

```

modelo      : 15
nombre del   : CPU Intel(R) Core(TM)2 Quad          Q6600 a 2,40 GHz
modelo paso  : 11
a paso cpu   : 1596.000
MHz tamaño   : 4096 KB
de la caché  : 0
id física    : 4
hermanos id   : 3
núcleo       : 4
núcleos cpu   : 3
apicid
apicid inicial : 3
fpu           : sí
fpu_exception : yes
nivel cpuid    : 10

wp           Sí
bande        : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts
acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx lm constant_tsc arch_perfmon pebs bts rep_good pni dtes64
monitor ds_cpl vmx est tm2 ssse3 cx16 xtpr pdcm lahf_lm tpr_shadow vnmi flexpriority bogomips : 4800.30
clflush tamaño : 64
cache_alignment : 64
dirección tamaños : 36 bits física, 48 bits virtual
gestión de energía:

```

procesador de recuento (incluidos los núcleos):

```
$ grep -c procesador /proc/cpuinfo
```

Sección 6.5: Seguimiento del proceso y recopilación de información

En general, usted tiene dos maneras de supervisar los procesos en linux host

Control estático

El comando más utilizado es `ps` (es decir, estado del proceso), que se utiliza para proporcionar información sobre los procesos que se están ejecutando actualmente, incluidos sus números de identificación de proceso (PID).

He aquí algunas opciones útiles para recabar información específica.

Listar procesos en una jerarquía

```
$ ps -e -o pid,args --forest
```

Lista de procesos ordenados por % de uso de cpu

```
$ ps -e -o pcpu,cpu, nice,state,cputime,args --sort pcpu | sed '/^ 0.0 /d'
```

Lista de procesos ordenados por uso de mem (KB).

```
$ ps -e -orss=,args = | sort -b -k1,1n | pr -TW$COLUMNS
```

Listar todos los hilos de un proceso en particular (proceso "firefox-bin" en el ejemplo)

```
$ ps -C firefox-bin -L -o pid,tid,pcpu,state
```

Después de encontrar un proceso específico, puede obtener información relacionada con él utilizando `lssof` para listar las rutas que el id del proceso tiene abiertas

```
$ lsof $$
```

O basándose en la ruta encuentre la lista de procesos que tienen abierta la ruta especificada

```
$ lso~
```

Supervisión interactiva

La herramienta más conocida para la supervisión dinámica es:

```
$ top
```

Que en su mayoría por defecto de comandos que tienen gran cantidad de opciones para filtrar y representar la información en tiempo real (en comparación con el comando `ps`).

Aún así, hay opciones más avanzadas que pueden considerarse e instalarse como reemplazo `superior`

```
$ htop -d 5
```

O

```
$ sobre
```

Que tiene la capacidad de registrar todas las actividades en el archivo de registro (por defecto `atop` registrará toda la actividad en cada 600 segundos) A esta lista hay algunos comandos especializados como `iotop` o `iftop`

```
$ sudo iotop
```

Capítulo 7: comando ls

Sección 7.1: Opciones del comando ls

Lista completa de opciones:

ls -a lista todos los archivos incluyendo los archivos ocultos que empiezan por '.'

ls --color lista coloreada [=siempre/nunca/auto]

ls -d lista directorios - con '*'

ls -F añade un carácter de */=>@| a enteries

ls -i lista el número de índice del inodo del archivo

ls -l lista con formato largo - muestra permisos

ls -la lista con formato largo incluyendo archivos

ocultos **ls -lh** lista con formato largo con tamaño

de archivo legible **ls -ls** lista con formato largo

con tamaño de archivo

ls -r lista en orden inverso

ls -R lista recursivamente el árbol de directorios

ls -s lista el tamaño del archivo

ls -S ordenar por tamaño de archivo

ls -t ordenar por fecha y hora

ls -X ordenar por nombre de extensión

Sección 7.2: comando ls con las opciones más utilizadas

ls muestra archivos y directorios en el directorio de trabajo actual. (si no se pasan argumentos.) (No muestra los archivos ocultos que empiezan por . por defecto.)

```
usuario@ubuntu14:/usr$ ls
bin juegos include lib lib32      local sbin share src
```

Para ver todos los archivos (también los archivos/carpetas ocultos). Utilice **ls -a** O **ls -all**

```
user@ubuntu14:/usr$ ls -a
. .. bin juegos include lib lib32      local sbin share src
```

Para diferenciar entre archivos y carpetas y enlaces simbólicos y otros, utilice **ls -F** O **ls --classify**

```
user@ubuntu14:~$ ls -F
bash_profile_course  chat_apps/      Escritori  Descargas      foxitsoftware/
Público             test/          bin/       ClionProjeéts/ Documentos/    IDE/         Música
Imágenes/ Plantillas/ Vídeos/
```


Aquí, los caracteres finales se utilizan para distinguir archivos y carpetas.

"/" sugiere directorio. "*"

sugiere ejecutables. "@"

sugiere enlaces simbólicos.

Para obtener más detalles sobre los archivos y directorios, utilice `ls -l`

```
user@ubuntu14:~/ejemplo$ ls -l
total 6464

-rw-r--r-- 1 dave dave 41 Dic 24 12:19 Z.txt
drwxr-xr-x 2 usuar grupo 4096 Dic 24 12:00 a_directorio
embre
-rw-r--r-- 1 usuar grupo 6 Dic 24 12:01 archivo_a
embre
lrwxrwxrwx 1 usuar grupo 6 Dic 24 12:04 a_link -> a_file
embre
-rw-r--r-- 1 usuar grupo 6 Dic 24 12:03 a_archivo_nuevo
embre
-rw-r----- 1 usuar grupo 6586816 Dic 24 12:07 grande.zip
embre
```

En este ejemplo, el tamaño total del contenido es de 6460KB.

A continuación, hay una entrada para cada archivo/directorio en orden alfabético con mayúsculas antes que minúsculas. El primer carácter es el tipo (por ejemplo, d - directorio, l - enlace).

Los siguientes 9 caracteres muestran los permisos para el usuario, grupo y otros.

A continuación, se indica el número de enlaces duros y, por último, el nombre y el grupo del propietario.

El siguiente campo es el tamaño en bytes. Esto se puede mostrar de forma amigable añadiendo la opción `-h`, por ejemplo, 6586816 se muestra como 6.3M

A continuación aparece una marca de tiempo (normalmente la hora de modificación).

El último campo es el nombre. Nota: los enlaces también muestran el destino del enlace.

Capítulo 8: Compresión de archivos con el comando 'tar

Opciones comunes	-
-c --crear	Crea un nuevo archivo.
-x --extract	Extrae ficheros de un archivo.
-t --list	Lista el contenido de un archivo.
-f --file=ARCHIVO	Utiliza el archivo o dir ARCHIVE.
-v --verbose	Lista de forma detallada los archivos procesados.
Opciones de compresión -	
-a --auto-compress	Utiliza el sufijo del archivo para determinar el programa de compresión.
-j --bzip2	Filtra el archivo mediante bzip2.
-J --xz --lzma	Filtra el archivo mediante xz.
-z --gzip	Filtra el archivo mediante gzip.

