Red Hat Linux 9

Manual de referencia de Red Hat Linux

Red Hat Linux 9: Manual de referencia de Red Hat Linux


Red Hat, Inc.

1801 Varsity Drive
Raleigh NC 27606-2072
USA
Teléfono: +1 919 754 3700
Teléfono: 888 733 4281
Fax: +1 919 754 3701
PO Box 13588
Research Triangle Park NC
27709 USA

rhl-rg(ES)-9-Print-RHI (2003-02-13T19:20)

La marca de GPG de la clave security@redhat.com es:
CA 20 86 86 2B D6 9D FC 65 F6 EC C4 21 91 80 CD DB 42 A6 0E

Tabla de contenidos

Introducción

Bienvenido al *Manual de referencia de Red Hat Linux*.

El *Manual de referencia de Red Hat Linux* contiene información muy útil sobre el sistema Red Hat Linux. Desde conceptos fundamentales, tales como la estructura del sistema de archivos de Red Hat Linux, a temas más delicados como la seguridad del sistema y el control de autenticación, esperamos que este manual sea un recurso valioso para usted.

Este manual es para usted si desea aprender un poco más sobre el funcionamiento de su sistema Red Hat Linux. Podrá profundizar en los siguientes temas:

- Estructura del sistema de archivos
- Proceso de arranque
- El sistema X Window
- Herramientas de seguridad
- Servicios de red

1. Cambios realizados en este manual

Este manual ha sido reorganizado y actualizado con las últimas características de Red Hat Linux 9. Algunos cambios son los siguientes:

Se ha actualizado el capítulo de El sistema X Window

> *El sistema X Window* se ha revisado completamente y ha sido reorganizado para mayor claridad. Se ha añadido nuevas instrucciones sobre configuraciones de fuentes.

Nuevo capítulo para `sysconfig`

> La sección `sysconfig` del capítulo *Proceso de arranque, inicio y cierre* ha sido expandida y convertida en su propio capítulo.

Se ha actualizado el capítulo de TCP Wrappers y `xinetd`

> El capítulo *TCP Wrappers y* `xinetd` ha sido revisado y reorganizado completamente para mayor claridad.

El capítulo Usuarios y grupos ha sido actualizado

> El capítulo *Usuarios y grupos* ha sido clarificado, actualizado y reorganizado.

El capítulo Interfaces de red ha sido actualizado

> El capítulo *Interfaces de red* ha sido actualizado y reorganizado.

Se ha actualizado el capítulo Servidor Apache HTTP

> La guía para la migración desde la versión 1.3 a la versión 2.0 de Servidor Apache HTTP ha sido actualizado. La lista de opciones de configuración han sido modificadas y reorganizadas. Se agradece muy especialmente a **Gary Benson** y a **Joe Orton** por su esfuerzo en la preparación de la guía de migración de Servidor Apache HTTP.

Antes de leer este manual, debería estar familiarizado con los contenidos del *Manual de instalación de Red Hat Linux* concerniente a los temas de instalación, el *Manual del principiante de Red Hat Linux* para los conceptos básicos de Linux y el *Manual de personalización de Red Hat Linux* para

instrucciones generales de personalización. El *Manual de referencia de Red Hat Linux* contiene información sobre tópicos para los usuarios avanzados.

Las versiones HTML y PDF de todos los manuales de Red Hat Linux están disponibles en línea en: http://www.redhat.com/docs

Nota

Aunque este manual contiene la información más actual, lea las *Notas de última hora* de Red Hat Linux si desea obtener más información. Se encuentran en el CD #1 de Red Hat Linux y en:

`http://www.redhat.com/docs/manuals/linux`

2. Cómo encontrar la documentación apropiada

Necesita documentación apropiada a su nivel de experiencia con Linux. De lo contrario, se sentirá abrumado o no encontrará la información necesaria para responder a sus dudas. El *Manual de referencia de Red Hat Linux* trata de aspectos más técnicos y de opciones de su sistema Red Hat Linux. Esta sección le ayudará a decidir, dependiendo de la información que necesite, si leer este manual u otros manuales Red Hat Linux, incluidos los recursos en línea.

Se pueden establecer tres categorías de personas que usan Red Hat Linux, e intentar ser más explícitos en cuanto a la documentación y fuentes de información necesarias. Puede empezar viendo el nivel de conocimiento que tiene:

Nuevo en Linux

Nunca ha usado el sistema operativo Linux o similar; o tiene muy pocos datos acerca de él. Tiene o no ha tenido experiencia usando otros sistemas operativos (como por ejemplo Windows). ¿Es ésta su situación? Si es así, por favor salte a Sección 2.1.

Alguna experiencia con Linux

Ha instalado con éxito Linux y lo ha usado con anterioridad (pero no Red Hat Linux). O bien ha tenido experiencias equivalentes con otros sistemas operativos parecidos a Linux ¿Se encuentra usted entre este tipo de personas? Si es así, vaya a Sección 2.2

Usuario avanzado

Ha instalado y usado Red Hat Linux con éxito en otras ocasiones. Si es así lea Sección 2.3

2.1. Documentación para usuarios principiantes de Linux

Para alguien nuevo en Linux, la cantidad de información disponible sobre cada tema, como imprimir, arrancar el sistema o particionar su disco duro, puede ser abrumadora. Es conveniente que primero adquiera una buena base de conocimientos centrados en cómo funciona Linux antes de entrar en temas más avanzados.

Su primer objetivo debería ser el de obtener documentación útil. De lo contrario se sentiría frustrado nada más empezar.

Trate de adquirir el siguiente tipo de documentación:

- *Una breve historia de Linux* — Muchos aspectos de Linux están ligados a precedentes históricos. Un poco de cultura sobre Linux puede ser útil a la hora de solventar problemas potenciales antes de que surjan.
- *Explicación acerca de cómo funciona Linux* — Aunque no es necesario profundizar en la mayoría de los aspectos del kernel Linux, es conveniente saber algo sobre cómo ha surgido Linux. Puede ser especialmente importante si está trabajando con otros sistemas operativos, ya que algunas de las suposiciones que tiene sobre cómo funcionan los ordenadores pueden no cumplirse en Linux.
- *Una revisión general de los comandos (con ejemplos)* — Probablemente esto es lo más importante a buscar en la documentación linux. La filosofía de Linux es que es mejor usar pequeños comandos conectados de diferentes modos, que utilizar pocos comandos amplios (y complejos) que hagan todo el trabajo por si mismos. Sin algunos ejemplos que ilustren el acercamiento a Linux para hacer cosas, se puede sentir intimidado por el gran número de comandos disponibles en el sistema Red Hat Linux.

  Tenga en cuenta que no tiene que memorizar todos los comandos Linux. Existen diversas técnicas para ayudarle a encontrar el comando específico que necesita para realizar un tarea determinada. Tan sólo necesita saber el modo en que Linux funciona, lo que necesita llevar a cabo y cómo acceder a la herramienta que le dará las instrucciones exactas para ejecutar el comando.

El *Manual de instalación de Red Hat Linux* constituye una referencia excelente de ayuda para instalar y configurar con éxito su sistema Red Hat Linux. El *Manual del principiante de Red Hat Linux* cubre los comandos de sistema básicos, el entorno de escritorio gráfico y otros muchos conceptos fundamentales. Debería empezar con estos dos libros y usarlos para conseguir una base de conocimiento sobre su sistema Red Hat Linux. Verá como después los conceptos más complicados empezarán a tener sentido, una vez que tenga las conceptos básicos claros.

Además de leer los manuales Red Hat Linux, existen otras fuentes excelentes de documentación disponibles por poco dinero o gratis.

2.1.1. Introducción a sitios Web Linux

- http://www.redhat.com — En el sitio Web, podrá encontrar enlaces al Proyecto de documentación Linux (LDP), versiones en línea de los manuales de Red Hat Linux, versiones en línea de las FAQs (preguntas y respuestas más frecuentes), una base de datos que puede ayudarle en la búsqueda de Grupos de usuarios Linux cercanos a usted, información técnica en la base de conocimientos de soporte de Red Hat y mucho más.
- http://www.linuxheadquarters.com — El sitio Web de la sede central de Linux le ofrece guías fáciles para una variedad de tareas Linux.

2.1.2. Introducción a los grupos de noticias de Linux

Puede participar en los grupos de noticias viendo las conversaciones de otros intentando solventar problemas, o bien puede participar activamente preguntando y contestando. Los usuarios experimentados de Linux son famosos por ser extremadamente colaboradores cuando tratan de ayudar a nuevos usuarios con problemas de Linux — especialmente si sus preguntas van a parar al punto de reunión justo. Si no tiene acceso a una aplicación de lector de noticias, puede acceder a esta información vía web en http://groups.google.com/. Existen docenas de grupos de noticias relacionadas a Linux, incluyendo las siguientes:

- linux.help — Un buen lugar donde encontrar ayuda de compañeros usuarios de Linux.
- linux.redhat — Este newsgroup cubre aspectos específicos a Red Hat Linux.
- linux.redhat.install — Para preguntas sobre instalación o para ver cómo otros han resuelto problemas similares.

- linux.redhat.misc — Preguntas o peticiones de ayuda que no encajan en ninguna de las categorías tradicionales.
- linux.redhat.rpm — Sitio donde dirigirse si tiene problemas con el uso de **RPM** para conseguir algún propósito en particular.

2.1.3. Libros sobre Linux

- *Red Hat Linux for Dummies, 2ª edición* de Jon "maddog" Hall; IDG
- *Special Edition Using Red Hat Linux* de Alan Simpson, John Ray y Neal Jamison; Que
- *Running Linux* de Matt Welsh y Lar Kaufman; O'Reilly & Associates
- *Red Hat Linux 8 Unleashed* por Bill Ball y Hoyle Duff; Pearson Education

Los libros aquí sugeridos constituyen fuentes excelentes de información para un conocimiento básico del sistema Red Hat Linux. Para una información más detallada sobre los diversos temas que aparecerán a través del libro, muchos de los capítulos listan títulos de libros específicos, habitualmente en la parte de *Recursos adicionales*.

2.2. Para los más experimentados

Si ha utilizado otras distribuciones Linux, tendrá un dominio básico de los comandos usados más frecuentemente. Puede que haya instalado su propio sistema Linux e incluso haya descargado y creado software que ha encontrado en Internet. Después de instalar Linux, no obstante, los puntos sobre configuración pueden ser confusos.

The *Manual de personalización de Red Hat Linux* está diseñado para ayudar a explicar los diversos modos en que su sistema Red Hat Linux puede ser configurado. Utilice este manual para aprender las opciones de configuración y cómo ponerlas en práctica.

Cuando instale software que no aparezca en el *Manual de personalización de Red Hat Linux*, le será útil ver lo que otra gente en las mismas circunstancias ha hecho. Los documentos HOWTO del LDP (proyecto de documentación de Linux), disponibles en http://www.redhat.com/mirrors/LDP/HOWTO/HOWTO-INDEX/howtos.html, documentan aspectos particulares de Linux, desde cambios del kernel hasta el uso de Linux en una estación de trabajo "radio amateur".

2.3. Documentación para gurús de Linux

Si es un usuario Red Hat Linux desde hace tiempo, sabrá probablemente que uno de los mejores modos para entender un programa en particular es leyendo su código fuente y/o sus archivos de configuración. Una gran ventaja de Red Hat Linux es la disponibilidad total del código fuente.

Obviamente, no todo el mundo es programador, por lo que puede que el código fuente no le sea de gran ayuda. Sin embargo, si tiene los conocimientos y la habilidad necesarias para leerlo, el código fuente alberga todas las respuestas.

3. Convenciones del documento

Cuando lea este manual, verá que algunas palabras están representadas en fuentes, tipos de letra, tamaño y peso diferentes. Esta forma de evidenciar es sistemática; se representan diferentes palabras

con el mismo estilo para indicar su pertenencia a una categoría específica. A continuación tiene una lista de los tipos de palabras representados de una manera determinada:

`comando`

Los comandos en Linux (y otros sistemas operativos) se representan de esta manera. Este estilo le indica que puede escribir la palabra o frase en la línea de comandos y pulsar [Intro] para aplicar el comando. A veces un comando contiene palabras que aparecerían con un estilo diferente si fueran solas (p.e, nombres de archivos). En estos casos, se las considera como parte del comando, de manera que toda la frase aparece como un comando. Por ejemplo:

Utilice el comando `cat testfile` para ver el contenido de un archivo, llamado `testfile`, en el directorio actual.

`nombre del archivo`

Los nombres de archivos, nombres de directorios, rutas y nombres de rutas y paquetes RPM aparecen siempre en este modo. Este estilo indica que un archivo o directorio en particular existe con ese nombre en su sistema Red Hat Linux. Ejemplos:

El archivo `.bashrc` en su directorio principal contiene definiciones de la shell de bash y alias para su propio uso.

El archivo `/etc/fstab` contiene información sobre diferentes dispositivos del sistema y sistemas de archivos.

Instale el RPM `webalizer` si quiere utilizar un programa de análisis del archivo de registro del servidor Web.

aplicación

Este estilo indica que el programa es una aplicación de usuario final (lo contrario a software del sistema). Por ejemplo:

Use **Mozilla** para navegar por la Web.

[tecla]

Una tecla del teclado aparece en el siguiente estilo. Por ejemplo:

Para utilizar [Tab], introduzca un carácter y pulse la tecla [Tab]. Aparecerá una lista de archivos en el directorio que empiezan con esa letra. Su terminal visualizará la lista de archivos en el directorio que empieza con esa letra.

[tecla]-[combinación]

Una combinación de teclas aparece de la siguiente manera. Por ejemplo:

La combinación de teclas [Ctrl]-[Alt]-[Backspace] le hará salir de la sesión gráfica y volver a la pantalla gráfica de login o a la consola.

texto de una interfaz gráfica (GUI)

Un título, palabra o frase dentro de una pantalla o ventana de interfaz gráfica GUI aparecerá de la siguiente manera. La finalidad del texto escrito en este estilo es la de identificar una pantalla GUI o un elemento e una pantalla GUI en particular (p.e, un texto relacionado con una casilla de verificación o un campo). Ejemplos:

Seleccione la casilla de verificación **Pedir contraseña** si quiere que su salvapantallas pida una contraseña antes de terminar.

nivel superior de un menú en una pantalla o ventana GUI

Cuando vea una palabra con este estilo, significa que la palabra está en el nivel superior de un menú desplegable. Si hace click sobre la palabra en la pantalla GUI, aparecerá el resto del menú. Por ejemplo:

Bajo **archivo** en una terminal de GNOME verá los siguientes elementos en el menú: opción **Nueva pestaña** que le permite abrir múltiples intérpretes de comandos de la shell en la misma ventana.

Si tiene que escribir una secuencia de comandos desde un menú GUI, aparecerán como en el siguiente ejemplo:

Vaya a **Botón del menú principal** (en el Panel) => **Programación** => **Emacs** para iniciar el editor de textos **Emacs**.

botón en una pantalla o ventana GUI

Este estilo indica que el texto se encuentra en un botón que se pulse en una pantalla GUI. Por ejemplo:

Pulse el botón **Anterior** para volver a la última página Web que haya visitado.

`salida de pantalla`

Cuando vea el texto en este estilo, significa que verá una salida de texto en la línea de comandos. Verá respuestas a comados que haya escrito, mensajes de error e intérpretes de comandos para la entrada de datos durante los scripts o programas mostrados de esta manera. Por ejemplo:

Utilice `ls` para visualizar los contenidos de un directorio:

```
$ ls
Desktop                about.html       logs          paulwesterberg.png
Mail                   backupfiles      mail          reports
```

La salida de pantalla que le devuelvan como respuesta al comando (en este caso, el contenido del directorio) se mostrará en este estilo.

`intérprete de comandos`

El intérprete de comandos es el modo en el que el ordenador le indica que está preparado para que usted introduzca datos, aparecerá con el siguiente estilo. Ejemplos:

```
$
#
[stephen@maturin stephen]$
leopard login:
```

`entrada del usuario`

El texto que el usuario tiene que escribir, ya sea en la línea de comandos o en una casilla de texto de una pantalla GUI, se visualizará en este estilo. En el siguiente ejemplo, **`text`** se visualiza en este estilo:

Para arrancar su sistema en modo texto de su programa de instalación, necesitará escribir en el comando **`text`** en el intérprete de comandos `boot:`.

Adicionalmente, usamos diferentes tipos de estrategias para llamar su atención para determinados tipos de información. Dependiendo de lo importante que esta información sea para su sistema, estos elementos serán marcados como nota, atención o aviso. Por ejemplo:

Nota

Recuerde que Linux es sensible a mayúsculas y minúsculas. En otras palabras, rosa no es lo mismo que ROSA o rOsA.

Sugerencia

El directorio `/usr/share/doc` contiene documentación adicional para paquetes instalados en su sistema.

Importante

Si modifica el archivo de configuración DHCP, los cambios no surtirán efecto hasta que el demonio DHCP se reinicie.

Atención

No lleve a cabo tareas rutinarias como root — utilice una cuenta de usuario normal a menos que necesite usar una cuenta de usuario para administrar su sistema.

Aviso

Si escoge no particionar de forma manual, una instalación de tipo servidor borrará todas las particiones ya existentes en los discos duros instalados. No escoja este tipo de instalación a menos que esté seguro de que no desea guardar los datos.

4. Uso del ratón

Red Hat Linux está diseñado para utilizar un ratón de tres botones. Si tiene un ratón de dos botones, debería haber seleccionado la emulación de tres botones durante el proceso de instalación. Si pulsa los dos botones a la vez, estará emulando el tercer botón, el del medio.

Si se le indica que pulse un elemento con el ratón, se da por descontado que nos referimos al botón izquierdo. Si necesita utilizar el botón del medio o el de la derecha, le será indicado explícitamente. (Esto será justamente lo contrario si ha configurado su ratón para que sea usado por una persona zurda.)

La frase "arrastrar y soltar" le debería ser familiar. Si se le indica que arrastre y suelte un elemento en su escritorio gráfico, haga click en el elemento y mantenga el botón del ratón pulsado. Mientras que lo mantiene pulsado, suelte el elemento moviendo el ratón a una nueva ubicación, dejando de presionar el botón para soltar el elemento.

5. Copiar y pegar un texto con X

Copiar y pegar un texto es fácil mediante el uso del ratón y del sistema X Window. Para copiar un texto, sencillamente haga click con el ratón y arrástrelo a lo largo del texto para evidenciarlo. Para pegar el texto en algún otro sitio, pulse el botón central del ratón en el lugar en el que quiere copiar el texto.

6. Y además...

El *Manual de referencia de Red Hat Linux* forma parte del compromiso que Red Hat tiene con los usuarios al proporcionarles soporte útil y puntual de Red Hat Linux. Las ediciones futuras contendrán más información sobre los cambios en la estructura del sistema y en la organización, nuevas y potentes herramientas de seguridad y otros recursos para ayudar a expandir la potencia del sistema Red Hat Linux — y su capacidad para usarlo.

That is where you can help.

6.1. Necesitamos su opinión

Si encuentra algún error en el *Manual de referencia de Red Hat Linux*, o si piensa que se necesitan hacer cambios, por favor mándenos su opinión a Bugzilla (http://bugzilla.redhat.com/bugzilla) contra el componente *rhl-rg*.

Asegúrese de mencionar el indentificador del manual:

```
rhl-rg(ES)-9-Print-RHI (2003-02-13T19:20)
```

Así podremos saber de qué versión del manual se trata.

Si tiene alguna sugerencia, descríbanosla y si ha encontrado algún error indique la sección y parte del texto en el que se encuentre para facilitarnos la búsqueda.

7. Regístrese para el soporte

Si tiene una edición de Red Hat Linux 9, recuerde que para beneficiarse de las ventajas que le corresponden como cliente de Red Hat, deberá registrarse.

Tiene derecho a disfrutar las siguientes ventajas, dependiendo del producto Red Hat Linux que haya comprado:

- Soporte Red Hat — Obtenga ayuda con las preguntas de instalación del equipo de soporte de Red Hat, Inc..
- Red Hat Network — Actualice de forma sencilla los paquetes y reciba avisos de seguridad personalizados para su sistema. Vaya a http://rhn.redhat.com para más detalles.
- *Under the Brim: Boletín de Red Hat* — Obtenga mensualmente las últimas noticias e información sobre el producto directamente desde Red Hat.

Para registrarse vaya a http://www.redhat.com/apps/activate/. Encontrará el ID de su producto en una tarjeta negra, roja y blanca dentro de la caja de su Red Hat Linux.

Para leer más acerca del soporte técnico para Red Hat Linux remítase al apéndice *Obtener soporte técnico* en el *Manual de instalación de Red Hat Linux*.

¡ Buena suerte y gracias por haber escogido Red Hat Linux!

El equipo de documentación de Red Hat

I. Referencia del sistema

Para administrar el sistema efectivamente, es de suma importancia conocer sus componentes y cómo ellos funcionan juntos. Esta parte resalta muchos aspectos importantes del sistema. Cubre el proceso de arranque, la distribución básica del sistema de archivos, la ubicación de directorios y archivos del sistema importantes y y los conceptos básicos detrás de los usuarios y grupos. Adicionalmente, se explica en detalle el sistema X Window.

Tabla de contenidos

redhat.

Capítulo 1.

Proceso de arranque, inicio y cierre del sistema

Una de las características más importantes y poderosas de Red Hat Linux es el método abierto y configurable para el inicio y cierre del sistema operativo. Los usuarios son libres de configurar muchos aspectos del proceso de arranque, incluyendo qué programas se lanzarán al momento de arranque. De forma parecida, el cierre del sistema finaliza los procesos de forma organizada y configurable, aunque la personalización de este proceso casi nunca es necesaria.

Entender el funcionamiento del proceso de arranque y cierre no solo le permitirá personalizar fácilmente Red Hat Linux, sino que también le facilitará resolver problemas relacionados con el inicio y el cierre del sistema.

1.1. Proceso de arranque

A continuación obtendrá las etapas básicas del proceso de arranque para un sistema x86:

1. La BIOS del sistema comprueba y lanza la primera etapa del gestor de arranque del MBR del disco duro primario.
2. La primera etapa del gestor de arranque se autocarga en memoria y lanza la segunda etapa del gestor de arranque desde la partición `/boot/`.
3. La segunda etapa del gestor de arranque carga el kernel en memoria, lo cual en su momento carga los módulos necesarios y monta la partición root para sólo-lectura.
4. El kernel transfiere el control del proceso de arranque al programa `/sbin/init`.
5. El programa `/sbin/init` carga todos los servicios y herramientas de espacio del usuario y monta todas las particiones listadas en `/etc/fstab`.
6. El usuario se presenta con un intérprete de comandos de registro para el sistema Linux apenas arrancado.

Ya que la configuración del proceso de arranque es más común que la del proceso de cierre, en el resto del capítulo se discutirá el modo en el que el proceso de arranque funciona y cómo se puede personalizar para satisfacer sus necesidades.

1.2. Vista detallada del proceso de arranque

El inicio del proceso de arranque varía dependiendo de la plataforma de hardware usada. Sin embargo, una vez que se encuentra el kernel y se carga el sistema, el proceso de arranque por defecto es idéntico a través de todas las arquitecturas. Este capítulo se basa en la arquitectura x86.

1.2.1. La BIOS

Cuando un ordenador x86 se carga, el procesador busca al final de la memoria del sistema por *Basic Input/Output System* o el programa *BIOS* y lo ejecuta. La BIOS controla no sólo el primer paso del proceso de arranque, sino que también proporciona una interfaz de bajo nivel para dispositivos periféricos. Por este motivo se escribe tan sólo en modo lectura, memoria permanente y está siempre disponible para el uso.

Otras plataformas usan programas diferentes para ejecutar tareas a bajo nivel equivalentes a aquellas de la BIOS en el sistema x86. Por ejemplo, los ordenadores basados en Itanium usan *Interfaz de Firmware extensible* (*EFI*) *Shell*, mientras que los sistemas Alpha usan *SRM console*.

Una vez que se haya cargado, la BIOS chequea los periféricos y localiza un dispositivo con el que arrancar el sistema. Habitualmente, en primer lugar comprueba cualquier disquete y unidades de CD-ROM presente por los medios de arranque, y a continuación si esto falla, echa un vistazo a las unidades de disco duro del sistema. El orden de las unidades necesario para arrancar puede ser controlado con una configuración de la BIOS. La BIOS carga en memoria cualquier programa que resida en el primer sector de este dispositivo, llamado *Master Boot Record* o *MBR*. La MBR sólo tiene 512 bytes de tamaño y contiene las instrucciones de código de máquina para el arranque del equipo, llama un gestor de arranque así como también la tabla de particiones. Una vez que la BIOS haya encontrado y cargado el gestor de arranque en memoria, le deja el control del proceso de arranque a éste.

1.2.2. El gestor de arranque

Esta sección revisa los gestores de arranque para la plataforma x86. Dependiendo de la arquitectura del sistema, el proceso de arranque diferirá ligeramente. Consulte Sección 1.2.2.1 para una descripción general de los gestores de arranque para otras arquitecturas.

Bajo Red Hat Linux están disponibles dos gestores de arranque: *GRUB* o *LILO*. GRUB es el gestor de arranque por defecto, pero LILO está disponible para los usuarios que lo necesiten o prefieran. Para obtener más información sobre la configuración y el uso de GRUB o LILO, consulte Capítulo 2.

Los gestores de arranque de Linux para la plataforma x86 se dividen en dos etapas. La primera es un código binario de máquina pequeña en el MBR. Su única función es la de localizar el gestor de arranque de la segunda etapa y cargar la primera parte de éste en memoria.

GRUB es el gestor de arranque más nuevo y tiene la ventaja de ser capaz de leer particiones ext2 y ext3 [1] y cargar su archivo de configuración — `/boot/grub/grub.conf` — al momento de arranque. Consulte Sección 2.7 para detalles sobre cómo modificar este archivo.

Con LILO, la segunda etapa del gestor de arranque es usar la información del MBR para determinar las opciones de arranque disponibles para el usuario. Esto significa que cada vez que se produzca un cambio en la configuración o actualice el kernel de forma manual, debe ejecutar el comando `/sbin/lilo -v -v` para escribir la información apropiada al MBR. Para obtener más detalles sobre como realizarlo, consulte Sección 2.8.

Sugerencia

Si actualiza el kernel mediante el uso de **Agente de actualización de Red Hat**, el archivo de configuración es actualizado automáticamente. Se puede encontrar más información sobre Red Hat Network en el siguiente URL: https://rhn.redhat.com.

Una vez que el gestor de arranque de la segunda etapa está en memoria, presenta al usuario con la pantalla inicial, gráfica de Red Hat Linux mostrando los diferentes sistemas operativos o kernels que para los que ha sido configurado para arrancar. En esta pantalla el usuario puede usar las flechas direccionales para escoger el sistema operativo o kernel con el que desea arrancar y presione la tecla [Intro]. Si no se presiona ninguna tecla, el gestor de arranque carga la selección predeterminada luego de un período de tiempo de espera (también configurable).

1. GRUB lee el sistema de archivos ext3 así como también ext2, sin importar el archivo journal. Consulte el capítulo titulado *El sistema de archivos ext3* en el *Manual de personalización de Red Hat Linux* para más información sobre el sistema de archivos ext3.

Nota

Si ha instalado el soporte para el kernel Symmetric Multi-Processor (SMP), verá más de una opción la primera vez que arranque el sistema. Bajo LILO verá `linux`, el cual es el kernel SMP, y `linux-up`, el cual es para procesadores únicos. GRUB mostrará `Red Hat Linux (<kernel-version>-smp)`, el cual es el kernel SMP, y `Red Hat Linux (<kernel-version>)`, la cual es para procesadores únicos.

Si surge cualquier problema con el kernel SMP, trate de seleccionar un kernel que no sea SMP antes de rearrancar.

Una vez que el gestor de arranque de la segunda etapa haya determinado qué kernel arrancar, localizará el binario del kernel correspondiente en el directorio `/boot/`. El kernel binario es llamado usando el siguiente formato — `/boot/vmlinuz-<kernel-version>` (donde `<kernel-version>` corresponde a la versión del kernel especificada en las configuraciones del gestor de arranque).

Para instrucciones sobre el uso del gestor de arranque para proveer argumentos de comandos al kernel, consulte el Capítulo 2. Para información sobre el cambio del nivel de ejecución en la línea de comandos de GRUB o LILO, vea Sección 2.10.

El gestor de arranque luego coloca la imagen apropiada de *initial RAM disk*, conocida como `initrd`, en la memoria. El `initrd` es usado por el kernel para cargar controladores necesarios para arrancar el sistema. Esto es muy importante si posee unidades de disco duro SCSI o si está usando el sistema de ficheros ext3. [2].

Aviso

No elimine el directorio `/initrd/` del sistema de ficheros bajo ningún concepto. Si lo elimina su sistema le dará un mensaje de error de pánico en el momento de arranque.

Una vez que el kernel y la imagen *initrd* se cargan en memoria, el gestor de arranque controla el proceso de arranque para el kernel.

Para una descripción más detallada sobre los gestores de arranque GRUB y LILO, consulte el Capítulo 2.

1.2.2.1. Gestores de arranque para otras arquitecturas

Una vez que el kernel de Red Hat Linux arranca y pasa el proceso de arranque al comando `init`, los mismos acontecimientos suceden en cada arquitectura exactamente en el mismo modo. La única diferencia entre el proceso de arranque de cada arquitectura está en la aplicación que se usa para encontrar y cargar el kernel.

Por ejemplo, la arquitectura Alpha usa el gestor de arranque `aboot`, mientras que Itanium usa el gestor de arranque ELILO.

Consulte el *Manual de instalación de Red Hat Linux* específico para estas plataformas para obtener información sobre la configuración de sus gestores de arranque.

2. Para más detalles sobre *initrd*, consulte el capítulo llamado *El sistema de archivos ext3* en el *Manual de personalización de Red Hat Linux*.

1.2.3. El kernel

Cuando el kernel se carga, inmediatamente se inicializa y configura la memoria del ordenador y los diferentes hardware conectado al sistema, incluyendo procesadores, subsistemas de entrada/salida y dispositivos de almacenamiento. A continuación buscará la imagen `initrd` en una ubicación predeterminada en memoria, la descomprimirá, la montará y cargará todos los controladores necesarios. A continuación inicializa los dispositivos virtuales relacionados con el sistema de ficheros, tal como LVM o software RAID antes de desmontar la imagen del disco `initrd` y liberar toda la memoria que la imagen del disco ocupó anteriormente.

El kernel luego crea un dispositivo root, monta la partición root como sólo lectura y libera cualquier memoria no utilizada.

Llegados a este punto, el kernel está cargado en memoria y operativo. Sin embargo, como no hay aplicaciones de usuario que permitan la entrada significativa de datos al sistema, no se puede hacer mucho más.

Para configurar el entorno de usuario, el kernel inicia el programa `/sbin/init`.

1.2.4. Programa `/sbin/init`

El programa `/sbin/init` (también llamado `init`) coordina el resto del proceso de arranque y configura el ambiente del usuario.

Cuando el comando `init` arranca, se vuelve el padre o abuelo de todos los procesos que comienzan automáticamente en el sistema Red Hat Linux. Primero, ejecuta el script `/etc/rc.d/rc.sysinit`, que establece la ruta a otros programas, activa el swap, controla los sistemas de fichero y y se encarga de todo lo que el sistema necesita tener hecho al momento de la inicialización. Por ejemplo, la mayoría de los sistemas usan un reloj, por lo tanto, en ellos, el `rc.sysinit` tendrá una referencia `/etc/sysconfig/clock` para inicializar el reloj. Otro ejemplo es si hay procesos en los puertos seriales especiales que deben ser inicializados, `rc.sysinit` ejecutará el archivo `/etc/rc.serial`.

El comando `init` luego ejecuta el script `/etc/inittab`, que describe cómo el sistema debería configurarse en cada *nivel de ejecución* de *SysV init* [3]. Entre otras cosas, `/etc/inittab` configura el nivel de ejecución por defecto y establece que `/sbin/update` debería de ejecutarse cuando se arranque un nivel de ejecución en concreto. [4].

A continuación, el comando `init` configura la librería de función de fuente, `/etc/rc.d/init.d/functions`, para el sistema. Esto indica el modo en que empezar o matar un programa y cómo determinar el PID del programa.

El programa `init` inicia todos los procesos de fondo buscando en el directorio apropiado `rc` por el nivel de ejecución especificado por defecto en `/etc/inittab`. Los directorios `rc` están numerados para corresponder al nivel de ejecución que represente. Por ejemplo, `/etc/rc.d/rc5.d/` es el directorio para el nivel de ejecuación 5.

Cuando se arranca el nivel de ejecución 5, el programa `init` consulta el directorio `/etc/rc.d/rc5.d/` para determinar qué procesos iniciar o parar.

A continuación un ejemplo de listado del directorio `/etc/rc.d/rc5.d/`:

```
K05innd->../init.d/innd
K05saslauthd->../init.d/saslauthd
K10psacct->../init.d/psacct
K12cWnn->../init.d/cWnn
K12FreeWnn->../init.d/FreeWnn
K12kWnn->../init.d/kWnn
K12mysqld->../init.d/mysqld
```

3. Para más información sobre los niveles de ejecución de SysV init, consulte Sección 1.4.
4. El comando `update` se usa para eliminar buffers contaminados del disco.

```
K12tWnn->../init.d/tWnn
K15httpd->../init.d/httpd
K15postgresql->../init.d/postgresql
K16rarpd->../init.d/rarpd
K20bootparamd->../init.d/bootparamd
K20iscsi->../init.d/iscsi
K20netdump-server->../init.d/netdump-server
K20nfs->../init.d/nfs
K20rstatd->../init.d/rstatd
K20rusersd->../init.d/rusersd
K20rwalld->../init.d/rwalld
K20rwhod->../init.d/rwhod
K24irda->../init.d/irda
K25squid->../init.d/squid
K28amd->../init.d/amd
K34dhcrelay->../init.d/dhcrelay
K34yppasswdd->../init.d/yppasswdd
K35atalk->../init.d/atalk
K35dhcpd->../init.d/dhcpd
K35smb->../init.d/smb
K35vncserver->../init.d/vncserver
K35winbind->../init.d/winbind
K40mars-nwe->../init.d/mars-nwe
K45arpwatch->../init.d/arpwatch
K45named->../init.d/named
K45smartd->../init.d/smartd
K46radvd->../init.d/radvd
K50netdump->../init.d/netdump
K50snmpd->../init.d/snmpd
K50snmptrapd->../init.d/snmptrapd
K50tux->../init.d/tux
K54pxe->../init.d/pxe
K55routed->../init.d/routed
K61ldap->../init.d/ldap
K65identd->../init.d/identd
K65kadmin->../init.d/kadmin
K65kprop->../init.d/kprop
K65krb524->../init.d/krb524
K65krb5kdc->../init.d/krb5kdc
K70aep1000->../init.d/aep1000
K70bcm5820->../init.d/bcm5820
K74ntpd->../init.d/ntpd
K74ups->../init.d/ups
K74ypserv->../init.d/ypserv
K74ypxfrd->../init.d/ypxfrd
K84bgpd->../init.d/bgpd
K84ospf6d->../init.d/ospf6d
K84ospfd->../init.d/ospfd
K84ripd->../init.d/ripd
K84ripngd->../init.d/ripngd
K85zebra->../init.d/zebra
K90isicom->../init.d/isicom
K92ipvsadm->../init.d/ipvsadm
K95firstboot->../init.d/firstboot
S00microcode_ctl->../init.d/microcode_ctl
S05kudzu->../init.d/kudzu
S08ip6tables->../init.d/ip6tables
S08ipchains->../init.d/ipchains
```

```
S08iptables->../init.d/iptables
S09isdn->../init.d/isdn
S10network->../init.d/network
S12syslog->../init.d/syslog
S13portmap->../init.d/portmap
S14nfslock->../init.d/nfslock
S17keytable->../init.d/keytable
S20random->../init.d/random
S24pcmcia->../init.d/pcmcia
S25netfs->../init.d/netfs
S26apmd->../init.d/apmd
S28autofs->../init.d/autofs
S44acpid->../init.d/acpid
S55sshd->../init.d/sshd
S56rawdevices->../init.d/rawdevices
S56xinetd->../init.d/xinetd
S80sendmail->../init.d/sendmail
S80spamassassin->../init.d/spamassassin
S84privoxy->../init.d/privoxy
S85gpm->../init.d/gpm
S90canna->../init.d/canna
S90crond->../init.d/crond
S90cups->../init.d/cups
S90xfs->../init.d/xfs
S95anacron->../init.d/anacron
S95atd->../init.d/atd
S97rhnsd->../init.d/rhnsd
S99local->../rc.local
S99mdmonitor->../init.d/mdmonitor
```

Como puede ver, ninguno de los scripts que inician y cierran los servicios están localizados en el directorio `/etc/rc.d/rc5.d/`. Casi todos los ficheros en `/etc/rc.d/rc5.d/` son *enlaces simbólicos* apuntando a los scripts localizados en el directorio `/etc/rc.d/init.d/`. Los enlaces simbólicos se usan en cada uno de los directorios `rc` de manera que los niveles de ejecución puedan ser reconfigurados al crear, modificar y eliminar los enlaces simbólicos sin que afecte a los scripts actuales a los que se refiere.

El nombre de cada enlace simbólico inicia con `K` o `S`. Los enlaces `K` son procesos eliminados en ese nivel de ejecución, mientras que aquellos que inician por `S` son procesos iniciados.

El comando `init` en primer lugar detiene todos los enlaces simbólicos de `K` en el directorio mediante la ejecución del comando `/etc/rc.d/init.d/<`*`command`*`> stop`, en el que `<`*`command`*`>` es el proceso a matar. A continuación inicia todos los enlaces simbólicos `S` al ejecutar `/etc/rc.d/init.d/<`*`command`*`>. start`.

Sugerencia

Una vez que el sistema haya acabado el arranque podrá registrarse como root y ejecutar los mismos scripts para iniciar y parar los servicios. Por ejemplo, el comando `/etc/rc.d/init.d/httpd stop` paralizará el servidor Web Apache.

Cada uno de los enlaces simbólicos se numera para dictaminar el orden de inicio. Puede cambiar el orden en el que los servicios inician o paran al cambiar el número. Mientras más bajo es el número, más rápido se arrancará. Los enlaces simbólicos con el mismo número se inician de modo alfabético.

Nota

Una de las últimas cosas que el programa `init` ejecuta es el archivo `/etc/rc.d/rc.local`. Este archivo es útil para la personalización del sistema. Consulte Sección 1.3 para más información sobre el uso del archivo `rc.local`.

Después que el comando `init` ha progresado a través del directorio adecuado `rc` para el nivel de ejecución, el script `/etc/inittab` bifurca los procesos `/sbin/mingetty` para cada consola virtual (intérpretes de comando de registro, login) ubicada para el nivel de ejecución. Los niveles de ejecución del 2 al 5 obtienen todas las seis consolas virtuales, mientras que el nivel de ejecución 1 (modo usuario único) obtiene tan sólo uno y lo niveles de ejecución del 0 al 6 no obtienen ninguno. El proceso `/sbin/mingetty` abre las rutas de la comunicación para los dispositivos *tty* [5], establece sus modos, imprime el indicador de inicio de sesión, toma el nombre del usuario, e inicia el proceso de inicio de sesión para el usuario.

En el nivel de ejecución 5, el `/etc/inittab` ejecuta un script llamado `/etc/X11/prefdm`. El script ejecuta `prefdm` ejecuta su gestor de pantalla preferido para X — `gdm`, `kdm`, o `xdm`, dependiendo de los contenidos del archivo `/etc/sysconfig/desktop`.

En este punto, el sistema está operando en el nivel de ejecución nivel 5 y mostrando la pantalla de inicio de sesión.

1.3. Ejecutar programas adicionales en el momento de arranque

El script `/etc/rc.d/rc.local` lo ejecuta el comando `init` en tiempo de arranque, o cuando se cambien niveles de ejecución. El agregar comandos a este script es una forma fácil de realizar tareas necesarias como arrancar servicios especiales o inicializar dispositivos sin tener que escribir scripts complejos de inicialización en el directorio `/etc/rc.d/init.d/` y creando los enlaces simbólicos.

El script `/etc/rc.serial` es usado si se deben configurar puertos seriales en el momento de arranque. Este script ejecuta los comandos `setserial` para configurar los puertos seriales del sistema. Consulte la página man de `setserial` para más información.

1.4. Niveles de ejecución de SysV Init

El sistema de niveles de ejecución SysV init provee de un proceso estándar para controlar cuáles programas `init` lanza o detiene cuando se inicializa un nivel de ejecución. SysV init fué escogido porque es más fácil de usar y más flexible que el proceso tradicional init estilo BSD.

Los ficheros de configuración para SysV init están en el directorio `/etc/rc.d/`. Dentro de este directorio, se encuentran los scripts `rc`, `rc.local`, `rc.sysinit`, y, opcionalmente, los scripts `rc.serial` así como los siguientes directorios:

```
init.d/
rc0.d/
rc1.d/
rc2.d/
rc3.d/
rc4.d/
rc5.d/
rc6.d/
```

5. Consulte Sección 5.3.11 para más información sobre dispositivos `tty`.

El directorio `init.d/` contiene los scripts usados por el comando `/sbin/init` cuando se controlan los servicios. Cada uno de los directorios numerados representa los seis niveles de ejecución predeterminados configurados por defecto bajo Red Hat Linux.

1.4.1. Niveles de ejecución

Los niveles de ejecución son un estado, o *modo*, definido por los servicios listados en el SysV directorio `/etc/rc.d/rc<x>.d/`, donde *<x>* es el número de nivel de ejecución.

La idea detrás de los niveles de ejecución de SysV init gira alrededor del hecho que sistemas diferentes se pueden usar de formas diferentes. Por ejemplo, el servidor corre de forma más eficiente sin tener que arrastrar recursos del sistema creados por el sistema X. Otras veces, el administrador del sistema puede necesitar operar el sistema en un nivel más bajo de ejecución para realizar tareas de diagnóstico, como reparar corrupción del disco duro, cuando no es posible que ningún otro usuario esté usando el sistema.

Las características de un nivel de ejecución dado determinan qué servicios son detenidos o iniciados por `init`. Por ejemplo, el nivel de ejecución 1 (modo único usuario) detiene cualquier servicio de red, mientras que el nivel 3 arranca estos servicios. Asignando servicios específicos a ser detenidos o arrancados en un nivel dado, `init` puede fácilmente cambiar el modo de la máquina sin que el usuario tenga que manualmente arrancar o detener servicios.

Los siguientes niveles de ejecución están definidos por defecto para Red Hat Linux:

- `0` — Parar
- `1` — Modo texto usuario único
- `2` — Sin usar (usuario-definible)
- `3` — Modo texto multiusuario completo
- `4` — Sin usar (usuario-definible)
- `5` — Modo gráfico multiusuario completo (con una pantalla de inicio de sesión basada en X)
- `6` — Rearrancar

Generalmente, los usuarios utilizan Red Hat Linux al nivel de ejecución 3 o nivel de ejecución 5 — ambos modos multiusuario. Ya que los niveles de ejecución 2 y 4 no son usados, los usuarios a veces personalizan estos niveles para cubrir necesidades específicas.

El nivel de ejecución por defecto para el sistema está listado en `/etc/inittab`. Para saber el nivel de ejecución por defecto de un sistema, busque por la línea similar a la que se muestra abajo cerca de la parte superior de `/etc/inittab`:

```
id:5:initdefault:
```

El nivel de ejecución predeterminado en el ejemplo de arriba es cinco, como indica el número después del punto y coma. Para cambiarlo, modifique `/etc/inittab` como usuario root.

Aviso

Tenga mucho cuidado cuando esté modificando `/etc/inittab`. Errores simples de tipeo pueden hacer que su sistema no arranque nuevamente. Si esto ocurre, use un disquete de arranque, entre a modo de usuario único o entre en modo de rescate y repare el archivo.

Para más información sobre los modos de usuario único y de rescate, consulte el capítulo llamado *Modo rescate* en el *Manual de personalización de Red Hat Linux*.

Es posible cambiar al nivel de ejecución por defecto al momento de arranque modificando los argumentos pasados del gestor de arranque al kernel. Para información sobre el cambio de niveles de ejecución al momento de arranque, consulte Sección 2.10.

1.4.2. Utilidades de los niveles de ejecución

Una de las mejores formas de configurar los niveles de ejecución es usando *initscript utility*. Estas herramientas están diseñadas para simplificar las tareas de mantener archivos en la jerarquía del directorio SysV init y descargan a los administradores de sistemas de tener que directamente manipular numerosos enlaces simbólicos en los subdirectorios de `/etc/rc.d/`.

Red Hat Linux ofrece tres de tales utilidades:

- `/sbin/chkconfig` — La utilidad `/sbin/chkconfig` es una herramienta de línea de comandos sencilla para mantener la jerarquía del directorio `/etc/rc.d/init.d`.
- **/sbin/ntsysv** — La utilidad basada en ncurses **/sbin/ntsysv** provee de una interfaz interactiva basada en texto, que muchos encuentran más fácil de usar que `chkconfig`.
- **Herramienta de configuración de servicios** — El programa de interfaz gráfica **Herramienta de configuración de servicios** (`redhat-config-services`) es una utilidad flexible basada en GTK2 para la configuración de niveles de ejecución.

Remítase al capítulo titulado *Control de acceso a servicios* en el *Manual de personalización de Red Hat Linux* para obtener más información relacionada con estas herramientas.

1.5. Apagar

Para apagar Red Hat Linux, el usuario root puede ejecutar el comando `/sbin/shutdown`. La página man para `shutdown` tiene una lista completa de opciones, pero las dos más usadas son:

```
/sbin/shutdown -h now
/sbin/shutdown -r now
```

Después de apagar todo, la opción `-h` detendrá la máquina, y la opción `-r` la reiniciará.

Los usuarios no root pueden usar los comandos `reboot` y `halt` para apagar el equipo mientras se está en niveles de ejecución 1 hasta 5. Sin embargo, no todos los sistemas operativos Linux soportan esta característica.

Si la computadora no se apaga, tenga cuidado de no apagar la computadora hasta que aparezca un mensaje indicando que el sistema ha sido detenido.

Si no espera por este mensaje puede significar que no todas las particiones de discos duros han sido desmontadas, y puede llevar a un sistema de archivos corrupto.

redhat.

Capítulo 2.

Gestores de arranque

Antes de poder ejecutar Red Hat Linux, debe ser cargado en memoria por un programa especial llamado *gestor de arranque*. El programa de gestor de arranque existe en el disco duro primario del sistema (o en otros dispositivos) y es responsable de la carga del kernel de Linux con sus archivos necesarios, o, en algunos casos, de otros sistemas operativos en la memoria.

2.1. Gestores de arranque y arquitectura del sistema

Cada arquitectura de sistemas que pueda ejecutar Red Hat Linux usa un gestor de arranque diferente. Por ejemplo, la arquitectura Alpha usa el gestor de arranque `aboot`, mientras que la arquitectura Itanium usa el gestor de arranque `ELILO`.

Este capítulo explica comandos y opciones de configuración para los dos cargadores de arranque suministrados con Red Hat Linux para la arquitectura x86: GRUB y LILO.

2.2. GRUB

GNU GRand Unified Boot loader o GRUB es un programa que habilita al usuario a seleccionar qué sistema operativo instalado o kernel descargar en el momento de arranque del sistema. Permite también que el usuario transmita argumentos al kernel.

2.2.1. Proceso de arranque en un sistema x86 y GRUB

Esta sección explica con más detalle el papel específico que desempeña GRUB al arrancar un sistema x86. Para hacerse una idea del proceso de arranque, vea Sección 1.2.

GRUB se carga asímismo en la memoria en las diferentes etapas:

1. *La etapa 1 o cargador de arranque primario se lee en la memoria con el BIOS desde el MBR*[1].El gestor de arranque primario existe en menos de 512 bytes de espacio en disco entre el MBR y es capaz de cargar bien sea la etapa 1.5 o la etapa 2 del gestor de arranque.
2. *El gestor de arranque de la etapa 1 lee en la memoria al gestor de arranque de la etapa 1.5. si es necesario* Determinado hardware requiere un paso intermedio para obtener el cargador de arranque de la etapa 2. Esto sucede a menudo cuando la partición `/boot` está por encima de 1024 cilindros de disco duro o cuando se usa el modo LBA. Este cargador de arranque de la etapa 1.5 se encuentra en la partición `/boot` o en una pequeña parte del MBR y la partición `/boot`.
3. *La etapa 2 o el gestor de arranque secundario se lee en la memoria.* El gestor de arranque secundario visualiza el menú GRUB y el entorno del comando. Esta interfaz le permite seleccionar qué sistema operativo o kernel de Linux arrancar, pasar argumentos al kernel o ver los parámetros del sistema, tales como la RAM disponible.
4. *El gestor de arranque secundario lee el sistema operativo o el kernel y* `initrd` *en la memoria.* Una vez que GRUB determina qué sistema operativo iniciar, éste lo carga en la memoria y transfiere el control de la máquina a dicho sistema operativo.

El método de arranque usado para arrancar Red Hat Linux se conoce como método de *carga directa* porque el gestor de arranque carga el sistema operativo directamente. No existe punto medio entre el gestor de arranque y el kernel.

1. Para obtener más información sobre BIOS y el MBR, vaya a Sección 1.2.1.

El proceso de arranque usado por otros sistemas operativos puede variar. Por ejemplo, los sistemas operativos de Microsoft DOS y Windows, así como otros sistemas operativos de propietarios, se cargan mediante un método de arranque de *carga encadenada*. Bajo este método, el MBR señala el primer sector de la partición que tiene el sistema operativo. Allí encuentra los archivos necesarios para arrancar el sistema operativo.

GRUB soporta ambos métodos de arranque, directo y carga encadenada, permitiendo arrancar desde casi cualquier sistema operativo.

Aviso

Durante la instalación, la instalación de DOS de Microsoft y Windows sobreescriben completamente el MBR, destruyendo cualquier cargador de arranque ya existente. Si crea un sistema de arranque dual, es preferible que instale el sistema operativo Microsoft de primero. Para obtener instrucciones sobre como llevarlo a cabo, vea el apéndice titulado *Instalación de Red Hat Linux en un entorno de doble arranque* en el *Manual de instalación de Red Hat Linux*.

2.2.2. Funciones de GRUB

GRUB contiene una serie de funciones que lo convierten en el método favorito respecto al resto de gestores de arranque disponibles para la arquitectura x86. A continuación tiene una lista de las características más importantes:

- *GRUB proporciona un entorno pre-OS basado en comandos verdaderos para máquinas x86.* Esto proporciona una flexibilidad máxima en la carga de los sistemas operativos con determinadas opciones o con la recopilación de información sobre el sistema. Durante muchos años arquitecturas que no son x-86 han usado entornos previos al sistema operativo que permiten arrancar el sistema desde una línea de comandos. Mientras que algunas características del comando están disponbiles con LILO y otros gestores de arranque para x86, GRUB tiene una mayor variedad de características.
- *GRUB soporta el modo Direccionamiento Lógico de Bloques (LBA).* El modo LBA coloca la conversión de direccionamiento utilizada para buscar archivos en la unidad de disco duro del firmware y se utiliza en muchos discos IDE y en todos los discos duros SCSI. Antes de LBA, los gestores de arranque encontraban la limitación del cilindro 1024 del BIOS, donde el BIOS no podía encontrar un archivo después de ese cabezal de cilindro del disco. El soporte LBA permite que GRUB arranque los sistemas operativos desde las particiones más allá del límite de 1024 cilindros, siempre y cuando el BIOS del sistema soporte el modo LBA. Las revisiones más modernas de la BIOS soportan el modo LBA.
- *GRUB puede leer las particiones ext2.* Esto permite que GRUB acceda a su archivo de configuración, `/boot/grub/grub.conf`, cada vez que el sistema arranca, eliminando la necesidad que tiene el usuario de escribir una nueva versión de la primera etapa del gestor de arranque al MBR en caso de que se produzcan cambios de la configuración. El único caso en el que el usuario necesitaría reinstalar GRUB en el MBR es en caso de que la localización física de la partición `/boot` se traslade en el disco. Para más detalles sobre la instalación de GRUB en el MBR, consulte Sección 2.3.

2.3. Instalación de GRUB

Si durante el proceso de instalación de Red Hat Linux no instaló GRUB, se puede hacer después. Una vez instalado se convierte en el gestor de arranque por defecto.

Antes de instalar GRUB, debería asegurarse de que cuenta con el último paquete disponible de GRUB o use el paquete GRUB desde los CD-ROMs de instalación de Red Hat Linux. Para instrucciones sobre la instalación de paquetes, vea el capítulo titulado *Gestión de paquetes con RPM* en el *Manual de personalización de Red Hat Linux*.

Una vez que el paquete GRUB esté instalado, abra un intérprete de comandos de la shell, y ejecute el comando `/sbin/grub-install <location>`, donde `<location>` es la localización de la etapa 1 de GRUB en la que el gestor de arranque debería ser instalado .

El siguiente comando instala GRUB en el MBR del dispositivo IDE maestro en el bus IDE primario: `/sbin/grub-install /dev/hda`

La próxima vez que arranque el sistema, el menú del gestor de arranque gráfico GRUB aparecerá antes del que el kernel se cargue en memoria.

2.4. Terminología de GRUB

Una de las cuestiones más importantes que deben entenderse antes de utilizar GRUB es cómo el programa hace referencia a los dispositivos, por ejemplo, a los discos duros y a las particiones. Esta información es muy importante si desea configurar GRUB para arrancar varios sistemas operativos.

2.4.1. Nombres de dispositivos

Suponga que un sistema tiene más de un disco duro. El primer disco duro del sistema es llamado `(hd0)` por GRUB. La primera partición en ese disco es llamada `(hd0,0)`, y la quinta partición en el segundo disco duro es llamada `(hd1,4)`. En general, la nomenclatura utilizada para los sistemas de archivos al usar GRUB se desglosa del siguiente modo:

`(<type-of-device><bios-device-number>,<partition-number>)`

Los paréntesis y las comas son muy importantes en el nombre. `<tipo-de-dispositivo>` hace referencia a si es un disco duro (`hd`) o una unidad de disquete (`fd`).

`<número-dispositivo-bios>` es el número de dispositivo según la BIOS del sistema, empezando desde `0`. El disco duro IDE principal tiene asignado el número `0` y el disco duro IDE secundario el número `1`. El orden es aproximadamente equivalente al modo en el que el kernel de Linux organiza los dispositivos con letras, donde la letra `a` en `hda` corresponde al número `0`, y la letra `b` en `hdb` corresponde al número `1`, y así sucesivamente.

Nota

El sistema de numeración de GRUB para los dispositivos empieza por `0` y no por `1`. Este es uno de los errores que cometen con más frecuencia los usuarios que empiezan a utilizar GRUB.

`<número-partición>` hace referencia al número de una partición concreta en dicho dispositivo. Al igual que en el caso de `<número-dispositivo- bios>`, la numeración de las particiones empieza por `0`. Aunque la mayoría de las particiones se especifican con números, si el sistema usa particiones BSD a éstas se hará referencia con letras, por ejemplo `a` o `c`.

GRUB usa las reglas siguientes para denominar los dispositivos y las particiones:

- No es relevante si los discos duros que utiliza son IDE o SCSI. Todos los discos duros empiezan con `hd`. Las unidades de disquete empiezan con `fd`.

- Para especificar todo un dispositivo sin respetar sus particiones, simplemente debe suprimir la coma y el número de partición. Esto es importante para indicarle a GRUB que configure el registro MBR para un disco concreto. Por ejemplo, `(hd0)` especifica la MBR en el primer dispositivo y `(hd3)` especifica la MBR en el cuarto dispositivo.
- Si tiene varios discos duros, es muy importante saber el orden de la unidad de arranque de la BIOS. Esto es muy sencillo si sólo tiene discos IDE o SCSI, pero si tiene una combinación de ambos, el asunto se complica un poco.

2.4.2. Nombres de archivos y listas de bloqueo

Al escribir comandos en GRUB que hagan referencia a un archivo, como una lista de menús que debe usarse para permitir el arranque de varios sistemas operativos, debe incluir el archivo inmediatamente después de especificar el dispositivo y la partición.

Una especificación de archivo de ejemplo que haga referencia a un nombre de archivo absoluto se organiza del modo siguiente:

```
(<type-of-device><bios-device-number>,<partition-number>)/path/to/file
```

La mayoría de las veces, un usuario especificará los archivos por la ruta del directorio en esa partición más el nombre del archivo.

También puede especificar archivos a GRUB que no aparecen realmente en el sistema de archivos, tal como un gestor de arranque de cadena que aparece en los primeros bloques de la partición. Para especificar estos archivos, deberá indicar una *lista de bloques*, que indique a GRUB, bloque por bloque, la ubicación exacta del archivo en la partición. Puesto que un archivo puede estar formado por varios conjuntos de bloques, hay un modo específico de escribir listas de bloques. Cada ubicación de sección de archivo se describe con un número de desplazamiento de bloques seguido de un número de bloques de ese punto de desplazamiento, y las secciones se colocan juntas de forma ordenada y separadas por comas.

La siguiente es una lista de bloques de ejemplo:

```
0+50,100+25,200+1
```

Esta lista de bloques indica a GRUB que debe utilizar un archivo que empieza en el primer bloque de la partición y que usa los bloques del 0 al 49, del 99 al 124, y el 199.

Saber cómo escribir listas de bloques es útil al utilizar GRUB para cargar sistemas operativos que usan el método de carga encadenada, como Microsoft Windows. Puede suprimir el número de desplazamiento de bloques si empieza por el bloque 0. Por ejemplo, el archivo de carga encadenada de la primera partición del primer disco duro tendrá el nombre siguiente:

```
(hd0,0)+1
```

Lo siguiente muestra el comando `chainloader` con una designación de lista de bloques similar en la línea de comandos de GRUB después de establecer el dispositivo correcto y la partición adecuada como raíz:

```
chainloader+1
```

2.4.3. Sistema de archivos raíz de GRUB

Algunos usuarios se confunden con el uso del término "sistema de archivos" en GRUB. Es importante recordar que el sistema de archivos raíz de GRUB no tiene nada que ver con el sistema de archivos raíz de Linux.

El sistema de archivos raíz de GRUB es la partición raíz de un dispositivo concreto. GRUB usa esta información para montar el dispositivo y carga los archivos desde el mismo.

Con Red Hat Linux, una vez que GRUB ha cargado la partición raíz (que es lo mismo que la partición `/boot` y contiene el kernel de Linux), el comando `kernel` puede ejecutarse con la localización del archivo del kernel como una opción. Una vez que el kernel de Linux inicia, establece el sistema de archivos raíz con los cuales los usuarios de Linux están familiarizados. El sistema de archivos root de GRUB original y los montajes deben olvidarse en este punto; la única finalidad de su existencia era arrancar el archivo del kernel.

Consulte las notas sobre los comandos `root` y `kernel` en Sección 2.6 para obtener más información.

2.5. Interfaces de GRUB

GRUB dispone de tres interfaces eficaces que proporcionan distintos niveles de funcionalidad. Cada una de estas interfaces le permite arrancar el kernel de Linux u otros sistemas operativos.

Las interfaces son como sigue:

Interfaz de menú

Si el programa de instalación de Red Hat Linux ha configurado automáticamente GRUB, ésta es la interfaz que ya conoce. En esta interfaz hay un menú de sistemas operativos o kernels preconfigurados con sus propios comandos de arranque en forma de lista ordenada por nombre, después de arrancar el sistema por primera vez. Puede utilizar las teclas de flecha para seleccionar una opción en lugar de la selección por defecto y pulsar la tecla [Intro] para arrancar el sistema. Como alternativa, se puede establecer un período de inactividad, de modo que GRUB inicie la carga de la opción por defecto.

Presione la tecla [e] para entrar en la interfaz del editor o la tecla [c] para cargar la interfaz de línea de comandos.

Consulte Sección 2.7 para obtener más información sobre la configuración de esta interfaz.

Interfaz del editor de menú de entrada

Para tener acceso al editor de entradas del menú, presione la tecla [e] desde el menú del gestor de arranque. Los comandos de GRUB de dicha entrada se muestran aquí y puede alterar estas líneas de comandos antes de arrancar el sistema operativo agregando una línea de comandos ([o] inserta una nueva línea después de la línea actual y [O] inserta una nueva línea antes de ella), modificandola ([e]), o borrando una ([d]).

Una vez realizados los cambios, la tecla [b] ejecuta los comandos y arranca el sistema operativo. Con la tecla [Esc] se omiten los cambios y el usuario vuelve a la interfaz de menú estándar. Con la tecla [c] se carga la interfaz de línea de comandos.

Sugerencia

Para más información sobre el cambio de niveles de ejecución con GRUB usando el editor de entradas de menú, consulte Sección 2.10.

Interfaz de línea de comandos

Esta es la interfaz de GRUB más básica, pero también la que proporciona un mayor control. En esta interfaz puede escribir cualquier comando de GRUB seguido de la tecla [Intro] para proceder a la ejecución correspondiente. Esta interfaz cuenta con algunas funciones similares a las de shell avanzadas, incluyendo el uso de [Tab] para autocompletar, y las combinaciones de

teclas con [Ctrl] al escribir comandos, tales como [Ctrl]-[a] para moverse al comienzo de la línea y [Ctrl]-[e] para moverse al final. Además, las teclas de flecha, [Inicio], [Fin], y [Supr] funcionan de forma similar al `bash` shell.

Vaya a Sección 2.6, para obtener una lista de los comandos más comunes.

2.5.1. Orden de uso de Interfaces

Cuando GRUB carga la segunda etapa de su gestor de arranque, primero busca por su archivo de configuración. Cuando lo encuentra, lo utiliza para crear la lista de menú y despliega la interfaz de menú.

Si no puede encontrar el archivo de configuración o si éste no se puede leer, GRUB carga la interfaz de línea de comandos para permitirle al usuario escribir manualmente los comandos necesarios para completar el proceso de arranque.

En el caso de que el archivo de configuración no sea válido, GRUB imprimirá el error y solicitará la introducción de valores. Esto puede ser muy útil, porque podrá ver con exactitud donde está el problema y corregirlo en el archivo. Si pulsa cualquier tecla se volverá a cargar la interfaz de menú, donde podrá modificar la opción de menú y corregir el problema según el error que GRUB haya notificado. Si la corrección falla, GRUB informa del error y puede empezar de nuevo.

2.6. Comandos de GRUB

GRUB permite varios comandos en su línea de comandos. Algunos de los comandos aceptan opciones después del nombre y estas opciones deben ir separadas del comando por comas y de otras opciones de esa línea por caracteres de espacio.

En la lista siguiente se indican los comandos más útiles:

- `boot` — Arranca el sistema operativo o gestor de encadenamiento que se ha especificado y cargado previamente.
- `chainloader` <*`nombre-archivo`*> — Carga el archivo especificado como gestor de encadenamiento. Para extraer el archivo en el primer sector de la partición especificada, puede utilizar `+1` como nombre de archivo.
- `displaymem` — Muestra el uso actual de memoria, en función de la información de la BIOS. Esto es útil si no está seguro de la cantidad de RAM que tiene un sistema y todavía tiene que arrancarlo.
- `initrd` <*`nombre-archivo`*> — Le permite especificar un disco RAM inicial para utilizarlo al arrancar. `initrd` es necesario cuando el kernel necesita ciertos módulos para poder arrancar adecuadamente, tales como cuando la partición se formatea con el sistema de archivos ext3.
- `install` <*`stage-1`*> <*`install-disk`*> <*`stage-2`*> `p` <*`config-file`*> — Instala GRUB en la MBR del sistema.

  Cuando esté usando el comando `install`, el usuario debe especificar lo siguiente:

  - <*`stage-1`*> — Significa un dispositivo, partición y archivo donde el primer gestor de arranque puede ser encontrado, tal como `(hd0,0)/grub/stage1`.
  - <*`install-disk`*> — Especifica el disco donde la etapa 1 del gestor de arranque debería ser instalado, tal como `(hd0)`.
  - <*`stage-2`*> — Pasa la ubicación de la etapa 2 del gestor de arranque a la etapa 1, tal como `(hd0,0)/grub/stage2`.
  - `p` <*`config-file`*> — Esta opción le indica al comando `install` que busque por el archivo de configuración de menú especificado por <*`config-file`*>. Un ejemplo de una ruta válida al archivo de configuración es `(hd0,0)/grub/grub.conf`.

Aviso

Este comando sobrescribirá cualquier información del MBR. Si se ejecuta, cualquier herramienta utilizada para arrancar el sistema operativo que no sea GRUB se perderá.

- `kernel <kernel-file-name> <option-1> <option-N>` — Especifica el archivo del kernel a cargar desde el sistema de archivos raíz de GRUB cuando se esté usando la carga directa para arrancar el sistema operativo. Las opciones pueden estar después del comando `kernel` y se pasarán al kernel cuando éste se cargue.

  Para el sistema Red Hat Linux, es probable que tenga una línea similar a la siguiente:
  `kernel/vmlinuzroot=/dev/hda5`

  Esta línea especifica que el archivo `vmlinuz` se carga desde un sistema de archivos raíz de GRUB, por ejemplo, `(hd0,0)`. También se transfiere una opción al kernel que especifica que el sistema de archivos raíz del kernel del Linux debe encontrarse, al cargarse, en `hda5`, la quinta partición en el primer disco duro IDE. Después de esta opción se pueden insertar varias opciones, si es necesario.

- `root <device-and-partition>` — Configura la partición raíz de GRUB para que sea el dispositivo y la partición concreta, por ejemplo, `(hd0,0)`, y monta la partición de modo que se puedan leer los archivos.

- `rootnoverify <device-and-partition>` — Realiza las mismas funciones que el comando `root` pero no monta la partición.

Hay otros comandos disponibles aparte de los indicados. Escriba `info grub` para obtener una lista completa de los comandos.

2.7. archivo de configuración de menú de GRUB

El archivo de configuración (`/boot/grub/grub.conf`), usado para crear la lista en la interfaz de menú de GRUB de los sistemas operativos para el arranque, básicamente permite al usuario seleccionar un grupo predefinido de comandos para su ejecución. Pueden utilizarse los comandos que se indican en Sección 2.6, así como algunos comandos especiales disponibles tan sólo en el archivo de configuración.

2.7.1. Comandos especiales del archivo de configuración

Los comandos siguientes sólo pueden usarse en el archivo de configuración de menú de GRUB:

- `color <normal-color> <selected-color>` — Le permite configurar los colores específicos que se usarán en el menú. Se configuran dos colores: uno de fondo y otro de primer plano. Use nombres de colores simples, tales como `red/black`. Por ejemplo:
  `colorred/blackgreen/blue`

- `default <nombre-título>` — Nombre del título por defecto de la entrada que se cargará si se supera el tiempo de inactividad de la interfaz de menú.

- `fallback <nombre-título>` — Si se utiliza, el nombre de título de la entrada que deberá probarse si falla el primer intento.

- `hiddenmenu` — Si se utiliza, no se podrá mostrar la interfaz de menú de GRUB ni cargar la entrada `default` si caduca el período `timeout`. El usuario puede ver el menú estándar de GRUB si pulsa la tecla [Esc].

- `password <contraseña>` — Si se utiliza, el usuario que no conozca la contraseña no podrá modificar las entradas de esta opción de menú.

  Opcionalmente, puede especificar un archivo de configuración de menú alternativo después de la `password <contraseña>`. En este caso, GRUB reiniciará el Nivel 2 del gestor de arranque y utilizará este archivo de configuración alternativo para crear el menú. Si se omite este archivo de

configuración alternativo del comando, el usuario que sepa la contraseña podrá modificar el archivo de configuración actual.

- `timeout` — Si se utiliza se establece la cantidad de tiempo, en segundos, antes de que GRUB cargue la entrada designada por el comando `default`.
- `splashimage` — Especifica la ubicación de la imagen de pantalla splash que se utilizará al arrancar.
- `title` — Establece el título que se utilizará con un grupo de comandos concreto para cargar un sistema operativo.

El carácter (#) se puede usar al principio de una línea para insertar comentarios en el archivo de configuración de menú.

2.7.2. Estructura del archivo de configuración

El archivo de configuración de menú de GRUB es `/boot/grub/grub.conf`. Los comandos para configurar las preferencias globales para la interfaz de menú están ubicados al inicio del archivo, seguido de las diferentes entradas para cada sistema operativo o kernels listados en el menú.

El siguiente es un ejemplo de archivo de configuración de menú muy básico diseñado para arrancar bien sea Red Hat Linux o Microsoft Windows 2000:

```
default=0
timeout=10
splashimage=(hd0,0)/grub/splash.xpm.gz

#sectiontoloadlinux
titleRedHatLinux(2.4.18-5.47)
root(hd0,0)
kernel/vmlinuz-2.4.18-5.47roroot=/dev/sda2
initrd/initrd-2.4.18-5.47.img

#sectiontoloadWindows2000
titlewindows
rootnoverify(hd0,0)
chainloader+1
```

Este archivo indicará a GRUB que cree un menú con Red Hat Linux como el sistema operativo predeterminado y que establezca un arranque automático después de 10 segundos. Se proporcionan dos secciones, una para cada entrada de sistema, con comandos específicos para la tabla de partición de cada sistema.

Nota

Observe que la opción predeterminada está especificada como un número. Esto hace referencia a la primera línea `title` con la que GRUB se encuentra. Si desea que `windows` sea predeterminada, cambie el valor `default=0` a `default=1`.

Este capítulo no abarca la configuración de un archivo de configuración del menú de GRUB para arrancar sistemas operativos múltiples. Consulte Sección 2.11 para una lista de los recursos adicionales.

2.8. LILO

LILO es un acrónimo de *LI*nux *LO*ader (cargador) y ha sido usado para arrancar linux en sistemas x86 por muchos años. Aunque ahora GRUB es el gestor de arranque por defecto, algunos prefieren usar LILO porque les es más familiar y otros porque GRUB puede causar problemas al arrancar determinado tipo de hardware.

2.8.1. LILO y el proceso de arranque de x86

Esta sección trata en más detalle el rol específico que LILO desempeña al arrancar el sistema x86. Para ver con más detenimiento el proceso de arranque, vea Sección 1.2.

LILO se carga asímismo en la memoria casi de forma idéntica a GRUB, con la diferencia de que es un gestor de dos etapas.

1. *La etapa 1 o el gestor de arranque primario se lee en la memoria por la BIOS desde el MBR* [2]. El gestor de arranque primario existe en menos de 512 bytes de espacio en disco dentro del MBR. Su función es cargar la etapa 2 del gestor de arranque y pasarle la información de la geometría del disco.
2. *La etapa 2 o el gestor de arranque secundario se lee en memoria.*. El gestor de arranque secundario visualiza la pantalla inicial de Red Hat Linux. Esta pantalla le permite seleccionar el sistema operativo o el kernel de Linux que desee arrancar.
3. *La etapa 2 lee el sistema operativo o el kernel y lleva a cabo* `initrd` *en memoria.* Una vez que LILO determina qué sistema operativo iniciar, éste lo carga en la memoria y lleva el control de la máquina a ese sistema operativo.

Una vez que se ha llevado a cabo la etapa 2 en memoria, LILO visualiza la pantalla inicial de Red Hat Linux con los diferentes sistemas operativos o kernel que han sido configurados para arrancar. Por defecto, si Red Hat Linux es el único sistema instalado, **linux** será la única opción disponible. Si el sistema tiene múltiples procesadores habrá una opción **linux-up** para el kernel del procesador único y una opción **linux** para los kernel de múltiples procesadores (SMP). Si LILO está configurado para arrancar otros sistemas operativos, estas entradas de arranque también aparecerán en pantalla.

Las flechas direccionales permiten al usuario resaltar el sistema operativo deseado y la tecla [Intro] comenzará el proceso de arranque.

Para accesar una línea de comandos `boot:`, presione [Ctrl]-[X].

2.8.2. LILO versus GRUB

En general, LILO funciona de forma parecida a GRUB a excepción de tres diferencias:

- No posee ninguna interfaz del comando interactiva.
- Almacena información sobre la localización del kernel o de si otro sistema operativo se debe cargar en el MBR.
- No puede leer las particiones ext2.

El primer punto significa que el intérprete de comandos para LILO no es interactivo y permite tan sólo un comando con argumentos.

Los últimos dos puntos significan que si usted cambia el archivo de configuración de LILO o instala un kernel nuevo, debe reescribir el gestor de arranque LILO de la etapa 1 al MBR llevando a cabo el comando siguiente:

2. Para obtener más información sobre la BIOS del sistema y el MBR, vaya a Sección 1.2.1.

```
/sbin/lilo-v-v
```

Este método es más arriesgado que el de GRUB, porque un MBR que no haya sido configurado adecuadamente deja el sistema sin poder arrancar. Con GRUB, si el archivo de configuración está configurado de forma errónea, se disparará por defecto la interfaz de la línea de comandos de modo que el usuario pueda arrancar el sistema manualmente.

Sugerencia

Si actualiza el kernel usando **Agente de actualización de Red Hat**, el MBR será actualizado automáticamente. Para obtener más información sobre RHN, remítase a la siguiente URL: https://rhn.redhat.com

2.9. Opciones en `/etc/lilo.conf`

El archivo de configuración de LILO es `/etc/lilo.conf`. El comando `/sbin/lilo` usa este archivo para determinar que información debe escribir al MBR.

Aviso

Si desea mofificar `/etc/lilo.conf`, asegúrese de que ha hecho una copia de seguridad del archivo antes de cualquier otro cambio. Asegúrese de que posee un disquete de arranque que funcione de manera que sea capaz de arrancar el sistema y realizar cambios en el MBR si existe algún problema. Consulte las páginas de manual para `mkbootdisk` para obtener más información en la creación de un disco de arranque.

El archivo `/etc/lilo.conf` es usado por el comando `/sbin/lilo` para determinar qué sistema operativo o kernel iniciar, así como para saber donde instalarlo.

Un archivo de ejemplo para `/etc/lilo.conf` será muy parecido a:

```
boot=/dev/hda
map=/boot/map
install=/boot/boot.b
prompt
timeout=50
message=/boot/message
lba32
default=linux

image=/boot/vmlinuz-2.4.0-0.43.6
label=linux
initrd=/boot/initrd-2.4.0-0.43.6.img
read-only
root=/dev/hda5

other=/dev/hda1
label=dos
```

Este ejemplo le muestra un sistema configurado para arrancar dos sistemas operativos: Red Hat Linux y DOS. A continuación una vista más detallada de las líneas de este archivo:

- `boot=/dev/hda` — Instruye a LILO para que se instale en el primer disco duro del primer controlador IDE.
- `map=/boot/map` — localiza el archivo del mapa. En uso normal, esto no se debería modificar.
- `install=/boot/boot.b` — Hace que LILO instale el archivo específico como el nuevo sector de arranque de boot. En un uso normal, esto no debería ser alterado. Si falta la línea `install`, LILO asumirá que el archivo `/boot/boot.b` es el predeterminado a utilizar.
- `prompt` — Instruye a LILO a que muestre cualquier cosa que sea referenciado en la línea `message`. No se le recomienda que elimine la línea `prompt`, si la elimina, todavía podrá tener acceso a un intérprete manteniendo pulsada la tecla [Shift] mientras que su máquina empieza a arrancar.
- `timeout=50` — Configura la cantidad de tiempo que LILO esperará la entrada del usuario antes de proceder con el arranque de la entrada de la línea `default`. Esto se mide en décimas de segundo, con 50 por defecto.
- `message=/boot/message` — Se refiere a la pantalla que LILO visualiza para permitirle seleccionar el sistema operativo o el kernel a arrancar.
- `lba32` — Describe la geometría del disco para LILO. Otra entrada común es `linear`. No debería cambiar esta línea a menos de que esté bien seguro de lo que está haciendo. De lo contrario, pondría su sistema en un estado de no arranque.
- `default=linux` — Se refiere al sistema operativo por defecto que LILO arrancará de acuerdo a las opciones listadas bajo esta línea. El nombre `linux` se refiere a la línea `label` bajo cada una de las opciones de arranque.
- `image=/boot/vmlinuz-2.4.0-0.43.6` — Especifica el kernel de linux para arrancar con esta opción de arranque en particular.
- `label=linux` — Nombra la opción del sistema operativo en la pantalla de LILO. En este caso, es también el nombre al que se refiere la línea `default`.
- `initrd=/boot/initrd-2.4.0-0.43.6.img` — Se refiere a la imagen *initial ram disk* que se usa en el tiempo de arranque para inicializar los dispositivos que hacen que el inicio del kernel sea posible. El disco ram inicial es una colección de controladores específicos de cada máquina necesarios para operar una tarjeta SCSI, una unidad de disco duro o cualquier otro dispositivo necesario para cargar el kernel. No debería nunca intentar compartir los discos ram iniciales entre las máquinas.
- `read-only` — Especifica que la partición root (consulte la línea `root` más abajo) es de sólo lectura y no puede ser alterada durante el proceso de arranque.
- `root=/dev/hda5` — Especifica cual partición de disco a usar como la partición raíz.
- `other=/dev/hda1` — Especifica la partición que contiene DOS.

2.10. Cambiar los niveles de ejecución en el tiempo de arranque

Bajo Red Hat Linux, es posible cambiar el nivel de ejecución predeterminado en el momento de arranque.

Si usa LILO, acceda al intérprete de comandos `boot:` presionando [Ctrl]-[X]. Luego escriba:

linux<***runlevel-number***>

En este comando, reemplace el número <*runlevel-number*> con el número de nivel de ejecución a arrancar (de **1** a **5**), o las palabras **single** o **emergency**.

Si está usando GRUB como gestor de arranque, siga los pasos siguientes:

- En la pantalla gráfica del gestor de arranque GRUB, seleccione la etiqueta de arranque **Red Hat Linux** y pulse [e] para modificarla.
- Vaya en la parte inferior a la línea del kernel y pulse [e] para modificarla.
- En el intérprete de comandos, escriba el número del nivel de ejecución en el que desea arrancar (desde **1** a **5**), o las palabras **single** o **emergency** y presione [Intro].
- Volverá a la pantalla de GRUB con la información sobre el kernel. Pulse [b] para arrancar el sistema.

Para obtener más información sobre los niveles de ejecución, consulte Sección 1.4.1.

2.11. Recursos adicionales

El objetivo de este capítulo sólo es servir de introducción a GRUB y a LILO. Consulte los siguientes recursos para descubrir más cosas sobre cómo funcionan GRUB y LILO.

2.11.1. Documentación instalada

- `/usr/share/doc/grub-<version-number>/` — Este directorio contiene muy buena información sobre el uso y configuración de GRUB. El *<version-number>* en la ruta a este archivo corresponde a la versión del paquete de GRUB instalado.
- La página de información de GRUB, a la que se puede acceder si se escribe el comando `info grub`, contiene un tutorial, un manual de referencia para el usuario, un manual de referencia para el programador y un documento de Preguntas más frecuentes (FAQ) sobre GRUB y su uso.
- `/usr/share/doc/lilo-<version-number>/` — Este directorio contiene información sobre el uso y la configuración de LILO. En particular, el subdirectorio `doc/` contiene un archivo postscript llamado `User_Guide.ps` que da mucha información. El número *<version-number>* en la ruta a este directorio, corresponde al número de versión del paquete LILO instalado.

2.11.2. Sitios Web útiles

- http://www.gnu.org/software/grub/ — Página principal del proyecto GNU GRUB. Este sitio contiene información sobre el estado de desarrollo de GRUB y una sección de Preguntas más frecuentes (FAQ).
- http://www.uruk.org/orig-grub/ — La documentación original GRUB antes de que el proyecto se entregue a la Free Software Foundation para su posterior desarrollo.
- http://www.redhat.com/mirrors/LDP/HOWTO/mini/Multiboot-with- GRUB.html — Investiga los distintos usos de GRUB, incluido el arranque de sistemas operativos que no son Linux.
- http://www.linuxgazette.com /issue64/kohli.html — Artículo de introducción en el que se describe cómo configurar GRUB en el sistema desde el principio y en el que se ofrece una introducción general a las opciones de la línea de comandos de GRUB.
- http://www.tldp.org/HOWTO/mini/LILO.html — Este mini-HOWTO habla de los diferentes usos para LILO, incluyendo los sistemas operativos de arranque diferentes a Linux.

redhat.

Capítulo 3.

Estructura del sistema de archivos

3.1. Por qué compartir una estructura común

Una estructura de sistema de archivos de un sistema operativo es el nivel más básico de organización. Casi siempre un sistema operativo interactúa con sus usuarios, aplicaciones y modelos de seguridad que dependen de la manera en que almacena los archivos en un dispositivo de almacenamiento primario (normalmente una unidad de disco duro). Por varios motivos, es muy importante que los usuarios, así como los programas para la instalación y demás, sean capaces de referirse a unas pautas comunes para saber donde escribir y leer los archivos binarios, la configuración, registro y otros archivos.

Un sistema de archivos se podría resumir en términos de dos categorías diferentes de archivos:

- archivos compartibles vs. no compartibles
- archivos variables vs. estáticos

Los archivos *compartibles* son aquéllos a los que se puede acceder desde varios hosts; mientras que los archivos *no compartibles* no están disponibles a todos los hosts. Los archivos *Variables* pueden cambiar en cualquier momento sin una intervención del gestor de sistemas; los archivos *estáticos*, tales como documentación de sólo lectura y binarios, no cambian sin una actuación por parte del administrador de sistemas o de una agente que el administrador de sistemas haya escogido para realizar esta tarea.

El hecho de que estos archivos sean vistos de esta manera es para ayudar a correlacionar la función del archivo con los permisos otorgados a los directorios que los sostienen. El modo en que el sistema operativo y sus usuarios interactúan con un archivo dado determina el directorio en el cual estos archivos estan ubicados, si ese directorio está montado como de sólo lectura o sólo escritura y el nivel de acceso que cada usuario tiene a ese archivo. El nivel superior de esta organización es crucial, como el acceso a los directorios inferiores pueden estar restringidos o se pueden manifestar problemas de seguridad si el nivel superior es dejado sin organizar o sin una estructura ampliamente usada.

No obstante, el hecho de tener simplemente una estructura no significa mucho a menos que ésta sea estándar. Las estructuras competitivas pueden causar más problemas de los que solucionan. Por esta razón, Red Hat ha escogido la estructura de sistema de archivos usada más ampliamente y la ha extendido ligeramente para acomodar los archivos especiales usados en Red Hat Linux.

3.2. Vista preliminar del estándar de jerarquía del sistema de archivos (FHS)

Red Hat está comprometido a respetar el *Estándar de Jerarquía del Sistema archivos* (*FHS* del inglés Filesystem Hierarchy Standard), un documento de consenso que define los nombres y la ubicación de muchos archivos y directorios.

El documento que define el FHS es la referencia autorizada para cualquier sistema compatible FHS, sin embargo el estándar da pie a la extensibilidad de unas áreas o no define otras. En esta sección se proporciona un resumen del estándar y una descripción de aquellas partes del sistema de archivos que no cubre el estándar.

El estándar completo está disponible en:

http://www.pathname.com/fhs

El cumplimiento del estándar significa varias cosas, los dos aspectos más importantes son la compatibilidad con otros sistemas que siguen el estándar y la capacidad de poder montar la partición `/usr` en modo sólo lectura pues contiene ejecutables comunes y no está pensado para ser alterada por los usuarios. Por este motivo, `/usr` puede ser montado directamente desde el CD-ROM o desde otro ordenador vía NFS en modo sólo lectura.

3.2.1. Organización de FHS

Los directorios y archivos aquí anotados, son sólo un subconjunto de los especificados por el FHS. Véase la última versión del FHS para una descripción detallada.

3.2.1.1. El directorio `/dev/`

El directorio `/dev/` contiene entradas del sistema de archivos que representan dispositivos del sistema. Estos archivos son esenciales para el correcto funcionamiento del sistema.

3.2.1.2. El directorio `/etc/`

El directorio `/etc/` está reservado para archivos de configuración que son locales a su ordenador. No deben colocarse binarios en `/etc`. Los binarios que antiguamente se colocaban en `/etc` deberían estar en `/sbin` o posiblemente en `/bin`.

Los directorios `X11/` y `skel/` son subdirectorios del directorio `/etc/`:

```
/etc
  |- X11/
  |- skel/
```

El directorio `/etc/X11/` es para archivos de configuración de X11 como `XF86Config`. El directorio `/etc/skel` es para archivos "esqueleto" (del inglés "skeleton") de usuarios, archivos que se utilizan para rellenar el directorio principal de un usuario cuando éste es creado.

3.2.1.3. El directorio `/lib/`

El directorio `/lib/` debería contener sólo las librerías necesarias para ejecutar los binarios en `/bin` y `/sbin`. Estas imágenes de librerías compartidas son particularmente importantes para arrancar el sistema y ejecutar comandos en el sistema de archivos de root.

3.2.1.4. El directorio `/mnt/`

El directorio `/mnt/` se refiere a sistemas de archivos montados temporalmente, tales como CD-ROMs y disquetes.

3.2.1.5. El directorio `/opt/`

El directorio `/opt/` proporciona un área para almacenar habitualmente paquetes de software de una aplicación estática y amplia.

Un paquete colocando archivos en el directorio `/opt/` crea un directorio con el mismo nombre del paquete. Este directorio en su lugar guarda archivos que de otra forma estarías esparcidos por el sistema de archivos, dándole así al administrador del sistema una forma fácil de determinar el papel de cada archivo dentro de un paquete particular.

Por ejemplo, si `sample` fuese el nombre de un paquete de software particular localizado en el directorio `/opt/`, todos sus archivos podrían ser emplazados en directorios dentro de

/opt/sample/, tales como /opt/sample/bin/ para binarios y /opt/sample/man/ para páginas de manual.

Los paquetes grandes que abarcan diferentes subpaquetes, cada uno de los cuales desempeñan una tarea específica, también también se ubican dentro de /opt/, aportando a este gran paquete un modo estándar de organizarse. De este modo, el paquete sample puede tener diferentes herramientas que cada una irá en sus propios subdirectorios, tales como /opt/sample/tool1/ y /opt/sample/tool2/, cada uno de los cuales puede tener su propio bin/, man/ y otros directorios similares.

3.2.1.6. El directorio /proc/

El directorio /proc/ contiene "archivos" especiales que o bien extraen información del kernel o bien la envían a éste.

Debido a la gran variedad de datos que contiene el directorio /proc/ y a la gran cantidad de maneras utilizadas para comunicar con el kernel, se ha dedicado un capítulo entero a este tema. Para mayor información vea el Capítulo 5.

3.2.1.7. El directorio /sbin/

El directorio /sbin/ es para ejecutables usados sólo por el usuario root. Los ejecutables en /sbin sólo se usan para arrancar y montar /usr y ejecutar operaciones de recuperación del sistema. El FHS dice:

> "/sbin contiene típicamente archivos esenciales para arrancar el sistema además de los binarios en /bin. Cualquier archivo ejecutado tras /usr, será montado (si no surge ningún problema) y ubicado en /usr/sbin. Los binarios de administración de sistema local solamente, deberían ser ubicados en /usr/local/sbin."

Los siguientes programas deberían encontrarse, al menos, en /sbin/:

```
arp, clock,
getty, halt,
init, fdisk,
fsck.*, grub,
ifconfig, lilo,
mkfs.*, mkswap,
reboot, route,
shutdown, swapoff,
swapon, update
```

3.2.1.8. El directorio /usr/

El directorio /usr es para archivos que puedan ser compartidos a través de todo el sitio. El directorio /usr habitualmente tiene su propia partición y debería ser montable en sólo lectura. Como mínimo, los siguientes directorios deberían ser subdirectorios de /usr:

```
/usr
  |- bin/
  |- dict/
  |- doc/
  |- etc/
  |- games/
  |- include/
```

```
|- kerberos/
|- lib/
|- libexec/
|- local/
|- sbin/
|- share/
|- src/
|- tmp -> ../var/tmp/
|- X11R6/
```

El directorio `bin/` contiene ejecutables, `dict/` contiene páginas de documentación incompatibles con FHS, `etc/` contiene archivos de configuración de sistema, `games` es para juegos, `include/` contiene los archivos de cabecera C, `kerberos/` contiene binarios y muchos más archivos de Kerberos y `lib/` contiene archivos objeto y librerías que no están diseñadas para ser directamente utilizadas por usuarios o scripts de shell. El directorio `libexec/` contiene pequeños programas de ayuda llamados por otros programas, `sbin/` es para los binarios de administración del sistema (aquéllos que no pertenecen a `/sbin/` `share/` contiene archivos que no son de una arquitectura específica, `src/` es para código fuente y `X11R6/` es para el sistema X Window (**XFree86** de Red Hat Linux).

3.2.1.9. El directorio `/usr/local/`

El FHS dice:

> "La jerarquía `/usr/local` es para uso del administrador del sistema al instalar localmente el software. Necesita ser seguro para ser sobreescrito cuando el software del sistema es compartible entre un grupo de hosts, pero no se encuentra en `/usr`."

El directorio `/usr/local/` es similar en estructura al directorio `/usr/`. Tiene los siguientes subdirectorios, que son similares en propósito a los del directorio `/usr/`:

```
/usr/local
       |- bin/
       |- doc/
       |- etc/
       |- games/
       |- include/
       |- lib/
       |- libexec/
       |- sbin/
       |- share/
       |- src/
```

3.2.1.10. El directorio `/var/`

Ya que el FHS requiere que Linux sea capaz de montar `/usr/` en sólo lectura, cualquier programa que escriba archivos log o que necesite los directorios `spool/` o `lock/` debería escribirlos en el directorio `/var/`. El FHS especifica que `/var/` es para:

> "... archivos de datos variables. Esto incluye archivos y directorios spool, datos de administración, de registro y archivos temporales."

Los siguientes directorios deberían ser subdirectorios de `/var/`:

```
/var
```

```
|- account/
|- arpwatch/
|- cache/
|- crash/
|- db/
|- empty/
|- ftp/
|- gdm/
|- kerberos/
|- lib/
|- local/
|- lock/
|- log/
|- mail -> spool/mail/
|- mailman/
|- named/
|- nis/
|- opt/
|- preserve/
|- run/
+- spool/
     |- anacron/
     |- at/
     |- cron/
     |- fax/
     |- lpd/
     |- mail/
     |- mqueue/
     |- news/
     |- rwho/
     |- samba/
     |- slrnpull/
     |- squid/
     |- up2date/
     |- uucp/
     |- uucppublic/
     |- vbox/
     |- voice/
|- tmp/
|- tux/
|- www/
|- yp/
```

Los archivos log de sistema tales como `messages/` y `lastlog/` están en el directorio `/var/log/`. El directorio `/var/lib/rpm/` también contiene el las bases de datos RPM. Los archivos lock van en `/var/lock/`, habitualmente en directorios particulares para el programa en el uso del archivo. El directorio `/var/spool/` tiene subdirectorios para varios sistemas que necesitan almacenar los archivos de datos.

3.2.2. `/usr/local/` en Red Hat Linux

En Red Hat Linux, el propósito del uso del directorio `/usr/local/` ligeramente diferente de lo especificado por FHS. El FHS establece que en `/usr/local/` debería memorizarse el software que permanece seguro en las actualizaciones de software de sistemas. Ya que las actualizaciones de sistemas de Red Hat Linux se han realizado de forma segura con el comando `rpm` y la aplicación gráfica **Herramienta de administración de paquetes**, no es necesario proteger archivos poniéndolos en `/usr/local/`. En vez de esto, el directorio `/usr/local/` es usado para software que es local a la máquina.

Por ejemplo, si usted ha montado `/usr/` sólo lectura de NFS desde un host remoto, aún es posible instalar un paquete o programa bajo el directorio `/usr/local/`.

3.3. Directorios especiales de Red Hat Linux

Red Hat Linux extiende un poco la estructura de FHS para acomodar archivos especiales.

La mayor parte de los archivos que pertenecen al *Administrador de paquetes* (*RPM*) se encuentran en el directorio `/var/lib/rpm/`. Para mayor información consulte el capítulo *Administración de paquetes con RPM* en el *Manual de personalización de Red Hat Linux*.

El directorio `/var/spool/up2date/` contiene los archivos que usa la aplicación **Agente de actualización de Red Hat**, incluyendo la información de cabecera de RPM para el sistema. Esta ubicación se puede usar temporalmente para almacenar los RPMs descargados durante la actualización del sistema. Para mayor información sobre la Red Hat Network, vaya al sitio en https://rhn.redhat.com/.

Otra de las ubicaciones específica de Red Hat Linux es el directorio `/etc/sysconfig/`. Este directorio almacena una variedad información de la configuración. Muchos scripts que se ejecutan al iniciar el sistema, usan los archivos de este directorio. Consulte el Capítulo 4 para más información sobre lo que se encuentra dentro directorio y el papel de estos archivos en el proceso de arranque.

Finalmente, otro directorio es el directorio `/initrd/`. Está vacío pero se usa como punto de montaje crítico durante el proceso de arranque.

Aviso

No elimine el directorio `/initrd/` por ningún motivo. Al eliminar este directorio causará que el sistema falle al arrancar con un mensaje de pánico del kernel.

Capítulo 4.

El directorio `sysconfig`

El directorio `/etc/sysconfig/` es donde se almacenan una gran variedad de archivos de configuración para Red Hat Linux.

Este capítulo resalta algunos de los archivos encontrados en el directorio `/etc/sysconfig/`, su función, y sus contenidos. La información en este capítulo no pretende ser exhaustiva, pues muchos de estos archivos tienen una variedad de opciones que sólo son usadas en circunstancias muy específicas.

4.1. Archivos en el directorio `/etc/sysconfig/`

Los siguientes archivos son normalmente encontrados en el directorio `/etc/sysconfig/`:

- `amd`
- `apmd`
- `arpwatch`
- `authconfig`
- `cipe`
- `clock`
- `desktop`
- `dhcpd`
- `firstboot`
- `gpm`
- `harddisks`
- `hwconf`
- `i18n`
- `identd`
- `init`
- `ipchains`
- `iptables`
- `irda`
- `keyboard`
- `kudzu`
- `mouse`
- `named`
- `netdump`
- `network`
- `ntpd`
- `pcmcia`
- `radvd`

- rawdevices
- redhat-config-securitylevel
- redhat-config-users
- redhat-logviewer
- samba
- sendmail
- soundcard
- spamassassin
- squid
- tux
- ups
- vncservers
- xinetd

Nota

Si alguno de los archivos aquí listados no está presente en el directorio /etc/sysconfig/, entonces el programa correspondiente lo más probable es que tampoco esté instalado.