Sección 8.1: Comprimir una carpeta

Esto crea un archivo simple de una carpeta :

```
tar -cf . /mi-archivo.tar . /mi-carpeta/
```

La salida detallada muestra qué ficheros y directorios se añaden al archivo; utilice la opción -v:

```
tar -cvf . /mi-archivo.tar . /mi-carpeta/
```

Para archivar una carpeta comprimida con 'gzip', hay que utilizar la opción -z :

```
tar -czf . /mi-archivo.tar.gz . /mi-carpeta/
```

En su lugar, puede comprimir el archivo con 'bzip2', utilizando la opción -j:

```
tar -cjf . /mi-archivo.tar.bz2 . /mi-carpeta/
```

O comprimir con 'xz', utilizando la opción -J:

```
tar -cjf . /mi-archivo.tar.xz . /mi-carpeta/
```

Sección 8.2: Extraer una carpeta de un archivo

Hay un ejemplo para extraer una carpeta de un archivo en la ubicación actual :

```
tar -xf nombre-archivo.tar
```

Si desea extraer una carpeta de un archivo a un destino específico :

```
tar -xf nombre-archivo.tar -C ./directorio/destino
```

Sección 8.3: Listar el contenido de un archivo

Listar el contenido de un archivo sin :

```
tar -tf archive.tar.gz
Carpeta-En-Archivo/
Carpeta-En-
Archivo/fichero1
Carpeta-En-Archivo/Otra-Carpeta/ Carpeta-
En-Archivo/Otra-Carpeta/Archivo2
```

Sección 8.4: Listar el contenido del archivo

Hay un ejemplo de listado de contenidos :

```
tar -tvf archivo.tar
```

La opción `-t` se utiliza para el listado. Para listar el contenido de un archivo `tar.gz`, hay que utilizar la `-z` más :

```
tar -tzvf archivo.tar.gz
```

Sección 8.5: Comprimir y excluir una o varias carpetas

Si desea extraer una carpeta, pero excluir una o varias carpetas durante la extracción, puede utilizar la opción Opción `--exclude` .

```
tar -cf archive.tar . /mi-carpeta/ --exclude= "mi-carpeta/sub1" --exclude= "mi-carpeta/sub3"
```

Con este árbol de carpetas :

```
mi-carpeta/
sub1/
sub2/
sub3/
```

El resultado será :

```
./archivo.tar
mi-carpeta/
sub2/
```

Sección 8.6: Desmontar los componentes principales

Para eliminar cualquier número de componentes principales, utilice la opción `--strip-components`:

```
--strip-components=N ÚMERO
eliminar los componentes iniciales de N ÚMERO de los nombres de archivo al extraerlos
```

Por ejemplo, para eliminar la carpeta inicial, utilice:

```
tar -xf --strip-components=1 nombre-archivo.tar
```

Capítulo 9: Servicios

Sección 9.1: Lista de servicios en ejecución en Ubuntu

Para obtener una lista del servicio en su sistema, puede ejecutar:

```
servicio --status-all
```

La salida de `service --status-all` lista el estado de los servicios controlados por System V.

El `+` indica que el servicio está en ejecución, `-` indica un servicio detenido. Puede comprobarlo ejecutando `service SERVICENAME status` para un servicio `+` y `-`.

Algunos servicios son gestionados por Upstart. Puede comprobar el estado de todos los servicios Upstart con `sudo initctl list`. Cualquier servicio gestionado por Upstart también se mostrará en la lista proporcionada por `service --status-all` pero se marcará con una `?`.

ref: <https://askubuntu.com/questions/407075/how-to-read-service-status-all-results>

Sección 9.2: Gestión de servicios Systemd

Servicios de listado

- `systemctl` Para listar los servicios en ejecución
- `systemctl --failed` Para listar los servicios fallidos

Gestión de objetivos (similar a los niveles de ejecución en SysV)

- `systemctl get-default` Para encontrar el objetivo por defecto de su sistema
- `systemctl set-default <target-name>` Para establecer el destino predeterminado de su sistema

Gestión de servicios en tiempo de ejecución

- `systemctl start [nombre-servicio]` Para iniciar un servicio
- `systemctl stop [nombre-servicio]` Para detener un servicio
- `systemctl restart [nombre-servicio]` Para reiniciar un servicio
- `systemctl reload [service-name]` Para solicitar al servicio que recargue su configuración
- `systemctl status [service-name]` Para mostrar el estado actual de un servicio

Gestión del inicio automático de servicios

- `systemctl is-enabled [service-name]` Para mostrar si un servicio está habilitado en el arranque del sistema
- `systemctl is-active [service-name]` Para mostrar si un servicio está actualmente activo (ejecutándose)
- `systemctl enable [service-name]` Para habilitar un servicio en el arranque del sistema
- `systemctl disable [service-name]` Para desactivar un servicio en el arranque del sistema

Servicios de enmascaramiento

- `systemctl mask [nombre-servicio]` Para enmascarar un servicio (dificulta el inicio de un servicio por error)
- `systemctl unmask [service-name]` Para desenmascarar un servicio

Reiniciar systemd

```
systemctl daemon-reload
```

Capítulo 10: Gestión de servicios

Apartado 10.1: Diagnosticar un problema con un servicio

En sistemas que utilizan systemd, como Fedora>= 15, Ubuntu (Server y Desktop)>= 15.04, y RHEL/CentOS>= 7:

```
systemctl status [nombre del servicio]
```

...donde [servicename] es el servicio en cuestión; por ejemplo, `systemctl status sshd`. Esto

mostrará información básica de estado y cualquier error reciente registrado.

Puedes ver más errores con `journalctl`. Por ejemplo, `journalctl -xe` cargará los últimos 1000 registrados en un paginador (como `less`), saltando hasta el final. También puede utilizar `journalctl -f`, que seguirá los mensajes de registro a medida que entran.

Para ver los registros de un servicio concreto, utilice el indicador `-t`, como se indica a continuación:

```
journalctl -f -t sshd
```

Otras opciones útiles incluyen `-p` para prioridad (`-p` advertencias para ver sólo advertencias y superiores), `-b` para "desde el último arranque", y `-S` para "desde" - juntando todo esto, podríamos hacer

```
journalctl -p err -S ayer
```

para ver todos los elementos registrados como errores desde ayer.

Si `journalctl` no está disponible, o si está siguiendo registros de errores de aplicaciones que no utilizan el diario del sistema, el comando `tail` se puede utilizar para mostrar las últimas líneas de un archivo. Una bandera útil para `tail` es `-f` (para "seguir"), que hace que `tail` continúe mostrando los datos a medida que se añaden al archivo. Para ver los mensajes de la mayoría de los servicios del sistema:

```
tail -f /var/log/messages
```

O, si el servicio es privilegiado y puede registrar datos sensibles:

```
tail -f /var/log/secure
```

Algunos servicios tienen sus propios archivos de registro, un buen ejemplo es `auditd`, el demonio de auditoría de Linux, que tiene sus registros almacenados en `/var/log/audit/`. Si no ve la salida de su servicio en `/var/log/messages`, intente buscar registros específicos del servicio en `/var/log/`.