4.1.1. /etc/sysconfig/amd

El archivo /etc/sysconfig/amd contiene varios parámetros usados por amd, que permiten el montaje y desmontaje automático de sistemas de archivos.

4.1.2. /etc/sysconfig/apmd

El archivo /etc/sysconfig/apmd es usado por apmd como una configuración para la determinación de qué configuraciones de energía usar en inicio/parada/cambio en el estado suspendido o reanudar. Está configurado para apagar o encender apmd al momento de arranque, dependiendo de si el hardware soporta *Advanced Power Management* - (*APM*), administración avanzada de energía o si el usuario ha configurado o no el sistema para usarla. El demonio apm es un programa de monitoreo que funciona con el código de administración de energía dentro del kernel de Linux. Es capaz de alertar a los usuarios sobre la condición de energía baja en la batería en las computadoras portátiles y otras configuraciones relacionadas con la energía del sistema.

4.1.3. /etc/sysconfig/arpwatch

El archivo /etc/sysconfig/arpwatch es usado para pasar argumentos al demonio arpwatch en el momento de ejecución. El demonio arpwatch mantiene una tabla de direcciones MAC Ethernet y sus direcciones pares IP. Para más información sobre los parámetros disponibles para este archivo, vea la página del manual de arpwatch. Por defecto, este archivo coloca como propietario del proceso arpwatch al usuario pcap.

4.1.4. /etc/sysconfig/authconfig

El archivo /etc/sysconfig/authconfig configura el tipo de autorización a ser usada en el host. Contiene una o más de las líneas siguientes:

- USEMD5=<*value*>, donde <*value*> es uno de los siguientes:
  - yes — Se usa MD5 para la autentificación.
  - no — No se usa MD5 para la autentificación.

- USEKERBEROS=<*value*>, donde <*value*> es uno de los siguientes:
  - yes — Kerberos es usado para la autentificación.
  - no — Kerberos no es usado para la autentificación.

- USELDAPAUTH=<*value*>, donde <*value*> es uno de los siguientes:
  - yes — LDAP es usado para la autentificación.
  - no — LDAP no es usado para la autentificación.

4.1.5. /etc/sysconfig/clock

El archivo /etc/sysconfig/clock controla la interpretación de los valores leídos desde el reloj del sistema.

Los valores correctos son:

- UTC=<*value*>, donde <*value*> es uno de los siguientes valores boleanos:
  - true o yes — El reloj del hardware está configurado a Universal Time.
  - false o no — El reloj del hardware está configurado a la hora local.

- ARC=<*value*>, donde <*value*> es lo siguiente:
  - true o yes — El desplazamiento de hora (time offset) de 42 años de la consola ARC, está en efecto. Esta configuración es sólo para los sistemas Alpha ARC o AlphaBIOS. Cualquier otro valor indica que se usa la época normal de UNIX.

- SRM=<*value*>, donde <*value*> es lo siguiente:
  - true o yes — Está en efecto la época 1900 de la consola SRM. Esta configuración es solamente para sistemas Alpha basados en SRM. Cualquier otro valor indica la época normal de UNIX.

- ZONE=<*filename*> — El archivo de zona horaria bajo /usr/share/zoneinfo del cual /etc/localtime es una copia. El archivo contiene información tal como:
  ZONE="America/New York"

Ediciones previas de Red Hat Linux usaban los valores siguientes (las cuales ya no son aprobadas):

- CLOCKMODE=<*value*>, donde <*value*> es uno de los siguientes:
  - GMT — El reloj está colocado al Universal Time (Greenwich Mean Time).

- ARC — El desplazamiento (time offset) de 42 años de la consola ARC está en efecto (sólo para sistemas basados en Alpha).

4.1.6. /etc/sysconfig/desktop

El archivo /etc/sysconfig/desktop especifica el administrador de escritorio a ser ejecutado, tal como:

```
DESKTOP="GNOME"
```

4.1.7. /etc/sysconfig/dhcpd

El archivo /etc/sysconfig/dhcpd es usado para pasar argumentos al demonio dhcpd en el momento de arranque. El demonio dhcpd implementa el Dynamic Host Configuration Protocol (DHCP) y el Internet Bootstrap Protocol (BOOTP). DHCP y BOOTP asignan nombres de host a las máquinas en la red. Para más información sobre qué parámetros están disponibles en este archivo, consulte la página del manual de dhcpd.

4.1.8. /etc/sysconfig/firstboot

Comenzando con Red Hat Linux 8.0, la primera vez que el sistema arranca, el programa /sbin/init llama al script etc/rc.d/init.d/firstboot lanzar **Agente de configuración**. Esta aplicación permite al usuario instalar las últimas actualizaciones así como también cualquier aplicación o documentación adicional.

El archivo /etc/sysconfig/firstboot le dice a la aplicación **Agente de configuración** que no se ejecute en los subsecuentes reinicios. Para ejecutarlo la próxima vez que el sistema arranca, elimine /etc/sysconfig/firstboot y ejecute chkconfig --level 5 firstboot on.

4.1.9. /etc/sysconfig/gpm

El archivo /etc/sysconfig/gpm es usado para pasar argumentos al demonio gpm en el momento de ejecución. El demonio gpm es el servidor del ratón que permite la aceleración del ratón y el pegado con el botón del medio. Para más información sobre qué parámetros están disponibles para este archivo, consulte la página del manual de gpm. Por defecto, se configura el dispositivo del ratón a /dev/mouse.

4.1.10. /etc/sysconfig/harddisks

El archivo /etc/sysconfig/harddisks optimiza el/los disco(s) duro. El administrador también puede usar /etc/sysconfig/hardiskhd[a-h] para configurar parámetros de dispositivos específicos.

Aviso

<<<<<<< sysconfig.sgml No realice cambios a este archivo a la ligera. Si cambia los valores predeterminados almacenados aquí, podría corromper todos los datos de su(s) disco(s). ======= No realice cambios a este archivo sin una consideración cuidadosa. Al cambiar los valores por defecto, es posible dañar todos los datos en disco. >>>>>>> 1.58

El archivo /etc/sysconfig/harddisks puede contener lo siguiente:

- USE_DMA=1, cuando se configura este valor a 1 se activa DMA. Sin embargo, con algunos chipsets y combinaciones de disco duro, DMA puede causar corrupción de los datos. *Verifique la documentación del disco duro o del fabricante antes de activar esta opción.*
- Multiple_IO=16, donde una configuración a 16 permite múltiples sectores por interrupción de E/S. Cuando está activada, esta característica reduce la sobrecarga del sistema operativo en un 30-50%. *Úselo con precaución.*
- EIDE_32BIT=3 activa (E)IDE 32-bit soporte de E/S a una tarjeta de interfaz.
- LOOKAHEAD=1 activa read-lookahead de dispositivos.
- EXTRA_PARAMS= especifica donde agregar los parámetros extra.

4.1.11. /etc/sysconfig/hwconf

El archivo /etc/sysconfig/hwconf lista todo el hardware que kudzu detectó en su sistema, así como también los controladores usados, ID de los fabricantes e información de ID de los dispositivos. El programa kudzu detecta y configura el hardware nuevo o modificado en su sistema. El archivo /etc/sysconfig/hwconf se supone que no es para ser modificado manualmente. Si se edita, los dispositivos se pueden repentinamente mostrar como que han sido agregados o eliminados.

4.1.12. /etc/sysconfig/i18n

El archivo /etc/sysconfig/i18n configura el idioma predeterminado, cualquier idioma soportado y la fuente predeterminada del sistema. Por ejemplo:

```
LANG="en_US.UTF-8"
SUPPORTED="en_US.UTF-8:en_US:en"
SYSFONT="latarcyrheb-sun16"
```

4.1.13. /etc/sysconfig/identd

El archivo /etc/sysconfig/identd es usado para pasar argumentos al demonio identd al momento de arranque. El demonio identd devuelve el nombre del usuario de procesos con conexiones TCP/IP abiertas. Algunos servicios en la red, tal como servidores FTP y IRC, se quejaran y causarán respuestas lentas si identd no se está ejecutando. Pero en general, identd no es un servicio requerido, por lo tanto si la seguridad es una de sus preocupaciones, no lo ejecute. Para más información sobre qué parámetros están disponibles para este archivo, consulte la página del manual de identd. Por defecto, el archivo no contiene parámetros.

4.1.14. /etc/sysconfig/init

El archivo /etc/sysconfig/init controla cómo el sistema aparecerá y funcionará durante el momento de arranque.

Se usan los siguientes valores:

- BOOTUP=<*value*>, donde <*value*> es uno de los siguientes:
  - BOOTUP=color significa el color estándar del despliegue al momento de arranque, donde el éxito o falla de dispositivos y servicios al iniciarse es mostrado en diferentes colores.

  - BOOTUP=verbose es un tipo de despliegue viejo, que provee de más información que el simple mensaje de éxito o falla.
  - Cualquier otra cosa significa un nuevo despliegue, pero sin el formato ANSI.

- RES_COL=<*value*>, donde <*value*> es el número de la columna de la pantalla para comenzar las etiquetas de estado. Predeterminado a 60.
- MOVE_TO_COL=<*value*>, donde <*value*> mueve el cursor al valor en la línea RES_COL a través del comando echo -en.
- SETCOLOR_SUCCESS=<*value*>, donde <*value*> coloca el color a un color que indica el éxito a través del comando echo -en. El color predeterminado es verde.
- SETCOLOR_FAILURE=<*value*>, donde <*value*> configura el color a un color que indica la falla a través del comando echo -en. El color predeterminado es rojo.
- SETCOLOR_WARNING=<*value*>, donde <*value*> coloca el color para indicar advertencia a través del comando echo -en. Por defecto el color es amarillo.
- SETCOLOR_NORMAL=<*value*>, donde <*value*> reconfigura el color a "normal" a través de echo -en.
- LOGLEVEL=<*value*>, donde <*value*> configura el nivel de conexión de la consola inicial para el kernel. El valor por defecto es 3; 8 significa cualquier cosa (incluyendo depuración); 1 significa pánico del kernel. El demonio syslogd ignora esta configuración una vez que se ha arrancado.
- PROMPT=<*value*>, donde <*value*> es uno de los siguientes valores boleanos:
  - yes — Activa la verificación de claves para el modo interactivo.
  - no — Desactiva la verificación de claves para el modo interactivo.

4.1.15. /etc/sysconfig/ipchains

El archivo /etc/sysconfig/ipchains contiene información usada por el script de inicialización de ipchains cuando se esté configurando el servicio ipchains.

Este archivo es modificado escribiendo el comando /sbin/service ipchains save cuando existen reglas ipchains válidas. No modifique manualmente este archivo. En lugar de esto, use el comando /sbin/ipchains para configurar las reglas de filtro de paquetes necesarias y luego guarde las reglas a este archivo usando /sbin/service ipchains save.

No se recomienda el uso de ipchains para configurar las reglas del cortafuegos (firewall) pues está censurado y puede que desaparezca de los futuros lanzamientos de Red Hat Linux. Si se requiere un cortafuegos, use preferiblemente iptables.

4.1.16. /etc/sysconfig/iptables

De la misma forma que /etc/sysconfig/ipchains, el archivo /etc/sysconfig/iptables guarda información usada por el kernel para configurar los servicios de filtrado de paquetes en el momento de arranque o cuando se arranque un servicio.

No modifique este archivo manualmente a menos que esté bien familiarizado con la forma de construir reglas iptables. La forma más fácil de agragar reglas es usando **Herramienta de configuración de nivel de seguridad** (redhat-config-securitylevel), el comando /usr/sbin/lokkit, o la aplicación **GNOME Lokkit** para crear un cortafuegos. Estas aplicaciones automáticamente editan este archivo al final del proceso.

Las reglas se pueden crear manualmente usando /sbin/iptables: luego escribiendo /sbin/service iptables save para agregar las reglas al archivo /etc/sysconfig/iptables.

Una vez que este archivo existe, cualquier regla de firewall guardadas en él, persisten a través de los reinicios del sistema o de un servicio.

Para más información sobre iptables consulte el Capítulo 16.

4.1.17. /etc/sysconfig/irda

El archivo /etc/sysconfig/irda controla cómo los dispositivos infrarojos en el sistema son configurados en el arranque.

Se pueden usar los valores siguientes:

- IRDA=<*value*>, donde <*value*> es uno de los siguientes valores boleanos:
  - yes — irattach se ejecutará, lo que verifica periódicamente si hay algo tratándo de conectarse al puerto infrarojo, tal como otra laptop tratando de hacer una conexión de red. Para que los dispositivos infrarojos funcionen en su sistema, se debe colocar esta línea a yes.
  - no — irattach no se ejecutará, evitando la comunicación de dispositivos infrarojos.

- DEVICE=<*value*>, donde <*value*> es el dispositivo (usualmente un puerto serial) que maneja las conexiones infrarojas.
- DONGLE=<*value*>, donde <*value*> especifica el tipo de dongle que está siendo usado para la comunicación infraroja. Este valor existe para los casos en que se usan dongles seriales en vez de puertos infrarojos reales. Un dongle es un dispositivo que es conectado a un puerto serial tradicional para comunicar a través de infrarojo. Esta línea se coloca en comentarios por defecto porque las computadoras portátiles con puertos infrarojos reales son mucho más populares que las que tienen dongles agregados.
- DISCOVERY=<*value*>, donde <*value*> es uno de los siguientes valores boleanos:
  - yes — Arranca irattach en modo 'discovery', descubrimiento, lo que significa que está activamente chequeando por otros dispositivos infrarojos. Este valor necesita ser activado para que la máquina esté buscando activamente por una conexión infraroja (el par que no inicia la conexión).
  - no — No arranca irattach en modo discovery.

4.1.18. /etc/sysconfig/keyboard

El archivo /etc/sysconfig/keyboard controla el comportamiento del teclado. Se pueden usar los siguientes valores:

- KEYBOARDTYPE=sun|pc, el cual es usado solamente en SPARCs. sun significa que un teclado Sun está conectado en /dev/kbd, y pc significa que hay un teclado PS/2 keyboard conectado al puerto PS/2.
- KEYTABLE=<*file*>, donde <*file*> es el nombre de un archivo de tabla de teclas.

  Por ejemplo: KEYTABLE="us". Los archivos que pueden ser usados como tabla de teclas comienzan en /lib/kbd/keymaps/i386 y se extienden en diferentes disposiciones de teclados desde aquí, a todos los etiquetados <*file*>.kmap.gz. El primer archivo encontrado debajo /lib/kbd/keymaps/i386 que coincide con la configuración KEYTABLE es usado.

4.1.19. /etc/sysconfig/kudzu

El archivo /etc/sysconfig/kuzdu dispara una exploración segura del hardware del sistema mediante kudzu en el momento de arranque. time. Una exploración segura es una que desactiva el sondeo del puerto serial.

- SAFE=<*value*>, donde <*value*> es uno de los siguientes:
  - yes — kuzdu hace una exploración segura.
  - no — kuzdu realiza una exploración normal.

4.1.20. /etc/sysconfig/mouse

El archivo /etc/sysconfig/mouse es usado para especificar información sobre el ratón disponible. Se pueden usar los valores siguientes:

- FULLNAME=<*value*>, donde <*value*> se refiere al nombre completo del tipo de ratón que está siendo usado.
- MOUSETYPE=<*value*>, donde <*value*> es uno de los siguientes:
  - imps2 — Un ratón genérico USB.
  - microsoft — Un ratón Microsoft™.
  - mouseman — Un ratón MouseMan™.
  - mousesystems — Un ratón Systems™.
  - ps/2 — Un ratón PS/2.
  - msbm — Un ratón bus de Microsoft™.
  - logibm — Un ratón bus de Logitech™.
  - atibm — Un ratón bus de ATI™.
  - logitech — Un ratón Logitech™.
  - mmseries — Un ratón MouseMan™ más viejo.
  - mmhittab — Un ratón mmhittab.

- XEMU3=<*value*>, donde <*value*> es uno de los siguientes valores boleanos:
  - yes — El ratón solamente tiene dos botones, pero se pueden emular tres.
  - no — El ratón tiene tres botones.
- XMOUSETYPE=<*value*>, donde <*value*> se refiere al tipo de ratón usado cuando se está corriendo X. Las opciones aquí son las mismas que en el valor MOUSETYPE en este mismo archivo.
- DEVICE=<*value*>, donde <*value*> es el dispositivo de ratón.

Además, /dev/mouse es un enlace simbólico que apunta al dispositivo de ratón actual.

4.1.21. /etc/sysconfig/named

El archivo /etc/sysconfig/named es usado para pasar argumentos al demonio named en el momento de arranque. El demonio named es un servidor *Domain Name System* (*DNS*) que implementa la distribución *Berkeley Internet Name Domain* (*BIND*) versión 9. Este servidor mantiene una tabla de cuales hosts están asociados con direcciones IP en la red.

Actualmente, sólo los valores siguientes son usados:

- ROOTDIR="<*/some/where*>", donde <*/some/where*> se refiere a la ruta completa del directorio de un ambiente chroot bajo el cual named se ejecuta. Este ambiente chroot debe ser configurado primero. Escriba info chroot para ver más información.
- OPTIONS="<*value*>", donde <*value*> es cualquier opción listada en la página del manual para named excepto -t. En lugar de -t, use la línea ROOTDIR.

Para más información sobre qué parámetros están disponibles para este archivo, consulte la página de manual de named. Para información detallada sobre cómo configurar un servidor BIND DNS, vea el Capítulo 12. Por defecto, el archivo no contiene parámetros.

4.1.22. /etc/sysconfig/netdump

El archivo /etc/sysconfig/netdump es el archivo de configuración para el servicio /etc/init.d/netdump. El servicio netdump envía ambos datos oops y escombros de memoria sobre la red. En general, netdump no es un servicio requerido; sólo ejecútelo si es absolutamente necesario. Para más información sobre los parámetros disponibles para este archivo, consulte la página del manual de netdump.

4.1.23. /etc/sysconfig/network

El archivo /etc/sysconfig/network es usado para especificar información sobre la configuración de red deseada. Se pueden usar los valores siguientes:

- NETWORKING=<*value*>, donde <*value*> es uno de los siguientes valores boleanos:
  - yes — Se debería configurar el servicio de red.
  - no — No se debería configurar el servicio de red.

- HOSTNAME=<*value*>, donde <*value*> debería ser el *Fully Qualified Domain Name* (*FQDN*), nombre de dominio cualificado completo, tal como hostname.expample.com, pero puede ser cualquier nombre de host necesario.

Nota

Para garantizar la compatibilidad con software más viejo que la gente pueda instalar (tal como trn), el archivo /etc/HOSTNAME debería contener el mismo valor que aquí.

- GATEWAY=<*value*>, donde <*value*> es la dirección IP de la gateway (compuerta) de la red.
- GATEWAYDEV=<*value*>, donde <*value*> es el dispositivo gateway, tal como eth0.
- NISDOMAIN=<*value*>, donde <*value*> es el nombre del dominio NIS.

4.1.24. /etc/sysconfig/ntpd

El archivo /etc/sysconfig/ntpd es usado para pasar argumentos al demonio ntpd en el momento de arranque. El demonio ntpd configura y mantiene el reloj del sistema para sincronizar con un servidor de hora estándar de Internet. Implementa la versión 4 del protocolo de hora de red (Network Time Protocol, NTP). Para más información sobre los parámetros disponibles para este archivo, apunte su navegador al siguiente archivo: /usr/share/doc/ntp-<*version*>/ntpd.htm (donde <*version*> es el número de versión de ntpd). Por defecto, este archivo configura el propietario del proceso ntpd al usuario de ntp.

4.1.25. /etc/sysconfig/pcmcia

El archivo /etc/sysconfig/pcmcia es usado para especificar la información de configuración de PCMCIA. Los valores siguientes se pueden usar:

- PCMCIA=<*value*>, donde <*value*> es uno de los siguientes:
  - yes — Se debería activar el soporte a PCMCIA.
  - no — No se debería activar el soporte a PCMCIA.

- PCIC=<*value*>, donde <*value*> es uno de los siguientes:
  - i82365 — El computador tiene un chipset estilo i82365 con socket PCMCIA.
  - tcic — El computador tiene un chipset estilo tcic con socket PCMCIA.

- PCIC_OPTS=<*value*>, donde <*value*> es el parámetro de tiempo del controlador de socket (i82365 o tcic).
- CORE_OPTS=<*value*>, donde <*value*> es la lista de opciones pcmcia_core.
- CARDMGR_OPTS=<*value*>, donde <*value*> es la lista de opciones para la cardmgr PCMCIA (tal como -q para el modo tranquilo; -m para buscar por módulos del kernel cargables en el directorio especificado, etc.). Lea la página man de cardmgr para más información.

4.1.26. /etc/sysconfig/radvd

El archivo /etc/sysconfig/radvd es usado para pasar argumentos al demonio radvd en el momento de arranque. El demonio radvd escucha por peticiones del enrutador y envía notificaciones del enrutador para el protocolo IP versión 6. Este servicio permite a los host en una red cambiar dinámicamente sus enrutadores predeterminados basados en estas notificaciones del enrutador. Para más información sobre qué parámetros están disponibles para este archivo, vea la página del manual de radvd. Por defecto, este archivo coloca como propietario del proceso radvd al usuario radvd.

4.1.27. /etc/sysconfig/rawdevices

El archivo /etc/sysconfig/rawdevices es usado para configurar los enlaces de un dispositivo bruto, tal como:

```
/dev/raw/raw1 /dev/sda1
/dev/raw/raw2 8 5
```

4.1.28. `/etc/sysconfig/redhat-config-securitylevel`

El archivo `/etc/sysconfig/redhat-config-securitylevel` contiene todas las opciones escogidas por el usuario la última vez que fué ejecutada **Herramienta de configuración de nivel de seguridad** (`redhat-config-securitylevel`). Los usuarios no deberían modificar este archivo manualmente. Para más información sobre **Herramienta de configuración de nivel de seguridad**, consulte el capítulo de nombre *Configuración básica del cortafuegos* en el *Manual de personalización de Red Hat Linux*.

4.1.29. `/etc/sysconfig/redhat-config-users`

El archivo `/etc/sysconfig/redhat-config-users` es el archivo de configuración para la aplicación gráfica **Administrador de usuarios**. Bajo Red Hat Linux este archivo es usado para filtrar usuarios del sistema tal como `root`, `daemon`, o `lp`. Este archivo es editado mediante el menú desplegable **Preferencias => Filtrar usuarios y grupos del sistema** en la aplicación **Administrador de usuarios** y no debería ser modificado manualmente. Para más información sobre el uso de esta aplicación, vea el capítulo llamado *Configuración de usuarios y grupos* en el *Manual de personalización de Red Hat Linux*.

4.1.30. `/etc/sysconfig/redhat-logviewer`

El archivo `/etc/sysconfig/redhat-logviewer` es el archivo de configuración para la aplicación gráfica interactiva de visualización del registro, **Visor de registros del sistema**. Este archivo se puede modificar mediante el menú desplegable **Editar => Preferencias** en la aplicación **Visor de registros del sistema** y no debería ser modificado manualmente. Para más información sobre el uso de esta aplicación, consulte el capítulo llamado *Archivos de registro* en el *Manual de personalización de Red Hat Linux*.

4.1.31. `/etc/sysconfig/samba`

El archivo `/etc/sysconfig/samba` es usado para pasar argumentos a los demonios `smbd` y `nmbd` en el momento de arranque. El demonio `smbd` ofrece conectividad de archivos compartidos para los clientes Windows en la red. El demonio `nmbd` ofrece servicios NetBIOS sobre nombres IP. Para más información sobre los parámetros disponibles para este archivo, consulte la página de manual de `smbd`. Por defecto este archivo configura `smbd` y `nmbd` para que se ejecuten en modo demonio.

4.1.32. `/etc/sysconfig/sendmail`

El archivo `/etc/sysconfig/sendmail` permite enviar mensajes a uno o más recipientes, enrutando el mensaje sobre todas las redes que sean necesarias. El archivo configura los valores predeterminados para que la aplicación Sendmail se ejecute. Sus valores por defecto son ejecutarse como un demonio en el fondo y verificar su cola una vez cada hora en caso de que algo se haya acumulado.

Se usan los siguientes valores:

- `DAEMON=<value>`, donde `<value>` es uno de los siguientes valores boleanos:
  - `yes` — Sendmail debería ser configurado para escuchar en el puerto 25 para el correo entrante. `yes` implica el uso de las opciones de Sendmail `-bd`.
  - `no` — Sendmail no debería ser configurado para escuchar en el puerto 25 para el correo entrante.

- QUEUE=1h que es entregado a Sendmail como -q$QUEUE. La opción -q no es dada a Sendmail si /etc/sysconfig/sendmail existe y QUEUE es vacío o no está definida.

4.1.33. /etc/sysconfig/soundcard

El archivo /etc/sysconfig/soundcard es generado por sndconfig y no debería ser modificado. El único uso de este archivo es determinar qué entrada de tarjeta en el menú debe hacer pop up por defecto la próxima vez que sea ejecutado sndconfig. La información de configuración de la tarjeta de sonido está localizada en el archivo /etc/modules.conf.

Puede contener lo siguiente:

- CARDTYPE=<*value*>, donde <*value*> está configurada a, por ejemplo, SB16 para una tarjeta de sonido Soundblaster 16.

4.1.34. /etc/sysconfig/spamassassin

El archivo /etc/sysconfig/spamassassin es usado para pasar argumentos al demonio spamd (una versión endemoniada de Spamassassin) al momento del arranque. Spamassassin es una aplicación de filtro de correo spam. Para una lista de las opciones disponibles, consulte la página de manual de spamd. Por defecto, se configura spamd para ejecutarse en modo demonio, crear las preferencias del usuario y autocrear whitelists.

Para información sobre Spamassassin, consulte Sección 11.4.2.6.

4.1.35. /etc/sysconfig/squid

El archivo /etc/sysconfig/squid es usado para pasar argumentos al demonio squid al momento de arranque. El demonio squid es un servidor proxy caching para las aplicaciones cliente Web. Para más información sobre cómo configurar un servidor proxy squid, use un navegador Web para abrir el directorio /usr/share/doc/squid-<*version*>/ (reemplace <*version*> con el número de la versión de squid instalado en su sistema). Por defecto, este archivo configura squid para arrancar en modo demonio y establecer la cantidad de tiempo antes de que se cierre asímismo.

4.1.36. /etc/sysconfig/tux

El archivo /etc/sysconfig/tux es el archivo de configuración para el Acelerador de contenidos de Red Hat, en inglés Red Hat Content Accelerator (anteriormente conocido como TUX), el servidor Web basado en el kernel. Para más información sobre la configuración de Red Hat Content Accelerator, use un navegador de Web para abrir /usr/share/doc/tux-<*version*>/tux/index.html (reemplace <*version*> con el número de versión de TUX instalado en su sistema). Los parámetros disponibles para este archivo están listados en /usr/share/doc/tux-<*version*>/tux/parameters.html.

4.1.37. /etc/sysconfig/ups

El archivo /etc/sysconfig/ups es usado para especificar información sobre cualquier sistema contínuo de poder, *Uninterruptible Power Supplies* (*UPS*) conectado al sistema. Un UPS puede ser de gran utilidad para un sistema Red Hat Linux porque le dá tiempo al sistema para cerrarse correctamente en el caso de una interrupción de la energía. Se pueden usar los valores siguientes:

- SERVER=<*value*>, donde <*value*> es uno de los siguientes:

  - `yes` — Un dispositivo UPS está conectado al sistema.
  - `no` — No hay ningún dispositivo UPS conectado al sistema.

- `MODEL=<value>`, donde `<value>` debe ser uno de los siguientes o estar configurado a `NONE`, ninguno, si no hay ningún UPS conectado al sistema:
  - `apcsmart` — Un dispositivo APC SmartUPS™ o similar.
  - `fentonups` — Un Fenton UPS™.
  - `optiups` — Un dispositivo OPTI-UPS™.
  - `bestups` — Un UPS Best Power™.
  - `genericups` — Un UPS genérico.
  - `ups-trust425+625` — Un UPS Trust™.

- `DEVICE=<value>`, donde `<value>` especifica donde está conectado el UPS, tal como `/dev/ttyS0`.
- `OPTIONS=<value>`, donde `<value>` es un comando especial que necesita ser pasado al UPS.

4.1.38. `/etc/sysconfig/vncservers`

El archivo `/etc/sysconfig/vncservers` configura la forma en que el servidor *Virtual Network Computing* (*VNC*) arranca.

VNC es un sistema de despliegue remoto el cual permite a los usuarios ver el ambiente de escritorio no sólo en la máquina en que se está ejecutando sino también a través de las diferentes redes en una variedad de arquitecturas.

Puede contener lo siguiente:

- `VNCSERVERS=<value>`, donde `<value>` está configurado a algo parecido a `"1:fred"`, para indicar que el servidor VNC debería ser arrancado por el usuario fred en el despliegue :1. El usuario fred debe haber establecido una contraseña VNC usando `vncpasswd` antes de intentar conectarse al servidor VNC remoto.

Note que cuando esté usando un servidor VNC, la comunicación a través de él no está encriptada, por lo tanto no debería ser usado en una red insegura. Para instrucciones específicas concerniente al uso de SSH para asegurar la comunicación VNC, por favor lea la información encontrada en http://www.uk.research.att.com/vnc/sshvnc.html. Para saber más sobre SSH, consulte el Capítulo 18 o *Manual de personalización de Red Hat Linux*.

4.1.39. `/etc/sysconfig/xinetd`

El archivo `/etc/sysconfig/xinetd` es usado para pasar argumentos al demonio `xinetd` en el momento de arranque. El demonio `xinetd` arranca programas que proveen servicios de Internet cuando se recibe una petición en el puerto para ese servicio. Para más información sobre los parámetros disponibles para este archivo, consulte la página del manual de `xinetd`. Para más información sobre el servicio `xinetd`, consulte Sección 15.3.

4.2. Directorios en el directorio /etc/sysconfig/

Los siguientes directorios se encuentran normalmente en /etc/sysconfig/.

- apm-scripts/ — Este directorio contiene el script APM de suspender/reanudar de Red Hat. No modifique estos archivos directamente. Si es necesario realizar una personalización, cree un archivo llamado /etc/sysconfig/apm-scripts/apmcontinue y será llamado al final del script. También es posible controlar el script editando /etc/sysconfig/apmd.
- cbq/ — Este directorio contiene los archivos de configuración necesitados para hacer Class Based Queuing para la administración del ancho de banda en las interfaces de red.
- networking/ — Este directorio es usadp por la **Herramienta de administración de redes** (redhat-config-network), y sus contenidos no deberían ser modificados manualmente. Para más información sobre la configuración de interfaces de red usando **Herramienta de administración de redes**, consulte el capítulo llamado *Configuración de red* en el *Manual de personalización de Red Hat Linux*.
- network-scripts/ — Este directorio contiene los siguientes archivos de configuración relacionados a la red:
  - Archivos de configuración de red para cada interfaz de red configurada, tal como ifcfg-eth0 para la interfaz de red Ethernet eth0.
  - Scripts usado para subir y bajar interfaces de red, tales como ifup e ifdown.
  - Scripts usados para subir y bajar las interfaces ISDN, tales como ifup-isdn e ifdown-isdn.
  - Varios scripts de funciones de red compartidas los cuales no deberían ser modificados manualmente.

  Para más información sobre el directorio network-scripts, consulte el Capítulo 8.
- rhn/ — Este directorio contiene los archivos de configuración y claves GPG para la Red Hat Network. Ningún archivo en este directorio debería ser modificado manualmente. Para más información sobre Red Hat Network, consulte el sitio web de Red Hat Network en el siguiente URL: https://rhn.redhat.com.

4.3. Recursos adicionales

Este capítulo sólo tiene la intención de servir de introducción para los archivos en el directorio /etc/sysconfig/. Las siguientes fuentes contienen información más detallada.

4.3.1. Documentación instalada

- /usr/share/doc/initscripts-<*version-number*>/sysconfig.txt — Este archivo contiene un listado autorizado de los archivos encontrados en el directorio /etc/sysconfig/ y de las opciones de configuración disponibles para ellos. El <*version-number*> en la ruta a este archivo corresponde a la versión del paquete initscripts instalado.

redhat.

Capítulo 5.

El sistema de archivos `/proc`

Las funciones primarias del kernel de Linux son controlar el acceso a los dispositivos físicos del ordenador y establecer cuándo y cómo deben de tener lugar la interacción entre estos dispositivos.

Dentro del directorio `/proc/`, se puede encontrar una gran cantidad de información con detalles sobre el hardware del sistema y cualquier proceso que se esté ejecutando actualmente. Además, algunos de los archivos dentro del árbol de directorios `/proc/` pueden ser manipulados por los usuarios y aplicaciones para comunicar al kernel cambios en la configuración.

5.1. Sistema de archivos virtual

En Linux, todo se guarda en archivos. La mayoría de usuarios están familiarizados con los dos primeros tipos de archivos, de texto y binarios. Sin embargo, el directorio `/proc` contiene archivos que no son parte de ningún sistema de archivos asociado a los discos duros, CD-ROM o cualquier otro dispositivo físico de almacenamiento conectado a su sistema (excepto la RAM). Mejor dicho, estos archivos forman parte de un *sistema de archivos virtual* habilitado o deshabilitado en el kernel de Linux cuando está compilado.

Los archivos virtuales poseen cualidades únicas. En primer lugar, la mayoría de ellos tienen un tamaño de 0 bytes. Sin embargo, cuando se visualiza el archivo, éste puede contener una gran cantidad de información. Además, la mayoría de configuraciones del tiempo y las fechas reflejan el tiempo y fecha real, lo que es un indicativo de que están siendo constantemente modificados.

Archivos virtuales tales como `/proc/interrupts`, `/proc/meminfo`, `/proc/mounts`, y `/proc/partitions` proveen una vista rápida actualizada del hardware del sistema. Otros, como `/proc/filesystems` y el directorio `/proc/sys/` proveen información de configuración y de las interfaces.

Además, un gestor de sistemas puede utilizar `/proc` como método sencillo de información de acceso sobre el estado del kernel, los atributos de las máquinas, los estados de los procesos individuales y mucho más. La mayoría de archivos en este directorio, tales como `interrupts`, `meminfo`, `mounts` y `partitions` proporcionan una idea de lo que es un entorno de sistemas. Otros como `sistema de archivos` y el directorio `/proc/sys/` dan información sobre la configuración del software. Para facilitar las cosas, los archivos que contienen información sobre un tema parecido se agrupan en directorios virtuales y en subdirectorios, tales como `/proc/ide`.

5.1.1. Visualización de archivos virtuales

Mediante el uso de los comandos `cat`, `more`, o `less` en los archivos dentro del directorio `/proc/`, los usuarios pueden inmediatamente accesar una cantidad enorme de información acerca del sistema. Por ejemplo, para desplegar el tipo de CPU que tiene un equipo, escriba `cat /proc/cpuinfo` para recibir una salida similar a lo siguiente:

```
processor:0
vendor_id:AuthenticAMD
cpufamily:5
model:9
modelname:AMD-K6(tm)3D+Processor
stepping:1
cpuMHz:400.919
cachesize:256KB
fdiv_bug:no
hlt_bug:no
```

```
f00f_bug:no
coma_bug:no
fpu:yes
fpu_exception:yes
cpuidlevel:1
wp:yes
flags:fpuvmedepsetscmsrmcecx8pgemmxsyscall3dnowk6_mtrr
bogomips:799.53
```

Como puede ver en el sistema de archivos `/proc/`, alguna información tiene sentido, mientras que otras áreas aparecen en un código extraño. Por eso es que existen utilidades para extraer información de los archivos virtuales y mostrarla en una forma útil. Ejemplos de estas utilidades incluyen `lspci`, `apm`, `free`, y `top`.

Nota

Algunos archivos en el directorio `/proc/` están configurados para que se puedan leer sólo por el usuario root.

5.1.2. Cambiar archivos virtuales

En general, todos los archivos que se encuentran en el directorio `/proc` solamente se pueden leer. Sin embargo, algunos se pueden usar para ajustar la configuración del kernel. Esto ocurre con los archivos del subdirectorio `/proc/sys/`.

Para cambiar el valor de un archivo virtual use el comando `echo` y el símbolo > para redirigir el nuevo valor al archivo. De hecho, para cambiar el nombre del host escriba:

```
echo www.example.com>/proc/sys/kernel/hostname
```

Otros archivos actúan como intercambiadores binarios o boleanos. Si escribe `cat /proc/sys/net/ipv4/ip_forward` verá el valor `0` o el valor `1`. El valor `0` indica que el kernel no está realizando el forwarding de los paquetes. Si usa el comando `echo` para cambiar el valor del archivo `ip_forward` a `1`, el kernel activará inmediatamente el forwarding de los paquetes.

Sugerencia

Otro comando que se usa para cambiar la configuración del subdirectorio `/proc/sys/` es `/sbin/sysctl`. Para mayor información consulte Sección 5.4

Para consultar la lista de los archivos de configuración del kernel disponibles en `/proc/sys/` vaya a Sección 5.3.9.

5.2. Archivos de alto nivel en el sistema de archivos `proc`

La siguiente lista expone algunos de los archivos más comunes y útiles que se encuentran en el directorio `/proc`.

Nota

En la mayor parte de los casos, el contenido de los archivos que aparecen en esta sección no será el mismo que el de su máquina. Esto se debe a que la mayor parte de la información pertenece al hardware particular en el que esté ejecutando Red Hat Linux.

5.2.1. /proc/apm

Este archivo proporciona información acerca del estado de *Advanced Power Management (APM)* y es usado por el comando `apm`. Si un sistema sin batería está conectado a una fuente de poder AC, este archivo virtual se vería similar a:

```
1.16 1.2 0x07 0x01 0xff 0x80 -1% -1 ?
```

Al ejecutar un comando `apm -v` en estos sistemas resulta parecido a:

```
APM BIOS 1.2 (kernel driver 1.16)
AC on-line, no system battery
```

Para estos sistemas, que no usan una batería como fuente de poder, `apm` sólo será capaz de poner la máquina en modo *standby*, comúnmente se le conoce como "poner el sistema a dormir." El comando `apm` es mucho más útil en portátiles y otros sistemas Linux portables. Esto queda reflejado en los archivos `/proc/apm`. Esta es la salida de datos del comando `cat /proc/apm` desde un archivo de muestra en un portátil que está ejecutando Linux, mientras que está conectado a una toma de corriente:

```
1.16 1.2 0x03 0x01 0x03 0x09 100% - 1 ?
```

Cuando la misma portátil está desconectada de su fuente de energía durante algunos minutos, el contenido del archivo `apm` cambiará:

```
1.16 1.2 0x03 0x00 0x00 0x01 99% 1792 min
```

En este estado, el comando `apm -v` muestra información más útil tal como la siguiente:

```
APM BIOS 1.2 (kernel driver 1.16)
AC off-line, battery status high: 99% (1 day, 5:52)
```

5.2.2. /proc/cmdline

Este archivo muestra los parámetros pasados al kernel de Linux en el momento en que éste inicia. Ejemplo de archivo `/proc/cmdline`:

```
roroot=/dev/hda2
```

Esto nos dice que el kernel está montado como de sólo lectura (especificado por `(ro)`) fuera de la segunda partición en el primer dispositivo IDE (`/dev/hda2`).

5.2.3. /proc/cpuinfo

Este archivo virtual identifica el tipo de procesador usado por su sistema. A continuación se muestra un ejemplo de la salida típica de `/proc/cpuinfo`:

```
processor : 0
```

```
vendor_id : AuthenticAMD
cpufamily : 5
model  : 9
modelname : AMD-K6(tm) 3D+ Processor
stepping : 1
cpuMHz  : 400.919
cachesize : 256KB
fdiv_bug : no
hlt_bug  : no
f00f_bug : no
coma_bug : no
fpu  : yes
fpu_exception : yes
cpuidlevel : 1
wp  : yes
flags  : fpu vme de pse tsc msr mce cx8 pge mmx syscall 3dnow k6_mtrr
bogomips : 799.53
```

- `processor` — Proporciona un número de identificación para cada procesador. Si tan sólo posee un procesador, tan sólo verá un `0`.
- `cpu family` — Le da de forma autorizada el tipo de procesador que tiene en el sistema. Para un sistema basado en Intel, ponga el número delante del "86" para calcular el valor. Esto le servirá de ayuda si se está preguntando sobre el tipo de arquitectura de un sistema antiguo (686, 586, 486 or 386). Ya que los paquetes RPM están compilados para cada una de estas arquitecturas particulares, este valor le indica qué paquete instalar en el sistema.
- `model name` — Le indica el nombre conocido del procesador, incluyendo el nombre de proyecto.
- `cpu MHz` — Le muestra la velocidad precisa en megahertz de ese procesador en particular (en milésimas).
- `cache size` — Le indica la cantidad de memoria de nivel 2 de la caché disponible en el procesador.
- `flags` — Define un número de cualidades diferentes del procesador, como la presencia de una unidad de coma flotante (FPU) y la habilidad para procesar instrucciones MMX.

5.2.4. `/proc/devices`

Este archivo visualiza los diversos dispositivos de caracteres y de bloque actualmente configurados (no incluye dispositivos cuyos módulos no están cargados). El ejemplo de salida de datos de este archivo quedaría de la siguiente manera:

```
Character devices:
1 mem
2 pty
3 ttyp
4 ttyS
5 cua
7 vcs
10 misc
14 sound
29 fb
36 netlink
128 ptm
129 ptm
```

```
136 pts
137 pts
162 raw
254 iscsictl

Block devices:
1 ram disk
2 fd
3 ide0
9 md
22 ide1
```

La salida de datos desde `/proc/devices` incluye el número mayor y el nombre del dispositivo y se divide en dos secciones: `Dispositivos de caracteres` y `Dispositivos de bloque`.

Los *Dispositivos de caracteres* son similares a los *Dispositivos de bloque*, excepto por dos diferencias básicas:

1. En primer lugar, los dispositivos de bloque disponen de un buffer que les permite ordenar las peticiones antes de tratar con ellas. Esto es muy práctico con dispositivos diseñados para guardar información — tales como discos duros — porque la habilidad de ordenar la información antes de escribirla en el dispositivo permite que ésta se almacene de forma más eficiente. Los dispositivos de caracteres no requieren buffering.
2. Los dispositivos de bloque pueden enviar y recibir información en bloques de un tamaño particular, configurable para un dispositivo en particular. Los dispositivos de caracteres envían datos en los bytes necesarios, sin un tamaño preconfigurado.

Puede encontrar más información sobre los dispositivos en `/usr/src/linux-2.4/Documentation/devices.txt`.

5.2.5. `/proc/dma`

Este archivo contiene una lista de los canales de acceso de memoria directos (DMA) ISA registrados en uso. Ejemplo de los archivos `/proc/dma`:

```
4: cascade
```

5.2.6. `/proc/execdomains`

Este archivo lista los dominios de ejecución soportados en la actualidad por el kernel de Linux junto con la gama de personalidades que soportan.

```
0-0 Linux [kernel]
```

Piense en los *dominios de ejecución* como en una clase de "personalidad" de un sistema operativo en particular. Otros formatos binarios, como Solaris, UnixWare y FreeBSD pueden usarse con Linux. Al cambiar la personalidad de una tarea ejecutada bajo Linux, un programador puede cambiar el modo en el que el sistema operativo trata las llamadas del sistema particulares desde un binario. A excepción del dominio de ejecución `PER_LINUX`, el resto pueden ser implementados como módulos cargables de forma dinámica.

5.2.7. /proc/fb

Este archivo contiene una lista de dispositivos frame buffer, con el número del dispositivo frame buffer y el driver que lo controla. La salida de datos más común de `/proc/fb` para sistemas que contienen dispositivos de frame buffer es:

```
0 VESA VGA
```

5.2.8. /proc/filesystems

Este archivo visualiza una lista de los tipos del sistema de archivos actuales soportados por el kernel. A continuación tiene un ejemplo de salida de datos desde un archivo `/proc/filesystems` de un kernel genérico:

```
nodev rootfs
nodev bdev
nodev proc
nodev sockfs
nodev tmpfs
nodev shm
nodev pipefs
      ext2
nodev ramfs
      iso9660
nodev devpts
      ext3
nodev autofs
nodev binfmt_misc
```

La primera columna significa si el sistema de archivos está montado en un dispositivo de bloque. Aquellos que comiencen con `nodev` no están montados en un dispositivo. La segunda columna lista el nombre de los sistemas de archivos soportados.

El comando `mount` circula por estos sistemas de archivos listados aquí cuando uno no está especificado como un argumento.

5.2.9. /proc/interrupts

Este archivo graba el número de interrupciones por IRQ en la arquitectura x86. Ejemplo de un archivo `interrupts` estándar:

```
CPU0
0: 80448940 XT-PIC timer
1: 174412  XT-PIC keyboard
2: 0  XT-PIC cascade
8: 1  XT-PIC rtc
10: 410964  XT-PIC eth0
12: 60330  XT-PIC PS/2 Mouse
14: 1314121  XT-PIC ide0
15: 5195422  XT-PIC ide1
NMI: 0
ERR: 0
```

Para una máquina con múltiples procesadores, el archivo aparecerá de forma diferente:

```
 CPU0  CPU1
```

```
0: 1366814704 0 XT-PIC  timer
1: 128  340 IO-APIC-edge keyboard
2: 0  0 XT-PIC  cascade
8: 0  1 IO-APIC-edge rtc
12: 5323  5793 IO-APIC-edge PS/2 Mouse
13: 1  0 XT-PIC  fpu
16: 11184294     15940594   IO-APIC-level Intel EtherExpress Pro 10/100 Ethernet
20: 8450043      11120093 IO-APIC-level megaraid
30: 10432     10722 IO-APIC-level aic7xxx
31: 23  22 IO-APIC-levela ic7xxx
NMI: 0
ERR: 0
```

La primera columna se refiere al número de IRQ. Cada CPU del sistema tiene su propia columna y su propio número de interrupciones por IRQ. La columna siguiente le indica el tipo de interrupción y la última contiene el nombre del dispositivo que está localizado en ese IRQ.

Cada uno de los tipos de interrupciones vistos en este archivo, que son específicos para el tipo de arquitectura, significan algo diferente. Los siguientes valores son comunes para las máquinas x86:

- `XT-PIC` — Interrupciones del ordenador AT antiguo que se han producido por un largo periodo de tiempo.
- `IO-APIC-edge` — Señal de voltaje de las transacciones interrumpidas desde abajo hasta arriba, creando una *edge*, en la que la interrupción `IO-APIC-level`, tan sólo se dan a partir de procesadores 586 y superiores.
- `IO-APIC-level` — Genera interrupciones cuando su señal de voltaje se alza hasta que la señal desciende de nuevo.