Sección 10.2: Arrancar y parar servicios

En sistemas que utilizan scripts de inicio del tipo System-V, como RHEL/CentOS 6:

```
servicio< servicio> inicio
```

```
servicio< servicio> parada
```

En sistemas que utilizan systemd, como Ubuntu (Server y Desktop)>= 15.04, y RHEL/CentOS>= 7:

```
systemctl< servicio> dnsmasq
```

```
systemctl< servicio> dnsmasq
```

Sección 10.3: Obtener el estado de un servicio

En sistemas que utilizan scripts de inicio del tipo System-V, como RHEL/CentOS 6:

```
servicio< servicio> estado
```

En sistemas que utilizan systemd, como Ubuntu (Server y Desktop)>= 15.04, y RHEL/CentOS>= 7.0:

```
systemctl status < service>
```

Capítulo 11: Modificación de usuarios

Parámetro

Detalles

nombre de usuario El nombre del usuario. No use mayúsculas, no use puntos, no lo termine en guión, no debe incluir dos puntos, sin caracteres especiales. No puede empezar por un número.

Apartado 11.1: Establecer su propia contraseña

contraseña

Sección 11.2: Establecer la contraseña de otro

Ejecute lo siguiente como root:

nombre de usuario `passwd`

Sección 11.3: Añadir un usuario

Ejecute lo siguiente como root:

`useradd` nombre de usuario

Apartado 11.4: Eliminar un usuario

Ejecute lo siguiente como root:

`userdel` nombre de usuario

Sección 11.5: Eliminar un usuario y su carpeta de inicio

Ejecute lo siguiente como root:

`userdel -r` nombredeusuario

Sección 11.6: Listado de grupos a los que pertenece el usuario actual

grupos

Puede encontrar información más detallada sobre los ID numéricos de usuario y grupo con el comando `id`.

Sección 11.7: Listado de grupos a los que pertenece un usuario

`grupos` nombre de usuario

Puede encontrar información más detallada sobre los identificadores numéricos de usuario y grupo con `id` nombreusuario.

Capítulo 12: Pila LAMP

LAMP (Linux Apache MySQL PHP) se compone del sistema operativo Linux como entorno de desarrollo, el servidor HTTP Apache como servidor web, el sistema de gestión de bases de datos relacionales MySQL (RDBMS) como sistema de base de datos y el lenguaje de programación PHP como lenguaje de programación del lado del servidor (back-end).

LAMP se utiliza como una pila de tecnologías de código abierto solución para el área de desarrollo web. La versión para Windows de esta pila se llama WAMP (Windows Apache MySQL PHP).

Sección 12.1: Instalación de LAMP en Arch Linux

Con esta línea instalaremos todos los paquetes necesarios en un solo paso, y la última actualización:

```
pacman -Syu apache php php-apache mariadb
```

HTTP

Editar

```
/etc/httpd/conf/httpdconf
```

Cambie ServerAdmin you@example.com según sus necesidades.

La carpeta de las Páginas WEB por defecto es ServerRoot `"/etc/httpd"`. El directorio debe establecerse en la misma carpeta, así que cambie la línea

```
<Directory "/etc/httpd" >
```

Esta carpeta debe tener acceso de lectura y ejecución, por lo que

```
chmod o+x /etc/httpd
```

Cambie AllowOverride de none (por defecto) a All para que el .htaccess funcione.

Ahora necesitas la carpeta `/public_html` para cada usuario. (para obtener la página raíz de cada usuario como `http://localhost/~yourusername/`. Desmarque esta línea:

```
Incluir conf/extra/httpd-userdir.conf
```

Ahora como root necesitas crear el `/public_html` para cada usuario y cambiar el acceso a (755) de cada .

```
chmod 755 /home
chmod 755 /home/nombredeusuario
chmod 755 /home/username/public_html
```

Puede comentar esta línea si desea utilizar SSL:

```
LoadModule ssl_module modules/mod_ssl.so
```

Si necesita utilizar dominios virtuales, descomente la línea:

```
Incluir conf/extra/httpd-vhosts.conf
```

y en `/etc/httpd/conf/extra/httpd-vhosts.conf` debes añadir todos los dominios virtuales. (más en `/etc/hosts` si quieres probar esos virtuales)

Edite `/etc/httpd/conf/extra/httpd-default.conf` y cambie `ServerSignature` a `Off` y `ServerToken` a `Prod` para ocultar datos críticos.

PHP

Edite: `/etc/httpd/conf/httpd.conf`

: `LoadModule mpm_event_module modules/mod_mpm_event.so`

Descomentar: `LoadModule mpm_prefork_module modules/mod_mpm_prefork.so`

Como último elemento en la lista `LoadModule`, añada `LoadModule php7_module modules/libphp7.so`

Como último elemento de la lista `include`, añada `Include conf/extra/php7_module.conf`

Editar `/etc/php/php.ini`

Descomentar `extension=mysqli.so` y `extension=pdo_mysql.so`

Cambia la zona horaria según necesites, por ejemplo:

```
date.timezone= America/Argentina/Buenos_Aires, date.default_latitude= 0.0, date.default_longitude  
= 0.0
```

MySQL

Ejecutar como root:

```
mysql_install_db --user=mysql --basedir=/usr --datadir=/var/lib/mysql
```

Ahora tienes la raíz del servidor MySQL. Inicie el

demonio MySQL:

```
systemctl enable mysqld  
systemctl start mysqld
```

Por fin, corre:

```
sh /usr/bin/mysql_secure_installation
```

Todo eso para conseguir un servidor web listo para ser personalizado como necesites.

Sección 12.2: Instalación de LAMP en Ubuntu

Instalar apache:

```
sudo apt-get install apache2
```

Instalar MySql:

```
sudo apt-get install mysql-server
```

Instalar PHP:

```
sudo apt-get install php5 libapache2-mod-php5
```

Reinicia el sistema:

```
sudo systemctl restart apache2
```

Compruebe la instalación de PHP:

```
php -r 'echo "\n\nSu instalaci        ón de PHP funciona correctamente.\n\n";'
```

Sección 12.3: Instalación de la pila LAMP en CentoOS

Instalar el servidor web Apache

El primer paso es instalar el servidor web Apache.

```
sudo yum -y install httpd
```

Una vez instalado, active (para que se ejecute al inicio) e inicie el servicio de servidor web Apache.

```
sudo systemctl enable --now httpd
```

Dirija su navegador a: <http://localhost>

Verá la página por defecto del servidor web Apache.