5.2.10. `/proc/iomem`

Este archivo muestra el mapa actual de la memoria del sistema para los diversos dispositivos:

```
00000000-0009fbff : System RAM
0009fc00-0009ffff : reserved
000a0000-000bffff : Video RAM area
000c0000-000c7fff : Video ROM
000f0000-000fffff : System ROM
00100000-07ffffff : System RAM
  00100000-00291ba8 : Kernel code
  00291ba9-002e09cb : Kernel data
e0000000-e3ffffff : VIA Technologies, Inc. VT82C597 [Apollo VP3]
e4000000-e7ffffff : PCI Bus #01
  e4000000-e4003fff : Matrox Graphics, Inc. MGA G200 AGP
  e5000000-e57fffff : Matrox Graphics, Inc. MGA G200 AGP
e8000000-e8ffffff : PCI Bus #01
  e8000000-e8ffffff : Matrox Graphics, Inc. MGA G200 AGP
ea000000-ea00007f : Digital Equipment Corporation DECchip 21140 [FasterNet]
  ea000000-ea00007f : tulip
ffff0000-ffffffff : reserved
```

La primera columna muestra los registros de memoria utilizados para cada uno de los diferentes tipos de memoria. La segunda columna indica el tipo de memoria de dichos registros. En particular, esta columna le mostrará qué registros de memoria son usados por el kernel dentro de la RAM del sistema o, si tiene puertos Ethernet múltiples en su NIC, los registros de memoria asignados para cada puerto.

5.2.11. /proc/ioports

De forma similar a /proc/iomem, /proc/ioports proporciona una lista de puertos registrados actualmente utilizados para la comunicación de entrada y salida con un dispositivo. Este archivo puede ser muy largo y empezaría de la siguiente manera:

```
0000-001f:  dma1
0020-003f:  pic1
0040-005f:  timer
0060-006f:  keyboard
0070-007f:  rtc
0080-008f:  dma page reg
00a0-00bf:  pic2
00c0-00df:  dma2
00f0-00ff:  fpu
0170-0177:  ide1
01f0-01f7:  ide0
02f8-02ff:  serial (auto)
0376-0376:  ide1
03c0-03df:  vga+
03f6-03f6:  ide0
03f8-03ff:  serial (auto)
0cf8-0cff:  PCI conf1
d000-dfff:  PCI Bus #01
e000-e00f:  VIA Technologies, Inc. Bus Master IDE
e000-e007:  ide0
e008-e00f:  ide1
e800-e87f:  Digital Equipment Corporation DECchip 21140 [FasterNet]
e800-e87f:  tulip
```

La primera columna le indica el rango de direcciones de los puertos de entrada y salida reservado para el dispositivo listado en la segunda columna.

5.2.12. /proc/isapnp

Este archivo lista las tarjetas *Plug and Play (PnP)* en espacios ISA del sistema. Esto es mucho más habitual con las tarjetas de sonido, pero puede incluir cualquier número de dispositivos. Un archivo /proc/isapnp con una entrada Soundblaster en él, sería de la siguiente manera:

```
Card 1 'CTL0070:Creative ViBRA16C PnP' PnP version 1.0 Product version 1.0
  Logical device 0 'CTL0001:Audio'
    Device is not active
    Active port 0x220,0x330,0x388
    Active IRQ 5 [0x2]
    Active DMA 1,5
    Resources 0
      Priority preferred
      Port 0x220-0x220, align 0x0, size 0x10, 16-bit address decoding
      Port 0x330-0x330, align 0x0, size 0x2, 16-bit address decoding
      Port 0x388-0x3f8, align 0x0, size 0x4, 16-bit address decoding
      IRQ 5 High-Edge
      DMA 1 8-bit byte-count compatible
      DMA 5 16-bit word-count compatible
      Alternate resources 0:1
        Priority acceptable
        Port 0x220-0x280, align 0x1f, size 0x10, 16-bit address decoding
        Port 0x300-0x330, align 0x2f, size 0x2, 16-bit address decoding
```

```
Port 0x388-0x3f8, align 0x0, size 0x4, 16-bit address decoding
IRQ 5,7,2/9,10 High-Edge
DMA 1,3 8-bit byte-count compatible
DMA 5,7 16-bit word-count compatible
```

Este archivo podría ser bastante largo dependiendo del número de dispositivos visualizados y de los requisitos o peticiones de recursos.

Cada tarjeta lista su nombre, número de versión PnP y versión del producto. Si el dispositivo está activo y configurado, este archivo revelará el puerto y los números de IRQ para el dispositivo. Además, para asegurar una mejor compatibilidad, la tarjeta especificará los valores `preferred` y `acceptable` para un número de parámetros diferentes. El objetivo es el de permitir que las tarjetas PNP funcionen en base a otra y evitar los problemas de IRQ y puertos.

5.2.13. /proc/kcore

Este archivo representa la memoria física del sistema y se almacena en el formato del archivo central. A diferencia de la mayoría de archivos `/proc/`, `kcore` muestra un tamaño. Este valor se da en bytes y es igual al tamaño de la memoria física (RAM) utilizada más 4KB.

Sus contenidos están diseñados para que los examine un depurador, como por ejemplo `gdb`, y no es legible para humanos.

Aviso

Evite visualizar el archivo virtual `/proc/kcore`. Los contenidos de este archivo se saldrán del terminal. Si accidentalmente lo visualiza, pulse [Ctrl]-[C] para detener el proceso y luego escriba `reset` para volver a la línea de comandos del prompt en la estaba.

5.2.14. /proc/kmsg

Este archivo se utiliza para mantener mensajes generados por el kernel y otros programas toman dichos mensajes, como por ejemplo `/sbin/klogd`.

5.2.15. /proc/ksyms

Este archivo contiene las definiciones del símbolo exportado del kernel usadas por las herramientas de módulos para enlazar y dinámicamente módulos cargables.

```
e003def4 speedo_debug [eepro100]
e003b04c eepro100_init [eepro100]
e00390c0 st_template [st]
e002104c RDINDOOR [megaraid]
e00210a4 callDone [megaraid]
e00226cc megaraid_detect [megaraid]
```

La segunda columna se refiere al nombre de una función del kernel y la primera columna lista la dirección de la memoria para dicha función. La última columna revela el nombre del módulo cargado para proporcionar dicha función.

5.2.16. /proc/loadavg

Este archivo ofrece una vista preliminar del promedio de carga del procesador, del mismo modo que le ofrece datos adicionales utilizados por `uptime` y otros comandos. Un archivo de ejemplo `/proc/loadavg` sería similar a lo siguiente:

```
0.20 0.18 0.12 1/80 11206
```

Las primeras tres columnas miden el uso de una CPU en los últimos periodos de 1, 5 y 10 minutos. La cuarta columna le muestra el número de procesos en ejecución en la actualidad y el número total de los mismos. La última columna visualiza el último ID de proceso usado.

5.2.17. /proc/locks

Estos archivos muestran los archivos bloqueados en la actualidad por el kernel. El contenido de este archivo contiene datos internos de depuración y puede variar enormemente, dependiendo del uso del sistema. Este es un ejemplo de archivo `/proc/locks` de un sistema ligeramente cargado:

```
1: FLOCK  ADVISORY  WRITE 807 03:05:308731 0 EOF c2a260c0 c025aa48 c2a26120
2: POSIX  ADVISORY  WRITE 708 03:05:308720 0 EOF c2a2611c c2a260c4 c025aa48
```

A cada bloqueo se le asigna un único número al inicio de cada línea. La segunda columna se refiere a la clase de bloqueo utilizado; `FLOCK`, haciendo referencia al estilo antiguo de bloqueos de archivos desde una llamada de sistema `flock` y `POSIX` que representa los bloqueos nuevos POSIX desde la llamada de sistema `lockf`.

La tercera columna puede tener dos valores. `ADVISORY` o `MANDATORY`. `ADVISORY` significa que el bloqueo no impide que otras personas puedan acceder a los datos; tan sólo previene de otros intentos de bloqueo. `MANDATORY` significa que mientras que dura el bloqueo no se permite ningún otro acceso a los datos. La cuarta columna muestra si el bloqueo permite al responsable del mismo acceso de `READ` o `WRITE` al archivo. La quinta muestra el ID del proceso que tiene el bloqueo. La sexta columna muestra el ID del archivo bloqueado, en el formato de *MAJOR-DEVICE*:*MINOR-DEVICE*:*INODE-NUMBER*. La séptima muestra el inicio y el final de la región bloqueada del archivo. Las columnas restantes señalan las estructuras de los datos del kernel interno usadas para una depuración especializada y no hace falta tenerlas en cuenta.

5.2.18. /proc/mdstat

Este archivo contiene información actual sobre las configuración de discos múltiples, de RAID. Si su sistema no contiene dicha configuración, el archivo `/proc/mdstat` será parecido a:

```
Personalities:
read_ahead not set
unused devices: <none>
```

Este archivo se mantiene en el mismo estado que el mostrado arriba a menos que un software RAID o dispositivo `md` esté presente. En ese caso, visualice `/proc/mdstat` para ver el estado actual de los dispositivos RAID `md`*X*.

El archivo `/proc/mdstat` a continuación, muestra un sistema con su `md0` configurado como un dispositivo RAID 1, mientras está resincronizando los discos:

```
Personalities : [linear] [raid1]
read_ahead 1024 sectors
md0: active raid1 sda2[1] sdb2[0] 9940 blocks [2/2] [UU] resync=1%
finish=12.3min
```

```
algorithm 2 [3/3] [UUU]
unused devices: <none>
```

5.2.19. /proc/meminfo

Este es uno de los archivos más utilizados en el directorio /proc/, ya que proporciona mucha información importante sobre el uso actual de RAM en el sistema.

El siguiente ejemplo de archivo virtual /proc/meminfo es de un sistema con 256MB de RAM y 384MB de espacio swap:

```
total:    used:    free:  shared: buffers:  cached:
Mem:  261709824 253407232  8302592        0 120745984 48689152
Swap: 402997248     8192 402989056
MemTotal:       255576 kB
MemFree:          8108 kB
MemShared:           0 kB
Buffers:        117916 kB
Cached:          47548 kB
Active:         135300 kB
Inact_dirty:     29276 kB
Inact_clean:       888 kB
Inact_target:        0 kB
HighTotal:           0 kB
HighFree:            0 kB
LowTotal:       255576 kB
LowFree:          8108 kB
SwapTotal:      393552 kB
SwapFree:       393544 kB
```

El comando top utiliza la mayoría de la información. De hecho, la salida de datos del comando free es parecida, aparentemente, al contenido y estructura de meminfo. Si lee directamente meminfo conocerá muchos detalles sobre la memoria:

- Mem — Muestra el estado actual de RAM física en el sistema, incluyendo el uso en bytes de memoria total usada, libre, compartida, buffer y caché.
- Swap — Muestra la cantidad total de espacio swap libre y usado en bytes.
- MemTotal — Cantidad total de RAM física en kilo bytes.
- MemFree — Cantidad de RAM física, en kilobytes, sin utilizar por el sistema.
- MemShared — No se utiliza con 2.4 y kernels superiores pero se deja por motivos de compatibilidad con versiones del kernel precedentes.
- Buffers — Cantidad de RAM física, en kilobytes, usada para los archivos de buffers.
- Cached — Cantidad de RAM física en kilobytes usada como memoria caché.
- Active — Cantidad total de buffer o memoria caché de página, en kilobytes, que está en uso activo.
- Inact_dirty — Cantidad total de buffer y páginas de la caché, en kilobytes, que podrían quedar libres.
- Inact_clean — Cantidad total de buffer o páginas de la caché, en kilobytes, que están libres y disponibles.
- Inact_target — Cantidad neta de asignaciones por segundo, en kilobytes, con un promedio de un minuto.

- HighTotal y HighFree — Cantidad total de memoria libre, que no está mapeada en el espacio del kernel. El valor HighTotal puede variar dependiendo del tipo de kernel utilizado.
- LowTotal y LowFree — Cantidad total de memoria libre implantada directamente en el espacio del kernel. El valor LowTotal puede cambiar dependiendo del tipo de kernel utilizado.
- SwapTotal — Cantidad total de swap disponible, en kilobytes.
- SwapFree — Cantidad total de swap libre, en kilobytes.

5.2.20. /proc/misc

Este archivo lista varios controladores registrados en el principal dispositivo de misceláneos, que es el número 10:

```
135 rtc
  1 psaux
134 apm_bios
```

La primera columna es el número menor (minor) de cada dispositivo y la segunda le muestra el controlador en uso.

5.2.21. /proc/modules

Este archivo visualiza una lista de todos los módulos que han sido cargados por el sistema. Su contenido variará dependiendo de la configuración y uso de su sistema, pero debería organizarse de forma similar al siguiente ejemplo de salida de datos del archivo /proc/modules:

```
ide-cd                  27008   0 (autoclean)
cdrom                   28960   0 (autoclean) [ide-cd]
soundcore                4100   0 (autoclean)
agpgart                 31072   0 (unused)
binfmt_misc              5956   1
iscsi                   32672   0 (unused)
scsi_mod                94424   1 [iscsi]
autofs                  10628   0 (autoclean) (unused)
tulip                   48608   1
ext3                    60352   2
jbd                     39192   2 [ext3]
```

La primera columna contiene el nombre del módulo. La segunda se refiere al tamaño de memoria del módulo en bytes. La tercera le indica si el módulo está cargado (1) o descargado (0). La última columna le indica si el módulo puede descargarse por sí mismo automáticamente tras un periodo en el que no se ha usado (autoclean) o si no se está utilizando (unused). Cualquier módulo con una línea que contenga un nombre listado entre corchetes ([o]) le indica que este módulo depende de que otro esté presente para que funcione.

5.2.22. /proc/mounts

Este archivo proporciona una lista rápida de todos los montajes en uso:

```
rootfs / rootfs rw 0 0
/dev/hda2 / ext3 rw 0 0
/proc /proc proc rw 0 0
/dev/hda1 /boot ext3 rw 0 0
none /dev/pts devpts rw 0 0
```

```
none /dev/shm tmpfs rw 0 0
none /proc/sys/fs/binfmt_misc binfmt_misc rw 0 0
```

La salida de datos que encontramos se parece `/etc/mtab`, excepto que `/proc/mount` puede ser más actual.

La primera columna especifica el dispositivo que está montado y la segunda revela el punto de montaje. La tercera le indica el tipo de sistema de archivos y la cuarta si está montado en modo sólo lectura (`ro`) o sólo escritura (`rw`). La quinta y sexta columna son valores no válidos diseñados para hacer coincidir el formato usado en `/etc/mtab`.

5.2.23. /proc/mtrr

Este archivo se refiere a la memoria actual Memory Type Range Registers (MTRRs) en uso dentro del sistema. Si la arquitectura de su sistema soporta MTRRs, entonces el archivo `/proc/mtrr` será algo parecido a:

```
reg00: base=0x00000000 (0 MB), size= 64MB: write-back, count= 1
```

Los MTRRs se usan con la familia de procesadores Intel P6 (Pentium II y superior), y controlan el acceso del procesador a los rangos de memoria. Cuando utilice una tarjeta de vídeo en un PCI o un bus AGP, un archivo `/proc/mtrr` adecuadamente configurado puede incrementar la ejecución en un 150%.

La mayoría de las veces, este valor está por defecto configurado adecuadamente. Para más información sobre la configuración de este archivo de forma manual, vea el URL: http://web1.linuxhq.com/kernel/v2.3/doc/mtrr.txt.html.

5.2.24. /proc/partitions

La mayoría de la información no es relevante para los usuarios, a excepción de las siguientes líneas:

- `major` — Número principal del dispositivo con esta partición. El número principal en nuestro ejemplo (`3`) corresponde con el dispositivo `ide0` en `/proc/devices`, permitiéndonos conocer el tipo de controlador de dispositivo usado para interactuar con esa partición.
- `minor` — Número menor del dispositivo con esta partición. Separa las particiones en diferentes dispositivos físicos y los relaciona con el número al final del nombre de la partición.
- `#blocks` — Lista el número de bloques de disco físicos contenidos en una partición particular.
- `name` — Nombre de la partición.

5.2.25. /proc/pci

El archivo contiene una lista completa de cada dispositivo PCI de su sistema. Dependiendo del número de dispositivos PCI que posea, `/proc/pci` puede ser bastante largo. Ejemplo de este archivo en un sistema básico:

```
Bus  0, device   0, function  0:
  Host bridge: Intel Corporation 440BX/ZX - 82443BX/ZX Host bridge (rev 3).
    Master Capable.  Latency=64.
    Prefetchable 32 bit memory at 0xe4000000 [0xe7ffffff].
Bus  0, device   1, function  0:
  PCI bridge: Intel Corporation 440BX/ZX - 82443BX/ZX AGP bridge (rev 3).
    Master Capable.  Latency=64.  Min Gnt=128.
```

```
Bus  0, device   4, function  0:
  ISA bridge: Intel Corporation 82371AB PIIX4 ISA (rev 2).
Bus  0, device   4, function  1:
  IDE interface: Intel Corporation 82371AB PIIX4 IDE (rev 1).
    Master Capable.  Latency=32.
    I/O at 0xd800 [0xd80f].
Bus  0, device   4, function  2:
  USB Controller: Intel Corporation 82371AB PIIX4 USB (rev 1).
    IRQ 5.
    Master Capable.  Latency=32.
    I/O at 0xd400 [0xd41f].
Bus  0, device   4, function  3:
  Bridge: Intel Corporation 82371AB PIIX4 ACPI (rev 2).
    IRQ 9.
Bus  0, device   9, function  0:
  Ethernet controller: Lite-On Communications Inc LNE100TX (rev 33).
    IRQ 5.
    Master Capable.  Latency=32.
    I/O at 0xd000 [0xd0ff].
    Non-prefetchable 32 bit memory at 0xe3000000 [0xe30000ff].
Bus  0, device  12, function  0:
  VGA compatible controller: S3 Inc. ViRGE/DX or /GX (rev 1).
    IRQ 11.
    Master Capable.  Latency=32.  Min Gnt=4.Max Lat=255.
    Non-prefetchable 32 bit memory at 0xdc000000 [0xdfffffff].
```

Esta salida de datos muestra una lista de todos los dispositivos PCI, en orden de bus, dispositivo y función. Además de proporcionar el nombre y versión del dispositivo, esta lista le proporciona información de IRQ detallada y así un administrador puede rápidamente dar un vistazo para verificar conflictos.

Sugerencia

Para obtener una versión más fácil de leer, escriba:

```
/sbin/lspci -vb
```

5.2.26. /proc/slabinfo

Este archivo le da información sobre el uso de memoria en el nivel *slab*. Los kernels Linux superiores a la versión 2.2 usan *slab pools* para manejar memoria por encima del nivel de página. Los objetos utilizados habitualmente, tienen sus propios slab pools. A continuación le mostramos una parte de un típico archivo virtual `/proc/slabinfo`:

```
slabinfo - version: 1.1
kmem_cache            64      68    112    2    2    1
nfs_write_data         0       0    384    0    0    1
nfs_read_data          0     160    384    0   16    1
nfs_page               0     200     96    0    5    1
ip_fib_hash           10     113     32    1    1    1
journal_head          51    7020     48    2   90    1
revoke_table           2     253     12    1    1    1
revoke_record          0       0     32    0    0    1
clip_arp_cache         0       0    128    0    0    1
```

```
ip_mrt_cache           0      0     96    0    0    1
```

Los valores en este archivo acontecen en el siguiente orden: nombre de la caché, nombre de objetos activos, nombre de objetos totales, tamaño del objeto, nombre de slabs activos (bloques) de los objetos, número total de slabs de los objetos y el número de páginas por slab.

Cabe remarcar que la palabra *activo* significa que el objeto está en uso.

5.2.27. /proc/stat

Este archivo le aporta diferentes estadísticas sobre el sistema desde que fue reiniciado por última vez. El contenido de `/proc/stat` que puede ser muy largo, empieza de la siguiente manera:

```
cpu  1139111 3689 234449 84378914
cpu0 1139111 3689 234449 84378914
page 2675248 8567956
swap 10022 19226
intr 93326523 85756163 174412 0 3 3 0 6 0 1 0 428620 0 60330 0 1368304 5538681
disk_io: (3,0):(1408049,445601,5349480,962448,17135856)
ctxt 27269477
btime 886490134
processes 206458
```

Algunas de las estadísticas más populares incluyen:

- `cpu` — Mide el número de *jiffies* (1/100 de un segundo) que el sistema ha estado en modo usuario, modo usuario con prioridad baja, modo del sistema y tarea inactiva respectivamente. El total de todas las CPUs se da al inicio y cada CPU individual se lista debajo con sus propias estadísticas.
- `page` — Número de páginas que el sistema ha cargado o suprimido del disco.
- `swap` — Número de páginas swap que el sistema ha introducido o sacado.
- `intr` — Número de interrupciones que ha experimentado el sistema.
- `btime` — Tiempo de arranque, medido por el número de segundos desde el 1 de enero de 1970, conocido con el nombre de *epoch*.

5.2.28. /proc/swaps

Este archivo mide el espacio swap y su uso. Para un sistema con tan sólo una partición de espacio swap, la salida de datos de `/proc/swap` será:

```
Filename   Type  Size Used Priority
/dev/hda6                      partition 136512 20024 -1
```

Mientras que alguna de esta información se puede encontrar en otros archivos en el directorio `/proc/`, `/proc/swap` proporciona una instantánea rápida de cada nombre de archivo swap, tipo de espacio swap, el tamaño total, y la cantidad de espacio en uso (en kilobytes). La columna de prioridad es útil cuando múltiples archivos swap están en uso. Cuanto más baja es la prioridad, más probable es que se use el archivo swap.

5.2.29. /proc/uptime

El archivo contiene información sobre el tiempo que lleva encendido el sistema desde el último reinicio. La salida de datos de /proc/uptime es mínima:

```
350735.47 234388.90
```

El primer número le indica el número total de segundos que el sistema ha estado en funcionamiento. El segundo indica cuánto de ese tiempo, también en segundos, la máquina ha estado inactiva.

5.2.30. /proc/version

Estos archivos muestran las versiones del kernel de Linux y gcc en uso, así como la versión de Red Hat Linux instalada en el sistema:

```
Linux version 2.4.20-0.40 (user@foo.redhat.com)(gcc version 3.2.1 20021125
(Red Hat Linux 8.0 3.2.1-1)) #1 Tue Dec 3 20:50:18 EST 2002
```

Esta información se usa para diversos propósitos, incluyendo la aportación de datos de la versión en el intérprete de comandos de registro estándar.

5.3. Directorios en /proc

Grupos comunes de información referente al kernel agrupado en directorios y subdirectorios en /proc/.

5.3.1. Directorios de proceso

Cada directorio /proc/ contiene unos cuantos directorios nombrados con un número. Un listado de los mismos empezaría de la siguiente manera:

```
dr-xr-xr-x    3 root     root            0 Feb 13 01:28 1
dr-xr-xr-x    3 root     root            0 Feb 13 01:28 1010
dr-xr-xr-x    3 xfs      xfs             0 Feb 13 01:28 1087
dr-xr-xr-x    3 daemon   daemon          0 Feb 13 01:28 1123
dr-xr-xr-x    3 root     root            0 Feb 13 01:28 11307
dr-xr-xr-x    3 apache   apache          0 Feb 13 01:28 13660
dr-xr-xr-x    3 rpc      rpc             0 Feb 13 01:28 637
dr-xr-xr-x    3 rpcuser  rpcuser         0 Feb 13 01:28 666
```

A estos directorios se les llama *directorios de proceso*, ya que pueden hacer referencia a un ID de proceso y contener información específica para ese proceso. El propietario y grupo de cada directorio de proceso está configurado para que el usuario ejecute el proceso. Cuando se finaliza el proceso, el directorio del proceso /proc desaparece. Sin embargo, mientras que se está ejecutando el proceso, una gran cantidad de información específica a ese proceso está contenida en varios archivos del directorio de procesos.

Cada uno de los directorios de procesos contiene los siguientes archivos:

- cmdline — Contiene el comando que se ejecutó cuando se arrancó el proceso.
- cpu — Proporciona información específica sobre el uso de cada uno de las CPUs del sistema. Un proceso que se ejecuta en un sistema CPU dual produciría la siguiente salida de datos:

```
cpu  11 3
cpu0 0  0
```

```
cpu1 11 3
```

- `cwd` — Enlace simbólico al directorio actual en funcionamiento para el proceso.
- `environ` — Le da una lista de variables de entorno para el proceso. La variable de entorno en mayúsculas y el valor en minúsculas.
- `exe` — Enlace simbólico al ejecutable de este proceso.
- `fd` — Directorio que contiene todos los descriptores de archivos para un proceso en particular. Vienen dados en enlaces numerados:

```
total 0
lrwx------    1 root     root           64 May  8 11:31 0 -> /dev/null
lrwx------    1 root     root           64 May  8 11:31 1 -> /dev/null
lrwx------    1 root     root           64 May  8 11:31 2 -> /dev/null
lrwx------    1 root     root           64 May  8 11:31 3 -> /dev/ptmx
lrwx------    1 root     root           64 May  8 11:31 4 -> socket:[7774817]
lrwx------    1 root     root           64 May  8 11:31 5 -> /dev/ptmx
lrwx------    1 root     root           64 May  8 11:31 6 -> socket:[7774829]
lrwx------    1 root     root           64 May  8 11:31 7 -> /dev/ptmx
```

- `maps` — Contiene mapas de memoria para los diversos ejecutables y archivos de librería asociados con este proceso. Este archivo puede ser bastante largo, dependiendo de la complejidad del proceso. Una muestra de la salida de datos desde el proceso `sshd` empezaría de la siguiente manera:

```
08048000-08086000 r-xp 00000000 03:03 391479     /usr/sbin/sshd
08086000-08088000 rw-p 0003e000 03:03 391479     /usr/sbin/sshd
08088000-08095000 rwxp 00000000 00:00 0
40000000-40013000 r-xp 00000000 03:03 293205     /lib/ld-2.2.5.so
40013000-40014000 rw-p 00013000 03:03 293205     /lib/ld-2.2.5.so
40031000-40038000 r-xp 00000000 03:03 293282     /lib/libpam.so.0.75
40038000-40039000 rw-p 00006000 03:03 293282     /lib/libpam.so.0.75
40039000-4003a000 rw-p 00000000 00:00 0
4003a000-4003c000 r-xp 00000000 03:03 293218     /lib/libdl-2.2.5.so
4003c000-4003d000 rw-p 00001000 03:03 293218     /lib/libdl-2.2.5.so
```

- `mem` — Memoria del proceso.
- `root` — Enlace al directorio root del proceso.
- `stat` — Estado del proceso.
- `statm` — Estado de la memoria en uso por el proceso. Ejemplo de archivos `statm`:

```
263 210 210 5 0 205 0
```

  Las siete columnas se relacionan a diferentes estadísticas de memoria para el proceso. Dependiendo de como se visualizan, de derecha a izquierda, remiten diferentes aspectos de la memoria utilizada:

  1. Tamaño total del programa, en kilobytes
  2. Tamaño de porciones de memoria, en kilobytes
  3. Número de páginas compartidas
  4. Número de páginas en código
  5. Número de páginas de datos/grupos
  6. Número de páginas de librería
  7. Número de páginas sucias

- `status` — Proporciona el estado del proceso en una forma mucho más legible que `stat` o `statm`. Ejemplo de salida de datos de `sshd`:

```
Name: sshd
State: S (sleeping)
```

```
Tgid: 797
Pid: 797
PPid: 1
TracerPid: 0
Uid: 0 0 0 0
Gid: 0 0 0 0
FDSize: 32
Groups:
VmSize:     3072 kB
VmLck:         0 kB
VmRSS:       840 kB
VmData:      104 kB
VmStk:        12 kB
VmExe:       300 kB
VmLib:      2528 kB
SigPnd: 0000000000000000
SigBlk: 0000000000000000
SigIgn: 8000000000001000
SigCgt: 0000000000014005
CapInh: 0000000000000000
CapPrm: 00000000ffffffff
CapEff: 00000000ffffffff
```

La información en esta salida incluye el nombre y ID del proceso, el estado (tal como `S (sleeping)` o `R (running)`), ID del usuario/grupo ejecutando el proceso y detalles sobre el uso de la memoria.

5.3.1.1. /proc/self

El directorio `/proc/self` es un enlace al proceso en ejecución. Esto le permite verse a si mismo sin tener que conocer su ID de proceso.

Dentro de un entorno de la shell, una lista del directorio `/proc/self` produce el mismo contenido que una lista del directorio del proceso para ese proceso.

5.3.2. /proc/bus/

Este directorio contiene información específica sobre los diversos busses disponibles en el sistema. Por ejemplo, en un sistema estándar que contenga busses ISA, PCI y USB, los datos actuales de cada uno de estos busses están disponibles en el directorio bajo `/proc/bus/`.

Los contenidos de los subdirectorios y archivos disponibles varían según la configuración precisa de su sistema. No obstante, cada uno de los directorios para cada uno de los tipos de bus contiene al menos un directorio para cada bus de ese tipo. Estos directorios bus individuales, habitualmente referenciados con números, tales como `00`, contienen archivos binarios que se refieren a los varios dispositivos disponibles en ese bus.

Por ejemplo, un sistema con un bus USB sin ningún USB conectado a este tiene un directorio `/proc/bus/usb` que contiene varios archivos:

```
total 0
dr-xr-xr-x    1 root     root            0 May  3 16:25 001
-r--r--r--    1 root     root            0 May  3 16:25 devices
-r--r--r--    1 root     root            0 May  3 16:25 drivers
```

El directorio `/proc/bus/usb/` contiene archivos que hacen un seguimiento de los diferentes dispositivos en cualquier bus USB, así como los controladores requeridos para su uso. El directorio

/proc/bus/usb/001/ contiene todos los dispositivos del primer bus USB. Al mirar el contenido del archivo devices, se puede identificar el hub raíz USB en la tarjeta madre.

```
T:  Bus=01 Lev=00 Prnt=00 Port=00 Cnt=00 Dev#=  1 Spd=12  MxCh= 2
B:  Alloc=  0/900 us ( 0%), #Int=  0, #Iso=  0
D:  Ver= 1.00 Cls=09(hub  ) Sub=00 Prot=00 MxPS= 8 #Cfgs=  1
P:  Vendor=0000 ProdID=0000 Rev= 0.00
S:  Product=USB UHCI Root Hub
S:  SerialNumber=d400
C:* #Ifs= 1 Cfg#= 1 Atr=40 MxPwr=  0mA
I:  If#= 0 Alt= 0 #EPs= 1 Cls=09(hub  ) Sub=00 Prot=00 Driver=hub
E:  Ad=81(I) Atr=03(Int.) MxPS=   8 Ivl=255ms
```

5.3.3. /proc/driver/

Este directorio contiene información para drivers específicos que el kernel está utilizando.

rtc es un archivo habitual aquí que proporciona salida desde el driver para el *Real Time Clock (RTC)* del sistema, dispositivo que guarda el tiempo mientras que el sistema está apagado. Un ejemplo de salida de datos desde /proc/driver/rtc:

```
rtc_time : 01:38:43
rtc_date : 1998-02-13
rtc_epoch : 1900
alarm  : 00:00:00
DST_enable : no
BCD  : yes
24hr  : yes
square_wave : no
alarm_IRQ : no
update_IRQ : no
periodic_IRQ : no
periodic_freq : 1024
batt_status : okay
```

Para obtener más información sobre el RTC, revise /usr/src/linux-2.4/Documentation/rtc.txt.

5.3.4. /proc/fs

Este directorio muestra cómo se exportan los sistemas de archivos. Si está usando un servidor NFS, escriba cat /proc/fs/nfs/exports para visualizar los sistemas de archivos que se comparten y los permisos acordados a esos sistemas de archivos. Para mayor información sobre archivos compartidos con NFS, consulte el Capítulo 9.

5.3.5. /proc/ide/

Este directorio contiene una cantidad muy surtida de información sobre los dispositivos IDE del sistema. Cada canal IDE está representado como un directorio separado, como /proc/ide/ide0 y /proc/ide/ide1. Además, un archivo drivers también está disponible, al proporcionar el número de versión de varios controladores usados en los canales IDE:

```
ide-cdrom  version 4.59
ide-floppy  version 0.97
```

```
ide-disk    version 1.10
```

Muchos chipsets proporcionan un archivo de información en este directorio que aporta datos adicionales referentes a las unidades conectadas a través de los canales. Por ejemplo, un chipset genérico Intel PIIX4 Ultra 33 produce un `/proc/ide/piix` que le informará de si DMA o UDMA está o no habilitado para los dispositivos en los canales IDE:

```
                                Intel PIIX4 Ultra 33 Chipset.
------------- Primary Channel ---------------- Secondary Channel -------------

                 enabled                          enabled
------------- drive0 --------- drive1 -------- drive0 ---------- drive1 ------

DMA enabled:    yes              no              yes               no
UDMA enabled:   yes              no              no                no
UDMA enabled:   2                X               X                 X
UDMA
DMA
PIO
```

Al navegar en un directorio para un canal IDE, como `ide0` para el primer canal, obtendrá más información. El archivo `channel` proporciona el número de canal, mientras que `model` le facilita el tipo de bus para el canal (como `pci`).

5.3.5.1. Directorios de dispositivo

Dentro de cada directorio de canal IDE hay un directorio de dispositivos. El nombre del directorio de dispositivos corresponde a la letra de la unidad en el directorio `/dev/`. Por ejemplo, la primera unidad IDE en `ide0` sería `hda`.

Nota

Existe un enlace simbólico para cada uno de estos directorios de dispositivos en el directorio `/proc/ide/`.

Cada dispositivo, como un disco duro o un CD-ROM, tendrá en ese canal su propio directorio en el que están incluidas su propia recopilación de información y estadísticas. Los contenidos de esos directorios varían de acuerdo con el tipo de dispositivo conectado. Algunos de los archivos más útiles habituales en diferentes dispositivos incluyen:

- `cache` — La caché del dispositivo.
- `capacity` — La capacidad del dispositivo, en bloques de 512 bytes.
- `driver` — El controlador y la versión usados para controlar el dispositivo.
- `geometry` — La geometría física y lógica del dispositivo.
- `media` — El tipo de dispositivo, como por ejemplo un `disk`.
- `model` — El nombre del modelo del dispositivo.
- `settings` — Recopilación de parámetros actuales del dispositivo. Este archivo habitualmente contiene bastante información técnica útil. Un ejemplo de archivo `settings` para un disco duro IDE estándar:

```
name                value           min          max          mode
----                -----           ---          ---          ----
bios_cyl            784             0            65535        rw
bios_head           255             0            255          rw
bios_sect           63              0            63           rw
breada_readahead    4               0            127          rw
bswap               0               0            1            r
current_speed       66              0            69           rw
file_readahead      0               0            2097151      rw
ide_scsi            0               0            1            rw
init_speed          66              0            69           rw
io_32bit            0               0            3            rw
keepsettings        0               0            1            rw
lun                 0               0            7            rw
max_kb_per_request  64              1            127          rw
multcount           8               0            8            rw
nice1               1               0            1            rw
nowerr              0               0            1            rw
number              0               0            3            rw
pio_mode            write-only      0            255          w
slow                0               0            1            rw
unmaskirq           0               0            1            rw
using_dma           1               0            1            rw
```

5.3.6. /proc/irq/

Este directorio se usa para configurar la afinidad de una IRQ con una CPU, lo que le permite conectar una IRQ particular a una sola CPU. De manera alternativa, puede evitar que una CPU manipule cualquier IRQ.

Cada IRQ tiene su propio directorio, permitiendo que cada IRQ sea configurada individualmente. El archivo `/proc/irq/prof_cpu_mask` es una máscara de bits que contiene los valores predeterminados para el archivo `smp_affinity` en el directorio IRQ. Los valores en `smp_affinity` especifican qué CPUs manipulan esa IRQ en particular.

Para más información sobre el directorio `/proc/irq/`, consulte:

```
/usr/src/linux-2.4/Documentation/filesystems/proc.txt
```

5.3.7. /proc/net/

El directorio proporciona una visión exhaustiva de diversos parámetros y estadísticas de red. Cada uno de los archivos cubre una cantidad específica de información relacionada con la red en el sistema:

- `arp` — Contiene la tabla del kernel ARP. Este archivo es particularmente útil para conectar la dirección del hardware a una dirección IP de un sistema.
- `atm` — Directorio que contiene archivos con diversas configuraciones y estadísticas *Asynchronous Transfer Mode (ATM)* y las tarjetas ASDL.
- `dev` — Lista los diferentes dispositivos de red configurados en el sistema, complementado con estadísticas de transmisión y recepción. Este archivo le indica el número de paquetes que cada interfaz ha enviado y recibido, el número de paquetes entrantes y salientes, número de errores vistos, el número de paquetes entregados y mucho más.

- dev_mcast — Visualiza los diversos grupos Layer2 con destinatario múltiple a los que cada dispositivo escucha.
- igmp — Lista las direcciones IP con destinatarios múltiples (multicast) a las que el sistema se ha incorporado.
- ip_fwchains — Revela cualquier cadena del firewall actual.
- ip_fwnames — Lista todos los nombres de cadenas del firewall.
- ip_masquerade — Proporciona una tabla de información de masquerading.
- ip_mr_cache — Lista de la caché de routing de múltiple destinatario.
- ip_mr_vif — Lista las interfaces virtuales de múltiple destinatario (multicast).
- netstat — Contiene una amplia colección de estadísticas de red, incluyendo la temporización TCP, los cookies enviados y recibidos y mucho más.
- psched — Lista de parámetros de planificación global del paquete.
- raw — Lista las estadísticas de dispositivo brutos (raw).
- route — Visualiza la tabla de ruta del kernel.
- rt_cache — Contiene la caché de ruta actual.
- snmp — Lista de los datos del protocolo Simple Network Management Protocol (SNMP) para varios protocolos de red en uso.
- sockstat — Proporciona estadísticas de socket.
- tcp — Contiene información detallada del socket TCP.
- tr_rif — Lista la tabla de enrutamiento de token ring RIF.
- udp — Contiene información detallada del socket UDP.
- unix — Lista sockets de dominio UNIX.
- wireless — Lista datos de la interfaz de radio.

5.3.8. /proc/scsi/

Este directorio es análogo al directorio /proc/ide/, sin embargo, es sólo para dispositivos SCSI conectados.

El archivo primario aquí es /proc/scsi/scsi, que contiene una lista de cada dispositivo SCSI reconocido. A partir de esta lista se puede obtener el tipo de dispositivo, así como también el nombre del modelo, fabricante, canal SCSI y el ID.

Por ejemplo, si un sistema contiene un CD-ROM SCSI, unidad de cinta, un disco duro y un controlador RAID, este archivo se parecerá a:

```
Attached devices:
Host: scsi1 Channel: 00 Id: 05 Lun: 00
  Vendor: NEC      Model: CD-ROM DRIVE:466 Rev: 1.06
  Type:   CD-ROM                           ANSI SCSI revision: 02
Host: scsi1 Channel: 00 Id: 06 Lun: 00
  Vendor: ARCHIVE  Model: Python 04106-XXX Rev: 7350
  Type:   Sequential-Access                ANSI SCSI revision: 02
Host: scsi2 Channel: 00 Id: 06 Lun: 00
  Vendor: DELL     Model: 1x6 U2W SCSI BP  Rev: 5.35
  Type:   Processor                        ANSI SCSI revision: 02
Host: scsi2 Channel: 02 Id: 00 Lun: 00
  Vendor: MegaRAID Model: LD0 RAID5 34556R Rev: 1.01
  Type:   Direct-Access                    ANSI SCSI revision: 02
```

Además, cada driver SCSI usado por el sistema tiene su propio directorio en `/proc/scsi/`, que contiene archivos específicos para cada controlador SCSI que utiliza ese controlador. En el ejemplo anterior de sistema, los directorios `aic7xxx` y `megaraid` están presentes, como aquellos dos drivers que se están utilizando. Los archivos en cada uno de los directorios contienen típicamente un rango de E/S, información de IRQ, y estadísticas para el controlador SCSI particular usando ese controlador. Cada controlador puede remitir un tipo y cantidad de información diferente. El archivo del adaptador Adaptec AIC-7880 Ultra SCSI en este ejemplo produce una salida como la siguiente:

```
Adaptec AIC7xxx driver version: 5.1.20/3.2.4
Compile Options:
  TCQ Enabled By Default : Disabled
  AIC7XXX_PROC_STATS     : Enabled
  AIC7XXX_RESET_DELAY    : 5

Adapter Configuration:
           SCSI Adapter: Adaptec AIC-7880 Ultra SCSI host adapter
                         Ultra Narrow Controller
    PCI MMAPed I/O Base: 0xfcffe000
 Adapter SEEPROM Config: SEEPROM found and used.
      Adaptec SCSI BIOS: Enabled
                    IRQ: 30
                   SCBs: Active 0, Max Active 1,
                         Allocated 15, HW 16, Page 255
             Interrupts: 33726
      BIOS Control Word: 0x18a6
   Adapter Control Word: 0x1c5f
   Extended Translation: Enabled
Disconnect Enable Flags: 0x00ff
     Ultra Enable Flags: 0x0020
 Tag Queue Enable Flags: 0x0000
Ordered Queue Tag Flags: 0x0000
Default Tag Queue Depth: 8
    Tagged Queue By Device array for aic7xxx host instance 1:
      {255,255,255,255,255,255,255,255,255,255,255,255,255,255,255,255}
    Actual queue depth per device for aic7xxx host instance 1:
      {1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1}

Statistics:


(scsi1:0:5:0)
  Device using Narrow/Sync transfers at 20.0 MByte/sec, offset 15
  Transinfo settings: current(12/15/0/0), goal(12/15/0/0), user(12/15/0/0)
  Total transfers 0 (0 reads and 0 writes)
             < 2K      2K+     4K+     8K+    16K+    32K+    64K+   128K+
   Reads:       0        0       0       0       0       0       0       0
  Writes:       0        0       0       0       0       0       0       0


(scsi1:0:6:0)
  Device using Narrow/Sync transfers at 10.0 MByte/sec, offset 15
  Transinfo settings: current(25/15/0/0), goal(12/15/0/0), user(12/15/0/0)
  Total transfers 132 (0 reads and 132 writes)
             < 2K      2K+     4K+     8K+    16K+    32K+    64K+   128K+
   Reads:       0        0       0       0       0       0       0       0
  Writes:       0        0       0       0       1     131       0       0
```

En esta pantalla, se puede ver la velocidad de transmisión a los diversos dispositivos SCSI conectados al controlador basado en el canal ID, así como estadísticas detalladas referentes a la cantidad y tamaño de los archivos leídos o escritos por ese dispositivo. Por ejemplo, este controlador se está comunicando con el CD-ROM a 20 megabytes por segundo, mientras que la unidad de cinta sólo se está comunicando a 10 megabytes por segundo.

5.3.9. /proc/sys/

El directorio /proc/sys/ es diferente de otros en /proc/ porque no sólo proporciona información sobre el sistema pero también permite al administrador activar y desactivar inmediatamente características del kernel.

Aviso

Tenga mucho cuidado al cambiar la configuración de un sistema en producción usando los diversos archivos en el directorio /proc/sys/. La modificación del valor incorrecto puede dejar el kernel inestable, requiriendo que se reinicie el sistema.

Por esta razón, asegúrese de que las opciones sean válidas para ese archivo antes de intentar cambiar un valor en /proc/sys/.

Una buena forma de determinar si un archivo particular se puede configurar o si tan sólo está diseñado para proporcionar información, es listándolo con la opción -l en el intérprete de comandos de la shell. Si se puede escribir en el archivo, podrá utilizarlo para configurar el kernel de algún modo. Por ejemplo, un listado parcial de /proc/sys/fs sería de la siguiente manera:

```
-r--r--r--    1 root     root            0 May 10 16:14 dentry-state
-rw-r--r--    1 root     root            0 May 10 16:14 dir-notify-enable
-r--r--r--    1 root     root            0 May 10 16:14 dquot-nr
-rw-r--r--    1 root     root            0 May 10 16:14 file-max
-r--r--r--    1 root     root            0 May 10 16:14 file-nr
```

En este listado, los archivos dir-notify-enable y file-max pueden escribirse y, por consiguiente, usarse para la configuración del kernel. Los otros archivos sólo proporcionan retroalimentación sobre las configuraciones actuales.

Para cambiar un valor en el archivo /proc/sys tiene que repetir el valor nuevo en el archivo. Por ejemplo, para habilitar la System Request Key en un kernel en ejecución, escriba el comando:

```
echo1>/proc/sys/kernel/sysrq
```

Esto cambiará el valor para sysrq de 0 (off) a 1 (on).

El objetivo de la System Request Key es el de permitirle darle al kernel entrada inmediata mediante combinaciones simples de teclas. Por ejemplo, se puede usar el System Request Key para apagar de inmediato el sistema o reiniciarlo, sincronizar todos los sistemas de archivos montados o volcar información importante a la consola. Esta característica es más útil cuando se utiliza un kernel de desarrollo o si está experimentando bloqueos del sistema. Sin embargo, es considerado un riesgo de seguridad para una consola desatendida y por lo tanto es desactivado por defecto bajo Red Hat Linux.

Para obtener más información sobre la clave de petición de sistema, remítase a /usr/src/linux-2.4/Documentation/sysrq.txt.

Unos cuantos archivos de configuración /proc/sys contienen más de un valor. Para enviar valores correctamente, coloque un espacio entre cada valor traspasado con el comando echo, como se ha hecho a continuación.

```
echo 4 2 45 > /proc/sys/kernel/acct
```

Nota

Cualquier cambio de configuración que haga mediante el comando `echo` desaparecerá cuando vuelva a iniciarse el sistema. Para hacer que cambios de configuración tengan efecto después que el sistema es reiniciado, consulte Sección 5.4.

El directorio `/proc/sys` contiene directorios diferentes que controlan diferentes aspectos de la ejecución de un kernel.

5.3.9.1. `/proc/sys/dev/`

Este directorio proporciona parámetros para dispositivos particulares en el sistema. La mayoría de sistemas tienen al menos dos directorios `cdrom` y `raid`. Los kernels personalizados pueden tener otros directorios, `parport`, que proporciona la habilidad de compartir un puerto paralelo entre drivers de dispositivo múltiple.

El directorio `cdrom` contiene un archivo llamado `info`, que revela algunos parámetros importantes de CD-ROM:

```
CD-ROM information, Id:  cdrom.c 3.12  2000/10/18

drive name:  hdc
drive speed:  32
drive # of slots: 1
Can close tray:  1
Can open tray:  1
Can lock tray:  1
Can change speed: 1
Can select disk: 0
Can read multisession: 1
Can read MCN:  1
Reports media changed: 1
Can play audio:  1
Can write CD-R:  0
Can write CD-RW: 0
Can read DVD:  0
Can write DVD-R: 0
Can write DVD-RAM: 0
```

Este archivo se puede escanear con la finalidad de descubrir las cualidades de un CD-ROM desconocido. Si tiene a su disposición múltiples CD-ROMs en un sistema, cada dispositivo tendrá su propia columna de información.

Se pueden utilizar diversos archivos en `/proc/sys/dev/cdrom`, como `autoclose` y `checkmedia`, para controlar el CD-ROM del sistema. Use el comando `echo` para activar o desactivar estas características.

Si se compila el soporte RAID en el kernel, tendrá a su disposición un directorio `/proc/sys/dev/raid/` con, al menos dos archivos dentro del mismo: `speed_limit_min` y `speed_limit_max`. Estas configuraciones determinan la aceleración de los dispositivos RAID para tareas intensivas de E/S, tales como la resincronización de discos.

5.3.9.2. /proc/sys/fs/

Este directorio contiene un compendio de opciones y de información referente a varios aspectos del sistema de archivos, incluyendo la información de cuotas, manipulación del archivos, inode y dentry.

El directorio `binfmt_misc` se usa para proporcionar soporte del kernel para varios formatos binarios.

Los archivos importantes en `/proc/sys/fs` incluyen:

- `dentry-state` — Proporciona el estado de la caché del directorio. El archivo se vería de la siguiente manera:
  ```
  57411 52939 45 0 0 0
  ```
  El primer número revela el número total de las entradas de la caché del directorio, mientras que el segundo número visualiza el número de entradas inutilizadas. El tercero, le indica el número de segundos en que un directorio ha sido liberado y puede ser reclamado y el cuarto mide las páginas que han sido requeridas por el sistema en la actualidad. Los últimos dos números no están e uso y tan sólo visualizan ceros.
- `dquot-nr` — Muestra el número máximo de entradas de cuota de disco cacheado.
- `file-max` — Permite cambiar el número máximo de manipulaciones de archivos que asignará el kernel. Si incrementa el valor de este archivo resolverá los errores causados por la falta de manejadores de archivos disponibles.
- `file-nr` — Visualiza el número de manipulación de archivos asignados, manipulación de archivos usados, así como el número máximo de manipulación de archivos, en este orden.
- `overflowgid` y `overflowuid` — Define el ID de grupo establecido y el ID de usuario, respectivamente, para el uso con el sistema de archivos que tan sólo soporta los IDs de grupo y usuario de 16 bits.
- `super-max` — Controla el número máximo de superbloques disponible.
- `super-nr` — Visualiza el número actual de superbloques en uso.

5.3.9.3. /proc/sys/kernel/

Este directorio contiene una variedad de archivos de configuración diferentes que afectan directamente a la operación del kernel. Algunos de los archivos más importantes incluyen:

- `acct` — Controla la suspensión del proceso de contabilización basado en el porcentaje de espacio libre disponible en el sistema de archivos que contienen el log. El archivo aparecerá de la siguiente manera, por defecto:
  ```
  4 2 30
  ```
  El segundo valor fija el porcentaje de umbral de espacio libre cuando se suspende el proceso de registro, mientras que la primera columna indica el porcentaje de espacio libre requerido cuando se reanude el proceso de registro. El tercer valor fija el intervalo en segundos en que el kernel interroga al sistema de archivos para ver si el registro se suspende o continua.
- `cap-bound` — Controla las configuraciones de las *capability bounding*, que proporcionan una lista de capacidades para cualquier proceso en el sistema. Si una capacidad no está listada aquí, ningún proceso, por muy privilegiado que sea éste, puede realizarlo. La idea inicial es hacer que el sistema sea más seguro asegurando que no acontezcan ciertas cosas, por lo menos llegados a un cierto nivel del proceso de arranque.

  Para una lista válida de los valores para este archivo virtual, consulte `/usr/src/linux-2.4/include/linux/capability.h`. Encontrará más información sobre capability bounding en línea en el siguiente URL: http://lwn.net/1999/1202/kernel.php3.

- `ctrl-alt-del` — Controla si [Ctrl]-[Alt]-[Delete] reiniciará el ordenador adecuadaments mediante el uso de `init` (valor `0`) o si forzará un rearranque inmediato sin la sincronización de buffers modificados al disco (valor `1`).
- `domainname` — Configura el nombre de dominio del sistema, tal como `example.com`.
- `hostname` — Configura el nombre del sistema host, por ejemplo `www.example.com`.
- `hotplug` — Configura la utilidad para que ésta sea usada cuando se detecta un cambio en la configuración del sistema. Principalmente se usa con USB y Cardbus PCI. El valor por defecto de `/sbin/hotplug` no debería ser cambiado a menos que esté probando un nuevo programa para cumplir con este papel.
- `modprobe` — Fija la ubicación del programa a usar para cargar los módulos del kernel cuando éstos sean necesarios. El valor por defecto de `/sbin/modprobe` significa que `kmod` lo solicitará para cargar el módulo cuando un thread del kernel solicite `kmod`.
- `msgmax` — Fija el tamaño máximo de cualquier mensaje enviado desde un proceso a otro, que está fijado en `8192` bytes por defecto. Debería tener cuidado al alzar este valor, ya que los mensajes en cola entre procesos están almacenados en una memoria de kernel sin memoria de intercambio (swap). Cualquier incremento en `msgmax` incrementará los requerimientos de RAM de su sistema.
- `msgmnb` — Establece el número máximo de bytes en una única cola de mensajes. Por defecto, `16384`.
- `msgmni` — Establece el número máximo de identificadores de la cola de mensajes. Por defecto, `16`.
- `osrelease` — Lista el número de versión del kernel de Linux. Este archivo tan sólo puede ser alterado al cambiar la fuente del kernel y recompilarla.
- `ostype` — Visualiza el tipo de sistema operativo. Por defecto, este archivo está configurado para `Linux` y este valor tan sólo puede ser cambiado al cambiar la fuente del kernel y recompilarla.
- `overflowgid` y `overflowuid` — Define el ID de grupo establecido y el ID de usuario, respectivamente, para el uso con llamadas del sistema a arquitecturas que tan sólo soportan IDs de grupo y usuario de 16 bits.
- `panic` — Define el número de segundos que el kernel pospone el arranque del sistema cuando se experimenta una emergencia en el kernel. Por defecto, el valor está establecido en `0`, lo que deshabilita el rearranque automático tras una emergencia.
- `printk` — Este archivo controla una variedad de configuraciones relacionadas con la impresión o los mensajes de error de registro. Cada mensaje de error remitido por el kernel tiene un *loglevel* asociado a éste que define la importancia del mensaje. Los valores de loglevel aparecen en el orden siguiente:
  - `0` — Emergencia del Kernel. No se puede utilizar el sistema.
  - `1` — Alerta del kernel. Se debe actuar inmediatamente.
  - `2` — La condición del kernel se considera crítica.
  - `3` — Condición de error general del kernel.
  - `4` — Condición de aviso general del kernel.
  - `5` — Nota del kernel de una condición normal pero significativa.
  - `6` — Mensaje informativo del kernel.
  - `7` — Mensajes de depuración del kernel.

  En el archivo `printk` aparecen cuatro valores:
  ```
  6 4 1 7
  ```
  Cada uno de estos valores define una regla diferente para tratar con los mensajes de error. El primer valor, llamado *console loglevel*, define la prioridad más baja de mensajes que se imprimirán en la

consola. (Observe que, cuanto más baja sea ésta, más alto será el número de loglevel.) El segundo valor establece el loglevel por defecto para mensajes adjuntos a éstos sin un loglevel explícito. El tercer valor establece la configuración del loglevel lo más bajo posible para el loglevel de la consola. El último valor establece el valor por defecto para el loglevel de la consola.

- `rtsig-max` — Configura el número máximo de señales en tiempo real POSIX que el sistema podría haber puesto en cola en cualquier otro momento. El valor por defecto es `1024`.
- `rtsig-nr` — El número actual de señales POIX en tiempo real que el kernel ha puesto en cola.
- `sem` — Configura los valores de semáforo dentro del kernel. Un *semáforo* es un objeto System V IPC que es usado para controlar la utilización de un proceso particular.
- `shmall` — Establece la cantidad total de memoria que se puede utilizar de una sola vez en el sistema, en bytes. Por defecto, este valor es `2097152`.
- `shmmax` — Establece el mayor tamaño de segmento de memoria compartida que permite el kernel, en bytes. Por defecto, este valor es `33554432`. No obstante, el kernel soporta valores con mucho más margen.
- `shmmni` — Establece el número máximo de segmentos de memoria compartida para el sistema completo, en bytes. Por defecto, este valor es `4096`
- `sysrq` — Activa la clave de petición de sistema, si el valor difiere del establecido por defecto `0`. Ver la Sección 5.3.9 para mayor información sobre la System Request Key.
- `threads-max` — Establece el número máximo de threads que puede usar el kernel, con un valor por defecto de `2048`.
- `version` — Visualiza la fecha y la hora en los que el kernel fue compilado por última vez. El primer campo en este archivo, tal como `#3`, está relacionado con el número de veces que se ha construido un kernel desde la base de la fuente.

El directorio `random` almacena un número de valores relacionados a la generación de números aleatorios para el kernel.

5.3.9.4. `/proc/sys/net`

Este directorio contiene diversos subdirectorios que tratan tópicos sobre redes. Las diferentes configuraciones en el momento en que el kernel fue compilado colocan diferentes directorios aquí, tales como `appletalk`, `ethernet`, `ipv4`, `ipx`, y `ipv6`. Dentro de estos directorios, los administradores de sistemas podrán ajustar la configuración de la red en un sistema en funcionamiento.

Debido a la amplia variedad de posibles opciones de red disponibles con Linux, tan sólo se comentarán los directorios `/proc/sys/net/`.

El directorio `/proc/sys/net/core/` contiene una variedad de configuraciones que controlan la interacción entre el kernel y las capas de red. Los archivos más importantes son:

- `message_burst` — Décimas de segundos requeridos para escribir un mensaje nuevo de aviso. Se usa para prevenir ataques Denial of Service (DoS) y la configuración por defecto es `50`.
- `message_cost` — También se utiliza para prevenir ataques de DoS poniendo un coste a cada mensaje de aviso. Cuanto más alto es el valor de este archivo (por defecto `5`), más probable es que el aviso del mensaje sea ignorado.

  La idea de un ataque DoS es bombardear su sistema con peticiones que generen errores y llenen sus particiones con archivos logs o necesiten todos los recursos del sistema para manipular el registro de errores. Las configuraciones en `message_burst` y `message_cost` están diseñadas para ser modificadas basándose el riesgo aceptable del sistema contra la necesidad de un registro exhaustivo.

- `netdev_max_backlog` — Establece el número máximo de paquetes permitido para hacer cola cuando una interfaz en particular recibe paquetes a una velocidad superior a la que el kernel puede procesarlos. El valor por defecto para este archivo es `300`.
- `optmem_max` — Configura el tamaño máximo de buffer secundario por socket.
- `rmem_default` — Establece el tamaño por defecto del buffer de recepción del socket en bytes.
- `rmem_max` — Establece el tamaño máximo del buffer de recepción en bytes.
- `wmem_default` — Establece el tamaño por defecto del buffer de envíos del socket en bytes.
- `wmem_max` — Establece el tamaño máximo de la buffer de envíos del socket en bytes.

El directorio `/proc/sys/net/ipv4/` contiene configuraciones de red adicionales. Muchas de estas configuraciones, usadas en conjunto, son muy útiles para prevenir ataques al sistema o en usar el sistema para que actúe como un enrutador.

Atención

Un cambio erróneo en estos archivos puede afectar la conectividad remota del sistema.

Aquí tiene algunos de los archivos más importantes en el directorio `ipv4`:

- `icmp_destunreach_rate`, `icmp_echoreply_rate`, `icmp_paramprob_rate` y `icmp_timeexeed_rate` — Establece el ratio máximo de paquetes ICMP a enviar, en centésimas de un segundo, para hosts bajo ciertas condiciones. Una configuración de `0` elimina cualquier retraso y no es una buena idea.
- `icmp_echo_ignore_all` and `icmp_echo_ignore_broadcasts` — Permite que el kernel ignore paquetes ICMP ECHO desde cada host o tan sólo aquéllos que se originen desde direcciones broadcast y de destinatario múltiple, respectivamente. `0` permite que el kernel responda, mientras un `1` ignora los paquetes.
- `ip_default_ttl` — Establece el *Time To Live (TTL)* predeterminado, que limita el número de saltos que un paquete puede efectuar antes de alcanzar su destino. Si incrementa este valor, la ejecución del sistema puede disminuir.
- `ip_forward` — Permite interfaces en el sistema para reenviar paquetes a otro. Por defecto, este archivo está fijado en `0`. Si se configura este valor a `1` activa el reenvío de paquetes.
- `ip_local_port_range` — Especifica el rango de puertos a usar por TCP o UDP cuando se necesita un puerto. El primer número es el puerto más bajo que puede utilizar, y el segundo especifica el puerto más alto. Cualquier sistema que se crea que necesitará más puertos que los predeterminados 1024 hasta 4999 debería usar el rango 32768 hasta 61000 en este archivo.
- `tcp_syn_retries` — Proporciona un límite en el número de veces que el sistema retransmitirá un paquete SYN cuando se intenta establecer una conexión.
- `tcp_retries1` — Establece el número de retransmisiones permitidas que intentan responder una conexión de entrada. `3` por defecto.
- `tcp_retries2` — Establece el número de retransmisiones permitidas de paquetes TCP. `15` por defecto.

Para consultar la lista completa de archivos y opciones, vea `/usr/src/linux-2.4/Documentation/networking/ip-sysctl.txt`.

Existe un número de otros directorios dentro del directorio `/proc/sys/net/ipv4/` y cada uno cubre tópicos específicos. El directorio `/proc/sys/net/ipv4/conf/` permite a cada interfaz del sistema ser configurada en diferentes formas, incluyendo el uso de valores por defecto para dispositivos no configurados (en el subdirectorio `/proc/sys/net/ipv4/conf/default/`)

y configuraciones que invalidan todas las configuraciones especiales (en el subdirectorio /proc/sys/net/ipv4/conf/all/).

El directorio /proc/sys/net/ipv4/neigh/ contiene configuraciones para la comunicación con un host que está conectado directamente al sistema (llamado un vecino de red) y también contiene configuraciones diferentes para sistemas que están a más de un salto de distancia.

Routing por encima de IPV4 también tiene su propio directorio, /proc/sys/net/ipv4/route/. A diferencia de conf/ y neigh/, el directorio /proc/sys/net/ipv4/route/ contiene especificaciones que aplican al enrutamiento con cualquier interfaz en el sistema. Muchas de estas configuraciones, tales como max_size, max_delay, y min_delay, están relacionadas con el control del tamaño de la caché de enrutamiento. Para limpiar la caché de enrutamiento, escriba cualquier valor al archivo flush.

Encontrará información adicional sobre estos directorios y los posibles valores de sus archivos de configuración en /usr/src/linux-2.4/Documentation/filesystems/proc.txt.

5.3.9.5. /proc/sys/vm/

Este directorio facilita la configuración del subsistema de memoria virtual (VM) del kernel. El kernel hace un uso extensivo e inteligente de la memoria virtual, conocida comunmente como espacio swap.

Los siguientes archivos se encuentran habitualmente en el directorio /proc/sys/vm:

- bdflush — Establece varios valores relacionados con el demonio del kernel bdflush.
- buffermem — Le permite controlar la cantidad porcentual de la memoria total del sistema a usar para la memoria del buffer. Una salida de datos típica para este archivo:
  ```
  2 10 60
  ```
  Los primeros y últimos valores establecen el porcentaje mínimo y máximo de memoria que ha de usarse como memoria de buffer, respectivamente. El valor del medio establece el porcentaje de memoria del sistema dedicado a la memoria de buffer donde el subsistema de gestión de memoria iniciará el borrado de la caché del buffer más que otros tipos de memoria para compensar la falta en general de memoria libre.
- kswapd — Establece varios valores referentes al demonio de swap-out del kernel, kswapd. Este archivo tiene tres valores:
  ```
  512 32 8
  ```
  El primer valor establece el número máximo de páginas que kswapd intentará liberar en una sola vez. Cuanto mayor sea el número, más enérgico será el kernel al liberar páginas. El segundo valor establece el número mínimo de veces que kswapd intenta dejar libre una página. El tercer valor establece el número de páginas que kswapd intenta escribir en un solo intento. Una correcta sintonización de este valor final puede mejorar la ejecución de un sistema usando mucho espacio swap diciendole al kernel que escriba páginas en grandes cantidades, minimizando el número de búsquedas en disco.
- max_map_count — Configura el número máximo de áreas de mapa de memoria que puede tener un proceso. En la mayoría de los casos, el valor por defecto de 65536 es apropiado.
- overcommit_memory — Contiene un valor, que cuando no es el predeterminado 0, permite al kernel saltarse un chequeo estándar para ver si existe suficiente memoria antes de asignarlo.

  Si el valor de overcommit_memory es 1, entonces se incrementa la sobrecarga potencial del sistema pero también lo es el rendimiento para las tareas de uso intensivo de memoria, tales como aquellas de software científico.

  Para los usuarios que no quieren correr riesgos en los compromisos de memoria, existen dos opciones. Configurando overcommit_memory a 2 falla si una petición de memoria añade más

de la mitad de la memoria física RAM, más swap. Colocandola a `3` falla si una petición de memoria solicita más de lo que la swap sola pueda manejar.

- `pagecache` — Controla la cantidad de memoria usada por la caché de la página. Los valores en `pagecache` son porcentajes y funcionan de manera parecida a `buffermem` para reforzar los mínimos y los máximos de memoria caché de página disponible.
- `page-cluster` — Establece el número de páginas leídas en un solo intento. El valor por defecto de `4` establecido en 16 páginas, es apropiado para la mayoría de los sistemas.
- `pagetable_cache` — Controla el número de tablas de página que están cacheadas en una base de por-procesador. Los primeros y segundos valores están relacionados con el número mínimo y máximo de tablas de página a establecer aparte, respectivamente.

Encontrará información adicional sobre estos archivos en `/usr/src/linux-2.4/Documentation/sysctl/vm.txt`.

5.3.10. `/proc/sysvipc`

Este directorio contiene información sobre los recursos System V IPC. Los archivos de este directorio están relacionados con las llamadas al System V IPC de mensajes (`msg`), semáforos (`sem`), y memoria compartida (`shm`).

5.3.11. `/proc/tty/`

Este directorio contiene información sobre los *dispositivos tty* disponibles y usados actualmente en el sistema. Originalmente conocido como *dispositivos teletipo*, cualquier terminal de datos basado en caracteres se le conoce como dispositivos tty.

En Linux existen tres tipos diferentes de dispositivos tty. Los *Dispositivos serial* son usados con conexiones seriales, tales como un modem o un cable serial. Los *Terminales virtuales* crean las conexiones de consola comunes, tales como las consolas virtuales disponibles al pulsar [Alt]-[<F-key>] en la consola del sistema. Los *Pseudo terminales* crean una comunicación de dos sentidos que usan las aplicaciones de nivel alto, tales como XFree86. El archivo `drivers` es una lista de dispositivos tty actualmente en uso:

```
serial               /dev/cua        5  64-127 serial:callout
serial               /dev/ttyS       4  64-127 serial
pty_slave            /dev/pts      136   0-255 pty:slave
pty_master           /dev/ptm      128   0-255 pty:master
pty_slave            /dev/ttyp       3   0-255 pty:slave
pty_master           /dev/pty        2   0-255 pty:master
/dev/vc/0            /dev/vc/0       4       0 system:vtmaster
/dev/ptmx            /dev/ptmx       5       2 system
/dev/console         /dev/console    5       1 system:console
/dev/tty             /dev/tty        5       0 system:/dev/tty
unknown              /dev/vc/%d      4    1-63 console
```

El archivo `/proc/tty/driver/serial` lista las estadísticas en uso y el estado de cada una de las líneas de serie tty.

Para que se puedan utilizar los dispositivos tty de un modo similar a los dispositivos de red, el kernel de Linux reforzará la *disciplina de línea* Esto permite que el controlador coloque un tipo específico de encabezamiento con cada bloque de datos transmitido por el dispositivo, haciendo posible que el lado remoto de la conexión vea el bloque de datos como uno más en la línea de bloques de datos. SLIP y PPP son disciplinas de línea comunes y se usan a menudo para conectar sistemas en un enlace serial.

En el archivo ldiscs encontrará disciplinas de líneas registradas, con información detallada en el directorio ldisc.

5.4. Uso del comando sysctl

El comando /sbin/sysctl es usado para visualizar, configurar y automatizar configuraciones del kernel en el directorio /proc/sys/.

Para tener una vista rápida de todas las variables configurables en el directorio /proc/sys/, escriba el comando /sbin/sysctl -a como root. Esto creará una lista exhaustiva, a continuación le mostramos un pequeño ejemplo:

```
net.ipv4.route.min_delay = 2
kernel.sysrq = 0
kernel.sem = 250     32000     32     128
```

Esta es la misma información básica que vería si echara un vistazo a cada uno de los archivos individualmente. La única diferencia es la localización del archivo. El archivo /proc/sys/net/ipv4/route/min_delay está representado por net.ipv4.route.min_delay, con las barras oblicuas del directorio sustituídas por puntos y la porción asumida proc.sys.

El comando sysctl se puede usar en vez de echo para asignar valores a los archivos en los que se puede escribir en el directorio /proc/sys/ . Por ejemplo, en vez de usar este comando:

```
echo1>/proc/sys/kernel/sysrq
```

Puede usar el comando sysctl:

```
sysctl-w kernel.sysrq="1"
kernel.sysrq = 1
```

A pesar de que es muy útil durante las pruebas el poder rápidamente efectuar configuraciones de valores simples en /proc/sys/, esto no funciona bien en un ambiente de producción. Todas las configuraciones especiales /proc/sys/ se pierden cuando se vuelve a arrancar el sistema. Para conservar las configuraciones que quiere establecer como permanentes en su kernel, añádalas al archivo /etc/sysctl.conf.

Cada vez que el sistema arranque, init ejecuta el script /etc/rc.d/rc.sysinit. Este script contiene un comando para ejecutar sysctl mediante el uso de /etc/sysctl.conf como los valores a establecer. Por eso, cualquier valor añadido a /etc/sysctl.conf surtirá efecto una vez que el sistema arranque sin la necesidad de reconfigurar y reconstruir el kernel para incorporar los cambios.

5.5. Recursos adicionales

A continuación están las fuentes adicionales de información sobre el sistema de archivos proc.

5.5.1. Documentación instalada

La mayoría de la mejor documentación sobre /proc está disponible en su sistema.

- /usr/src/linux-2.4/Documentation/filesystems/proc.txt — Contiene información variada pero limitada sobre todos los aspectos de /proc.

- /usr/src/linux-2.4/Documentation/sysrq.txt — Vista preliminar de las opciones de System Request Key.
- /usr/src/linux-2.4/Documentation/sysctl/ — Un directorio conteniendo una variedad de sugerencias sysctl, incluyendo valores modificados que hacen referencia al kernel (kernel.txt), accediendo al sistema de archivos (fs.txt), y el uso de memoria virtual (vm.txt).
- /usr/src/linux-2.4/Documentation/networking/ip-sysctl.txt — Descripción de varias opciones IP de red y lo que significan para el kernel.
- /usr/src/linux-2.4/ — Una de las mejores fuentes de información de /proc/ se puede encontrar leyendo el código fuente del kernel. Asegúrese de que el RPM para kernel-source esté instalado en su sistema y revise el directorio /usr/src/linux-2.4/ por el código fuente.

5.5.2. Sitios web útiles

- http://www.linuxhq.com — Este sitio mantiene una base de datos completa de recursos, parches y documentación para varias versiones del kernel de Linux.

Capítulo 6.

Usuarios y grupos

El control de los *usuarios* y *grupos* es un elemento clave en la administración de sistemas de Red Hat Linux.

Los *usuarios* pueden ser gente real, es decir, cuentas ligadas a un usuario físico en particular o cuentas que existen para ser usadas por aplicaciones específicas.

Los *grupos* son siempre expresiones lógicas de organización, reuniendo usuarios para un propósito común. Los usuarios dentro de un mismo grupo pueden leer, escribir o ejecutar archivos que pertenecen al grupo.

Cada usuario y grupo tiene un número de identificación único llamado *userid* (*UID*) y un *groupid* (*GID*) respectivamente.

Cuando se crea un archivo se asigna a un usuario y a un grupo. De la misma manera se asignan los permisos de lectura, escritura y ejecución para el propietario del archivo, para el grupo y para cualquier otro usuario en un host. El usuario y el grupo de un archivo particular, así como los permisos en ese archivo, pueden ser cambiados por un root o, en la mayoría de los casos, por el creador del archivo.

Una de las tareas más importantes de cualquier administrador del sistema, es la de administrar adecuadamente usuarios y grupos, así como asignar y revocar permisos. Para una vista detallada de las estrategias para la administración de usuarios y grupos, consulte el capítulo titulado *Administración de cuentas y grupos* en el *Manual de administración del sistema de Red Hat Linux*.

6.1. Herramientas de administración de usuarios y grupos

La gestión de usuarios y grupos ha sido tradicionalmente tediosa, pero Red Hat Linux posee algunas herramientas y convenciones que facilitan a los administradores su gestión.

La forma más fácil de manejar usuarios y grupos es a través de la aplicación gráfica, **Administrador de usuarios** (`redhat-config-users`). Para más información sobre **Administrador de usuarios**, consulte el capítulo llamado *Configuración de usuarios y grupos* en el *Manual de personalización de Red Hat Linux*.

Las siguientes herramientas de línea de comandos también se pueden utilizar para manejar usuarios y grupos:

- `useradd`, `usermod` y `userdel` — Métodos estándar de la industria para añadir, eliminar y modificar cuentas de usuarios.
- `groupadd`, `groupmod` y `groupdel` — Métodos estándar de la industria para añadir, eliminar y modificar grupos de usuarios.
- `gpasswd` — Métodos estándar de la industria para administrar el archivo `/etc/group`.
- `pwck`, `grpck` — Herramientas para la verificación de contraseñas, grupo y archivos shadow asociados.
- `pwconv`, `pwunconv` — Herramientas para la conversión a contraseñas shadow y de vuelta a contraseñas estándar.

Para una visión general de la administración de usuarios y grupos, consulte *Manual de administración del sistema de Red Hat Linux*. Para una vista más detallada a las herramientas de línea de comandos para la administración de usuarios y grupos, consulte el capítulo llamado *Configuración de usuarios y grupos* en el *Manual de personalización de Red Hat Linux*.

6.2. Usuarios estándar

La Tabla 6-1 lista los usuarios estándar configurados en el archivo `/etc/passwd` para una instalación con "Todo". El groupid (GID) en esta tabla es *grupo primario* para el usuario. Vea Sección 6.3 para una lista de los grupos estándar.

Usuario	UID	GID	Directorio principal	Shell
root	0	0	`/root`	`/bin/bash`
bin	1	1	`/bin`	`/sbin/nologin`
daemon	2	2	`/sbin`	`/sbin/nologin`
adm	3	4	`/var/adm`	`/sbin/nologin`
lp	4	7	`/var/spool/lpd`	`/sbin/nologin`
sync	5	0	`/sbin`	`/bin/sync`
shutdown	6	0	`/sbin`	`/sbin/shutdown`
halt	7	0	`/sbin`	`/sbin/halt`
mail	8	12	`/var/spool/mail`	`/sbin/nologin`
news	9	13	`/var/spool/news`	
uucp	10	14	`/var/spool/uucp`	`/sbin/nologin`
operator	11	0	`/root`	`/sbin/nologin`
games	12	100	`/usr/games`	`/sbin/nologin`
gopher	13	30	`/usr/lib/gopher-data`	`/sbin/nologin`
ftp	14	50	`/var/ftp`	`/sbin/nologin`
nobody	99	99	`/`	`/sbin/nologin`
rpm	37	37	`/var/lib/rpm`	`/bin/bash`
vcsa	69	69	`/dev`	`/sbin/nologin`
ntp	38	38	`/etc/ntp`	`/sbin/nologin`
canna	39	39	`/var/lib/canna`	`/sbin/nologin`
nscd	28	28	`/`	`/bin/false`
rpc	32	32	`/`	`/sbin/nologin`
postfix	89	89	`/var/spool/postfix`	`/bin/true`
named	25	25	`/var/named`	`/bin/false`
amanda	33	6	`var/lib/amanda/`	`/bin/bash`
postgres	26	26	`/var/lib/pgsql`	`/bin/bash`
sshd	74	74	`/var/empty/sshd`	`/sbin/nologin`
rpcuser	29	29	`/var/lib/nfs`	`/sbin/nologin`
nsfnobody	65534	65534	`/var/lib/nfs`	`/sbin/nologin`

Usuario	UID	GID	Directorio principal	Shell
pvm	24	24	/usr/share/pvm3	/bin/bash
apache	48	48	/var/www	/bin/false
xfs	43	43	/etc/X11/fs	/sbin/nologin
desktop	80	80	/var/lib/menu/kde	/sbin/nologin
gdm	42	42	/var/gdm	/sbin/nologin
mysql	27	27	/var/lib/mysql	/bin/bash
webalizer	67	67	/var/www/html/usage	/sbin/nologin
mailman	41	41	/var/mailman	/bin/false
mailnull	47	47	/var/spool/mqueue	/sbin/nologin
smmsp	51	51	/var/spool/mqueue	/sbin/nologin
squid	23	23	/var/spool/squid	/dev/null
ldap	55	55	/var/lib/ldap	/bin/false
netdump	34	34	/var/crash	/bin/bash
pcap	77	77	/var/arpwatch	/sbin/nologin
ident	98	98	/	/sbin/nologin
privoxy	100	101	/etc/privoxy	
radvd	75	75	/	/bin/false
fax	78	78	/var/spool/fax	/sbin/nologin
wnn	49	49	/var/lib/wnn	/bin/bash

Tabla 6-1. Usuarios estándar

6.3. Grupos estándar

La Tabla 6-2 lista los grupos estándar configurados por una instalación con "Todo". Los grupos son almacenados bajo Red Hat Linux en el archivo /etc/group.

Grupo	GID	Miembros
root	0	root
bin	1	root, bin, daemon
daemon	2	root, bin, daemon
sys	3	root, bin, adm
adm	4	root, adm, daemon
tty	5	
disk	6	root
lp	7	daemon, lp
mem	8	

Grupo	GID	Miembros
kmem	9	
wheel	10	root
mail	12	mail
news	13	news
uucp	14	uucp
man	15	
games	20	
gopher	30	
dip	40	
ftp	50	
lock	54	
nobody	99	
users	100	
rpm	37	rpm
utmp	22	
floppy	19	
vcsa	69	
ntp	38	
canna	39	
nscd	28	
rpc	32	
postdrop	90	
postfix	89	
named	25	
postgres	26	
sshd	74	
rpcuser	29	
nfsnobody	65534	
pvm	24	
apache	48	
xfs	43	
desktop	80	
gdm	42	
mysql	27	
webalizer	67	

Grupo	GID	Miembros
mailman	41	
mailnull	47	
smmsp	51	
squid	23	
ldap	55	
netdump	34	
pcap	77	
ident	98	
privoxy	101	
radvd	75	
fax	78	
slocate	21	
wnn	49	

Tabla 6-2. Grupos estándar

6.4. Grupos de usuario privado

Red Hat Linux utiliza un esquema de *grupo de usuario privado* (*UPG*), lo que hace más fácil de manejar los grupos de UNIX.

Se crea un UPG siempre que se añade un nuevo usuario al sistema. Un UPG tiene el mismo nombre que el usuario para el cual se crea y ese usuario es el único miembro de ese UPG.

Los UPGs hacen que sea más seguro configurar los privilegios por defecto para un nuevo archivo o directorio lo que permite a ambos, tanto el usuario como al *grupo de ese usuario* hacer modificaciones al archivo o directorio.

El parámetro que determina qué permisos son aplicados a un nuevo archivo o directorio es llamado un *umask* y es configurado en el archivo `/etc/bashrc`. Tradicionalmente, en sistemas UNIX el `umask` es configurado a `022`, lo que sólo permite al usuario que creó el archivo o directorio realizar modificaciones. Bajo este esquema, todos los demás usuarios *incluyendo miembros del grupo del creador* no tienen derecho a realizar ninguna modificación. Sin embargo, bajo el esquema UPG, esta "protección de grupo" no es necesaria puesto que cada usuario tiene su propio grupo privado.

6.4.1. Directorios de grupos

Muchas organizaciones de IT (del inglés Information Technologies) prefieren crear un grupo para cada proyecto importante y luego asignar personas al grupo si estos necesitan acceso a los archivos de ese proyecto. Usando este esquema tradicional, el manejo de archivos ha sido difícil pues cuando alguien crea un archivo, este es asociado con el grupo primario al cual pertenece. Cuando una persona individual trabaja en múltiples proyectos, se hace difícil asociar los archivos correctos con el grupo correcto. Usando el esquema UPG, sin embargo, los grupos son automáticamente asignados a archivos creados dentro de un directorio con el bit setgid configurado, lo que hace muy simple el manejo de proyectos de grupos que comparten un directorio común.

Digamos, por ejemplo, que un grupo de personas trabajan con archivos en el directorio `/usr/lib/emacs/site-lisp/`. Algunas personas son de confianza como para modificar el

directorio, pero ciertamente no todos. Entonces primero cree un grupo `emacs`, como se muestra en el siguiente comando:

```
/usr/sbin/groupadd emacs
```

Para poder asociar los contenidos del directorio con el grupo `emacs`, escriba:

```
chown -R root.emacs /usr/lib/emacs/site-lisp
```

Ahora es posible añadir los usuarios adecuados al grupo con el comando `gpasswd`:

```
/usr/bin/gpasswd -a <username> emacs
```

Permita a los usuarios crear archivos dentro del directorio con el comando siguiente:

```
chmod 775 /usr/lib/emacs/site-lisp
```

Cuando un usuario crea un nuevo archivo, se le asigna el grupo del grupo por defecto privado del usuario. Luego, configure el bit setgid, el cual asigna que todo lo que se cree en el directorio la misma permisología de grupo del directorio mismo (`emacs`). Use el comando siguiente:

```
chmod 2775 /usr/lib/emacs/site-lisp
```

En este punto, puesto que cada usuario tiene por defecto su umask en 002, todos los miembros del grupo `emacs` pueden crear y modificar archivos en el directorio `/usr/lib/emacs/site-lisp/` sin que el administrador tenga que cambiar los permisos de los archivos cada vez que un usuario escriba nuevos archivos.

6.5. Contraseñas Shadow

En entornos multiusuario es muy importante utilizar *contraseñas shadow* (proporcionadas por el paquete `shadow-utils`). Haciendo esto se mejora la seguridad de los archivos de autenticación del sistema. Por esta razón, el programa de instalación de Red Hat Linux activa por defecto las contraseñas shadow.

Lo siguiente es una lista de las ventajas de las contraseñas shadow sobre el método antiguo de almacenar contraseñas en los sistemas basados en UNIX.

- Mejora la seguridad del sistema al mover las contraseñas encriptadas desde el archivo `/etc/passwd` que puede leer todo el mundo, a `/etc/shadow`, el cual sólo puede ser leído por el usuario root.
- Almacena información sobre la vigencias de las contraseñas.
- Permite el uso del archivo `/etc/login.defs` para reforzar las políticas de seguridad.

La mayoría de las utilidades proporcionadas por el paquete `shadow-utils` funcionan adecuadamente sin importar si las contraseñas shadow están activadas o no. Sin embargo, puesto que la información sobre la vigencia de las contraseñas es almacenada exclusivamente en el archivo `/etc/shadow`, cualquier comando que cree o modifique la información sobre la vigencia de las contraseñas, no funcionará.

Abajo se muestra una lista de los comandos que no funcionan a menos que se activen las contraseñas shadow:

- `chage`

- `gpasswd`
- Las opciones `/usr/sbin/usermod -e` o `-f`
- Las opciones `/usr/sbin/useradd -e` o `-f`

Capítulo 7.

El sistema X Window

Mientras que el corazón de Red Hat Linux es el kernel, para muchos usuarios, la cara del sistema operativo es el entorno gráfico proporcionando por el *Sistema X Window*, también llamado simplemente *X*.

En el mundo UNIX™, los entornos de ventanas han existido desde hace décadas, siendo éstos precursores de muchos de los utilizados en los sistemas operativos actuales. A través de los años el sistema X Window se ha convertido en el entorno gráfico (GUI) predominante para sistemas operativos del tipo UNIX.

El entorno gráfico para Red Hat Linux es suministrado por XFree86™, una implementación open source de X. XFree86 es un proyecto de larga escala que se apoya en un gran número de desarrolladores en todo el mundo. Presenta una amplia gama de soporte para diferentes dispositivos y arquitecturas gráficas, así como la posibilidad de ejecutarse en diferentes sistemas operativos y plataformas.

El sistema X Window utiliza una arquitectura cliente-servidor. El *servidor de X* escucha por conexiones desde las aplicaciones *cliente X* a través de la red o una interfaz local de loopback. El proceso servidor gestiona la comunicación con el hardware, como puede ser con una tarjeta gráfica, un monitor, un teclado o un ratón. Las aplicaciones cliente de X existen en el espacio del usuario, creando una *interfaz gráfica del usuario* (*GUI*) y pasando peticiones al servidor de X.

7.1. XFree86

Red Hat Linux 9 utiliza la versión 4.x de XFree86 como la base del sistema X Window, la cual incluye muchas mejoras de tecnología de punta para XFree86 tales como soporte para la aceleración de hardware 3D, la extensión XRender para fuentes anti-alias, un diseño basado en controladores modular y soporte para hardware de vídeo y dispositivos de entrada.

Importante

Red Hat Linux ya no proporciona los paquetes del servidor de XFree86 versión 3. Antes de actualizar a la última versión de Red Hat Linux, asegúrese de que la tarjeta de vídeo es compatible con la versión 4 de XFree86 verificando la Lista de compatibilidad del Hardware de Red Hat localizada en línea en http://hardware.redhat.com.

Los archivos relacionados a XFree86 residen principalmente en dos ubicaciones:

/usr/X11R6/

Contiene un servidor X y algunas aplicaciones cliente así como también archivos de cabecera X, librerías, módulos y documentación.

`/etc/X11/`

Contiene archivos de configuración para aplicaciones cliente y servidor de X. Esto incluye archivos de configuración para el servidor X mismo, el viejo servidor de fuentes `xfs`, los manejadores de display de X y muchos otros componentes base.

Es importante resaltar que el archivo de configuración para la arquitectura de fuentes basado en Fontconfig es `/etc/fonts/fonts.conf` (que deja obsoleto al archivo

`/etc/X11/XftConfig`). Para más información sobre la configuración y añadir fuentes, vea Sección 7.4.

Debido a que el servidor XFree86 realiza tareas avanzadas en una amplia variedad de arreglos de hardware, requiere una configuración detallada. El programa de instalación de Red Hat Linux instala y configura XFree86 automáticamente, a menos que los paquetes XFree86 no se seleccionen para la instalación. Sin embargo, si la tarjeta de vídeo o el monitor cambia, XFree86 necesitará ser reconfigurado. La mejor forma de hacer esto es usando la **Herramienta de configuración de X** (`redhat-config-xfree86`).

Para comenzar la **Herramienta de configuración de X** mientras se esté en una sesión activa de X, vaya al **Botón de menú principal** (en el Panel) => **Configuración del sistema** => **Visualización**. Después de usar la **Herramienta de configuración de X** durante una sesión de X, los cambios tendrán efecto después que se desconecte y se vuelva a conectar. Para más sobre el uso de la **Herramienta de configuración de X** refiérase al capítulo llamado *Audio, Vídeo y entretenimiento general* en el *Manual del principiante de Red Hat Linux*.

En algunas situaciones, la reconfiguración del servidor XFree86 puede requerir la edición manual de su archivo de configuración, `/etc/X11/XF86Config`. Para más información sobre la estructura de este archivo, consulte Sección 7.3.

7.2. Entornos de escritorio y gestores de ventanas

Una vez que un servidor XFree86 se esté ejecutando, las aplicaciones de cliente X pueden conectarlo y crear una GUI para el usuario. Un rango de GUIs están disponibles con Red Hat Linux, desde el rudimentario *Administrador de pestañas de ventanas* hasta un entorno de escritorio altamente desarrollado, interactivo como *GNOME*, con el que la mayoría de los usuarios de Red Hat Linux están familiarizados.

Para crear lo último, se deben conectar las dos clases principales de GUI más avanzadas de aplicaciones cliente X al servidor XFree86: un *entorno de escritorio* y un *gestor de ventanas*.

7.2.1. Entornos de escritorio

Un entorno de escritorio une diferentes clientes de X, los cuales cuando se usan juntos crean un ambiente de usuario gráfico común y una plataforma de desarrollo.

Los entornos de escritorio tienen características avanzadas las cuales permiten a los clientes X y a otros procesos comunicarse unos con otros y permitir a todas las aplicaciones escritas para funcionar en ese ambiente a que realicen tareas avanzadas, tales como operaciones de arrastrar y soltar.

Red Hat Linux proporciona dos entornos de escritorio:

- *GNOME* — Es el entorno de escritorio por defecto en Red Hat Linux basado en el conjunto de herramientas gráficas GTK+ 2.
- *KDE* — Un entorno de escritorio alternativo basado en el conjunto de herramientas gráficas Qt 3.

Ambos entornos GNOME y KDE tienen aplicaciones de productividad avanzadas, tales como procesadores de palabras, hojas de cálculo y navegadores Web así como herramientas para personalizar la apariencia de la GUI. Adicionalmente, si ambas librerías estan presentes, la GTK+ 2 y la Qt, las aplicaciones KDE pueden ejecutarse en GNOME y viceversa.

Para información sobre la personalización de los entornos de escritorio GNOME y KDE, refiérase al *Manual del principiante de Red Hat Linux*.

7.2.2. Gestores de ventanas

Los *gestores de ventanas* son programas clientes de X que son o parte del entorno de escritorio o, en otros casos, standalone. Su propósito principal es controlar la forma en que las ventanas gráficas son posicionadas, redimensionadas o movidas. Los manejadores de ventanas controlan las barras de títulos, el comportamiento del foco, los vínculos del botón del ratón y teclas especificadas por el usuario.

Se incluyen cinco gestores de ventanas con Red Hat Linux:

- `kwin` — El gestor de ventanas *KWin* es el manejador por defecto para el entorno KDE. Es un manejador de ventanas que soporta temas personalizados.
- `metacity` — El gestor de ventanas *Metacity* es el manejador por defecto del entorno GNOME. Es un manejador de ventanas simple y eficiente que también soporta temas personalizados.
- `mwm` — El gestor de ventanas *Motif*, es un gestor básico tipo standalone. Puesto que está diseñado para ser un gestor standalone, no se debería utilizar en conjunto con los entornos de escritorios GNOME o KDE.
- `sawfish` — *Sawfish* es un gestor de ventanas con características completas el cual era el manejador por defecto del entorno GNOME hasta la versión Red Hat Linux 8.0. Se puede usar bien sea standalone o con un entorno de escritorio.
- `twm` — El minimalista *Administrador de pestañas de ventanas*, el cual proporciona el conjunto de herramientas más básicas de cualquier gestor de ventanas y puede ser usado bien sea standalone o con un entorno de escritorio. Es instalado como parte de XFree86.

Estos gestores de ventanas pueden ejecutarse sin los entornos de escritorio para poder obtener una impresión de sus diferencias. Teclee el comando `xinit -e <path-to-window-manager>`, donde `<path-to-window-manager>` es la ubicación del archivo binario de gestor de ventanas. El archivo binario puede ser encontrado escribiendo `which <window-manager-name>`.

7.3. Archivos de configuración del servidor XFree86

El servidor XFree86 es un binario ejecutable (`/usr/X11R6/bin/XFree86`) que carga dinámicamente cualquier módulo de servidor X necesario en el momento de ejecución desde el directorio `/usr/X11R6/lib/modules/`. Algunos de estos módulos son cargados automáticamente por el servidor, mientras que otros son opcionales y deben ser especificados en el archivo de configuración de XFree86.

El servidor XFree86 y los archivos de configuración asociados son almacenados en el directorio `/etc/X11/`. El archivo de configuración para el servidor XFree86 es `/etc/X11/XF86Config`. Cuando se instala Red Hat Linux, los archivos de configuración para XFree86 son creados usando información necesaria reunida sobre el hardware del sistema durante el proceso de instalación.

7.3.1. `XF86Config`

Mientras que casi nunca se necesita editar manualmente el `/etc/X11/XF86Config`, es muy útil conocer sobre las varias secciones y los parámetros opcionales disponibles, especialmente cuando se esten solucionando problemas.

7.3.1.1. La estructura de XFree86

El archivo `/etc/X11/XF86Config` esta formado de muchas secciones diferentes las cuales hacen referencia a aspectos específicos del hardware del sistema.

Cada sección comienza con una línea `Section "<section-name>"` (donde <section-name> es el título para la sección) y termina con una línea `EndSection`. Dentro de cada sección, hay líneas conteniendo nombres de opciones y al menos un valor de opción, ocasionalmente visto en comillas.

Las líneas que comienzan con un símbolo de numeral o almohadilla [#] no son leídas por el servidor XFree86 y son usadas como comentarios.

Algunas opciones dentro del archivo `/etc/X11/XF86Config` aceptan un switch boleano el cual activa o desactiva la característica. Los valores boleanos son:

- `1`, `on`, `true`, o `yes` — Activa la opción.
- `0`, `off`, `false`, o `no` — Desactiva la opción.

Lo siguiente son algunas de las secciones más importantes ordenadas como aparecen en un archivo `/etc/X11/XF86Config` típico. Más información detallada sobre el archivo de configuración del servidor XFree86 se puede encontrar en la página man de `XF86Config`.

7.3.1.2. `ServerFlags`

La sección opcional `ServerFlags` contiene varios parámetros globales del servidor XFree86. Cualquier parámetro en esta sección puede ser sobreescrito por opciones colocadas en la sección `ServerLayout` (refiérase a Sección 7.3.1.3 para más detalles).

Las entradas dentro de la sección `ServerFlags` estan en sus propias líneas y comienzan con el término `Option` seguido por una opción encerrada en dobles comillas ["].

A continuación un ejemplo de la sección `ServerFlags`:

```
Section "ServerFlags"
      Option "DontZap" "true"
EndSection
```

La siguiente es una lista de algunas de las opciones más útiles:

- `"DontZap" "<boolean>"` — Cuando el valor de <boolean> está configurado a verdadero, esta configuración previene el uso de la combinación de teclas [Ctrl]-[Alt]-[Backspace] para terminar inmediatamente el servidor XFree86.
- `"DontZoom" "<boolean>"` — Cuando el valor de <boolean> está colocado a verdadero, esta configuración previene moverse a lo largo de las resoluciones de vídeo configuradas usando las combinaciones de teclas [Ctrl]-[Alt]-[Keypad-Plus] y [Ctrl]-[Alt]-[Keypad-Minus].

7.3.1.3. `ServerLayout`

La sección `ServerLayout` vincula los dispositivos de entrada y salida controlados por el servidor XFree86. Como mínimo, esta sección debe especificar un dispositivo de salida y al menos dos dispositivos de entrada (un teclado y un ratón).

El ejemplo siguiente ilustra una sección `ServerLayout` típica:

```
Section  "ServerLayout"
        Identifier     "Default Layout"
        Screen      0  "Screen0" 0 0
        InputDevice    "Mouse0" "CorePointer"
        InputDevice    "Keyboard0" "CoreKeyboard"
EndSection
```

Las entradas siguientes son usadas a menudo en la sección `ServerLayout`:

- `Identifier` — Especifica un nombre único para esta sección `ServerLayout`.
- `Screen` — Especifica el nombre de la sección `Screen` a ser usado con el servidor XFree86. Pueden estar presentes más de una opción `Screen`.

  Lo siguiente es un ejemplo de una entrada `Screen` típica:

  ```
  Screen      0  "Screen0" 0 0
  ```

  El primer número en esta entrada de ejemplo `Screen` (`0`) indica que el primer conector del monitor o *head* en la tarjeta de vídeo usa la configuración especificada en la sección `Screen` con el identificador `"Screen0"`.