Instalar el servidor MariaDB

El segundo paso es instalar MariaDB:

```
sudo yum -y install mariadb-server
```

A continuación, inicie y habilite (al iniciar) el servidor MariaDB:

```
sudo systemctl enable --now mariadb
```

Si es necesario, utilice `mysql_secure_installation` para asegurar su base de datos. Este script le permitirá hacer lo siguiente:

- Cambiar la contraseña del usuario root
- Eliminar las bases de datos de prueba
- Desactivar el acceso remoto

Instalar PHP

```
sudo yum -y install php php-common
```

A continuación, reinicie el servicio httpd de Apache.

```
sudo systemctl restart httpd
```

Para probar PHP, cree un archivo llamado `index.php` en

`/var/www/html`. A continuación, añada la siguiente línea al archivo:

A continuación, dirija su navegador a:

<http://localhost/index.php>

Debería ver información relacionada con su servidor. Si no es así, asegúrese de que php está instalado correctamente ejecutando el siguiente comando:

```
php --versi ón
```

Si recibes algo como:

```
PHP 5.4.16 (cli) (construido: Nov 6 2016 00:29:02) Copyright (c) 1997-2013 The PHP Group
```

Entonces PHP está instalado correctamente. Si este es el caso, por favor asegúrese de ha reiniciado su servidor web.

Capítulo 13: el comando tee

Opciones	Descripción
-a, --append	Añadir a los ARCHIVOS dados. No sobrescribir.
-i, --ignore-interrupts	Ignora las señales de interrupción.
--ayuda	Muestra un mensaje de ayuda y sale.
--versión	Muestra información sobre la versión y sale.

tee - leer de la entrada estándar y escribir en la salida estándar y archivos.

El comando tee debe su nombre al separador en T de la fontanería, que divide el agua en dos direcciones y tiene forma de T mayúscula.

tee copia datos de la entrada estándar a cada ARCHIVO, y también a la salida estándar. En efecto, tee duplica su entrada, dirigiéndola a múltiples salidas a la vez.

Sección 13.1: Escribir la salida a stdout, y también a un fichero

El siguiente comando muestra la salida sólo en la pantalla (stdout).

```
$ ls
```

El siguiente comando escribe la salida sólo en el archivo y no en la pantalla.

```
$ ls > archivo
```

El siguiente comando (con la ayuda del comando **tee**) escribe la salida tanto en la pantalla (stdout) como en el archivo.

```
$ ls | archivo tee
```

Sección 13.2: Escribir la salida desde el medio de una cadena de tuberías a un archivo y pasarlo de nuevo a la tubería.

También puede utilizar el comando **tee** para almacenar la salida de un comando en un archivo y redirigir la misma salida a otro comando.

El siguiente comando escribirá las entradas crontab actuales en un archivo crontab-backup.txt y pasará las entradas crontab al comando **sed**, que realizará la sustitución. Después de la sustitución, se añadirá como una nueva tarea cron.

```
$ crontab -l | tee crontab-backup.txt | sed 's/old/new/' | crontab -
```

Sección 13.3: escribir la salida en varios archivos

Puede enviar la salida a varios archivos (incluido el terminal) utilizando **tee** de la siguiente manera:

```
$ ls | tee fichero1 fichero2 fichero3
```

Sección 13.4: Ordenar al comando tee que añada al fichero

Por defecto, el comando **tee** sobrescribe el archivo. Puede ordenar a **tee** que añada datos al archivo utilizando la opción **-a** como se muestra a continuación

abajo.

```
$ ls | tee -a archivo
```

Capítulo 14: Secure Shell (SSH)

Un shell seguro se utiliza para acceder remotamente a un servidor desde un cliente a través de una conexión cifrada. OpenSSH se utiliza como alternativa a las conexiones Telnet, que consiguen un acceso shell remoto pero sin cifrar. El cliente OpenSSH está instalado por defecto en la mayoría de las distribuciones GNU/Linux y se utiliza para conectarse a un servidor. Estos ejemplos muestran cómo utilizar la suite SSH para aceptar conexiones SSH y conectarse a otro host.

Apartado 14.1: Conexión a un servidor remoto

Para conectarnos a un servidor debemos utilizar SSH en el cliente de la siguiente manera,

```
# ssh -p puerto usuario@dirección-servidor
```

- port - El puerto ssh de escucha del servidor (puerto 22 por defecto).
- user - Debe ser un usuario existente en el servidor con privilegios SSH.
- Dirección del servidor - La IP/Dominio del servidor.

En un ejemplo real, imaginemos estás creando una página web. La compañía que elegiste para alojar tu sitio te dice que el servidor está localizado en web-servers.com en un puerto personalizado de 2020 y tu nombre de cuenta usr1 ha sido elegido para crear un usuario en el servidor con privilegios SSH. En este caso el comando SSH utilizado sería el siguiente

```
# ssh -p 2020 usr1@web-servers.com
```

Si el nombre de cuenta en el sistema remoto es el mismo que el del cliente local, puede omitir el nombre de usuario. Así que si usted es usr1 en ambos sistemas, entonces usted puede simplemente utilizar web-servers.com en lugar de usr1@web-servers.com.

Cuando no pueda acceder directamente un servidor al que desea conectarse, puede intentar utilizar el conmutador ProxyJump para conectarse a él a través de otro servidor al que pueda acceder y que pueda conectarse al servidor deseado.

```
# ssh -J usr1@10.0.0.1:2020 usr2@10.0.0.2 -p 2222
```

Esto le permitirá conectarse al servidor 10.0.0.2 (ejecutando ssh en el puerto 2222) a través del servidor en 10.0.0.1 (ejecutando ssh en el puerto 2020). Por supuesto, necesitarás tener cuentas en ambos servidores. Ten en cuenta también que el parámetro -J se introdujo en la versión 7.3 de OpenSSH.

Sección 14.2: Instalación del paquete OpenSSH

Tanto la conexión a un servidor SSH eliminado como la aceptación de conexiones SSH requieren la instalación de openssh

Debian:

```
# apt-get install openssh
```

Arch Linux:

```
# pacman -S openssh
```

Yum:

```
# yum install openssh
```

Sección 14.3: Configurar un servidor SSH para aceptar conexiones

Primero debemos editar el archivo de configuración del demonio SSH. Aunque en diferentes distribuciones de Linux puede estar localizado en diferentes directorios, normalmente se almacena en `/etc/ssh/sshd_config`

Utilice su editor de texto para cambiar los valores establecidos en este archivo, todas las líneas que comienzan con `#` se comentan y deben tener este carácter eliminado para tener algún efecto. A continuación se muestra una lista de recomendaciones.

```
Puerto (elija un número entre 0 - 65535, normalmente mayor de cuatro dígitos)
PasswordAuthentication yes
PermitirUsuariosusuario1 usuario2 ...etc
```

Tenga en cuenta que es preferible desactivar los inicios de sesión con contraseña y utilizar claves SSH para mejorar la seguridad, como se explica en este documento.