  Si la tarjeta de vídeo tiene más de una cabeza, será necesaria otra entrada `Screen` con un número diferente y un identificador de sección `Screen`.

  Los números a la derecha de `"Screen0"` proporcionan las coordenadas absolutas X y Y para la esquina superior izquierda de la pantalla (`0 0` por defecto).
- `InputDevice` — Especifica el nombre de una sección `InputDevice` a ser usada con el servidor XFree86.

  Al menos deben haber dos entradas `InputDevice`: una para el ratón por defecto y una para el teclado por defecto. Las opciones `CorePointer` y `CoreKeyboard` indican que estos son el ratón y teclado principales.
- `Option "<option-name>"` — Una entrada opcional que especifica parámetros extra para esta sección. Cualquier sección listada aquí sobreescriben aquellas listadas en la sección `ServerFlags`.

  Reemplace *<option-name>* con una opción válida listada para esta sección en la página man XF86Config.

Es posible crear más de una sección `ServerLayout`. Sin embargo, el servidor sólo leerá la primera sección que aparezca a menos que se especifique una sección `ServerLayout` alterna como una línea de argumento.

7.3.1.4. `Files`

La sección `Files` configura la ruta para servicios vitales al servidor XFree86, tal como la ruta de la fuente.

El siguiente ejemplo ilustra una sección `Files`:

```
Section "Files"
        RgbPath      "/usr/X11R6/lib/X11/rgb"
        FontPath     "unix/:7100"
EndSection
```

Las siguientes entradas son usadas comúnmente en la sección `Files`:

- `RgbPath` — Especifica la ubicación de la base de datos de colores RGB. Esta base de datos define todos los esquemas de color en XFree86 y los junta para valores RGB especificos.
- `FontPath` — Especifica dónde el servidor XFree86 debe ser conectado para obtener las fuentes desde el servidor de fuentes `xfs`.

  Por defecto, la `FontPath` es `unix/:7100`. Esto le dice al servidor XFree86 para obtener información de fuentes usando sockets de dominio UNIX para la comunicación entre procesos (IPC) en el puerto 7100.

  Vea Sección 7.4 para más información sobre XFree86 y fuentes.
- `ModulePath` — Un parámetro opcional el cual especifica directorios alternativos el cual almacena módulos de servidor XFree86.

7.3.1.5. `Module`

La sección `Module` especifica cuales módulos del directorio `/usr/X11R6/lib/modules/` cargará el servidor XFree86. Los módulos añaden funcionalidad adicional al servidor XFree86.

El ejemplo siguiente ilustra una sección `Module` típica:

```
Section "Module"
 Load  "dbe"
 Load  "extmod"
 Load  "fbdevhw"
 Load  "glx"
 Load  "record"
 Load  "freetype"
 Load  "type1"
 Load  "dri"
EndSection
```

7.3.1.6. `InputDevice`

Cada sección `InputDevice` configura un dispositivo de entrada para el servidor XFree86. Los sistemas típicamente tienen al menos dos secciones `InputDevice`, un teclado y un ratón.

El ejemplo siguiente ilustra una sección `InputDevice` típica para un ratón:

```
Section "InputDevice"
 Identifier  "Mouse0"
 Driver      "mouse"
 Option     "Protocol" "IMPS/2"
 Option     "Device" "/dev/input/mice"
 Option     "Emulate3Buttons" "no"
EndSection
```

Las entradas siguientes son comúnmente usadas en la sección `InputDevice`:

- `Identifier` — Especifica un nombre único para esta sección `InputDevice`. Esto es una entrada requerida.
- `Driver` — Especifica el nombre del controlador del dispositivo que XFree86 debe cargar para el dispositivo.
- `Option` — Especifica las opciones necesarias pertinentes al dispositivo.

  Para un ratón, estas opciones incluyen:

  - `Protocol` — Indica el protocolo usado por el ratón, tal como `IMPS/2`.
  - `Device` — Indica la ubicación del dispositivo físico.
  - `Emulate3Buttons` — Especifica si se va a permitir a un ratón de dos botones a que se comporte como uno de tres cuando se presionen ambos botones simultáneamente.

  Consulte la página man de XF86Config para una lista de las opciones válidas para esta sección.

Por defecto la sección `InputDevice` tiene comentarios para permitir a los usuarios configurar opciones adicionales.

7.3.1.7. sección `Monitor`

Cada sección `Monitor` configura un tipo de monitor usado por el sistema. Mientras una sección `Monitor` es lo mínimo, pueden ocurrir varias instancias para cada tipo de monitor en uso con la máquina.

La mejor forma de configurar un monitor es configurando X durante la instalación o usando la **Herramienta de configuración de X**. Para más información sobre el uso de la **Herramienta de configuración de X** refiérase al capítulo llamado *Audio, Vídeo y entretenimiento general* en el *Manual del principiante de Red Hat Linux*.

Este ejemplo muestra una sección de `Monitor` típica:

```
Section "Monitor"
 Identifier   "Monitor0"
 VendorName   "Monitor Vendor"
 ModelName    "DDC Probed Monitor - ViewSonic G773-2"
 DisplaySize  320 240
 HorizSync    30.0 - 70.0
 VertRefresh  50.0 - 180.0
EndSection
```

Aviso

Tenga cuidado cuando modifique manualmente valores en la sección `Monitor` de `/etc/X11/XF86Config`. Valores inapropiados pueden dañar o destruir su monitor. Consulte la documentación sobre monitores para un listado de parámetros seguros.

A continuación se muestran entradas comunes usadas en la sección `Monitor`:

- `Identifier` — Proporciona un nombre único para esta sección `Monitor`. Esta es una entrada requerida.
- `VendorName` — Parámetro opcional que muestra el nombre del fabricante del monitor.
- `ModelName` — Parámetro opcional que muestra el nombre del modelo del monitor.
- `DisplaySize` — Un parámetro opcional que especifica, en milímetros, el tamaño físico del área de dibujo del monitor.
- `HorizSync` — Especifica el rango de la frecuencia de sincronización horizontal compatible con el monitor en kHz. Estos valores ayudan al servidor XFree86 a determinar la validez de las entradas `Modeline` especificadas para el monitor.
- `VertRefresh` — Lista de los rangos de frecuencias de refresco verticales soportados por el monitor, en Hz. Estos valores se usan como referencia para que el servidor XFree86 sepa cuando deberá utilizar cada una de las entradas que aparecen en `Modeline` con este monitor.
- `Modeline` — Un parámetro opcional el cual especifica los modos de vídeo adicionales para el monitor en resoluciones particulares, con ciertas resoluciones de refrescamiento de sincronización horizontal y vertical. Vea la página man de `XF86Config` para una explicación más detallada de las entradas `Modeline`.
- `Option "<option-name>"` — Una entrada opcional la cual especifica parámetros extra para la sección. Reemplace `<option-name>` con una opción válida listada para esta sección en la página man de XF86Config.

7.3.1.8. `Device`

Cada sección `Device` configura una tarjeta de vídeo en el sistema. Mientras una sección `Device` es lo mínimo, instancias adicionales pueden ocurrir para cada tarjeta de vídeo instalada en la máquina.

La mejor forma de configurar una tarjeta de vídeo es configurando X durante el proceso de instalación o usando la **Herramienta de configuración de X**. Para más detalles sobre el uso de la **Herramienta de configuración de X** consulte el capítulo llamado *Audio, Vídeo y entretenimiento general* en *Manual del principiante de Red Hat Linux*.

El siguiente ejemplo ilustra una sección `Device` típica para una tarjeta de vídeo:

```
Section "Device"
 Identifier  "Videocard0"
 Driver      "mga"
 VendorName  "Videocard vendor"
 BoardName   "Matrox Millennium G200"
 VideoRam    8192
        Option      "dpms"
EndSection
```

Las siguientes entradas son usadas comúnmente en la sección `Device`:

- `Identifier` — Especifica un nombre único para esta sección `Device`. Esta es una entrada requerida.
- `Driver` — Especifica cuál controlador debe cargar el servidor XFree86 para poder utilizar la tarjeta de vídeo. Se puede encontrar una lista de los controladores en `/usr/X11R6/lib/X11/Cards`, el cual es instalado con el paquete `hwdata`.
- `VendorName` — Un parámetro opcional el cual especifica el fabricante de la tarjeta de vídeo.
- `BoardName` — Un parámetro opcional el cual especifica el nombre de la tarjeta de vídeo.
- `VideoRam` — Un parámetro opcional el cual especifica la cantidad de RAM disponible en la tarjeta de vídeo en kilobytes. Este valor sólo es necesario para tarjetas de vídeo que el servidor XFree86 no puede probar para detectar la cantidad de RAM.
- `BusID` — Una entrada opcional la cual especifica la ubicación del bus de la tarjeta de vídeo. Esta opción es necesaria solamente para sistemas con múltiples tarjetas.
- `Screen` — Una entrada opcional la cual especifica que conector de monitor o cabezal en la tarjeta de vídeo configura la sección `Device`. Esta opción es útil solamente para tarjetas de vídeo con múltiples cabezales.

  Si múltiples monitores son conectados a diferentes cabezales en la misma tarjeta de vídeo, deben existir secciones `Device` separadas y cada una de estas secciones debe tener un valor `Screen` diferente.

  Los valores para la entrada `Screen` deben ser enteros. El primer cabezal en la tarjeta de vídeo tiene un valor de `0`. El valor para cada cabezal adicional incrementa este valor en uno.
- `Option "<option-name>"` — Una entrada opcional la cual especifica parámetros extra para la sección. Reemplace *<option-name>* con una opción válida listada para esta sección en la página man de XF86Config.

  Una de las opciones más comunes es `"dpms"`, la cual activa la conformidad de energía Service Star para el monitor.

7.3.1.9. `Screen`

Cada sección `Screen` vincula una tarjeta de vídeo (o cabezal) a un monitor referenciando la sección `Device` y la sección `Monitor` para cada uno. Mientras que una sección `Screen` es lo mínimo, pueden ocurrir instancias adicionales para cada combinación de tarjeta de vídeo y monitor presente en la máquina.

El ejemplo siguiente ilustra una sección `Screen` típica:

```
Section "Screen"
 Identifier "Screen0"
 Device     "Videocard0"
 Monitor    "Monitor0"
 DefaultDepth     16
 SubSection "Display"
  Depth     24
  Modes    "1280x1024" "1280x960" "1152x864" "1024x768" "800x600" "640x480"
 EndSubSection
 SubSection "Display"
  Depth     16
  Modes    "1152x864" "1024x768" "800x600" "640x480"
 EndSubSection
EndSection
```

Las siguientes entradas son usadas a menudo en la sección `Screen`:

- `Identifier` — Especifica un nombre único para esta sección `Screen`. Esta es una entrada requerida.
- `Device` — Especifica el nombre único de una sección `Device`. Esta es una entrada requerida.
- `Monitor` — Especifica el nombre único de una sección `Monitor`. Esta es una entrada requerida.
- `DefaultDepth` — Especifica la profundidad del color por defecto en bits. En el ejemplo anterior, el valor por defecto es `16`, lo que proporciona miles de colores. Múltiples entradas de `DefaultDepth` son permitidas, pero al menos una debe estar presente.
- `SubSection "Display"` — Especifica los modos de la pantalla disponibles en una profundidad de color particular. Una sección `Screen` puede tener múltiples subsecciones `Display`, pero debe haber al menos una para la profundidad de color especificada en la entrada `DefaultDepth`.
- `Option "<option-name>"` — Una entrada opcional que especifica parámetros extra para la sección. Reemplace *<option-name>* con una opción válida listada para esta sección en la página man de XF86Config.

7.3.1.10. `DRI`

La sección opcional `DRI` especifica parámetros para *Direct Rendering Infrastructure* (*DRI*). DRI es una interfaz que permite a las aplicaciones de software 3D sacar provecho de las capacidades de aceleración de hardware 3D incorporadas en la mayoría del hardware moderno de vídeo. Además, DRI puede mejorar el rendimiento de 2D a través de la aceleración de hardware, si es soportado por el controlador de la tarjeta.

Esta sección es ignorada a menos que DRI esté activada en la sección `Module`.

El ejemplo siguiente muestra una sección `DRI` típica:

```
Section "DRI"
        Group        0
        Mode         0666
EndSection
```

Puesto que tarjetas de vídeo diferentes utilizan DRI de formas diferentes, no modifique estos valores para esta sección sin primero referirse al archivo `/usr/X11R6/lib/X11/doc/README.DRI`.

7.4. Fuentes

Red Hat Linux utiliza dos métodos para manejar fuentes y mostrarlas bajo XFree86. El subsistema de fuentes más nuevo Fontconfig simplifica la gestión de fuentes y proporciona características avanzadas, tales como anti-aliasing. Este sistema es usado automáticamente para aplicaciones programadas usando el conjunto de herramientas Qt 3 o GTK+ 2.

Por compatibilidad, Red Hat Linux incluye el subsistema de fuentes original, llamado el subsistema de fuentes nucleo de X. Este sistema, el cual tiene más de 15, está basado en el *Servidor de fuentes de X* (*xfs*).

Esta sección discute cómo configurar fuentes para X usando ambos sistemas.

7.4.1. Fontconfig

El subsistema de fuentes Fontconfig permite a las aplicaciones accesar directamente fuentes en el sistema y usar Xft u otros mecanismos de traducción de fuentes para interpretar fuentes Fontconfig con anti-aliasing avanzados. Las aplicaciones gráficas pueden usar la librería Xft con Fontconfig para dibujar texto a la pantalla.

Con el tiempo, el subsistema de fuentes Fontconfig/Xft reemplazará el subsistema de fuentes base de X.

Importante

El subsistema de fuentes Fontconfig aún no funciona para **OpenOffice.org** y **Abiword**, las cuales tienen sus propias tecnologías de interpretación de fuentes.

Es importante resaltar que Fontconfig comparte el archivo de configuración `/etc/fonts/fonts.conf`, el cual sustituye al `/etc/X11/XftConfig`. El archivo de configuración Fontconfig no debería se modificado manualmente.

Sugerencia

Debido a la transición al nuevo sistema de fuentes, las aplicaciones GTK+ 1.2 no son afectadas por ningún cambio realizado a través del diálogo **Preferencias de tipografía** (accesado al seleccionar **Botón de menú principal** [en el Panel] => **Preferencias** => **Fuentes**). Para estas aplicaciones, se puede configurar una fuente añadiendo las líneas siguientes al archivo `~/.gtkrc.mine`:

```
style "user-font" {

fontset = "<font-specification>"

}

widget_class "*" style "user-font"
```

Sustituya *<font-specification>* con una especificación de fuente en el estilo utilizado por las aplicaciones X tradicionales, tales como `-adobe-helvetica-medium-r-normal--*-120-*-*-*-*-*-*`. Se puede obtener una lista completa de las fuentes base ejecutando `xlsfonts` o creándolas interactivamente usando `xfontsel`.

7.4.1.1. Añadir fuentes a Fontconfig

Añadir fuentes al subsistema Fontconfig es un proceso bastante directo.

1. Para añadir fuentes para todo el sistema, copie las nuevas fuentes en el directorio `/usr/share/fonts/local/`.

   Para añadir fuentes para un usuario individual, copie las nuevas fuentes en el directorio `.fonts/` en el directorio principal del usuario.

2. Utilice el comando `fc-cache` para actualizar la información caché de la fuente, como en el ejemplo siguiente:

   ```
   4fc-cache <path-to-font-directory>
   ```

   En este comando, sustituya *<path-to-font-directory>* con el directorio conteniendo las nuevas fuentes (bien sea `/usr/share/fonts/local/` o `~/.fonts/`).

Sugerencia

Usuarios individuales también pueden instalar fuentes gráficamente, navegando a `fonts:///` en Nautilus y arrastrando los nuevos archivos de fuentes allí.

Importante

Si el nombre del archivo de fuentes termina con una extensión `.gz`, está comprimido y no puede ser usado hasta que se descomprima. Para hacer esto, utilice el comando `gunzip` o haga doble-click sobre el archivo y arrastre la fuente a un directorio en **Nautilus**.

7.4.2. Sistema de fuentes base de X

Por compatibilidad, Red Hat Linux todavía proporciona el subsistema de fuentes base de X, el cual utiliza el servidor de fuentes X (`xfs`) para proporcionar fuentes a las aplicaciones clientes X.

El servidor XFree86 busca por un servidor de fuentes especificado en la entrada `FontPath` bajo la sección `Files` del archivo de configuración `/etc/X11/XF86Config`. Refiérase a Sección 7.3.1.4 para más información sobre la entrada `FontPath`.

El servidor XFree86 se conecta al servidor `xfs` en un puerto especificado para adquirir la información de fuentes. Por esta razón, el servicio `xfs` debe estar ejecutándose para que X pueda arrancar. Para más detalles sobre la configuración de servicios para un nivel de ejecución particular, refiérase al capítulo llamado *Controlar el acceso a servicios* en el *Manual de personalización de Red Hat Linux*.

7.4.2.1. Configuración de `xfs`

El script `/etc/rc.d/init.d/xfs` inicia el servidor `xfs`. Se pueden configurar muchas opciones en el archivo `/etc/X11/fs/config`.

La siguiente es una lista de las opciones más usadas:

- `alternate-servers` — Configura una lista de servidores alternativos de fuentes que podrán ser utilizados en el caso de que el servidor actual no esté disponible. Los diferentes servidores deberán estar separados por comas.
- `catalogue` — Lista ordenada de rutas que contienen las fuentes a utilizar. Cada ruta hacia las fuentes deberá estar separada por una coma antes de que comience otra nueva ruta en la lista.

  Puede utilizar la cadena `:unscaled` inmediatamente después de la ruta hacia las fuentes para hacer que las fuentes no escalables se carguen antes que el resto de las fuentes de la ruta. Entonces, podrá especificar la ruta completa de nuevo de tal forma que las otras fuentes que sean escalables puedan ser cargadas.
- `client-limit` — Configura el número de clientes que el servidor de fuentes podrá servir antes de comenzar a denegar las conexiones. El número por defecto es `10`.
- `clone-self` — Permite al servidor de fuentes clonar una nueva versión de sí mismo si se llega al límite definido por el parámetro `client-limit`. Por defecto, esta opción está configurada como `on`.
- `default-point-size` — Configura el tamaño de punto por defecto para cualquier fuente que no especifique este valor. El valor de esta opción está estimado en décimas de puntos. El valor por defecto de `120` se corresponde a fuentes de 12 puntos.
- `default-resolutions` — Especifica una lista de resoluciones soportadas por el servidor XFree86. Cada resolución de la lista debe estar separada por una coma.
- `deferglyphs` — Especifica si retrasar la carga de *glyphs* (el gráfico usado para visualmente representar una fuente). Para desactivar esta característica utilice `none`, para activarla para todas las fuentes utilice `all`, o para activar esta característica solamente para fuentes de 16-bit use `16`.
- `error-file` — Le permite especificar la ruta y el nombre de archivo donde se almacenarán los informes de error de `xfs`.
- `no-listen` — Dice a `xfs` que no escuche utilizando un protocolo en particular. Por defecto, esta opción está configurada con `tcp` para evitar que `xfs` escuche utilizando puertos TCP, por motivos de seguridad. Si planea utilizar`xfs` para servir fuentes a estaciones de trabajo en red, deberá borrar esta línea.
- `port` — Especifica el puerto TCP en el cual `xfs` escuchará si `no-listen` no existe o está entre comentarios.
- `use-syslog` — Especifica si utilizar el registro de errores del sistema.

7.4.2.2. Añadir fuentes a `xfs`

Para añadir fuentes al subsistema base de fuentes de X (`xfs`), siga los pasos siguientes:

1. Si aún no existe, cree un directorio llamado `/usr/share/fonts/local/` usando el comando siguiente como usuario root:
   ```
   mkdir /usr/share/fonts/local/
   ```
   Si es necesario la creación del directorio `/usr/share/fonts/local/`, se debe añadir a la ruta `xfs` usando el comando siguiente como root:
   ```
   chkfontpath --add /usr/share/fonts/local/
   ```
2. Copie el nuevo archivo de fuente en el directorio `/usr/share/fonts/local/`
3. Actualice la información de la fuente emitiendo el siguiente comando como root:
   ```
   ttmkfdir -d /usr/share/fonts/local/ -o /usr/share/fonts/local/fonts.scale
   ```
4. Reinicie el servidor de fuentes `xfs` utilizando el comando siguiente como root:
   ```
   service xfs reload
   ```

7.5. Niveles de ejecución y XFree86

En la mayoría de los casos, la instalación por defecto de Red Hat Linux configura una máquina para arrancar en un entorno de conexión gráfico, conocido como nivel de ejecución 5. Es posible, sin embargo, arrancar en el modo multiusuario de sólo texto llamado nivel de ejecución 3 y comenzar una sesión X desde allí.

Para más información sobre los niveles de ejecución, consulte Sección 1.4.

Esta sección revisa cómo XFree86 arranca en ambos niveles de ejecución 3 y 5.

7.5.1. Nivel de ejecución 3

Cuando estamos en el nivel de ejecución 3, la forma habitual de iniciar una sesión X es escribiendo el comando `startx`. El comando `startx` es un front-end del programa `xinit` el cual lanza el servidor XFree86 y conecta los clientes X al mismo. Ya que usted debe de haber entrado en el sistema con su usuario cuando realice este procedimiento a partir del nivel de ejecución 3, `startx` no lanzará un gestor de visualización o autenticará al usuario. Refiérase a Sección 7.5.2 para más información sobre los gestores de visualización.

Cuando `startx` comienza, busca un archivo `.xinitrc` en el directorio principal del usuario para definir el entorno de escritorio y posiblemente otras aplicaciones clientes X a ejecutar. Si este archivo `.xinitrc` no se encuentra, se utilizará el archivo por defecto `/etc/X11/xinit/xinitrc`.

El script por defecto `xinitrc` luego buscará por los archivos definidos por el usuario y archivos de sistema por defecto, incluyendo `.Xresources`, `.Xmodmap` y `.Xkbmap` en el directorio principal del usuario y `Xresources`, `Xmodmap` y `Xkbmap` en el directorio `/etc/X11/`. Los archivos `Xmodmap` y `Xkbmap`, si existen, son usados por la utilidad `xmodmap` para configurar el teclado. Los archivos `Xresources` son leídos para asignar valores de preferencia específicos a aplicaciones.

Después de configurar estas opciones, el script `xinitrc` ejecuta todos los scripts localizados en el directorio `/etc/X11/xinit/xinitrc.d/`. Un script muy importante en este directorio es `xinput`, el cual configura los parámetros tales como el idioma por defecto.

Luego, el script `xinitrc` intenta ejecutar `.Xclients` en el directorio principal del usuario y cambia a `/etc/X11/xinit/Xclients` si no lo puede encontrar. El propósito del archivo `Xclients` es arrancar el entorno de escritorio o posiblemente, sólo un gestor de ventanas básico. El script `.Xclients` en el directorio principal del usuario inicia el entorno de escritorio especificado por el usuario en el archivo `.Xclients-default`. Si `.Xclients` no existe en el directorio principal del usuario, el script estándar `/etc/X11/init/Xclients` intenta iniciar otro entorno de escritorio, intentando primero con GNOME y luego con KDE seguido por `twm`.

El usuario es devuelto a una sesión de modo texto después de desconectarse de X del nivel de ejecución 3.

7.5.2. Nivel de ejecución 5

Cuando el sistema arranca en el nivel de ejecución 5, se lanza una aplicación cliente de X especial, llamada un gestor de visualización. Un usuario debe autenticarse usando el gestor de visualización antes de que se inicien cualquier entorno de escritorio o gestores de ventanas.

Dependiendo de los entornos de escritorio instalados en su máquina, estan disponibles tres gestores de visualización diferentes para manejar la autenticación de los usuarios.

- `gdm` — Es el gestor de visualización por defecto para Red Hat Linux y permite que los usuarios puedan configurar los parámetros de idioma, cierre del sistema, reinicio o conexión al sistema.
- `kdm` — es el gestor de visualización de KDE que permite a los usuarios apagar, reiniciar o conectarse al sistema.

- `xdm` — Este es un gestor de visualización muy básico que sólo permite que el usuario se conecte al sistema.

Cuando arranque en el nivel de ejecución 5, el script `prefdm` determina el gestor de visualización preferido haciendo referencia al archivo `/etc/sysconfig/desktop`. Refiérase al archivo `/usr/share/doc/initscripts-<version-number>/sysconfig.txt` (donde *<version-number>* es el número de la versión del paquete `initscripts`) para ver un listado de las opciones disponibles para este archivo.

Cada uno de los gestores de visualización hace referencia al archivo `/etc/X11/xdm/Xsetup_0` para configurar la pantalla de conexión. Una vez que el usuario se conecte al sistema, el script `/etc/X11/xdm/GiveConsole` corre para asignar la propiedad de la consola para el usuario. Luego, el script `/etc/X11/xdm/Xsession` se ejecuta para llevar a cabo muchas de las tareas que son normalmente realizadas por el script `xinitrc` cuando arranca X desde el nivel de ejecución 3, incluyendo la configuración del sistema y los recursos del usuario, así como también ejecutar los scripts en el directorio `/etc/X11/xinit/xinitrc.d/`.

El usuario puede especificar cuál entorno de escritorio desea utilizar cuando se autentican usando los gestores de visualización `gdm` o `kdm` seleccionándolo desde el menú **Session** (accesado al seleccionar **Botón de menú principal** [en el Panel] => **Preferencias** => **Más Preferencias** => **Sesiones**). Si el entorno de escritorio no es especificado en el gestor de visualización, el script `/etc/X11/xdm/Xsession` verificará los archivos `.xsession` y `.Xclients` en el directorio principal del usuario para decidir cuál entorno de escritorio cargar. Como último recurso el archivo `/etc/X11/xinit/Xclients` es usado para seleccionar un entorno de escritorio o gestor de ventanas para usarse de la misma forma que en el nivel de ejecución 3.

Cuando el usuario termina una sesión X en la visualización por defecto (`:0`) y se desconecta, el script `/etc/X11/xdm/TakeConsole` se ejecuta y vuelve a asignar la propiedad de la consola al usuario root. El gestor de visualización original, que continúa ejecutándose después que el usuario se conecta, toma el control liberando un nuevo gestor de visualización. Esto reinicia el servidor XFree86, despliega una nueva ventana de conexión y reinicia el proceso completo otra vez.

El usuario es devuelto al gestor de visualización después de desconectarse de X desde el nivel de ejecución 5.

Para más información sobre cómo los gestores de visualización controlan la autenticación de los usuarios, consulte /usr/share/doc/gdm-*<version-number>*/README (donde *<version-number>* es el número de la versión para el paquete `gdm` instalado) y la página man de `xdm`.

7.6. Recursos adicionales

Se podría decir mucho más sobre el servidor XFree86, los clientes que se conectan a él y la variada gama de entornos de escritorio y gestores de ventanas.

7.6.1. Documentación instalada

- `/usr/X11R6/lib/X11/doc/README` — Describe brevemente la arquitectura de XFree86 y cómo obtener información adicional sobre el proyecto XFree86 como nuevo usuario.
- `/usr/X11R6/lib/X11/doc/RELNOTES` — Para usuarios avanzados que deseen leer más sobre las últimas características presentes en XFree86.
- `man XF86Config` — Contiene información sobre los archivos de configuración XFree86, incluyendo el significado y la sintaxis para las diversas secciones dentro de los archivos.

- `man XFree86` — La página man principal para toda la información de XFree86, detalla la diferencia entre conexiones de servidor X locales y de red, explora variables de entorno comunes, enumera opciones de líneas de comandos y proporciona combinaciones de teclas de gran ayuda.
- `man Xserver` — Describe el servidor de pantalla X.

7.6.2. Sitios Web útiles

- http://www.xfree86.org — Página principal del proyecto XFree86, que presenta la versión open source del sistema X Window. XFree86 se proporciona junto con Red Hat Linux para proporcionar el control sobre el hardware necesario y un entorno gráfico de usuario.
- http://dri.sourceforge.net — Página principal del proyecto DRI (Direct Rendering Infrastructure). DRI es el corazón del componente de aceleración 3D de XFree86.
- http://www.redhat.com/mirrors/LDP/HOWTO/XFree86-HOWTO — Un documento HOWTO detallando cómo realizar la instalación manual y la configuración personalizada de XFree86.
- http://www.gnome.org/ — Página principal del proyecto GNOME.
- http://www.kde.org/ — Página principal del entorno de escritorio KDE.
- http://nexp.cs.pdx.edu/fontconfig/ — Página principal del subsistema de fuentes Fontconfig para XFree86.

7.6.3. Libros relacionados

- *The Concise Guide to XFree86 for Linux* por Aron Hsiao; Que — Proporciona un punto de vista experto sobre la operación de XFree86 en sistemas Linux.
- *The New XFree86* por Bill Ball; Prima Publishing — Discute XFree86 y su relación con los entornos de escritorio más populares, como GNOME y KDE.
- *Beginning GTK+ and GNOME* por Peter Wright; Wrox Press, Inc. — Introduce a los programadores la arquitectura de GNOME, mostrando cómo iniciarse con GTK+.
- *GTK+/GNOME Application Development* por Havoc Pennington; New Riders Publishing — Una visión avanzada del corazón de la programación en GTK+, centrándose en código de ejemplo a través de una visión de las APIs disponibles.
- *KDE 2.0 Development* por David Sweet y Matthias Ettrich; Sams Publishing — Instruye a los programadores principiantes y avanzados sobre cómo sacarle partido a las diferentes guías de entorno necesarias para construir aplicaciones QT para KDE.

II. Referencia de servicios de red

Es posible implementar un gran variedad de servicios de red bajo Red Hat Linux. Esta parte describe cómo las interfaces de red son configuradas, así como también proporciona detalles sobre servicios de red críticos, tales como NFS, Servidor Apache HTTP, Sendmail, Fetchmail, Procmail, BIND y LDAP.

Tabla de contenidos

Capítulo 8.

Scripts de red

Usando Red Hat Linux, todas las comunicaciones de red acontecen entre *interfaces*, que son dispositivos de networking conectados al sistema, configurados de un modo determinado y usando un protocolo, al menos, para intercambiar datos con otros sistemas. Los diferentes tipos de interfaz que existen son tan variados como los dispositivos que los soportan.

Los ficheros de configuración para las diferentes interfaces de red y scripts para activarlos o desactivarlos están ubicados en el directorio `/etc/sysconfig/network-scripts`. Mientras que la existencia de ficheros de interfaces particulares puede diferir de sistema a sistema dependiendo del uso, los tres tipos de ficheros diferentes que existen en este directorio, *ficheros de configuración de interfaz*, *scripts de control de interfaz* y *ficheros de función de red*, funcionan conjuntamente para habilitar Red Hat Linux para el uso de diversos dispositivos de red disponibles.

Este capítulo explorará la relación entre estos ficheros y las diferentes opciones para su uso.

8.1. Ficheros de configuración de red

Antes de revisar los ficheros de configuración de interfaz estudiemos los ficheros de configuración principales que usa Red Hat Linux para configurar la red. La comprensión del papel que desemepeñan en la configuración de la red es fundamental para configurar el sistema.

Los principales ficheros de configuración de la red son los siguientes:

- `/etc/hosts` — El principal propóposito de este fichero es resolver los nombres de hosts que no se pueden resolver en otra manera. Se puede usar solamente para resolver nombres de hosts en pequeñas redes sin servidor DNS. Sin tener en cuenta el tipo de red que el ordenador use, este fichero contiene un línea que especifica la direcciól̀n IP del dispositivo loopback (`127.0.0.1`) como por ejemplo `localhost.localdomain`. Para mayor información consulte la página man del host.
- `/etc/resolv.conf` — Este fichero especifica las direcciones IP de los servidores DNS y el dominio de búsqueda. A menos que se haya configurado para algo diferente, los scripts de inicialización de la red llenan este fichero. Para mayor información consulte la página man resolv.conf.
- `/etc/sysconfig/network` — Especifica la información del routing y del host para todas las interfaces de red. Para mayor información sobre este fichero y las directivas que acepta consulte la Sección 4.1.23.
- `/etc/sysconfig/network-scripts/ifcfg-<interface-name>` — Para cada interfaz de red del sistema Red Hat Linux existe un script de configuración de interfaz para una interfaz de red determinada. Consulte la Sección 8.2 para mayor información.

Advertencia

El directorio `/etc/sysconfig/networking/` lo usa la herramienta **Herramienta de administración de redes** (`redhat-config-network`) y sus contenidos no se modifican manualmente. Para mayor información sobre la configuración de las interfaces de red usando la herramienta **Herramienta de administración de redes**, consulte el capítulo *Configuración de red* en el *Manual de personalización de Red Hat Linux*.

8.2. Ficheros de configuración de interfaz

Los ficheros de configuración de interfaz controlan la operación de un dispositivo de interfaz de red particular. Cuando su sistema Red Hat Linux arranca, utiliza estos ficheros para saber qué interfaces utilizar automáticamente y cómo configurarlas para que operen correctamente. Estos ficheros habitualmente se conocen como `ifcfg-<device>`, donde `<device>` hace referencia al nombre del dispositivo que controla el fichero de configuración.

8.2.1. Interfaces Ethernet

Uno de los ficheros de interfaz más comunes es `ifcfg-eth0`, que controla el primer NIC de un sistema. En un sistema con muchos NICs, tendrá ficheros `ifcfg-eth` múltiples, cada uno con nun número al final del nombre del fichero. Como cada dispositivo tiene su propio fichero de configuración, llevará un gran control sobre el modo en que funciona cada interfaz.

Un ejemplo `ifcfg-eth0` para un sistema que usa una dirección IP fija sería de la siguiente manera:

```
DEVICE=eth0
BOOTPROTO=none
ONBOOT=yes
NETWORK=10.0.1.0
NETMASK=255.255.255.0
IPADDR=10.0.1.27
USERCTL=no
```

Los valores que se requieren en un fichero de configuración de interfaz pueden cambiar basándose en otros valores. Por ejemplo, el fichero `ifcfg-eth0` para una interfaz que use DHCP aparecerá diferente, debido al hecho de que la información IP viene proporcionada por el servidor DHCP:

```
DEVICE=eth0
BOOTPROTO=dhcp
ONBOOT=yes
```

La mayoría del tiempo, deseará utilizar una utilidad GUI, como por ejemplo **Configurador de red** (`redhat-config-network`) o `netconfig` para hacer cambios en los diversos ficheros de configuración de interfaz. Vea el *Manual de personalización de Red Hat Linux* para más instrucciones sobre el uso de estas herramientas.

También puede modificar el fichero de configuración de un determinado dispositivo de red a mano. A continuación, le mostramos los parámetros que necesita para modificar el fichero de configuración.

Dentro de cada uno de los ficheros de configuración de la interfaz, son comunes los siguientes valores:

- `BOOTPROTO=<protocol>`, donde `<protocol>` es uno de los siguientes:
  - `none` — No se debería utilizar ningún protocolo de tiempo de arranque.
  - `bootp` — Se debería utilizar el protocolo BOOTP.
  - `dhcp` — Se debería utilizar el protocolo DHCP.

- `BROADCAST=<address>`, donde `<address>` es la dirección de broadcast.
- `DEVICE=<name>`, donde `<name>` es el nombre del dispositivo físico (a excepción de los dispositivos PPP asignados de forma dinámica donde es el *nombre lógico*).
- `DNS{1,2}=<address>`, donde `<address>` es la dirección del servidor de nombres que se tiene que colocar en `/etc/resolv.conf` si la directiva `PEERDNS` está activada.
- `IPADDR=<address>`, donde `<address>` es la dirección IP.

- `NETMASK=<mask>`, donde `<mask>` es el valor de la máscara de red.
- `NETWORK=<address>`, donde `<address>` es la dirección de red. Esta opción ya no se usa.
- `ONBOOT=<answer>`, donde `<answer>` es uno de los siguientes:
  - `sí` — El dispositivo debería activarse en el momento de arranque.
  - `no` — Este dispositivo no debería activarse en el momento de arranque.

- `PEERDNS=<answer>`, donde `<answer>` es uno de las siguientes:
  - `sí` — Modificar `/etc/resolv.conf` si la directiva DNS está activada. Si estáusando DCHP, la opción `sí` es la predeterminada.
  - `no` — No modificar `/etc/resolv.conf`.
  - `SRCADDR=<address>`, donde `<address>` es la dirección IP de una fuente específica para los paquetes externos.
  - `USERCTL=<answer>`, donde `<answer>` es uno de los siguientes:
    - `verdadero` — Se les permite controlar este dispostivo a todos los ususarios, aunque éstos no sean root.
    - `falso` — No se les permite controlar este dispositivo a los usuarios que no sean root.

8.2.2. Interfaces de acceso telefónico

Si se conecta a una red, como Internet, a través de la conexión de acceso telefónico PPP, necesitará un fichero de configuración para esa interfaz.

Este fichero se crea automáticamente cuando usa las aplicaciones `wvdial`, **Herramienta de administración de redes** o **Kppp** para crear una cuenta telefónica. Además, todos los cambios que se hagan en la cuentas telefónica se reflejan en estos ficheros de configuración de estos dispositivos. El *Manual del principiante de Red Hat Linux* contine las instrucciones para usar estas herramientas de conexión telefónica. También puede crear y modificar este fichero a mano. La muestra de ficheros `ifcfg-ppp0` sería de la siguiente manera:

```
DEVICE=ppp0
NAME=test
WVDIALSECT=test
MODEMPORT=/dev/modem
LINESPEED=115200
PAPNAME=test
USERCTL=true
ONBOOT=no
PERSIST=no
DEFROUTE=yes
PEERDNS=yes
DEMAND=no
IDLETIMEOUT=600
```

El *Protocolo de Internet de línea serial (SLIP)* es otra interfaz de acceso telefónico, aunque se usa menos. Los fichero SLIP tienen nombres de ficheros de configuración de interfaz tales como `ifcfg-sl0`.

Opciones que se pueden utilizar en estos ficheros:

- `DEFROUTE=<answer>`, donde `<answer>` es uno de las siguientes:
  - `sí` — Establece esta interfaz como la ruta por defecto.
  - `no` — No establece la interfaz como la ruta por defecto.

- `DEMAND=<answer>`, donde `<answer>` es una de las siguientes:
  - `sí` — Esta interfaz permitirá que `pppd` inicie una conexión cuando alguien está intentando utilizarla.
  - `no` — Se debe establecer una conexión de forma manual para esta interfaz.

- `IDLETIMEOUT=<value>`, donde `<value>` es el número de segundos de actividad improductiva antes de que la interfaz se desconecte.
- `INITSTRING=<string>`, donde `<string>` es la cadena init que pasa al dispositivo del módem. Esta opción se usa en primer lugar con las interfaces SLIP.
- `LINESPEED=<value>`, donde `<value>` es el ratio de baudios del dispositivo. Los posibles valores estándar incluyen `57600`, `38400`, `19200` y `9600`, entre otros.
- `MODEMPORT=<device>`, donde `<device>` es el nombre del dispositivo (habitualmente un módem) que se usa para establecer la conexión para la interfaz.
- `MTU=<value>`, donde `<value>` es la *unidad máxima de transferencia (MTU)* configurada para la interfaz. La MTU hace referencia a el mayor número de bytes de datos que puede abarcar un bloque, sin contar la información de encabezamiento y de final. En algunas situaciones de acceso telefónico, configurarlo hasta un valor de `576` dará un resultado de pocos paquetes eliminados y mejorará muy vagamente la capacidad de tratamiento para una conexión.
- `NAME=<name>`, donde `<name>` esla referencia al título que se le da al grupo de configuraciones de conexiones de acceso telefónico.
- `PAPNAME=<name>`, donde `<name>` es el nombre de usuario dado durante el intercambio del *Protocolo de autenticación de contraseña (PAP)* que le permite conectarse a un sistema remoto.
- `PEERDNS=<answer>`, donde `<answer>` es una de las siguientes:
  - `sí` — Esta interfaz modificará las entradas del fichero `/etc/resolv.conf` del sistema para el uso de los servidores DNS proporcionados por el sistema remoto cuando se establece una conexión.
  - `no` — El fichero `/etc/resolv.conf` no cambiará.

- `PERSIST=<answer>`, donde `<answer>` es una de las siguientes:
  - `sí` — Esta interfaz debería mantenerse siempre activa, incluso si se desactiva tras una detención del módem.
  - `no` — Esta interfaz no debería mantenerse siempre activa.

- `REMIP=<address>`, donde `<address>` es la dirección IP del sistema remoto. Habitualmente esto no se especifica.
- `WVDIALSECT=<name>`, donde `<name>` asocia esta interfaz con una configuración de marcado en `/etc/wvdial.conf`, que contiene el número de teléfono para que sea marcado y otra información importante para la interfaz.

8.2.3. Otras interfaces

Otro fichero de configuración de interfaz comunes que usan eestas opciones es el `ifcfg-lo`, que controla el dispositivo loopback local del protocolo IP, `ifcfg-irlan0`, que establece los parámetros para el primer dispositivo infrarojo, `ifcfg-plip0`, que controla el primer dispositivo PLIP, y `ifcfg-tr0`, que se usa con el primer dispositivo Token Ring.

A menudo se usa una *interfaz loopback* en las pruebas así como una variedad de aplicaciones que requieren una dirección IP que apunte al mismo sistema. Todos los datos que se mandan al dispositivo loopback vuelven inmediatamente a la red del host. layer.

Advertencia

No modifique nunca el script de la interfaz loopback `/etc/sysconfig/network-scripts/ifcfg-lo` manualmente. Si ocurre lo contrario, el sistema puede dejar de funcionar correctamente.

Una *interfaz de infrarojo* permite que se transmita información entre dispositivos como un portátil y una impresora y además se puede pasar a un enlace infrarojo que funciona como el dispositivo Ethernet excepto que se da en una conexión peer-to-peer.

La conexión *Parallel Line Interface Protocol (PLIP)* funciona de la misma manera, solamente que usa un paralelo. port.

Las topologías *Token Ring* no son tan frecuentes como las redes de área local como antes ocurría ya que han Ethernet las ha sustituido.

8.2.4. Ficheros alias y clon

Dos tipos menos usados de ficheros de configuración de interfaz que se encuentran en `/etc/sysconfig/network-scripts` son los ficheros *alias* y *clon*, que incluyen un componente adicional en el nombre del fichero más allá del nombre de la interfaz.

Los ficheros de configuración de la interfaz toman nombres en el formato de `ifcfg-<if-name>:<alias-value>` y permiten que un alias señale una interfaz. Por ejemplo, un fichero `ifcfg-eth0:0` podría estar configurado para especificar `DEVICE=eth0:0` y una dirección IP estática de 10.0.0.2, que sirva como un alias de una interfaz Ethernet que ya haya sido configuradad para recibir la información IP a través de DHCP en `ifcfg-eth0`. Llegado a ese punto, el dispositivo `eth0` está ligado a una dirección IP 10.0.0.2.