Apartado 14.4: Conexión sin contraseña (utilizando un par de claves)

En primer lugar necesitarás tener un par de claves. Si aún no tienes uno, echa un vistazo al tema 'Generar clave pública y privada'.

Tu par de claves está compuesto por una clave privada (`id_rsa`) y una clave pública (`id_rsa.pub`). Todo lo que tienes que hacer es copiar la clave pública en el host remoto y añadir su contenido al archivo `~/.ssh/authorized_keys`.

Una forma sencilla de hacerlo es

```
ssh< usuario>@<ssh-servidor > 'cat>> ~/.ssh/authorized_keys' < id_rsa.pub
```

Una vez que la clave pública esté colocada correctamente en el directorio principal de su usuario, sólo tiene que iniciar sesión utilizando la clave privada correspondiente:

```
ssh< usuario>@<ssh-servidor > -i id_rsa
```

Sección 14.5: Generar clave pública y privada

Para generar claves para el cliente SSH:

```
ssh-keygen [-t rsa | rsa1 | dsa ] [-C< comment> ] [-b bits ]
```

Por ejemplo:

```
ssh-keygen -t rsa -b 4096 -C myemail@email.com
```

La ubicación por defecto es `~/.ssh/id_rsa` para la clave privada y `~/.ssh/id_rsa.pub`

para la pública. Para más información, visite man.openbsd.org

Sección 14.6: Desactivar el servicio ssh

Esto desactivará el servicio del lado del servidor SSH, ya que si es necesario esto asegurará que los clientes no puedan conectarse a través de ssh

Ubuntu

```
sudo service ssh stop
```

```
sudo systemctl disable sshd.service
```

Debian

```
sudo /etc/init.d/ssh stop  
sudo systemctl disable sshd.service
```

Arch Linux

```
sudo killall sshd  
sudo systemctl disable sshd.service
```


Capítulo 15: SCP

Sección 15.1: Copia Segura

El comando scp se utiliza para copiar de forma segura un archivo hacia o desde un destino remoto. Si el archivo se encuentra directamente en el sistema de trabajo actual, basta con el nombre del archivo; de lo contrario, se requiere una ruta completa que incluya el nombre del host remoto, por ejemplo, `remote_user@some_server.org:/path/to/file`.

Copie el archivo local en su CWD al nuevo directorio

```
scp archivolocal.txt /home/amigo/compartir/
```

Copiar el archivo remoto al directorio de trabajo actual

```
scp rocky@arena51.net:/home/rocky/game/data.txt . /
```

Copiar archivo de una ubicación remota a otra ubicación remota

```
scp mars@universe.org:/beacon/light/bitmap.conf jupiter@universe.org:/beacon/night/
```

Para copiar directorios y subdirectorios utilice la opción recursiva '-r' de scp

```
scp -r user@192.168.0.4: ~/proyecto/* ./espacio de trabajo/
```

Sección 15.2: Uso básico

```
# Copy remote file to local dir
scp user@remotehost.com:/remote/path/to/foobar.md /local/dest

# Copy local file to remote dir
scp foobar.md user@remotehost.com:/remote/dest

# Se pueden usar ficheros de claves (igual que ssh)
scp -i mi_clave.pem foobar.md user@remotehost.com:/remote/dest
```

Capítulo 16: GnuPG (GPG)

GnuPG es un sofisticado sistema de gestión de claves que permite firmar o cifrar datos de forma segura. GPG es una herramienta de línea de comandos utilizada para crear y manipular claves GnuPG.

GnuPG es el más utilizado para tener conexiones SSH (Secure Shell) sin contraseña o cualquier medio de autenticación interactiva, lo que mejora el nivel de seguridad significativamente.

Las siguientes secciones describen las formas de crear, utilizar y mantener la seguridad de las claves GnuPG.

Sección 16.1: Exportar su clave pública

Para que tu par de claves pública-privada sea útil, debes poner tu clave pública a disposición de los demás. Asegúrate de que estás trabajando con tu clave pública, ya que *nunca* debes compartir tu clave privada. Puedes exportar tu clave pública con el siguiente comando:

```
gpg -armor -export DIRECCIÓN_EMAIL> public_key.asc
```

donde EMAIL_ADDRESS es la dirección de correo electrónico asociada a la clave

Alternativamente, puedes subir tu clave pública a un servidor de claves públicas como keys.gnupg.net para que otros puedan . Para ello, introduce lo siguiente en un terminal:

```
gpg -list-keys
```

A continuación, busque la cadena de 8 dígitos (el ID primario) asociada a la clave que desea exportar. A continuación, emita el comando:

```
gpg -send-keys PRIMARY_ID
```

donde PRIMARY_ID es el ID real de esa clave.

Ahora, la clave pública se ha subido al servidor de claves y está a disposición del público.

Sección 16.2: Crear y utilizar rápidamente una clave GnuPG

Instala haveged (ejemplo `sudo apt-get install haveged`) para acelerar el proceso de bytes aleatorios. Entonces:

```
gpg --gen-key  
gpg --list-keys
```

salidas:

```
pub      2048R/NNNNNNNN 2016-01-01  
uid           Nombre name@example.com  
sub      2048R/xxxxxxx 2016-01-01
```

Pues publícalo:

```
gpg --keyserver pgp.mit.edu --send-keys NNNNNNNNN
```

Entonces planea revocar: <https://www.hackdiary.com/2004/01/18/revoking-a-gpg-key/>

Capítulo 17: Configuración de la red

Este documento cubre los aspectos básicos de las redes TCP/IP, la administración de redes y la configuración del sistema. Linux puede soportar múltiples dispositivos de red. Los nombres de los dispositivos están numerados y comienzan en cero y cuentan hacia arriba. Por ejemplo, un ordenador con dos NICs tendrá dos dispositivos etiquetados como eth0 y eth1.

Sección 17.1: Resolución DNS local

Archivo: `/etc/hosts` contiene una lista de hosts que deben ser resueltos localmente(no por

DNS) Ejemplo de contenido del archivo:

```
127.0.0.1          tu-nombre-de-nodo.tu-dominio.com localhost.localdomain localhost
XXX.XXX.XXX.XXX   nombre-nodo
```

El formato del archivo hosts está especificado en [el RFC 952](#)

Sección 17.2: Configurar servidores DNS para la resolución de nombres de dominio

Archivo: `/etc/resolv.conf` contiene una lista de servidores DNS para la resolución de nombres

de dominio Muestra el contenido del archivo:

```
nameserver 8.8.8.8   # dirección IP del servidor de nombres primario
nameserver 8.8.4.4   # dirección IP del servidor de nombres secundario
```

En el caso de un servidor DNS interno, puede validar si este servidor resuelve los nombres DNS correctamente utilizando el comando **dig**:

```
$ dig google.com @tu.dns.servidor.com +short
```

Sección 17.3: Ver y manipular rutas

Manipular la tabla de enrutamiento IP utilizando **route**

Mostrar tabla de enrutamiento

```
$ route # Muestra la lista de rutas y también resuelve los nombres de host
$ route -n # Muestra la lista de rutas sin resolver los nombres de host para obtener resultados más rápidos.
```

Añadir/Borrar ruta

Opción	Descripción
<code>añadir</code> o <code>suprimir</code>	Añadir o eliminar una ruta
<code>-host x.x.x.x</code>	Añadir ruta a un único host identificado por la dirección IP.
<code>-red x.x.x.x</code>	Añadir ruta a una red identificada por la dirección de red.
<code>gw x.x.x.x</code>	Especifique la máscara de red de la pasarela de red x.x.x.x
<code>red x.x.x.x</code>	Especifique la máscara de red de la red por defecto
<code>por defecto</code>	Añada una ruta por defecto

Ejemplos:

- añadir ruta a un host\$ **route** add -host x.x.x.x eth1
- añadir ruta a una red\$ **route** add -net 2.2.2.0 netmask 255.255.255.0 eth0
- Alternativamente, también puede utilizar el formato cidr para añadir una ruta a la **red route** add -net 2.2.2.0/24 eth0
- add default gateway\$ **route** add default gw 2.2.2.1 eth0
- eliminar una ruta\$ **route** del -net 2.2.2.0/24

Manipular la tabla de enrutamiento IP utilizando **ip**

Mostrar tabla de enrutamiento

```
$ ip route show # Listar tabla de enrutamiento
```

Añadir/Borrar ruta

Opción	Descripción
add or del or change or append o sustituir	Cambiar a ruta
mostrar o borrar	el comando muestra el contenido de las tablas de enrutamiento o lo elimina
restaurar	restaurar la información de la tabla de enrutamiento desde stdin
obtener	este comando obtiene una única ruta a un destino e imprime su contenido exactamente como lo ve el núcleo

Ejemplos:

- Establezca la puerta de enlace predeterminada en 1.2.3.254\$ **ip route** add default via 1.2.3.254
- Añade una ruta por defecto (para todas las direcciones) a través de la puerta de enlace local 192.168.1.1 que se puede alcanzar en el dispositivo eth0
\$ **ip route** add default via 192.168.1.1 dev eth0

Sección 17.4: Configurar un nombre de host para otro sistema de la red

Puede configurar su sistema Linux (o macOS) para vincular un identificador **<hostname>** a la dirección IP de algún otro sistema de su red. Puede configurarlo:

- En todo el sistema. Debe modificar el archivo `/etc/hosts`. Sólo tienes que añadir a ese archivo una nueva línea que contenga:
 1. la dirección IP del sistema remoto **<ip_rem>**,
 2. uno o más espacios en blanco, y
 3. el identificador **<hostname>**.
- Para un solo usuario. Deberá modificar el archivo `~/.hosts` --- tendrá que . No es tan sencillo como para todo el sistema. [Aquí](#) puedes ver una explicación.

Por ejemplo, puedes añadir esta línea utilizando la herramienta **cat** de Unix. Supongamos que quieres hacer un **ping** a un PC en tu red local cuya dirección IP es 192.168.1.44 y quieres referirte a esa dirección IP sólo por `remote_pc`. Entonces debes escribir en tu shell

```
$ sudo cat 192.168.1.44 remote_pc
```

Entonces usted puede hacer que el ping sólo por:

```
$ ping_pc_remoto
```

Sección 17.5: Detalles de la interfaz

Ifconfig

Enumerar todas las interfaces disponibles en la máquina

```
$ ifconfig -a
```

Lista los detalles de una interfaz

especifica Sintaxis: `$ ifconfig<`

interface > Ejemplo:

```
$ ifconfig eth0
eth0      Link encap:Ethernet HWaddr xx:xx:xx:xx:xx:xx inet
          addr:x.x.x.x Bcast:x.x.x.x Mask:x.x.x.x
          inet6 addr: xxxx::xxx:xxxx:xxxx/64 Scope:Link UP
          BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX paquetes:4426618 errores:0 abandonados:1124 desbordamientos:0
          trama:0 TX paquetes:189171 errores:0 abandonados:0
          desbordamientos:0 portadora:0 colisiones:0 txqueuelen:1000
          RX bytes:382611580 (382,6 MB) TX bytes:36923665 (36,9 MB)
          Interrupt:16 Memory:fb5e0000-fb600000
```

Ethtool - consulta el controlador de red y la configuración del hardware

Sintaxis: `$ ethtool< interface >`

Ejemplo:

```
$ ethtool eth0

Configuración para eth0:
Puertos compatibles: [ TP ]
Modos de enlace admitidos: 10baseT/Media
10baseT/Completa 100baseT/Media 100baseT/Completa
1000baseT/Completo
Admite el uso de tramas de pausa: No
Admite negociación automática: Sí
Modos de enlace anunciados: 10baseT/Media 10baseT/Completa
100baseT/Media 100baseT/Completa
1000baseT/Completo
Uso de trama de pausa anunciado: No
Autonegociación anunciada: Sí
Velocidad: 1000Mb/s
Dúplex: Completo
Puerto: Par trenzado
PHYAD: 1
Transceptor: interno
Autonegociación: activado
MDI-X: activado (auto)
Soporta Wake-on: pumbg
Wake-on: g
Nivel de mensaje actual: 0x00000007 (7)
drv probe link
Enlace detectado: sí
```

ip - mostrar / manipular enrutamiento, dispositivos, políticas de enrutamiento y túneles

Sintaxis: `$ ip { link | ... | route | macsec } (consulte man ip para ver la lista completa de`

objetos) Ejemplos

Lista de interfaces de red

```
$ ip link show
```

Cambiar el nombre de la interfaz eth0 a wan

```
$ ip link set dev eth0 name wan
```

Subir (o bajar) la interfaz eth0

```
$ ip link set dev eth0 up
```

Lista de direcciones para interfaces

```
$ ip addr show
```

Añadir (o eliminar) ip y máscara (255.255.255.0)

```
$ ip add add 1.2.3.4/24 brd+ dev eth0
```

Sección 17.6: Añadir IP a una interfaz

La dirección IP de una interfaz puede obtenerse mediante DHCP o asignación estática.

DHCP Si está conectado a una red con un servidor DHCP en ejecución, el comando `dhclient` puede obtener una dirección IP para su interfaz

```
$ dhclient < interfaz >
```

o alternatively, puedes hacer un cambio en el archivo `/etc/network/interfaces` para que la interfaz se active al arrancar y obtenga la IP DHCP

```
auto eth0
iface eth0 inet dhcp
```

Configuración estática (cambio permanente) mediante el archivo `/etc/network/interfaces`

Si desea configurar estáticamente los ajustes de la interfaz (cambio permanente), puede hacerlo en el archivo Archivo `/etc/network/interfaces`.

Ejemplo:

```
auto eth0 # Activa la interfaz en el arranque
iface eth0 inet dirección
    estática 10.10.70.10
    máscara de red 255.255.0.0
    pasarela 10.10.1.1
    dns-nameservers 10.10.1.20
    dns-nameservers 10.10.1.30
```

Estos cambios persisten incluso después de reiniciar el sistema.

Configuración estática (cambio temporal) mediante la utilidad `ifconfig`

Se puede añadir una dirección IP estática a una interfaz utilizando la utilidad `ifconfig` de la siguiente manera

```
$ ifconfig< interfaz >< dirección-ip>/<máscara > up
```

Por ejemplo:

```
$ ifconfig eth0 10.10.50.100/16 up
```

Capítulo 18: Comandante de medianoche

Midnight Commander o mc es un gestor de archivos de consola. Este tema incluye la descripción de sus funcionalidades y ejemplos y consejos de cómo en todo su potencial.

Sección 18.1: Teclas de función del Midnight Commander en modo navegación

Aquí hay una lista de acciones que pueden ser activadas en el modo de exploración del sistema de archivos de Midnight Commander usando las teclas de función de tu teclado.

- F1** Muestra la ayuda
- F2** Abre el menú de
- F3** usuario
- F4** Muestra el contenido del archivo seleccionado
- F4** Abre el archivo seleccionado en el editor de archivos interno
- F5** Copia el fichero seleccionado en el directorio abierto en el segundo panel
- F6** Desplaza el fichero seleccionado al directorio abierto en el segundo panel
- F7** Crea un nuevo directorio en el directorio abierto en el panel actual
- F8** Elimina el fichero o directorio seleccionado
- F9** Se centra en el menú principal de la parte superior de
- F10** pantalla Sale de mc

Sección 18.2: Teclas de función de Midnight Commander en el modo de edición de archivos

Midnight Commander tiene un editor incorporado que se inicia con la tecla de función F4 cuando se encuentra sobre el archivo deseado en el modo browse. También puede invocarse en modo autónomo ejecutando

```
mcedit < nombrearchivo >
```

Esta es una lista de acciones que pueden activarse en el modo de edición.

- F1** Muestra la ayuda
- F2** Guarda el archivo
- F3** actual
- F4** Marca el inicio de la selección de texto. Mueva el cursor en cualquier dirección para seleccionar. La segunda pulsación marca el final de la selección.
- F4** Abre el cuadro de diálogo de búsqueda/sustitución de texto
- F5** Copia el texto seleccionado en la posición del cursor
- F6** (copiar/pegar) Desplaza el texto seleccionado a la posición del
- F7** cursor (cortar/pegar) Abre el cuadro de diálogo de búsqueda de
- texto

F8 Borra el texto seleccionado

F9 Se centra en el menú principal de la parte superior de la

F10 pantalla Sale del editor

Capítulo 19: Cambiar de raíz (chroot)

Cambiar raíz (chroot) es una operación que cambia el directorio raíz aparente para el proceso en ejecución actual y sus hijos. Un programa que se ejecuta en un entorno modificado de este tipo no puede acceder a archivos y comandos fuera de ese árbol de directorios del entorno.

Sección 19.1: Requisitos

- privilegios de root
- otro entorno Linux operativo, como el arranque desde Live CD o una distribución existente
- coincidencia de las arquitecturas de entorno de origen y destino **de chroot** (compruebe la arquitectura de entorno actual con **uname -m**)
- los módulos del kernel que pueda necesitar en un entorno **chroot** deben cargarse (por ejemplo, con **modprobe**)

Sección 19.2: Cambiar manualmente la raíz de un directorio

1. Asegúrese de que cumple todos los requisitos, según Requirements
2. Montar los sistemas de archivos temporales de la API:

```
cd /ubicación/de/nuevo/root
mount -t proc proc/ mount -
-rbind /sys sys/ mount --
rbind /dev dev/
mount --rbind /run run/ (opcionalmente )
```

3. Si necesita utilizar una conexión a Internet en el entorno chroot, copie los detalles DNS:

```
cp /etc/resolv.conf etc/resolv.conf
```

4. Cambia root a /ubicación/de/nuevo/root, especificando el shell (**/bin/bash** en este ejemplo):

```
chroot /ubicación/de/nuevo/root /bin/bash
```

5. Después del chrooting puede ser necesario cargar la configuración local de bash:

```
fuentes /etc/profile fuentes~
/.bashrc
```

6. Opcionalmente, crea un prompt único para poder diferenciar tu entorno chroot:

```
export PS1= "(chroot) $PS1"
```

7. Cuando termines con el chroot, puedes salir de él mediante:

```
salida
```

8. Desmonta los sistemas de archivos temporales:

```
cd /
umount --recursive /ubicación/de/nuevo/root
```

Sección 19.3: Razones para usar chroot

El cambio de root se realiza habitualmente para llevar a cabo el mantenimiento del sistema en sistemas en los que ya no es posible arrancar y/o iniciar sesión.

Ejemplos comunes son:

- reinstalación del gestor de arranque
 - reconstrucción de la imagen initramfs
 - actualizar o degradar paquetes•
- restablecer una contraseña olvidada
- creación de software en un entorno raíz limpio

Capítulo 20: Gestores de paquetes

Sección 20.1: Cómo actualizar paquetes con el gestor de paquetes apt

La herramienta avanzada de paquetes, llamada acertadamente gestor de paquetes 'apt', puede gestionar la instalación y eliminación de software en Debian, Slackware y otras distribuciones de Linux. A continuación se muestran algunos ejemplos sencillos de uso:

actualización

Esta opción recupera y escanea los archivos Packages.gz, de forma que la información sobre paquetes nuevos y actualizados esté disponible. Para ello, introduzca el siguiente comando:

```
sudo apt-get update
```

actualizar

Esta opción se utiliza para instalar las versiones más recientes de todos los paquetes actualmente instalados en el sistema. Los paquetes actualmente instalados con nuevas versiones disponibles se recuperan y actualizan; en ningún caso se eliminan los paquetes actualmente instalados, ni se recuperan e instalan los paquetes que aún no están instalados. Para actualizar, introduzca el siguiente comando:

```
sudo apt-get upgrade
```

dist-upgrade

Además de realizar la función de actualización, dist-upgrade también gestiona de forma inteligente las dependencias cambiantes con las nuevas versiones de los paquetes. Si es necesario, intentará actualizar los paquetes más importantes a expensas de los menos importantes. Para , introduzca el siguiente comando:

```
sudo apt-get dist-upgrade
```

Sección 20.2: Cómo instalar un paquete con el gestor de paquetes pacman

Para buscar paquetes en la base de datos, buscando tanto en los nombres como en las descripciones de los paquetes:

```
pacman -Ss cadena1 cadena2 ...
```

Para instalar un solo paquete o una lista de paquetes (incluidas las dependencias), ejecute el siguiente comando:

```
sudo pacman -S nombre_paquete1 nombre_paquete2 ...
```

fuentes

Sección 20.3: Cómo actualizar paquetes con el gestor de paquetes pacman

Para actualizar un programa específico:

```
sudo pacman -S < nombrePrograma >
```

Para actualizar todo el sistema:

```
sudo pacman -Syu
```

Sección 20.4: Cómo actualizar paquetes con yum

Yellowdog Updater, Modified, uno de los últimos vestigios de Yellow Dog Linux, es el gestor de paquetes utilizado por los sistemas Red Hat, Fedora y CentOS y sus derivados. Puede manejar la instalación y eliminación de software empaquetado como rpms para estas distribuciones de Linux. A continuación se muestran algunos ejemplos sencillos de uso:

busque en

Este comando intentará localizar paquetes de software en los repositorios de software configurados que coincidan con los criterios de búsqueda dados, y mostrará el nombre / versión / ubicación del repositorio de las coincidencias que encuentre. Para utilizarlo, introduzca el siguiente comando:

```
yum search< queryString >
```

instale

Este comando intentará localizar e instalar el software nombrado desde los repositorios de software configurados, localizando e instalando recursivamente también cualquier software prerequisite necesario. Para utilizarlo, introduzca el siguiente comando:

```
sudo yum install< nombredelpaquete>
```

actualización

Esta opción se utiliza para instalar las versiones más recientes de todos los paquetes instalados actualmente en el sistema. Los paquetes actualmente instalados con nuevas versiones disponibles se recuperan y actualizan; también se recuperan e instalan nuevos prerequisites según sea necesario, y se eliminan los paquetes reemplazados u obsoletos. Para actualizar, introduzca el siguiente comando:

```
sudo yum update
```

A diferencia de apt, la mayoría de los comandos yum también comprobarán automáticamente si hay actualizaciones en los metadatos del repositorio si no se ha realizado una comprobación recientemente (o si se ha forzado a hacerlo) y recuperarán y escanearán los metadatos actualizados para que la información sobre paquetes nuevos y actualizados esté disponible antes de realizar la operación solicitada.

Capítulo 21: Compilación del núcleo Linux

Sección 21.1: Compilación del núcleo Linux en Ubuntu

Atención: asegúrese de que dispone de al menos 15 GB de espacio libre en disco.

Compilación en Ubuntu >=13.04

Opción A) Usar Git

Usa git si quieres mantenerte sincronizado con el último código fuente del kernel de Ubuntu. Puede encontrar instrucciones detalladas en la Guía Git del núcleo. El repositorio git no incluye los archivos de control necesarios, por lo que debe construirlos por:

```
fakeroot debian/rules clean
```

Opción B) Descargar el archivo fuente

Descargar el archivo fuente - Esto es para los usuarios que quieren reconstruir los paquetes estándar de Ubuntu con parches adicionales. Utilice un comando follow para instalar las dependencias de compilación y extraer el código fuente (en el directorio actual):

1. Instale los siguientes paquetes:

```
sudo apt-get build-dep linux-image-`uname -r`
```

Opción C) Descargar el paquete fuente y compilar

Esto es para los usuarios que quieran modificar, o jugar, con el código fuente del kernel parcheado de Ubuntu.

1. Recupera la última fuente del kernel de kernel.org.
2. Extrae el archivo a un directorio y cd en él:

```
tar xf linux-*.tar.xz
cd linux-*
```

3. Construye la interfaz de configuración de ncurses:

```
hacer menuconfig
```

4. Para aceptar la configuración por defecto, pulse para resaltar < Salir de > y, a continuación, pulse .
5. Pulse de nuevo para guardar la configuración.
6. Usa **make** para construir el kernel:

```
escriba
```

Tenga en cuenta que puede utilizar la bandera -jem> para compilar archivos en paralelo y aprovechar múltiples núcleos.

La imagen comprimida del kernel puede encontrarse en **arch/[arch]/boot/bzImage**, donde **[arch]** es igual a **uname -a**.

Créditos

Muchas gracias a todas las personas de Stack Overflow Documentation que ayudaron a proporcionar este contenido, se pueden enviar más cambios aweb@petercv.com para que se publiquen o actualicen nuevos contenidos.

7theo.tk	Capítulo 1
Aaron Skomra	Capítulo 16
Ajay Sangale	Capítulos 1 y 9
Anagh Hegde	Capítulos 4 y 14
Ani Menon	Capítulos 2 y 4
Arden Shackelford	Capítulo 12
Armalj	Capítulo 1
Baard Kopperud	Capítulo 8
BrightOne	Capítulos 9, 13, 14 y 19
C.W.Holeman II	Capítulo 14
caped114	Capítulos 1 y 4
colelemonz	Capítulo 1
ctafur	Capítulo 17
DaveM	Capítulo 2
depperm	Capítulo 1
e.dan	Capítulo I
incrustado	Capítulo 8
Emmanuel Mathi	Capítulo 4
EsmaeelE	Capítulo 1
fdeslaur	Capítulo 3
Federico Ponzi	Capítulo 9
Filipe	Capítulo 14
Cables de llama	Capítulo 10
FOP	Capítulo 12
foxtrot9	Capítulo 7
geek1011	Capítulo 11
Jarryd	Capítulo 1
Jensd	Capítulos 1 y 4
KerDam	Capítulo 1
Kiran Vemuri	Capítulos 6 y 17
kuldeep mishra	Capítulo 13
Léo Léopold Hertz ☐☐	Capítulo 21
lardenn	Capítulos 2 y 4
leeor	Capítulo 9
likewhoa	Capítulo 1
manav m	Capítulo 15
Manuel	Capítulo 14
Marsso	Capítulo 8
Mateusz Piotrowski	Capítulo 1
mattdm	Capítulos 2, 10 y 11
mertyildiran	Capítulo 5
Mike P	Capítulo 1
Mohammad	Capítulo 1
Nathan Osman	Capítulo 21
Naveen Chakravarthy	Capítulo 1
Nikhil Raj	Capítulo 2
No22	Capítulo 8

<u>oznek</u>	Capítulo 4
<u>Paradoja</u>	Capítulos 16 y 20
<u>parkydr</u>	Capítulo 7
<u>Philip Kirkbride</u>	Capítulos 4, 7, 12 y 20
<u>Cuáquero</u>	Capítulo 4
<u>Rajesh Rengaraj</u>	Capítulo 14
<u>Riley Guerin</u>	Capítulo 15
<u>Rubio</u>	Capítulos 1, 4 y 20
<u>S.Rohit</u>	Capítulos 3, 6 y 12
<u>Sava</u>	Capítulos 1, 4, 14 y 18
<u>Sergey Stolyarov</u>	Capítulo 2
<u>Sudip Bhandari</u>	Capítulo 1
<u>Teddy</u>	Capitulo 2
<u>Tejus Prasad</u>	Capítulo 1
<u>TiansHUo</u>	Capítulo 1
<u>Todd</u>	Capítulos 1 y 14
<u>vishram0709</u>	Capítulo 1
<u>Whoami</u>	Capitulo 1
<u>Y4Rv1K</u>	Capítulos 6 y 17
<u>Zumo de Vidrio</u>	Capitulo 1
<u>zyio</u>	Capítulo 5

También le puede interesar