Un fichero de configuración de clon tiene un nombre parecido a `ifcfg-<if-name>-<clone-name>`. Un fichero alias es otro modo de referirse a un fichero de configuración de interfaz ya existente, mientras que un fichero clon se usa para especificar opciones adicionales al especificar una interfaz. Por ejemplo, si tiene una interfaz Ethernet DHCP estándar llamada `eth0`, será de la siguiente manera:

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=dhcp
```

Como `USERCTL` no está configurado para la opción `sí`, los usuarios no pueden activar y desactivar esta interfaz. Para que los usuarios gocen de esta habilidad, cree un clon llamado `user` de `ifcfg-eth0` que permita que un usuario active o no la interfaz `eth0`. El nombre que resulta del clon sería `ifcfg-eth0-user` y tan sólo necesitaría una línea:

```
USERCTL=yes
```

Cuando un usuario activa la interfaz `eth0` mediante el comando `ifup eth0-user`, las opciones de configuración desde `ifcfg-eth0` y `ifcfg-eth0-user` se usan conjuntamente. Aunque este ejemplo es muy sencillo, este método puede ser utilizado con una variedad de opciones e interfaces.

El modo más sencillo de crear ficheros de configuración de interfaces alias y clon es mediante el uso de la herramienta **Herramienta de administración de redes**. Para mayor información, consulte el capítulo *Configuración de red* en el *Manual de personalización de Red Hat Linux*.

8.3. Scripts de control de interfaz

Los scripts de control de interfaz controlan la activación y desactivación de las conexiones de interfaz. Existen dos scripts de control de la interfaz primarios, `/sbin/ifdown` y `/sbin/ifup` que utilizan otros scripts de control variados localizados en el directorio `/etc/sysconfig/network-scripts` para activar y desactivar las interfaces de red.

Los dos scripts de control de interfaz son `ifdown` y `ifup` y son enlaces simbólicos para los scripts en el directorio `/sbin`. Cuando se solicita cualquiera de estos scripts, aceptan el uso de un valor de la interfaz, como por ejemplo:

```
ifup eth0
Determining IP information for eth0... done.
```

At that point, the `/etc/rc.d/init.d/functions` and `/etc/sysconfig/network-scripts/network-functions` files are used to perform a variety of tasks. See Sección 8.4 for more information.

Tras haber verificado que se ha especificado una interfaz y que al usuario que ha ejecutado la petición se le permite activar o desactivar la interfaz, se solicita el script correcto para el tipo de dispositivo de interfaz. Los siguientes scripts de control de interfaz son los más habituales de este tipo:

- `ifup-aliases` — Configura los alias IP desde los ficheros de configuración de la interfaz cuando se asocia más de una dirección IP con una interfaz.
- `ifdown-cipcb` y `ifup-cipcb` — Se usan para activar y desactivar conexiones *Crypto IP Encapsulation (CIPE)*.
- `ifdown-ipv6` y `ifup-ipv6` — Contiene la llamada de funciones basadas en IPv6 que utilizan las variables de entorno en varios ficheros de configuración de la interfaz y `/etc/sysconfig/network`.
- `ifup-ipx` — Se usa para configurar una interfaz IPX.
- `ifup-plip` — Se usa para configurar una interfaz PLIP.
- `ifup-plusb` — Se usa para configurar una interfaz USB para conexiones de red.
- `ifdown-post` y `ifup-post` — Contiene comandos que se ejecutan después de que una interfaz particular haya sido activada o desactivada.
- `ifdown-ppp` y `ifup-ppp` — Se usa para activar o desactivar una interfaz PPP mediante el uso de un dispositivo en particular.
- `ifup-routes` — Añade rutas estáticas para un dispositivo en particular como si se activase su interfaz.
- `ifdown-sit` and `ifup-sit` — Contiene llamadas de funciones relacionadas con la activación y desactivación de un túnel IPv6 dentro de una conexión IPv4.
- `ifdown-sl` y `ifup-sl` — Se usa para activar o desactivar una interfaz SLIP.

Tenga en cuenta que si elimina o modifica estos scripts puede provocar varias conexiones de interfaz que pueden funcionar de forma extraña o incluso fallar, debido a que los scripts tienden a apoyarse

uno en el otro. Sin embargo, los usuarios avanzados pueden modificar los scripts relacionados con una interfaz específica para hacer que se produzcan pasos adicionales cuando esa interfaz se activa o desactiva.

También puede utilizar el script init `/etc/rc.d/init.d/network` para activar o desactivar todas las interfaces de red configuradas para iniciar en el momento de arranque con el comando:

```
/sbin/service network action
```

donde *`action`* es `start` para inciar las interfaces de red, `stop` para interrumpir las interfaces de red o `restart` para reiniciar las interfaces dered. También puede utilizar el comando `/sbin/service/network status` para visualizar una lista de dispositivos configurados y dispositivos activos en la actualidad.

8.4. Funciones de red

Red Hat Linux utiliza varios ficheros que contienen funciones importantes que se usan de modo diverso para activar o desactivar interfaces. Más que fozar cada fichero de control de interfaz para que contenga las mismas funciones que otros, estas funciones están agrupadas convenientemente juntas en algunos ficheros que se pueden suministrar cuando sean necesarios.

El fichero con las funciones de red más comunes es el `network-functions`, localizado en el directorio `/etc/sysconfig/network-scripts`. Este fichero contiene una variedad de funciones IPv4 comunes útil para muchos scripts de control de interfaz, como por ejemplo el contactar con programas en ejecución que han solicitado información sobre cambios en un estado en interfaces, configuración de nombres del host, encontrar dispositivos de puerta de enlace, ver si un dispositivo en particular está o no activado y añadir una ruta por defecto.

Debido a que las funciones solicitadas por las interfaces IPv6 son diferentes de las interfaces IPv4, existe específicamente un fichero `network-functions-ipv6` para sostener esta información. El soporte IPv6 debe ser habilitado en el kernel para comunicar a través de ese protocolo. Existe una función presente en este fichero que comprueba la presencia de soporte IPv6. Además de las funciones que configuran y borran las rutas IPv6 estáticas, crean y borran túneles, añaden y eliminan direcciones IPv6 para una interfaz y comprueban la existencia de una dirección IPv6 en una interfaz que también puede hallarse en este fichero.

8.5. Recursos adicionales

Puede encontrar más información sobre los scripts de red en los siguientes sitios.

- `/usr/share/doc/initscripts-<`*`version`*`>/sysconfig.txt` — Un manual amplio que estudia las opciones disponibles para los ficheros de configuración, incluidas las opciones IPv6 que no cubre este capítulo.
- `/usr/share/doc/iproute-<`*`version`*`>/ip-cref.ps` — Fichero Postscript™ que contiene mucha información sobre el comando `ip`, que se usa para manipular las tablas de routing, entre otras cosas. Use **ghostview** o **kghostview** para ver este fichero.

redhat.

Capítulo 9.

Network File System (NFS)

NFS (Network File System) permite a las máquinas montar particiones en un sistema remoto en concreto y usarlas como si estuvieran en el sistema de archivos local. Esto permite centralizar archivos en una localización, mientras se permite su acceso continuo a los usuarios autorizados.

Hay dos versiones de NFS actualmente en uso. La versión 2 de NFS (NFSv2), que tiene varios años, es ampliamente soportada por muchos sistemas operativos. La versión 3 (NFSv3) tiene más características, incluyendo tamaño variable del manejador de archivos y una mejor información de errores. Red Hat Linux soporta tanto NFSv2 como NFSv3, y usa NFSv3 por defecto cuando se conecta a un servidor que lo soporta.

Este capítulo se centra en la versión 2 de NFS, aunque muchos de los conceptos aquí discutidos también se aplican a la versión 3. Adicionalmente, sólo los conceptos fundamentales de NFS e información suplementaria serán proporcionados. Para instrucciones específicas con respecto a la configuración y operación de NFS en servidores o clientes, vea el capítulo titulado *Network File System (NFS)* en el *Manual de personalización de Red Hat Linux*.

9.1. Metodología

Linux usa una combinación de soporte a nivel de kernel y demonios en continua ejecución para proporcionar la compartición de archivos NFS, sin embargo, el soporte NFS debe estar activo en el kernel de Linux para que funcione. NFS usa *Remote Procedure Calls (RPC)* para enrutar peticiones entre clientes y servidores, implicando que el servicio `portmap` debe estar disponible y activo en los niveles de ejecución adecuados para que la comunicación NFS funcione. Trabajando con `portmap`, los procesos siguientes se aseguran que una conexión particular NFS esté permitida y pueda proceder sin error:

- `rpc.mountd` — El proceso que recibe la petición de montaje desde un cliente NFS y chequea para mirar si coincide con un sistema de archivos actualmente exportado.
- `rpc.nfsd` — El proceso que implementa los componentes del espacio del usuario del servicio NFS. Trabaja con el kernel Linux para satisfacer las demandas dinámicas de clientes NFS, tales como proporcionar procesos adicionales del servidor para que los clientes NFS lo utilicen.
- `rpc.lockd` — Un demonio innecesario en los kernels modernos. El bloqueo de archivos NFS ahora lo hace el kernel. Está incluido en el paquete `nfs-utils` para usuarios de versiones antiguas del kernel que no incluyen esta capacidad por defecto.
- `rpc.statd` — Implementa el protocolo RPC *Network Status Monitor (NSM)*. Esto proporciona notificación de reinicio cuando un servidor NFS es reiniciado luego de haber sido apagado abruptamente.
- `rpc.rquotad` — Un servidor RPC que proporciona información de cuotas de usuarios a usuarios remotos.

No todos estos programas son requeridos para el servicio NFS. Los únicos servicios que deben estar activos son `rpc.mountd`, `rpc.nfsd`, y `portmap`. Los otros demonios proporcionan funcionalidades adicionales y sólo deben usarse si el entorno de su servidor los requiere.

La versión 2 de NFS usa el *User Datagram Protocol (UDP)* para proporcionar una conexión de red sin estado entre el cliente y el servidor. La versión 3 de NFS puede usar UDP o TCP corriendo sobre una IP. La conexión UDP sin estado minimiza el tráfico de red, al mandar el servidor NFS una cookie al cliente, después de que el cliente sea autorizado a acceder al volumen compartido. Esta cookie es un valor aleatorio guardado en la parte del servidor y es pasado junto con las peticiones RPC desde

el cliente. El servidor NFS puede ser reiniciado sin afectar a los clientes y las cookies permanecen intactas.

Con NFS, la autentificación sólo se produce cuando el cliente intenta montar un sistema de archivos remoto. Para limitar el acceso, el servidor NFS utiliza en primer lugar envolturas TCP (TCP wrappers). Estas envolturas leen los archivos `/etc/hosts.allow` y `/etc/hosts.deny` para determinar si a un cliente particular le debe ser explícitamente permitido o denegado su acceso al NFS. Para más información sobre cómo configurar los controles de acceso con envolturas TCP (TCP wrappers), consulte el Capítulo 15.

Después de que al cliente se le permite acceso a una envoltura TCP, el servidor NFS recurre a su archivo de configuración, `/etc/exports`, para determinar si el cliente tiene suficientes privilegios para montar alguno de los sistemas de archivos exportados. Después de permitir el acceso, cualquier operación de archivos y directorios es mandada al servidor usando llamadas de procedimiento remotas.

Aviso

Los privilegios de montajes NFS son permitidos específicamente a clientes, no a usuarios. Los sistemas de archivos exportados pueden ser accesados por cualquier usuario en la máquina remota.

Al configurar el archivo `/etc/exports`, sea extremadamente cuidadoso al otorgar permisos de lectura y escritura (`rw`) a un sistema de archivos exportado.

9.1.1. NFS y `portmap`

NFS se apoya en las llamadas de procedimientos remotos (RPC) para funcionar. Se requiere `portmap` para trazar las peticiones RPC a los servicios correctos. Los procesos RPC notifican a `portmap` cuando comienzan, revelando el número de puerto que ellos están monitorizando y el número de programas RPC que esperan servir. El sistema cliente entonces contacta con el `portmap` del servidor con un número de programa RPC particular. Entonces `portmap` redirecciona al cliente al número del puerto apropiado para que se comunique con el servicio adecuado.

Como los servicios basados en RPC confian en `portmap` para hacer todas las conexiones con las peticiones de clientes entrantes, `portmap` debe estar disponible antes que cualquiera de esos servicios comience. Si, por alguna razón, el servicio `portmap` inesperadamente se quita, reinicie `portmap` y cualquier servicio que estuviera ejecutándose entonces.

El servicio `portmap` puede ser usado con los archivos de accesos de envolturas TCP (`/etc/hosts.allow` y `/etc/hosts.deny`) para controlar a qué sistemas remotos les son permitidos usar servicios basados en RPC en el servidor. Vea el Capítulo 15 para más información. Las reglas de control de acceso para `portmap` afectarán a todos los servicios basados en RPC. Alternativamente, puede especificar a cada uno de los demonios RPC NFS que serán afectados por una regla de control específica. Las páginas man para `rpc.mountd` y `rpc.statd` contienen información relativa a la sintaxis precisa de estas reglas.

9.1.1.1. Resolución de problemas de NFS con `portmap`

Como `portmap` proporciona la coordinación entre servicios RPC y los números de puertos usados para comunicarlos, es útil poder visualizar el estado de los servicios RPC actuales usando `portmap` cuando estamos resolviendo algún problema. El comando `rpcinfo` muestra cada servicio basado en RPC con su número de puerto, número de programa RPC, versión y tipo de protocolo (TCP o UDP).

Para asegurarse que los servicios NFS basados en RPC están activos para `portmap`, use el comando `rpcinfo -p`:

```
program vers proto   port
```

```
100000    2   tcp    111  portmapper
100000    2   udp    111  portmapper
100024    1   udp   1024  status
100024    1   tcp   1024  status
100011    1   udp    819  rquotad
100011    2   udp    819  rquotad
100005    1   udp   1027  mountd
100005    1   tcp   1106  mountd
100005    2   udp   1027  mountd
100005    2   tcp   1106  mountd
100005    3   udp   1027  mountd
100005    3   tcp   1106  mountd
100003    2   udp   2049  nfs
100003    3   udp   2049  nfs
100021    1   udp   1028  nlockmgr
100021    3   udp   1028  nlockmgr
100021    4   udp   1028  nlockmgr
```

La opción `-p` prueba el portmap de la máquina especificada, o en la máquina local por defecto si no se especifica ninguna máquina. Otras opciones están disponibles en la página manual de `rpcinfo`.

De la salida anterior, varios servicios NFS pueden verse ejecutándose. Si uno de los servicios NFS no comienza correctamente, `portmap` puede ser incapaz de corresponder las peticiones RPC con sus respectivos puertos. En muchos casos, reiniciando NFS como root (`/sbin/service nfs restart`) provocará que estos servicios funcionen correctamente con `portmap` y empiecen a funcionar.

9.2. Archivos de configuración del servidor NFS

Es sencillo configurar un sistema para compartir archivos y directorios usando NFS. Cada Sistema de archivos que se exporta a usuarios remotos vía NFS, así como los derechos de acceso relativos a ellos, es localizado en el archivo `/etc/exports`. Este archivo es leído por el comando `exportfs` para dar a `rpc.mountd` y a `rpc.nfsd` la información necesaria para permitir el montaje remoto de un sistema de archivos por una máquina autorizada.

El comando `exportfs` permite a root exportar o no directorios concretos sin reiniciar los servicios NFS. Cuando se le pasan las opciones apropiadas a `exportfs`, el sistema de archivos a exportar es incluido en `/var/lib/nfs/xtab`. Como `rpc.mountd` se refiere al archivo `xtab` para decidir privilegios de acceso a un sistema de archivos, los cambios en la lista de sistemas de archivos exportados toman efecto inmediatamente.

Hay varias opciones disponibles cuando usamos `exportfs`:

- `-r` — Provoca que todos los directorios listados en `/etc/exports` sean exportados construyendo una nueva lista de exportación en `/etc/lib/nfs/xtab`. Esta opción refresca la lista de exportación con cualquier cambio que hubiéramos realizado en `/etc/exports`.
- `-a` — Provoca que todos los directorios sean exportados o no, dependiendo de qué otras opciones hemos pasado a `exportfs`.
- `-o` *opciones* — Permite al usuario especificar directorios a exportar que no estén listados en `/etc/exports`. Estos sistemas de archivos adicionales compartidos deben ser escritos de la misma forma que son especificados en `/etc/exports`. Esta opción es usada para probar un sistema de archivos antes de añadirlo permanentemente a la lista de sistemas a exportar.
- `-i` — Ignora `/etc/exports`; sólo las opciones dadas desde la línea de comandos son usadas para definir los sistemas de archivos exportados.

- `-u` — Termina de exportar directorios que puedan ser montados por usuarios remotos. El comando `exportfs -ua` suspende la compartición de archivos NFS mientras que mantiene los demonios activos. Para continuar con la compartición NFS, teclee `exportfs -r`.
- `-v` — Operación descriptiva, donde los sistemas de archivos exportados o dejados de exportar son mostrados en gran detalle al ejecutarse el comando `exportfs`.

Si no se pasan opciones al comando `exportfs`, mostrará una lista de los sistemas de archivos actualmente exportados.

Los cambios efectuados a `/etc/exports` pueden ser leídos al recargar el servicio NFS con el comando `service nfs reload`. Esto deja a los demonios NFS ejecutándose mientras reexporta el archivo `/etc/exports`.

9.2.1. `/etc/exports`

El archivo `/etc/exports` controla cuáles sistemas de archivos son exportados a las máquinas remotas y especifica opciones particulares que controlen todo. Las líneas en blanco son ignoradas, se pueden comentar líneas con el símbolo `#` y las líneas largas pueden ser divididas con una barra invertida (`\`). Cada sistema de archivos exportado debe tener su propia línea. La lista de máquinas autorizadas colocada después de un sistema de archivos exportado, debe estar separada por un espacio. Las opciones para cada uno de las máquinas deben ser colocadas entre paréntesis directamente detrás del identificador de la máquina, sin ningún espacio de separación entre la máquina y el primer paréntesis.

De esta sencilla manera, `/etc/exports` sólo necesita saber el directorio a exportar y las máquinas que pueden usarlo:

```
/some/directory bob.example.com
/another/exported/directory 192.168.0.3
```

Después de reexportar `/etc/exports` con el comando `/sbin/service nfs reload`, la máquina `bob.example.com` será capaz de montar `/some/directory` y `192.168.0.3` podrá montar `/another/exported/directory`. Como no hay opciones especificadas en este ejemplo, varias preferencias por defecto toman efecto:

- `ro` — Sólo lectura (read-only). Las máquinas que monten este sistema de archivos no podrán cambiarlo. Para permitirlas que puedan hacer cambios en el sistema de archivos, debe especificar la opción `rw` (lectura-escritura, read-write).
- `async` — Permite al servidor escribir los datos en el disco cuando lo crea conveniente. Mientras que esto no tiene importancia en un sistema de sólo lectura, si una máquina hace cambios en un sistema de archivos de lectura-escritura y el servidor se cae o se apaga, se pueden perder datos. Especificando la opción `sync`, todas las escrituras en el disco deben hacerse antes de devolver el control al cliente. Esto puede que disminuya el rendimiento.
- `wdelay` — Provoca que el servidor NFS retrase el escribir a disco si sospecha que otra petición de escritura es inminente. Esto puede mejorar el rendimiento reduciendo las veces que se debe acceder al disco por comandos de escritura separados. Use `no_wdelay` para desactivar esta opción, la cual sólo funciona si está usando la opción `sync`.
- `root_squash` — Previene a los usuarios root conectados remotamente de tener privilegios como root asignándole el userid de 'nobody'. Esto reconvierte el poder del usuario root remoto al de usuario local más bajo, previniendo que los usuarios root remotos puedan convertirse en usuarios root en el sistema local. Alternativamente, la opción `no_root_squash` lo desactiva. Para reconvertir a todos los usuarios, incluyendo a root, use la opción `all_squash`. Para especificar los ID de usuario y grupo para usar con usuarios remotos desde una máquina particular, use las opciones `anonuid` y `anongid`, respectivamente. De esta manera, puede crear una cuenta de usuario especial para usuarios NFS remotos para compartir y especificar

`(anonuid=`*<`uid-value`>*`,anongid=`*<`gid-value`>*`)`, donde *<`uid-value`>* es el número ID de usuario y *<`gid-value`>* es el número ID de grupo.

Para saltarse estas opciones predeterminadas, debe especificar una opción que tome su lugar. Por ejemplo, si no especifica la opción `rw`, entonces se exportará en sólo lectura. Cada opción predeterminada para cada sistema de archivos exportado, debe ser explícitamente ignorada. Adicionalmente, hay otra opciones que están disponibles que no tienen especificado un valor predeterminado. Estas incluyen desactivar el navegar por subdirectorios, permitir el acceso a puertos inseguros, y permitir bloquear archivos inseguros (necesario para algunas implementaciones antiguas de clientes NFS). Vea la página man de `exports` para estas opciones menos usadas.

Cuando especifique los nombres de máquinas, use los métodos siguientes:

- *una sola máquina* — Cuando una máquina en particular es especificada con nombre completo de dominio, nombre de máquina o dirección IP.
- *comodines* — Cuando usamos un carácter `*` o `?` para referirnos a un grupo de nombres completos de dominio o direcciones IP o que coincidan con una cadena particular de letras.

  Sin embargo, tenga cuidado cuando especifique comodines con nombres de dominio completos, pues tienden a ser más exactos de lo que usted cree. Por ejemplo, el uso de `*.example.com` como comodín, permitirá a ventas.domain.com acceder al sistema de archivos exportado, pero no a bob.ventas.domain.com. Para permitir ambas posibilidades, así como a sam.corp.domain.com, debería usar `*.example.com *.*.example.com`.
- *redes IP* — Permite el acceso a máquinas basadas en sus direcciones IP dentro de una red más grande. Por ejemplo, `192.168.0.0/15` permite al acceso a las primeras 16 direcciones IP, desde la 192.168.0.0 a la 192.168.0.15, acceder al sistema de archivos, pero no a la 192.168.0.16 y superiores.
- *grupos de redes* — Permite que un nombre de grupo de red NIS, escrita como `@`*<`group-name`>*, sea usada. Esto pone al servidor NIS controlando el acceso de este sistema de archivos, donde los usuarios pueden ser añadidos o borrados de un grupo NIS sin que afecte a `/etc/exports`.

Aviso

La manera en que el archivo `/etc/exports` está organizado es muy importante, particularmente lo que concierne a los espacios en blanco. Recuerde separar siempre los sistemas de archivos exportados de una máquina a la otra, con un espacio. Sin embargo, no debería haber otros espacios en el archivo a menos que se usen en líneas comentadas.

Por ejemplo, las siguientes dos líneas significan cosas distintas:

```
/home bob.example.com(rw)
/home bob.example.com (rw)
```

La primera línea permite sólo a los usuarios de `bob.example.com` acceder en modo de lectura-escritura al directorio `/home`. La segunda línea permite a los usuarios de `bob.example.com` montar el directorio de solo lectura (el predeterminado), pero el resto del mundo puede instalarlo como lectura-escritura.

9.3. Archivos de configuración de clientes NFS

Cualquier compartición NFS puesta a disposición por un servidor puede ser montada usando varios métodos. La compartición puede ser montada manualmente, usando el comando `mount`. Sin embargo, esto requiere que el usuario root teclee el comando `mount` cada vez que el sistema reinicie. Los

dos métodos de configurar las comparticiones NFS para que sean montadas automáticamente en el momento de arranque incluyen la modificación de `/etc/fstab` o el uso del servicio `autofs`.

9.3.1. `/etc/fstab`

Colocando una línea adecuadamente formada en el archivo `/etc/fstab` tiene el mismo efecto que el montaje manual del sistema de archivos exportado. El archivo `/etc/fstab` es leído por el script `/etc/rc.d/init.d/netfs` cuando arranca el sistema y cualquier compartición NFS listada será montada.

Un ejemplo de línea `/etc/fstab` para montar un NFS exportado será parecida a:

```
<server>:</path/of/dir> </local/mnt/point> nfs <options> 0 0
```

La opción *<server-host>* tiene que ver con el nombre de la máquina, dirección IP o nombre de dominio totalmente cualificado del servidor que exporta el sistema de archivos.

La opción *</path/of/directory>* es la ruta al directorio exportado.

La opción *</local/mount/point>* especifica dónde montar en el sistema de archivos local el directorio exportado. Este punto de montaje debe existir antes de que `/etc/fstab` sea leído o el montaje fallará.

La opción `nfs` especifica el tipo de sistema de archivos que esta siendo montado.

El área *<options>* especifica como el sistema de archivos es montado. Por ejemplo, si las opciones indican `rw,suid`, el sistema de archivos exportado será montado en modo de lectura-escritura y los ID de usuario y grupo puestos por el servidor serán usados. Aquí no se usan paréntesis. Para más opciones de montaje, vea Sección 9.3.3.

9.3.2. `autofs`

Una desventaja de usar `/etc/fstab` es que, sin tener en cuenta con que frecuencia se use este sistema de archivos montado, su sistema debe dedicar recursos para mantener este montaje en su sitio. Esto no es un problema con uno o dos montajes, pero cuando su sistema está manteniendo montajes a una docena de sistemas al mismo tiempo, el rendimiento global puede decaer. Una alternativa a `/etc/fstab` es usar la utilidad basada en el kernel `automount`, la cual monta y desmonta sistemas de archivos NFS automáticamente, ahorrando recursos.

El script `autofs`, localizado en `/etc/rc.d/init.d/`, es usado para controlar a `automount` a través del archivo de configuración primario `/etc/auto.master`. Mientras que `automount` puede ser especificado en la línea de comandos, es más conveniente especificar los puntos de montaje, nombres de máquinas, directorios exportados y opciones en un conjunto de archivos que teclearlo todo a mano. Ejecutando `autofs` como servicio que empieza y termina en sus niveles de ejecución designados, las configuraciones de montaje en los diferentes archivos pueden ser implementadas automáticamente.

Los archivos de configuración `autofs` están fijados en una relación padre-hijo. Un archivo principal de configuración (`/etc/auto.master`) se refiere a los puntos de montaje de su sistema que están enlazados a un particular *tipo de mapa*, el cual toma la forma de otros archivos de configuración, programas, mapas NIS y otros métodos de montaje menos comunes. El archivo `auto.master` contiene líneas referidas a cada punto de montaje, organizadas como:

```
<mount-point>
<map-type>
```

El elemento *<mount-point>* de esta línea indica la ubicación del montaje en el sistema de archivos local. *<map-type>* se relaciona con la forma en la cual el punto de montaje será montado. El método más común para el automontaje de las exportaciones NFS es usar un archivo como la guía para un punto de montaje particular. El mapa del archivo, usualmente llamado `auto.`*<mount-point>*,

donde <*mount-point*> es el punto de montaje designado en `auto.master`, contiene líneas que se ven similar a:

```
<directory>
<mount-options>
<host>:<exported-file-system>
```

<*directory*> se refiere al directorio dentro del punto de montaje donde el sistema de archivos exportado debe ser montado. Como en un comando estándar `mount`, la máquina que exporta el sistema de archivos y el sistema de archivos que está siendo exportado, son requeridos en la sección <*host*>:<*exported-file system*>. Para especificar las opciones particulares para montar un sistema de archivos exportado, colóquelas en la sección <*mount-options*>, separadas por comas. Para montajes NFS que usen `autofs`, coloque `-fstype=nfs` en la sección <*mount-options*>.

Mientras que los archivos de configuración `autofs` pueden ser usados por una variedad de montajes de muchos tipos de dispositivos y sistemas de archivos, son particularmente útiles para crear montajes NFS. Por ejemplo, algunas organizaciones guardan un directorio `/home/` en un servidor central a través de una compartición NFS. Entonces, configuran el archivo `auto.master` en cada una de las estaciones de trabajo para que apunten a un archivo `auto.home` que contiene como montar el directorio `/home/` vía NFS. Esto permite al usuario acceder a sus datos personales y archivos de configuración en su directorio `/home/` conectándose desde cualquier sitio de la red interna. El archivo `auto.master` en esta situación debería parecerse a:

```
/home   /etc/auto.home
```

Esto hace que el punto de montaje `/home/` en el sistema local sea configurado mediante el archivo `/etc/auto.home`, el cual se vería como:

```
*  -fstype=nfs,soft,intr,rsize=8192,wsize=8192,nosuid  server.example.com:/home
```

Esta línea establece que cualquier directorio que un usuario intente accesar bajo el directorio local `/home/` (debido al asterisco), debe resultar en un punto de montaje NFS en el sistema server.domain.com dentro del sistema de archivos exportado `/home/`. Las opciones de montaje especifican que cada directorio `/home/` montado via NFS debe usar una colección particular de opciones. Para más información de las opciones de montaje, incluyendo las usadas en este ejemplo, vea Sección 9.3.3.

9.3.3. Opciones de montaje NFS comunes

Aparte de montar un sistema de archivos via NFS en una máquina remota, existe un número de diferentes opciones que pueden ser especificadas en tiempo de montaje que pueden ser más fáciles de usar. Estas opciones pueden usarse con el comando manual `mount`, configuraciones `/etc/fstab`, `autofs` y otros métodos de montaje.

Las siguientes opciones son las más populares para montajes NFS:

- `hard` o `soft` — especifican si el programa que usa un archivo vía conexión NFS debe parar y esperar a que el servidor vuelva a estar en línea si la máquina que exporta ese sistema de archivos no está disponible (`hard`), o bien debe informar de un error (`soft`).

  Si se especifica la opción `hard`, el usuario no podrá parar el proceso que está esperando la comunicación NFS a menos que especifique la opción `intr`.

  Si usa `soft`, puede usar la opción adicional `timeo=`<*value*>, donde <*value*> especifica el número de segundos que deben pasar antes de informar del error.

- `intr` — permite a las peticiones NFS ser interrumpidas si el servidor se cae o no puede ser accedido.

- `nolock` — es requerido a veces cuando conectamos a servidores NFS antiguos. Para requerir el bloqueo, use la opción `lock`.
- `noexec` — no permite la ejecución de binarios en el sistema de archivos montado. Esto es útil si el sistema está montando un sistema de archivos no Linux a través de NFS que contiene binarios incompatibles.
- `nosuid` — no permite que los bits set-user-identifier o set-group-identifier tomen efecto.
- `rsize=8192` y `wsize=8192` — pueden acelerar la comunicación NFS tanto para leer (`rsize`) como para escribir (`wsize`), configurando un tamaño de bloque de datos mayor, en bytes, que serán transferidos de una sola vez. Tenga cuidado al cambiar estos valores; algunos kernels antiguos de Linux y tarjetas de red pueden no trabajar bien con grandes tamaños de bloques.
- `nfsvers=2` o `nfsvers=3` — especifica que versión del protocolo NFS usar.

Hay muchas más opciones en la página del manual de `mount`, incluyendo opciones para montar sistemas de archivos que no sean NFS.

9.4. Asegurar NFS

NFS trabaja muy bien compartiendo sistemas de archivos enteros con un gran número de máquinas conocidas de una manera muy transparente. Muchos usuarios que acceden a archivos sobre un punto de montaje NFS pueden no estar atentos a que el sistema de archivos que están usando no está en su sistema local. Sin embargo, esta facilidad de uso trae una variedad de potenciales problemas de seguridad.

Los puntos siguientes deberían ser considerados cuando se exporte sistemas de archivos NFS en un servidor o cuando se monten en un cliente. Haciendo esto reducirá los riesgos de seguridad NFS y protegerá mejor los datos en el servidor.

9.4.1. Acceso al sistema

NFS controla quien puede montar y exportar sistemas de archivos basados en la máquina que lo pide, no el usuario que utilizará el sistema de archivos. Las máquinas tienen que tener los derechos para montar los sistemas de archivos exportados explícitamente. El control de acceso no es posible para usuarios, aparte de los permisos de archivos y directorios. En otras palabras, cuando exporta un sistema de archivos vía NFS, cualquier usuario en cualquier máquina remota conectada al servidor NFS puede acceder a los datos compartidos. Para limitar estos riesgos potenciales, los administradores sólo pueden permitir acceso de sólo-lectura o reducir a los usuarios a un usuario común y groupid. Pero estas soluciones pueden impedir que la compartición NFS sea usada de la forma en que originalmente se pensó.

Adicionalmente, si un atacante gana el control del servidor DSN usado por el sistema que exporta el sistema de archivos NFS, el sistema asociado con un nombre de máquina concreto o nombre de dominio totalmente cualificado, puede ser dirigido a una máquina sin autorización. En este punto, la máquina desautorizada *es* el sistema que tiene permitido montar la compartición NFS, ya que no hay intercambio de información de nombre de usuario o contraseña para proporcional seguridad adicional al montaje NFS. Los mismos riesgos corre el servidor NIS, si los nombres de red NIS son usados para permitir a ciertas máquinas montar una compartición NFS. Usando direcciones IP en `/etc/exports`, esta clase de ataques son más difíciles.

Los comodines o metacaracteres deben ser usados lo menos posible cuando garantizamos el acceso a una compartición NFS. El uso de los comodines puede permitir el acceso a sistemas que puede no saber que existen y que no deberían montar el sistema de archivos.

Para más información sobre la seguridad en NFS, consulte el capítulo titulado *Seguridad del servidor* en el *Manual de seguridad de Red Hat Linux*.

9.4.2. Permisos de archivos

Una vez que el sistema de archivos es montado como lectura-escritura por una máquina remota, la única protección que tiene cada archivo son sus permisos. Si dos usuarios que comparten el mismo valor de userid montan el mismo NFS, ellos podran modificar sus archivos mutuamente. Adicionalmente, cualquiera con acceso root en el sistema cliente puede usar el comando `su -` para volverse un usuario que tenga acceso a determinados archivos a través de la compartición NFS. Para más detalles sobre los conflictos en NFS y userid, consulte el capítulo llamado *Administración de cuentas y grupos* en el *Manual de administración del sistema de Red Hat Linux*.

El comportamiento por defecto cuando se está exportando un sistema de archivos a través NFS es usar *root squashing*. Esto coloca el userid de cualquiera que esté accesando la compartición NFS como el usuario root en su máquina local al valor de la cuenta de 'nobody'. Nunca desactive el aplastamiento (squashing) de root.

Si se está exportando una compartición NFS como de sólo lectura, considere usar la opción `all_squash`, la cual hace que todos los usuarios accesando el sistema de archivos exportado tomen el userid del usuario nobody.

9.5. Recursos adicionales

Administrar un servidor NFS puede ser un desafío. Muchas opciones, incluyendo algunas no mencionadas en este capítulo, están disponibles para exportar sistemas de archivos NFS o montarlos como cliente. Consulte las siguientes fuentes de información para más detalles.

9.5.1. Documentación instalada

- `/usr/share/doc/nfs-utils-<version-number>/` — Reemplace *<version-number>* con el número de la versión del paquete NFS. Este directorio contiene una riqueza de información sobre la implementación NFS para Linux, incluyendo una vista a varias configuraciones NFS y su impacto en el rendimiento de la transferencia de archivos.
- `man mount` — Contiene una vista completa de las opciones de montaje para configuraciones tanto de servidor como de cliente NFS.
- `man fstab` — Otorga detalles para el formato del archivo `/etc/fstab` usado para montar sistemas de archivos en el momento de arranque.
- `man nfs` — Proporciona detalles de opciones de montaje y de exportación de sistemas de archivos específicos NFS.
- `man exports` — Muestra opciones comunes usadas en el archivo `/etc/exports` cuando exportamos sistemas de archivos NFS.

9.5.2. Libros relacionados

- *Managing NFS and NIS* por Hal Stern, Mike Eisler, y Ricardo Labiaga; O'Reilly & Associates — Es una guía de referencia excelente para las diferentes opciones NFS disponibles de montaje y exportación.
- *NFS Illustrated* por Brent Callaghan; Addison-Wesley Publishing Company — Proporciona comparaciones entre NFS y otros sistemas de archivos de red y muestra, en detalle, como las comunicaciones NFS funcionan.

Capítulo 10.

Servidor Apache HTTP

El Servidor Apache HTTP es un servidor Web de tecnología Open Source sólido y para uso comercial desarrollado por la Apache Software Foundation (http://www.apache.org). Red Hat Linux incluye el Servidor Apache HTTP versión 2.0 así como también una serie de módulos de servidor diseñados para mejorar la funcionalidad.

El archivo de configuración predeterminado instalado en el Servidor Apache HTTP funciona en la mayor parte de los casos. Este capítulo subraya cómo personalizar el archivo de configuración (`/etc/httpd/conf/httpd.conf`) de Servidor Apache HTTP para ayudar a aquellos que requieren una configuración personalizada o necesitan convertir un archivo de configuración del formato más antiguo del Servidor Apache HTTP 1.3.

Aviso

Si utiliza la **Herramienta de configuración de HTTP** (`redhat-config-httpd`), *no* cambie el archivo de configuración del Servidor Apache HTTP manualmente pues la **Herramienta de configuración de HTTP** vuelve a generar este archivo cada vez que se usa.

Si desea más información sobre la **Herramienta de configuración de HTTP**, consulte el capítulo *Configuración del Servidor Apache HTTP* en el *Manual de personalización de Red Hat Linux*.

10.1. Servidor Apache HTTP 2.0

Existen diferencias importantes entre el Servidor Apache HTTP versión 2.0 y la versión 1.3 (la versión 1.3 venía con Red Hat Linux 7.3 y versiones anteriores). Esta sección revisa algunas de las nuevas características de la versión 2.0 del Servidor Apache HTTP y subraya las principales diferencias. Si necesita migrar una versión 1.3 del archivo de configuración al formato 2.0, consulte Sección 10.2.

10.1.1. Características del Servidor Apache HTTP 2.0

La versión 2.0 del Servidor Apache HTTP contiene muchas características nuevas. Entre ellas se encuentran las siguientes:

- *Nuevos módulos Apache API* — se utiliza un nuevo conjunto de interfaces de aplicación (APIs).

Importante

Los módulos creados con la versión 1.3 del Servidor Apache HTTP no funcionan si no se llevan al nuevo API. Si no está seguro de si se ha llevado un determinado módulo, consulte el mantenedor de paquetes antes de la actualización.

- *Filtrado* — Los módulos pueden actuar como filtros de contenido. Consulte Sección 10.2.4 para mayor información.
- *Soporte para IPv6* — Se tiene soporte para la nueva generación de las direcciones IP.
- *Directivas simplificadas* — Se han eliminado una serie de directivas complicadas y otras se han simplificado. Consulte Sección 10.5 para mayor información sobre directivas específicas.

- *Respuestas a errores en diversos idiomas* — Cuando usa documentos *Server Side Include* (*SSI*), las páginas de errores personalizables se pueden entregar en diversos idiomas
- *Soporte a múltiples protocolos* — Se soportan diversos protocolos.

En el siguiente sitio web se muestra una lista completa de los cambios realizados: http://httpd.apache.org/docs-2.0/.

10.1.2. Cambios en los paquetes del Servidor Apache HTTP 2.0

A partir de la versión 8.0 de Red Hat Linux, los paquetes del Servidor Apache HTTP han sido renombrados. Además otros paquetes se han eliminado y otros se han introducido en otros paquetes.

La siguiente lista contiene los cambios de los paquetes:

- Los paquetes `apache`, `apache-devel` y `apache-manual` fueron renombrados a `httpd`, `httpd-devel` y `httpd-manual` repectivamente.
- El paquete `mod_dav` se ha incorporado al paquete `httpd`.
- Los paquetes `mod_put` y `mod_roaming` se han eliminado, ya que su funcionalidad aparece recogida en `mod_dav` (el cual forma parte ahora del paquete `httpd`).
- Los paquetes `mod_auth_any` y `mod_bandwidth` se han eliminado.
- El número de versión del paquete `mod_ssl` se ha sincronizado con el paquete `httpd`. Esto significa que el paquete `mod_ssl` para el Servidor Apache HTTP 2.0 tiene un número de versión *menor* que el paquete `mod_ssl` del Servidor Apache HTTP 1.3.

10.1.3. Cambios en el sistema de archivos de la versión 2.0 del Servidor Apache HTTP

Ocurren los siguientes cambios en la presentación del sistema de archivos cuando se actualiza a la versión 2.0 del Servidor Apache HTTP:

- *Se ha añadido un nuevo directorio de configuración `/etc/httpd/conf.d/`.* — Este nuevo directorio de usa para alamacenar archivos de configuración para módulos individuales como `mod_ssl`, `mod_perl` y `php`. Se instruye al servidor para que cargue archivos de configuración desde esta ubicación con la directiva `Include conf.d/*.conf` dentro del archivo de configuración del Servidor Apache HTTP, `/etc/httpd/conf/httpd.conf`.

Importante

Es fundamental que se introduzca esta línea cuando migre una configuración ya existente.

- *Se han trasladado los programas `ab` y `logresolve`.* — Estos programas se han trasladado desde el directorio `/usr/sbin/` al directorio `/usr/bin/`. Esto provoca que scripts con rutas absolutas para estos binarios fallen.
- *Se ha sustituido el comando `dbmmanage`.* — El comando `dbmmanage` ha sido reemplazado con `htdbm`. Consulte Sección 10.2.4.4 para mayor información.
- *Se ha cambiado el nombre del archivo de configuración `logrotate`.* — Se le ha dado el nombre de `/etc/logrotate.d/httpd`.

La siguiente sección explica cómo migrar al nuevo formato de la versión 2.0.

10.2. Migración de los archivos de configuración de la versión del Servidor Apache HTTP 1.3

Si ha actualizado desde Red Hat Linux 7.3 o una versión anterior en el cual el Servidor Apache HTTP ya se había instalado, entonces el nuevo archivo de configuración para el paquete Servidor Apache HTTP 2.0 esta instalado como `/etc/httpd/conf/httpd.conf.rpmnew` y la versión original 1.3 `httpd.conf` no se toca. Por supuesto, depende absolutamente de usted, si elije migrar a la nueva versión y migrar los viejos cambios o si usar el archivo ya existente y modificarlo para que se adapte; sin embargo, algunas partes del archivo se han cambiado más que otras y lo mejor es llegar a un punto intermedio. Los archivos de configuración para ambas versiones están divididos en tres secciones. El objetivo de este manual es sugerirle la manera más sencilla.

Si el archivo `/etc/httpd/conf/httpd.conf` es una versión modificada de la versión por defecto de Red Hat Linux y ha guardado una copia del original, entonces le será más fácil invocar el comando `diff`, como se muestra a continuación:

```
diff -u httpd.conf.orig httpd.conf | less
```

Este comando subraya los cambios realizados. Si no tiene una copia del archivo original, cójalo del paquete RPM usando los comandos `rpm2cpio` y `cpio`, como en el ejemplo siguiente:

```
rpm2cpio apache-<version-number>.i386.rpm | cpio -i --make
```

En el comando de arriba, sustituya *<version-number>* con el número de versión para el paquete `apache`.

Finalmente, es útil saber que el Servidor Apache HTTP tiene un modo de prueba para verificar si hay errores en la configuración. Para ello, escriba el siguiente comando:

```
apachectl configtest
```

10.2.1. Configuración del entorno a nivel global

La sección del entorno global del archivo de configuración contiene directivas que afectan a todas las áreas que cubre el Servidor Apache HTTP como por ejemplo el número de peticiones que puede recibir al mismo tiempo y las localizaciones de varios archivos que usa. Esta sección requiere un gran número de cambios comparado con las otras y por ello se recomienda que base esta sección en el archivo de configuración de la versión 2.0 del Servidor Apache HTTP y que migre sus configuraciones anteriores en el.

10.2.1.1. Selección de las interfaces y de los puertos a los que vincularse

Ya no existen las directivas `BindAddress` y `Port`; porque quedan recogidas en la directiva `Listen`.

Si tenía configurado el `Puerto 80` en el archivo de configuración de la versión 1.3, debe cambiarlo a `Listen 80` en el archivo de configuración 2.0. Si el valor del `Puerto` estaba configurado a un valor *diferente que 80*, tiene que poner el número del puerto a los contenidos de la directiva `ServerName`.

Por ejemplo, el siguiente ejemplo es una directiva de la versión 1.3 del Servidor Apache HTTP:

```
Port 123
ServerName www.example.com
```

Para migrar esta configuración a la versión 2.0 del Servidor Apache HTTP use la siguiente estructura:

```
Listen 123
ServerName www.example.com:123
```

Para mayor información sobre este tema, consulte la siguiente documentación en el sitio web de la Apache Software Foundation:

- `http://httpd.apache.org/docs-2.0/mod/mpm_common.html#listen`
- `http://httpd.apache.org/docs-2.0/mod/core.html#servername`

10.2.1.2. Regulación del tráfico de peticiones-respuestas del servidor

En la versión 2.0 del Servidor Apache HTTP se han creado un grupo de módulos llamados *Módulos de procesos múltiples* (*MPMs*), que se encargan de aceptar las peticiones y de dar procesos hijo para gestionarlos. A diferencia de otros módulos, el Servidor Apache HTTP solamente puede cargar un módulo del grupo MPM. Hay tres módulos MPM que incluye la versión 2.0: `prefork`, `worker` y `perchild`.

El comportamiento original del Servidor Apache HTTP 1.3 se ha convertido en `prefork` MPM. Actualmente solamente está disponible el MPM `prefork` en el Red Hat Linux aunque los otros estaran disponibles más adelante.

El MPM `prefork` acepta las mismas directivas que el Servidor Apache HTTP 1.3, por tanto las siguientes directivas se pueden migrar directamente:

- `StartServers`
- `MinSpareServers`
- `MaxSpareServers`
- `MaxClients`
- `MaxRequestsPerChild`

Para mayor información consulte la documentación en el sitio web de la Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mpm.html

10.2.1.3. Soporte del Dynamic Shared Object (DSO) (Objeto dinámico compartido)

Se tienen que realizar muchos cambios aquí, por eso se recomienda que para modificar la configuración del Servidor Apache HTTP 1.3 para pasar a la 2.0 se copie esta sección del archivo de configuración del Red Hat Linux Servidor Apache HTTP 2.0.

Aquellos que no deseen copiar la sección desde el tronco del Servidor Apache HTTP 2.0 deberían tomar en cuenta lo siguiente:

- Las directivas `AddModule` y `ClearModuleList` ya no existen. Estas directivas eran usadas para asegurar que se pudiesen activar los módulos en el orden correcto. La API del Servidor Apache HTTP 2.0 permite a los módulos especificar su orden, eliminando la necesidad de estas dos directivas.
- El orden de las líneas `LoadModule` ya no es relevante.
- Se han añadido muchos módulos, otros han sido eliminados, renombrado, dividido o incorporados con otros.
- Ya no son necesarias las líneas `LoadModule` para los módulos empaquetados en los propios RPMs (`mod_ssl`, `php`, `mod_perl` y similares) ya que se pueden encontrar en el directorio `/etc/httpd/conf.d/`.
- Las definiciones `HAVE_XXX` ya no existen.

Importante

Si se está modificando el archivo original, por favor tenga en cuenta que es de suma importancia que `httpd.conf` contenga la directiva siguiente:

```
Include conf.d/*.conf
```

La omisión de esta directiva podría resultar en la falla de todos los módulos enpaquetados en sus propios RPMs (tales como `mod_perl`, `php` y `mod_ssl`).

10.2.1.4. Otros cambios en el entorno global

Se han eliminado las siguientes directivas de la configuración del Servidor Apache HTTP 2.0:

- *`ServerType`* — El Servidor Apache HTTP se puede ejecutar solamente como `ServerType standalone` por lo que esta directiva es irrelevante.
- *`AccessConfig`* y *`ResourceConfig`* — Se han eliminado estas directivas porque su funcionalidad aparece ya en la directiva `Include`. Si las directivas `AccessConfig` y `ResourceConfig` son configuradas entonces reemplácelas por las directivas `Include`.

  Para asegurarse que estos archivos se lean en el orden de las antiguas directivas, las directivas `Include` se deberían colocar al final de `httpd.conf`, con la correspondiente a `ResourceConfig` precediendo la que corresponde a `AccessConfig`. Si se estan usando los valores por defecto, inclúyalos explícitamente como archivos `conf/srm.conf` y `conf/access.conf`.

10.2.2. Configuración del servidor principal

La sección de la configuración del servidor principal del archivo de configuración configura el servidor principal que responde a todas aquellas peticiones que no maneja un host virtual definido dentro de un contenedor <`VirtualHost`>. Los valores aquí también proporcionan valores por defecto para cualquier contenedor <`VirtualHost`> definido.

Las directivas de esta sección han cambiado ligeramente respecto a las de la versión 1.3. Si la configuración del servidor principal está fuertemente personalizada le será fácil modificar la configuración existente para que se adapte a la versión 2.0 del Servidor Apache HTTP. Los usuarios con secciones del servidor principal ligeramente personalizadas deberían migrar sus cambios al archivo de configuración de Apache 2.0 por defecto.

10.2.2.1. Asignaciones UserDir

La directiva `UserDir` se usa para permitir asignaciones de URLs tales como `http://example.com/~bob/` a subdirectorios dentro del directorio principal del usuario `bob`, tal como `/home/bob/public_html`. Un efecto secundario de esta característica es que un atacante potente puede determinar si un nombre de usuario está en el sistema, por esta razón la configuración por defecto para Servidor Apache HTTP 2.0 desactiva esta directiva.

Para habilitar la asignación de `UserDir`, cambie la directiva en `httpd.conf` desde:

```
UserDir disable
```

a lo siguiente:

```
UserDir public_html
```

Para mayor información sobre este tema, consulte la documentación en el sitio de la Apache Software Foundation, http://httpd.apache.org/docs-2.0/mod/mod_userdir.html#userdir.

10.2.2.2. Conexión

Se han eliminado las siguientes directivas de conexión:

- `AgentLog`
- `RefererLog`
- `RefererIgnore`

Sin embargo, las conexiones agent y referrer estan disponibles usando las directivas `CustomLog` y `LogFormat`.

Para mayor información, consulte los siguientes sitios web de la Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_log_config.html#customlog
- http://httpd.apache.org/docs-2.0/mod/mod_log_config.html#logformat

10.2.2.3. Índice de directorios

Se ha eliminado la directiva `FancyIndexing`. La misma funcionalidad se encuentra ahora en `FancyIndexing` *option* dentro de la directiva `IndexOptions`.

La nueva opción `VersionSort` para la directiva `IndexOptions` causa que los archivos conteniendo números de versiones sean ordenados de una forma más natural. Por ejemplo, `httpd-2.0.6.tar` aparece antes de `httpd-2.0.36.tar` en una página de índices de directorio.

Las directivas predeterminadas `ReadmeName` y `HeaderName` han sido cambiadas desde `README` y `HEADER` a `README.html` y `HEADER.html`.

Para mayor información sobre este tema, consulte los siguientes sitios web de la Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_autoindex.html#indexoptions
- http://httpd.apache.org/docs-2.0/mod/mod_autoindex.html#readmename
- http://httpd.apache.org/docs-2.0/mod/mod_autoindex.html#headername

10.2.2.4. Negociación de contenido

La directiva `CacheNegotiatedDocs` toma ahora el argumento `on` o `off`. Las instancias existentes de `CacheNegotiatedDocs` deberían ser cambiadas con `CacheNegotiatedDocs on`.

Para mayor información sobre este tema, refiérase a la documentación siguiente en los sitios web de la Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_negotiation.html#cachenegotiateddocs

10.2.2.5. Documentos de error

Para usar un mensaje codificado con la directiva `ErrorDocument`, el mensaje tiene que aparecer en un par de dobles comillas ["], en vez de estar simplemente precedido por las comillas como en el Servidor Apache HTTP 1.3.

Para migrar la configuración de `ErrorDocument` a Servidor Apache HTTP 2.0, use la siguiente estructura:

```
ErrorDocument 404 "The document was not found"
```

Observe que se han puesto las dobles comillas en el ejemplo anterior.

Para mayor información sobre este tema, consulte la documentación siguiente en los sitios web de la Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/core.html#errordocument

10.2.3. Configuración de las máquinas virtuales

Los contenidos de los contenedores `<VirtualHost>` se tienen que migrar de la misma manera que en la sección del servidor principal como se describió en Sección 10.2.2.

Importante

Observe que la configuración de las máquinas virtuales SSL/TLS se han quitado del archivo de configuración del servidor principal al archivo `/etc/httpd/conf.d/ssl.conf`.

Para mayor información, consulte el capítulo llamado *Configuración del Servidor HTTP de Apache* en el *Manual de personalización de Red Hat Linux* y la documentación en línea en el siguiente URL:

- http://httpd.apache.org/docs-2.0/vhosts/

10.2.4. Módulos y el Servidor Apache HTTP 2.0

En la versión 2.0 del Servidor Apache HTTP, el sistema de módulos se ha cambiado para permitir que los módulos se encadenen o se combinen en maneras nuevas e interesantes. Los scripts CGI (*Common Gateway Interface*), por ejemplo, pueden generar documentos HTML interpretados por el servidor que luego pueden ser procesados por `mod_include`. Esto abre una gran cantidad de posibilidades en lo que respecta a cómo los módulos pueden combinarse para llevar a cabo una meta determinada.

La forma en que esto funciona es que cada petición es servida por exáctamente un módulo *handler* seguido por cero o más módulos *filtro*.

Bajo el Servidor Apache HTTP 1.3, por ejemplo, un script PHP es manejado completamente por el módulo PHP. En la versión 2.0 del Servidor Apache HTTP, la petición la gestiona inicialmente el módulo principal — que sirve archivos estáticos — y que es luego *filtrado* por el módulo PHP.

Exactamente cómo se lleva esto a cabo y otras de las nuevas características del Servidor Apache HTTP 2.0, estan más allá del ámbito de este documento; sin embargo, el cambio tiene ramificaciones si ha usado la directiva `PATH_INFO`, que contiene información del recorrido después del nombre del archivo verdadero, en un documento que se gestiona con un módulo que se usa como filtro. EL módulo principal no entiende por defecto `PATH_INFO` y devuelve la petición como error `404 Not Found`.

Puede usar la directiva `AcceptPathInfo` como alternativa para obligar al módulo principal a que acepte peticiones con `PATH_INFO`.

A continuación se presenta un ejemplo de esta directiva:

```
AcceptPathInfo on
```

Para mayor información sobre este tema, revise los documentos siguientes en los sitios web de la Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/core.html#acceptpathinfo
- http://httpd.apache.org/docs-2.0/handler.html
- http://httpd.apache.org/docs-2.0/filter.html

10.2.4.1. El módulo `mod_ssl`

La configuración del módulo `mod_ssl` ahora está en el archivo `/etc/httpd/conf.d/ssl.conf`. Para cargar este archivo y hacer que `mod_ssl` funcione, tiene que tener la declaración `Include conf.d/*.conf` en `httpd.conf` como se describe en Sección 10.2.1.3.

Las directivas `ServerName` en las máquinas virtuales SSL tienen que especificar el número del puerto.

Por ejemplo, este es un ejemplo de la directiva de la versión 1.3 del Servidor Apache HTTP:

```
<VirtualHost _default_:443>
    # General setup for the virtual host
    ServerName ssl.example.name
    ...
</VirtualHost>
```

Para migrar esta configuración a la versión 2.0, use la siguiente estructura:

```
<VirtualHost _default_:443>
    # General setup for the virtual host
    ServerName ssl.host.name:443
    ...
</VirtualHost>
```

Para mayor información, refiérase a la documentación siguiente en los sitios web de la Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_ssl.html
- http://httpd.apache.org/docs-2.0/vhosts/

10.2.4.2. El módulo `mod_proxy`

Las declaraciones de control del acceso proxy se encuentran ahora en el bloque `<Proxy>` en vez de en `<Directory proxy:>`.

La funcionalidad de caché del antiguo `mod_proxy` se ha dividido en tres módulos siguientes:

- `mod_cache`
- `mod_disk_cache`
- `mod_file_cache`

Estos generalmente usan las mismas directivas o similares que las versiones anteriores del módulo `mod_proxy`.

Para mayor información, consulte los siguientes sitios web de la Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_proxy.html

10.2.4.3. El módulo `mod_include`

El módulo `mod_include` es ahora implementado como un filtro y por tanto se activa de una forma diferente. Consulte Sección 10.2.4 para más información sobre filtros.

Por ejemplo, a continuación se muestra una directiva del Servidor Apache HTTP 1.3:

```
AddType text/html .shtml
AddHandler server-parsed .shtml
```

Para cambiar esta configuración al Servidor Apache HTTP 2.0, use la estructura siguiente:

```
AddType text/html .shtml
AddOutputFilter INCLUDES .shtml
```

Observe que como antes, la directiva `Options +Includes` es aún requerida para el contenedor `<Directory>` o en el archivo `.htaccess`.

Para mayor información, consulte la documentación en los siguientes sitios web de la Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_include.html

10.2.4.4. Los módulos `mod_auth_dbm` y `mod_auth_db`

El Servidor Apache HTTP 1.3 soportaba dos módulos de autenticación, `mod_auth_db` y `mod_auth_dbm`, que usaba las bases de datos Berkeley y las DBM respectivamente. Estos módulos se han combinado en un único módulo que se llama `mod_auth_dbm` en el Servidor Apache HTTP 2.0, que puede acceder a las diferentes bases de datos. Para migrar desde `mod_auth_db`, los archivos de configuración se tienen que ajustar reemplazando `AuthDBUserFile` y `AuthDBGroupFile` con los equivalentes: `mod_auth_dbm`: `AuthDBMUserFile` y `AuthDBMGroupFile`. También, se debe añadir la directiva `AuthDBMType DB` para indicar el tipo de archivo de base de datos en uso.

El ejemplo siguiente muestra una configuración `mod_auth_db` de ejemplo para el Servidor Apache HTTP 1.3:

```
<Location /private/>
  AuthType Basic
  AuthName "My Private Files"
  AuthDBUserFile /var/www/authdb
  require valid-user
</Location>
```

Para migrar esta configuración a la versión 2.0 del Servidor Apache HTTP, use la estructura siguiente:

```
<Location /private/>
  AuthType Basic
  AuthName "My Private Files"
  AuthDBMUserFile /var/www/authdb
  AuthDBMType DB
  require valid-user
```

```
</Location>
```

Observe que la directiva `AuthDBMUserFile` también puede ser usada en archivos `.htaccess`.

El script Perl `dbmmanage`, usado para manipular bases de datos de nombres de usuarios y contraseñas, ha sido reemplazado por `htdbm` en Servidor Apache HTTP 2.0. El programa `htdbm` ofrece una funcionalidad equivalente y como `mod_auth_dbm` puede operar en una variedad de formatos de bases de datos; la opción `-T` se puede usar en la línea de comandos para especificar el formato a utilizar.

Tabla 10-1 muestra cómo migrar desde un formato de base de datos DBM al formato `htdbm` usando `dbmmanage`.

Acción	comando dbmmanage (1.3)	comando equivalente htdbm (2.0)
Añade un usuario a la base de datos (usando la contraseña dada)	`dbmmanage authdb add username password`	`htdbm -b -TDB authdb username password`
Añade un usuario a la base de datos (le pide la contraseña)	`dbmmanage authdb adduser username`	`htdbm -TDB authdb username`
Eliminar el usuario de la base de datos	`dbmmanage authdb delete username`	`htdbm -x -TDB authdb username`
Listar usuarios en la base de datos	`dbmmanage authdb view`	`htdbm -l -TDB authdb`
Verificar una contraseña	`dbmmanage authdb check username`	`htdbm -v -TDB authdb username`

Tabla 10-1. Migración del `dbmmanage` a `htdbm`

Las opciones `-m` y `-s` trabajan con `dbmmanage` y con `htdbm`, permitiendo el uso de los algortimos MD5 o SHA1 para las contraseñas hashing, respectivamente.

Cuando cree una nueva base de datos con `htdbm`, use la opción `-c`.

Para mayor información, consulte los siguientes sitios web de la Apache Software Foundation:

- http://httpd.apache.org/docs-2.0/mod/mod_auth_dbm.html

10.2.4.5. El módulo `mod_perl`

La configuración del módulo `mod_perl` se ha pasado al archivo `/etc/httpd/conf.d/perl.conf`. Para cargar este archivo, y hacer funcionar `mod_perl`, se debe incluir la declaración `Include conf.d/*.conf` en el `httpd.conf` como se describe en Sección 10.2.1.3.

Las ocurrencias del `Apache::` en el `httpd.conf` tienen que ser sustituídas por `ModPerl::`. Además se ha cambiado el modo en que los gestores se graban.

Ejemplo de configuración del módulo `mod_perl` en el Servidor Apache HTTP 1.3:

```
<Directory /var/www/perl>
    SetHandler perl-script
    PerlHandler Apache::Registry
    Options +ExecCGI
</Directory>
```

Este es el equivalente del `mod_perl` para el Servidor Apache HTTP 2.0:

```
<Directory /var/www/perl>
    SetHandler perl-script
    PerlModule ModPerl::Registry
    PerlHandler ModPerl::Registry::handler
    Options +ExecCGI
</Directory>
```

La mayoría de los módulos para `mod_perl` 1.x deberían funcionar sin modificación con los módulos `mod_perl` 2.x. Los módulos XS requieren recompilación y quizás algunas modificaciones menores de Makefile.

10.2.4.6. El módulo `mod_python`

La configuración para `mod_python` ha sido movida desde `httpd.conf` al archivo `/etc/httpd/conf.d/python.conf`. Para que se cargue este archivo y por tanto funcione `mod_python`, se debe incluir la declaración `Include conf.d/*.conf` en el `httpd.conf` como se describe en Sección 10.2.1.3.

10.2.4.7. PHP

La configuración del PHP ha sido movida de `httpd.conf` al archivo `/etc/httpd/conf.d/php.conf`. Para cargar este archivo, tiene que tener la declaración `Include conf.d/*.conf` en `httpd.conf` tal y como se describe en Sección 10.2.1.3.

El PHP es ahora implementado como un filtro y tiene que ser habilitado en una manera diferente. Consulte la Sección 10.2.4 para mayor información.

Bajo el Servidor Apache HTTP 1.3, PHP era implementado usando las directivas siguientes:

```
AddType application/x-httpd-php .php
AddType application/x-httpd-php-source .phps
```

Bajo el Servidor Apache HTTP 2.0, utilice las directivas siguientes:

```
<Files *.php>
    SetOutputFilter PHP
    SetInputFilter PHP
</Files>
```

En PHP 4.2.0 y en las versiones sucesivas, el conjunto predeterminado de las variables predefinidas que están disponibles en el ámbito global han cambiado. Las entradas individuales y las variables del servidor ya no se colocan directamente en el ámbito global. Este cambio puede hacer que se rompan los scripts. Tiene que invertir al antiguo comportamiento poniendo `register_globals` a `On` en el archivo `/etc/php.ini`.

Para mayor información sobre estos temas, consulte los siguientes sitios web:

- http://www.php.net/release_4_1_0.php

10.3. Después de la instalación

Tras haber instalado el paquete `httpd`, podrá encontrar la documentación sobre el Servidor Apache HTTP instalando el paquete `httpd-manual` y apuntando el navegador de web al

sitio http://localhost/manual/ o puede navegar en la documentación de Apache en línea en http://httpd.apache.org/docs-2.0/.

La documentación del Servidor Apache HTTP contiene una lista completa con descripciones detalladas de todas las opciones de configuración. Para ayudarle, este capítulo contiene pequeñas descripciones de las directivas de configuración que usa la versión 2.0 del Servidor Apache HTTP.

La versión del Servidor Apache HTTP incluida en Red Hat Linux ofrece la posibilidad de configurar servidores Web seguros con la función robusta de encriptación SSL propocionada por los paquetes `mod_ssl` y `openssl`. Cuando lea los archivos de configuración, tenga en cuenta que la configuración predeterminada incluye tanto el servidor de web seguro como el no seguro. El servidor seguro se ejecuta como una máquina virtual, que aparece configurada en el archivo `/etc/httpd/conf.d/ssl.conf`. Para más información sobre máquinas virtuales, vea Sección 10.8. Para mayor información sobre la configuración de un servidor virtual seguro, vea Sección 10.8.1. Para información sobre la configuración del Servidor seguro HTTP de Apache vea el capítulo llamado *Configuración del servidor seguro HTTP de Apache* en el *Manual de personalización de Red Hat Linux*.

Nota

Red Hat, Inc. no se distribuye con extensiones de FrontPage pues la licencia de Microsoft™ prohibe la inclusión de estas extensiones en un producto de terceros. Para obtener más información sobre las extensiones de Frontpage y el Servidor Apache HTTP, consulte: http://www.rtr.com/fpsupport/.

10.4. Arrancar y detener `httpd`

El RPM de `httpd` instala el script `/etc/rc.d/init.d/httpd`, el cual se puede accesar usando el comando `/sbin/service`.

Para iniciar el servidor, como root escriba:

```
/sbin/service httpd start
```

Para detener el servidor, como root escriba:

```
/sbin/service httpd stop
```

La opción `restart` es un truco para detener y luego arrancar el Servidor Apache HTTP.

Para reiniciar el servidor, como root escriba:

```
/sbin/service httpd restart
```

Nota

Si está usando el Servidor Apache HTTP como un servidor seguro se le pedirá la contraseña del servidor siempre que use las opciones `start` o `restart`.

Sin embargo, luego de modificar el archivo `httpd.conf`, no es necesario que explícitamente detenga e inicie el servidor. Para esto use la opción `reload`.

Para volver a cargar el archivo de configuración, como usuario root escriba:

```
/sbin/service httpd reload
```

Nota

Si está ejecutando el Servidor Apache HTTP como un servidor seguro, la contraseña del servidor *no* se necesita cuando esté usando la opción `reload`.

Por defecto, el servicio `httpd` *no* se iniciará automáticamente en el momento de arranque. Puede configurar el servicio `httpd` para que se inicie en el momento de arranque usando una utilidad de tipo initscript, tal como `/sbin/chkconfig`, **/sbin/ntsysv** o la **Herramienta de configuración de servicios**. Refiérase al capítulo llamado *Control de acceso a servicios* en *Manual de personalización de Red Hat Linux* para más información sobre estas herramientas.

Nota

Si esta ejecutando el Servidor Apache HTTP como un servidor seguro, se le pedirá la contraseña del servidor seguro después que la máquina arranca, a menos que se haya especificado un tipo de clave especial para el servidor.

Para más información sobre la configuración de un servidor seguro HTTP de Apache consulte el capítulo llamado *Configuración de un servidor HTTP de Apache* en el *Manual de personalización de Red Hat Linux*.

10.5. Directivas de configuración en `httpd.conf`

El archivo de configuración del Servidor Apache HTTP es `/etc/httpd/conf/httpd.conf`. El archivo `httpd.conf` está bien comentado y es bastante autoexplicativo. Su configuración por defecto funciona para la mayoría de los casos; sin embargo, quizás quiera conocer el resto de las opciones de configuración más importantes.

Aviso

Con la versión 2.0 del Servidor Apache HTTP, han cambiado muchas opciones de configuración. Si necesita migrar de la versión 1.3 al nuevo formato, consulte Sección 10.2.

10.5.1. Sugerencias de configuración generales

Si necesita configurar Servidor Apache HTTP sólo tiene que modificar el archivo `/etc/httpd/conf/httpd.conf` y después recargar o bien apagar y arrancar el proceso del comando `httpd` como se describe en Sección 10.4.

Antes de modificar el archivo `httpd.conf`, primero haga una copia del archivo original. Si crea una copia de respaldo, podrá recuperar el sistema de posibles errores cometidos antes al editar el nuevo archivo de configuración.

Si comete un error y su servidor de web no funciona correctamente, lo primero que debe realizar es revisar lo que lo que acaba de modificar en `httpd.conf` para ver si no hay errores de transcripción.

Después consulte el archivo de registro de errores del servidor web, `/var/log/httpd/error_log`. Este puede ser difícil de interpretar, todo depende del nivel de experiencia. Si acaba de tener problemas, de todas formas, las últimas entradas deberían de ayudarle a saber lo que ha pasado.

Las siguientes secciones proporcionan una breve descripción de las directivas incluídas en el archivo `httpd.conf`. Estas descripciones no estan completas. Para más información, consulte la documentación de Apache proporcionada en formato HTML en http://localhost/manual/ o en línea en la siguiente dirección: http://httpd.apache.org/docs-2.0/.

Para más información sobre las directivas `mod_ssl`, refiérase a la documentación incluida en formato HTML en http://localhost/mod/mod_ssl.html o en línea en: http://httpd.apache.org/docs-2.0/mod/mod_ssl.html.

10.5.2. `ServerRoot`

La directiva `ServerRoot` especifica el directorio de nivel superior que tiene el contenido web. Por defecto, `ServerRoot` está configurado a `"/etc/httpd"` para servidores seguros y no seguros.

10.5.3. `ScoreBoardFile`

`ScoreBoardFile` almacena información de procesos internos del servidor, la cual es usada para la comunicación entre el servidor padre y sus procesos hijos. Red Hat Linux utiliza memoria compartida para almacenar el `ScoreBoardFile`, el valor por defecto de `/etc/httpd/logs/apache_runtime_status` sólo es usado como un apoyo.

10.5.4. `PidFile`

`PidFile` nombra el archivo en el que el servidor graba su ID de proceso (pid). Por defecto, el PID es listado en `/var/run/httpd.pid`.

10.5.5. `Timeout`

`Timeout` define, en segundos, el tiempo que el servidor esperará para recibir y enviar peticiones durante la comunicación. Específicamente, `Timeout` define cuánto esperará el servidor para recibir peticiones GET, cuánto esperará para recibir paquetes TCP en una petición POST o PUT y cuánto esperará entre ACKs respondiendo a otros paquetes TCP. `Timeout` está configurado por defecto a `300`, lo cual es apropiado para la mayoría de las situaciones.

10.5.6. `KeepAlive`

`KeepAlive` determina si el servidor permitirá más de una petición por conexión y se puede usar para prevenir a un cliente consumir demasiados recursos del servidor.

Por defecto `Keepalive` está configurado a `off`. Si `Keepalive` está en `on` y el servidor se vuelve muy ocupado, este puede rápidamente generar el máximo número de procesos hijos. En esta situación, el servidor se volverá significativamente lento. Si se activa `Keepalive`, es una buena idea configurar el `KeepAliveTimeout` a un valor bajo (consulte Sección 10.5.8 para más información sobre la directiva `KeepAliveTimeout`) y controle el archivo de registro `/var/log/httpd/error_log` en el servidor. Este registro informa cuando el servidor se esta quedando corto de procesos hijos.

10.5.7. MaxKeepAliveRequests

Esta directiva establece el número máximo de peticiones permitidas por cada conexión persistente. El Proyecto Apache recomienda un valor alto, lo que mejoraría el rendimiento del servidor. El valor predeterminado de `MaxKeepAliveRequests` es de `100` que debería bastar en la mayoría de los casos.

10.5.8. KeepAliveTimeout

La directiva `KeepAliveTimeout` establece el número de segundos que el servidor esperará a la siguiente petición, tras haber dado servicio a una petición, antes de cerrar la conexión. Una vez recibida la petición, se aplica la directiva `Timeout` en su lugar. `KeepAliveTimeout` está configurado a 15 segundos por defecto.

10.5.9. MinSpareServers y MaxSpareServers

El Servidor Apache HTTP se adapta dinámicamente a la carga percibida manteniendo un número apropiado de procesos de servidores libres basado en el tráfico. El servidor comprueba el número de servidores que esperan peticiones y elimina algunos si el número es más alto que `MaxSpareServers` o crea algunos si el número de servidores es menor que `MinSpareServers`.

El valor predeterminado de `MinSpareServers` es `5` y el de `MaxSpareServers` es `20`. Estos valores predeterminados son suficientes en la mayoría de los casos. El número de `MinSpareServers` no debería de ser elevado ya que creará una gran carga incluso cuando el tráfico fuese bajo.

10.5.10. StartServers

`StartServers` establece cuántos procesos serán creados al arrancar. Ya que el servidor Web crea y elimina dinámicamente servidores según el tráfico, no se necesitará cambiar este parámetro. El servidor está configurado para arrancar ocho procesos al arrancar.

10.5.11. MaxClients

La directiva `MaxClients` establece un límite al total de los procesos del servidor (o clientes conectados simultáneamente) que se pueden ejecutar a la vez. El propósito principal de esta directiva es prevenir que un Servidor Apache HTTP descontrolado vuelva inestable al sistema operativo. Para los servidores muy ocupados este valor se debería colocar a un valor alto. El valor por defecto es 150. No se recomienda que el valor del comando `MaxClients` supere `256`.

10.5.12. MaxRequestsPerChild

`MaxRequestsPerChild` establece el número máximo de peticiones que cada proceso hijo procesa antes de morir. La principal razón para configurar `MaxRequestsPerChild` es evitar que procesos de larga vida gasten memoria. El valor predeterminado de `MaxRequestsPerChild` para el servidor es de `1000`.

10.5.13. Listen

El comando `Listen` identifica los puertos en los que el servidor Web aceptará las peticiones entrantes. Por defecto, el Servidor Apache HTTP está configurado para escuchar en el puerto 80 para comunicaciones Web no seguras y (en `/etc/httpd/conf.d/ssl.conf` el cual define cualquier servidor seguro) en el puerto 443 para comunicaciones seguras.

Si el Servidor Apache HTTP está configurado para escuchar a un puerto por debajo del 1024, se necesita al usuario root para iniciarlo. Para los puertos 1024 y superiores, `httpd` puede ser arrancado por cualquier usuario.

La directiva `Listen` también se puede usar para especificar direcciones IP particulares sobre las cuales el servidor aceptará conexiones.

10.5.14. `Include`

`Include` permite que se incluyan otros archivos de configuración en el tiempo de ejecución.

La ruta a estos archivos de configuración pueden ser absolutas o relativas con respecto al `ServerRoot`.

Importante

Para que el servidor use módulos de paquetes individuales, como `mod_ssl`, `mod_perl` y `php`, la siguiente directiva tiene que estar en `Section 1: Global Environment` del `httpd.conf`:

```
Include conf.d/*.conf
```

10.5.15. `LoadModule`

`LoadModule` es usada para cargar módulos Dynamic Shared Object (DSO). Se puede encontrar más información sobre el soporte del Servidor Apache HTTP para DSO, incluyendo exáctamente cómo utilizar la directiva `LoadModule`, en Sección 10.7. Observe, que ya *no es importante* el orden en que se cargan estos módulos con el Servidor Apache HTTP 2.0. Consulte Sección 10.2.1.3 para más información sobre el soporte DSO del Servidor Apache HTTP 2.0.

10.5.16. `ExtendedStatus`

La directiva `ExtendedStatus` controla si Apache generará información básica del estado del servidor (`off`) o detallada (`on`), cuando el manejador `server-status` es llamado. El manejador `Server-status` es llamado usando etiquetas de `Location`. Se incluye más información sobre `server-status` en Sección 10.5.63.

10.5.17. `IfDefine`

Las etiquetas `<IfDefine>` y `</IfDefine>` envuelven directivas de configuración que son aplicadas si el "test" establecido en la etiqueta `<IfDefine>` es verdadero. Las directivas no se tienen en cuenta si el test es falso.

El test en las etiquetas `<IfDefine>` es un nombre de parámetro (por ejemplo, `HAVE_PERL`). Si el parámetro está definido, es decir, si se da como argumento al comando de arranque del servidor, entonces el test es verdadero. En este caso, cuando se arranca el servidor Web, el test es verdadero y se aplican las directivas contenidas en las etiquetas `IfDefine`.

Por defecto, las etiquetas `<IfDefine HAVE_SSL>` rodean las etiquetas de la máquina virtual para el servidor seguro. Las etiquetas `<IfDefine HAVE_SSL>` también rodean las directivas `LoadModule` y `AddModule` para el `ssl_module`.

10.5.18. `User`

La directiva `User` establece el nombre de usuario para el proceso del servidor y determina qué archivos puede accesar el servidor. Cualquier archivo que no esté accesible a este usuarip tampoco estará disponible para los clientes del Servidor Apache HTTP.

Por defecto `User` es configurado a `apache`.

Nota

Por razones de seguridad, el Servidor Apache HTTP se negará a ejecutarse como el usuario root.

10.5.19. `Group`

Especifica el nombre del grupo de los procesos Servidor Apache HTTP.

Por defecto `Group` está configurado a `apache`.

10.5.20. `ServerAdmin`

Configure la directiva `ServerAdmin` a la dirección de correo electrónico del administrador del servidor Web. Esta dirección de correo aparecerá en los mensajes de error en las páginas generadas por el servidor Web, de tal manera que los usuarios pueden comunicar errores enviando correo al administrador.

Por defecto, `ServerAdmin` es configurado a `root@localhost`.

Una forma típica de configurar `ServerAdmin` es situarlo en la dirección `webmaster@ejemplo.com`. Después cree un alias del `webmaster` para la persona responsable del servidor Web en `/etc/aliases` y ejecute `/usr/bin/newaliases`.

10.5.21. `ServerName`

Use la directiva `ServerName` para configurar un nombre de servidor y un número de puerto (que coincida con la directiva `Listen`) para el servidor. El `ServerName` no necesita coincidir con el nombre real de la máquina. Por ejemplo, el servidor Web puede ser `www.example.com` pero el nombre del servidor es en realidad `foo.example.com`. El valor especificado en `ServerName` debe ser un nombre de Domain Name Service válido (DNS) que pueda ser resuelto por el sistema — no invente algo.

Lo siguiente es una directiva `ServerName` de ejemplo:

```
ServerName www.example.com:80
```

Cuando especifique un `ServerName`, asegúrese de que el par de la dirección IP y el nombre del servidor esten incluidos en el archivo `/etc/hosts`.

10.5.22. `UseCanonicalName`

Cuando se configure esta directiva a `on`, se está indicando al Servidor Apache HTTP a que se referencie asímismo usando el valor especificado en las directivas `ServerName` y `Port`. Cuando `UseCanonicalName` es configurada a `off`, el servidor usará el valor usado por el cliente solicitante cuando se refiera a el.

`UseCanonicalName` está configurada a `off` por defecto.

10.5.23. `DocumentRoot`

`DocumentRoot` es el directorio que contiene la mayoría de los archivos HTML que se entregarán en respuesta a peticiones. El directorio predeterminado `DocumentRoot` para servidores seguros y no seguros es `/var/www/html`. Por ejemplo, el servidor puede recibir una petición para el siguiente documento:

```
http://example.com/foo.html
```

El servidor busca por el archivo siguiente en el directorio por defecto:

```
/var/www/html/foo.html
```

Si se quiere cambiar `DocumentRoot` para que no lo compartan los servidores seguros y no seguros, vea Sección 10.8.

10.5.24. `Directory`

Las etiquetas `<Directory /path/to/directory>` y `</Directory>` se usan para crear lo que se conoce como un *contenedor* y se usan para agrupar un grupo de directivas de configuración que sólo se aplican a ese directorio y sus subdirectorios. Cualquier directiva aplicable a un directorio puede usarse en las etiquetas `<Directory>`.

Por defecto, se aplican parámetros muy restrictivos al directorio raíz (`/`), utilizando las directivas `Options` (consulte Sección 10.5.25) y `AllowOverride` (vea Sección 10.5.26). Con esta configuración, cualquier directorio del sistema que necesite valores más permisivos ha de ser configurado explícitamente con esos valores.

En la configuración por defecto, otro contenedor `Directory` es configurado para el `DocumentRoot` el cual asigna parámetros menos rígidos al árbol del directorio para que el Servidor Apache HTTP pueda accesar archivos que residan allí.

El contenedor `Directory` también se puede usar para configurar directorios adicionales `cgi-bin` para las aplicaciones del servidor fuera del directorio especificado en la directiva `ScriptAlias` (consulte a Sección 10.5.44 para más información sobre esta directiva `ScriptAlias`).

Para lograr esto, el contenedor `Directory` debe configurar la opción `ExecCGI` para ese directorio.

Por ejemplo, si los scripts CGI están localizados en `/home/my_cgi_directory`, añada el contenedor siguiente `Directory` al archivo `httpd.conf`:

```
<Directory /home/my_cgi_directory>
    Options +ExecCGI
</Directory>
```

Luego, necesitará anular el comentario de la directiva `AddHandler` para identificar archivos con la extensión `.cgi` como scripts CGI. Consulte Sección 10.5.59 para saber cómo configurar el `AddHandler`.

Para que esto funcione, los permisos para los scripts CGI y la ruta completa a los scripts, se deben colocar a 0755.

10.5.25. `Options`

La directiva `Options` controla características del servidor que están disponibles en un directorio en particular. Por ejemplo, en los parámetros restrictivos especificados para el directorio raíz, el comando `Options` sólo permite `FollowSymLinks`. No hay características activadas, salvo que el servidor puede seguir enlaces simbólicos en el directorio raíz.

Por defecto, el directorio `DocumentRoot`, `Options` está configurado para incluir `Indexes` y `FollowSymLinks`. `Indexes` permite al servidor generar un listado de un directorio si no se especifica el `DirectoryIndex` (por ejemplo, `index.html`) es especificado. `FollowSymLinks` permite al servidor seguir enlaces simbólicos en ese directorio.

Nota

Las declaraciones `Options` desde la sección de configuración del servidor principal necesita ser replicado a cada contenedor `VirtualHost` individualmente. Refiérase a Sección 10.5.69 para más información sobre los contenedores `VirtualHost`.

10.5.26. `AllowOverride`

La directiva `AllowOverride` indica si puede o no sobreescribir `Options` por las declaraciones en un archivo `.htaccess`. Por defecto, tanto el directorio raíz como `DocumentRoot` están configurados para no permitir la sobreescritura `.htaccess`.

10.5.27. `Order`

La directiva `Order` controla el orden en el cual las directivas `allow` y `deny` son evaluadas. El servidor es configurado para evaluar las directivas `Allow` antes de las directivas `Deny` para el directorio `DocumentRoot`.

10.5.28. `Allow`

`Allow` especifica cual solicitante pueda accesar a un directorio dado. El solicitante puede ser `all`, un nombre de dominio, una dirección IP, una dirección IP parcial, un par de red/máscara de la red, etc. El directorio `DocumentRoot` esta configurado para `Allow` (permitir) a `all`, es decir, que todos tienen acceso.

10.5.29. `Deny`

`Deny` funciona igual que `Allow`, excepto que especifica a quién se le niega el acceso. `DocumentRoot` no es configurado para negar `Deny` peticiones a ninguno por defecto.

10.5.30. `UserDir`

`UserDir` es el nombre del subdirectorio dentro del directorio de cada usuario dónde estarán los archivos HTML personal que serán servidos por el servidor de Web. Esta directiva esta configurada por defecto a `disable`.

El nombre para el subdirectorio esta configurado a `public_html` en la configuración por defecto. Por ejemplo, el servidor puede recibir la siguiente petición:

```
http://example.com/~username/foo.html
```

El servidor buscará por el archivo:

```
/home/username/public_html/foo.html
```

En el ejemplo de arriba, `/home/username/` es el directorio principal del usuario (observe que la ruta por defecto al directorio principal del usuario puede variar).

Hay que asegurarse que los permisos de los directorios de usuario sean correctos. El valor de los permisos deben ser de 0711. Los bits de lectura (r) y ejecución (x) deben estar activados en el directorio del usuario `public_html` (0755 también funcionará). El valor de los permisos con que se servirán los archivos desde `public_html` debe ser 0644 por lo menos.

10.5.31. `DirectoryIndex`

`DirectoryIndex` es la página por defecto que entrega el servidor cuando hay una petición de índice de un directorio especificado con una barra (/) al final del nombre del directorio.

Por ejemplo, cuando un usuario pide la página http://*example*/*this_directory*/, recibe la página `DirectoryIndex` si existe, o un listado generado por el servidor. El valor por defecto para `DirectoryIndex` es `index.html` y el tipo de mapa `index.html.var`. El servidor intentará encontrar cualquiera de estos archivos y entregará el primero que encuentre. Si no encuentra ninguno de estos archivos y `Options Indexes` esta configurado para ese directorio, el servidor genera y devuelve una lista, en formato HTML, de los subdirectorios y archivos del directorio, a menos que la característica de listar directorios esté desactivada.

10.5.32. `AccessFileName`

`AccessFileName` denomina el archivo que el servidor utilizará para información de control de acceso en cada directorio. Por defecto, el servidor utilizará `.htaccess`.

Justo tras `AccessFileName`, un conjunto de indicadores de `Files` aplican el control de acceso a cualquier archivo comenzando con un `.ht`. Estas directivas niegan el acceso Web a cualquier archivo `.htaccess` (o otros archivos que comiencen con `.ht`) por razones de seguridad.

10.5.33. `CacheNegotiatedDocs`

Por defecto, el servidor Web requiere a los "proxies" que no hagan caché de los documentos que se negocian en base al contenido (pueden cambiar en el tiempo o según la entrada de quien los solicita). Si se configura `CacheNegotiatedDocs` a `on`, se desactiva la función y se permite a los servidores proxy hacer caché de los documentos.

10.5.34. `TypesConfig`

`TypesConfig` nombra el archivo que configura la lista por defecto de asignaciones tipo MIME (extensiones de nombres de archivo a tipos de contenido). El archivo predeterminado `TypesConfig` es `/etc/mime.types`. En vez de modificar el `/etc/mime.types`, la forma recomendada de añadir asignaciones de tipo MIME es usando la directiva `AddType`.

Para más información sobre `AddType`, refiérase a Sección 10.5.58.

10.5.35. DefaultType

DefaultType establece el tipo de contenido por defecto que el servidor utilizará para documentos cuyos tipos MIME no puedan ser determinados. Por defecto es text/plain.

10.5.36. IfModule

Las etiquetas <IfModule> y </IfModule> crean un contenedor condicional que sólo es activado si el módulo especificado es cargado. Las directivas contenidas entre etiquetas IfModule son procesadas bajo una de dos condiciones. Las directivas son procesadas si el módulo entre la etiqueta de comienzo <IfModule> es cargado. O, si un símbolo de exclamación [!] aparece antes del nombre del módulo, las directivas son procesadas sólo si el módulo especificado en la etiqueta <IfModule> *no* es cargado.

Para más información sobre los módulos del Servidor Apache HTTP, refiérase a Sección 10.7.

10.5.37. HostnameLookups

HostnameLookups se puede configurar a on, off o double. Si se configura HostnameLookups a on, el servidor automáticamente resuelve las direcciones IP para cada conexión. Resolver las direcciones IP significa que el servidor hace una o más conexiones a un servidor DNS, añadiendo sobrecarga por procesamiento. Si HostnameLookups es configurado a double, el servidor realiza búsquedas inversa doble añadiendo aún más sobrecarga.

Para ahorrar recursos en el servidor, HostnameLookups es configurado a off por defecto.

Si se requieren nombres de host en los archivos de registro, considere ejecutar una de los muchas herramientas de análisis de log que llevan a cabo las búsquedas de DNS de forma mucho más eficiente y por montones cuando se este rotando los archivos de log del servidor Web.

10.5.38. ErrorLog

ErrorLog especifica el archivo donde se guardan los errores del servidor . Por defecto, esta directiva es configurada a /var/log/httpd/error_log.

10.5.39. LogLevel

LogLevel establece que tantos detalles tendrán los registros de mensajes de error. LogLevel se puede configurar (desde el que tiene menos detalles a los más detallados) a emerg, alert, crit, error, warn, notice, info o debug. El valor predeterminado de LogLevel es warn.

10.5.40. LogFormat

La directiva LogFormat pone el formato para los archivos de registro del servidor Web. El comando LogFormat usado en realidad depende de la configuración dada en la directiva CustomLog (consulte Sección 10.5.41).

Las siguientes son las opciones de formato si la directiva CustomLog es configurada a combined:

%h (dirección IP del host remoto o nombre de la máquina)

Lista la dirección IP de la máquina remota del cliente solicitante. Si HostnameLookups es configurada a on, el nombre de máquina del cliente es registrado a menos que no este disponible desde el DNS.

`%l` (rfc931)

No se usa. Un guión [-] aparece en el campo de registro para este campo.

`%u` (usuario autenticado)

Si se requiere autenticación, lista el nombre del usuario registrado. Usualmente, esto no se usa, por tanto aparece un guión [-] en el archivo de registro para este campo.

`%t` (fecha)

Lista la fecha y hora de la solicitud.

`%r` (cadena de la solicitud)

Lista la cadena de la solicitud exactamente como viene del navegador o cliente.

`%s` (estado)

Lista el estado de código HTTP el cual fue devuelto al host cliente.

`%b` (bytes)

Lista el tamaño del documento.

`%\"%{Referer}i\"` (referencia)

Lista la dirección URL de la página web que refiere el máquina cliente al servidor Web.

`%\"%{User-Agent}i\"` (agente usuario)

Lista el tipo de navegador Web que está realizando la solicitud.

10.5.41. `CustomLog`

`CustomLog` identifica el archivo de registro y su formato. Por defecto, el registro es guardado al archivo `/var/log/httpd/access_log`.

El formato por defecto `CustomLog` es `combined`. Lo siguiente ilustra el formato del archivo de registro `combined`:

```
remotehost rfc931 user date "request" status bytes referrer user-agent
```

10.5.42. `ServerSignature`

La directiva `ServerSignature` añade una línea que contiene la versión del Servidor Apache HTTP y el `ServerName` para cualquier documento generado por el servidor, tales como mensajes de error devueltos a los clientes. Por defecto `ServerSignature` está configurada a `on`.

También se puede configurar a `off` o a `EMail`. `EMail`, agrega una etiqueta HTML `mailto:ServerAdmin` a la línea de la firma de las respuestas autogeneradas.

10.5.43. `Alias`

La directiva `Alias` permite que haya directorios fuera del `DocumentRoot` a los que puede acceder el servidor. Cualquier URL que termine en el alias será automáticamente traducido a la ruta del alias. Por defecto, ya existe un alias configurado para un directorio `icons`. El servidor web puede acceder al directorio `icons` pero el directorio no está en `DocumentRoot`.

10.5.44. `ScriptAlias`

La directiva `ScriptAlias` define dónde pueden encontrarse los scripts CGI (u otros scripts). Normalmente, no es una buena idea colocar los scripts CGI dentro de `DocumentRoot`. Si los scripts CGI se encontrasen en `DocumentRoot`, podrían, potencialmente, ser considerados como documentos de texto. Por esta razón, la directiva `ScriptAlias` diseña un directorio especial fuera del directorio `DocumentRoot` para contener ejecutables del servidor y scripts. Este directorio es conocido como un `cgi-bin` y se configura por defecto a `/var/www/cgi-bin/`.

Es posible establecer directorios para almacenar ejecutables fuera del directorio `cgi-bin`. Para más instrucciones sobre cómo hacer esto, refiérase a Sección 10.5.59 y Sección 10.5.24.

10.5.45. `Redirect`

Cuando se mueve una página web, se puede utilizar `Redirect` para crear asignaciones de la ubicación del archivo a un nuevo URL. El formato es como sigue:

```
Redirect /<old-path>/<file-name> http://<current-domain>/<current-path>/<file-name>
```

En este ejemplo, sustituya *<old-path>* con la vieja información de la ruta por *<file-name>* y *<current-domain>* y *<current-path>* con el dominio actual y la información de la ruta para *<file-name>*.

En este ejemplo, cualquier petición *<file-name>* en la vieja ubicación será redirigida automáticamente a la nueva ubicación.

Para técnicas más avanzadas de redireccionamiento, use el módulo `mod_rewrite` incluido con el Servidor Apache HTTP. Para más información sobre la configuración del módulo `mod_rewrite`, refiérase a la documentación de la Apache Software Foundation en http://httpd.apache.org/docs-2.0/mod/mod_rewrite.html.

10.5.46. `IndexOptions`

`IndexOptions` controla la apariencia de los listados generados por el servidor, al añadir iconos y texto descriptivo, etc. Si `Options Indexes` está configurado (see Sección 10.5.25), el servidor Web server genera un listado de directorio cuando el servidor Web recibe una petición HTTP para un directorio sin un índice.

Primero el servidor Web busca en el directorio solicitado un archivo que coincida los nombres listados en la directiva `DirectoryIndex` (usualmente, `index.html`). Si el servidor no encuentra un archivo `index.html`, el Servidor Apache HTTP genera un listado del directorio en HTML. La apariencia del listado de este directorio es controlada, en parte, por la directiva `IndexOptions`.

La configuración predeterminada activa `FancyIndexing`. Esto significa que un usuario puede reordenar un listado de directorio haciendo click en las cabeceras de columnas. Otro click en la misma cabecera cambiará del orden ascendente al descendente. `FancyIndexing` también muestra iconos diferentes para diferentes archivos, basados en las extensiones de archivos.

La opción `AddDescription`, cuando se utiliza junto con `FancyIndexing`, presenta una descripción corta para el archivo en los listados de directorios generados por el servidor.

`IndexOptions` tiene otros parámetros que pueden activarse para controlar la apariencia de los listados. Los parámetros incluyen `IconHeight` y `IconWidth`, para hacer que el servidor incluya etiquetas HTML `HEIGHT` y `WIDTH` para los iconos en las páginas generadas por el servidor; `IconsAreLinks`, hace que los iconos actúen como parte del enlace HTML junto con el nombre del archivo, y otros.

10.5.47. AddIconByEncoding

Esta directiva denomina qué iconos se mostrarán con los archivos según su codificación MIME, en los listados de directorio. Por ejemplo, por defecto, el servidor muestra el icono `compressed.gif` junto a archivos con codificación MIME x-compress y x-gzip en los listados de directorio.

10.5.48. AddIconByType

Esta directiva denomina qué iconos se mostrarán con los archivos con codificación MIME, en los listados del directorio. Por ejemplo, por defecto, el servidor muestra el icono `text.gif` junto a archivos con tipo MIME `text` en los listados del directorio.

10.5.49. AddIcon

`AddIcon` dice al servidor qué icono mostrar en los listados del directorio para ciertos tipos de archivos según la extensión. Por ejemplo, el servidor Web muestra el icono `binary.gif` para archivos con extensiones `.bin` o `.exe`.

10.5.50. DefaultIcon

`DefaultIcon` especifica el icono desplegado en el listado generado por el servidor para archivos que no tienen otro icono especificado. El archivo de imagen por defecto es `unknown.gif`.

10.5.51. AddDescription

Cuando utilice `FancyIndexing` como un parámetro de `IndexOptions`, la directiva `AddDescription` se puede usar para mostrar descripciones especificadas por el usuario para ciertos archivos o tipos de archivo en un listado de directorio generado por el servidor. La directiva `AddDescription` soporta el listado de archivos específicos, expresiones con comodines o extensiones de archivos.

10.5.52. ReadmeName

La directiva `ReadmeName` determina el archivo (si existe dentro del directorio) que se adjuntará a los listados de los directorios. El servidor Web intentará primero incluirlo como documento HTML y luego como texto. El valor predeterminado de `ReadmeName` es `README.html`.

10.5.53. HeaderName

La directiva `HeaderName` dicta el archivo (si existe dentro del directorio) que se antepondrá al comienzo de los listados de los directorios. Al igual que con `ReadmeName`, el servidor intentará incluirlo como documento HTML si es posible, o en caso contrario, como texto.

10.5.54. IndexIgnore

`IndexIgnore` lista las extensiones de archivo, los nombres de los archivos parciales, las expresiones con comodines o los nombres completos. El servidor Web no incluirá los archivos que encajen en estos patrones en los listados de directorios.

10.5.55. AddEncoding

La directiva AddEncoding dice qué extensiones especifican un tipo particular de codificación. AddEncoding se puede usar para decirle a los navegadores (no a todos) que descompriman ciertos archivos mientras los descargan.

10.5.56. AddLanguage

La directiva AddLanguage asocia extensiones a contenidos específicos de idiomas. Esta directiva es útil para la negociación de contenidos, cuando el Servidor Apache HTTP devuelve contenidos en diferentes idiomas dependiendo de la configuración del idioma del navegador Web.

10.5.57. LanguagePriority

La directiva LanguagePriority permite dar la prioridad para diferentes idiomas en caso de que el navegador Web no especique la preferencia de idioma.

10.5.58. AddType

Utilice la directiva AddType para definir parejas de tipo MIME y sus extensiones. Por ejemplo, si usa el PHP4, utilice AddType para que el servidor Web reconozca archivos con extensiones PHP (.php4, .php3, .phtml y .php) como tipos MIME PHP. La directiva siguiente le indica al Servidor Apache HTTP que reconozca la extensión de archivo .shtml:

```
AddType text/html .shtml
AddHandler server-parsed .shtml
```

10.5.59. AddHandler

La directiva AddHandler mapea extensiones de archivos a manejadores específicos. Por ejemplo, el manejador cgi-script puede mapearse con la extensión .cgi para que automáticamente trate a cualquier archivo con un nombre que termine en .cgi como un script CGI. A continuación se presenta un ejemplo de una directiva AddHandler para la extensión .cgi.

```
AddHandler cgi-script .cgi
```

Esta directiva habilita a CGIs fuera del cgi-bin a que funcionen en cualquier directorio en el servidor que tengan la opción ExecCGI dentro del contenedor de directorios. Refiérase a Sección 10.5.24 para más información sobre la configuración de la opción ExecCGI para un directorio.

Además de los scripts CGI, la directiva AddHandler es usada para procesar archivos de mapas de imagen y HTML analizados por el servidor.

10.5.60. Action

Action especifica parejas tipo contenido MIME y script CGI, para que cuando un archivo de ese tipo de media sea solicitado, se ejecute un script CGI particular.

10.5.61. `ErrorDocument`

La directiva `ErrorDocument` asocia un código de respuesta HTTP con un mensaje o un URL para que sea devuelto al cliente. Por defecto, el servidor Web produce una salida simple de mensaje de error cuando ocurre alguno. La directiva `ErrorDocument` obliga a que el servidor Web envie una salida de mensaje personalizado o redirije al cliente a un URL local o externo.

Importante

Para que el mensaje sea válido se *debe* rodear en un par de dobles comillas ["].

10.5.62. `BrowserMatch`

La directiva `BrowserMatch` permite al servidor definir variables de entorno y/o tomar acciones según sea el campo de cabecera User-Agent del HTTP, que identifica el tipo de navegador Web del cliente. Por defecto, el servidor usa `BrowserMatch` para denegar la conexión a navegadores con problemas conocidos y para desactivar "keepalives" y vaciados de cabecera de HTTP para navegadores que se sabe tienen problemas con acciones de ese tipo.

10.5.63. `Location`

Las etiquetas `<Location>` y `</Location>` permiten crear un contenedor en el cual se puede especificar el control de acceso basado en URL.

Por ejemplo, para permitir a personas conectarse desde dentro del dominio del servidor para ver informes de estado, utilice las directivas siguientes:

```
<Location /server-status>
    SetHandler server-status
    Order deny,allow Deny from all
    Allow from <.example.com>
</Location>
```

Reemplace *<.example.com>* con el nombre de dominio de segundo nivel para el servidor Web.

Para proporcionar informes de configuración del servidor (incluyendo los módulos instalados y las directivas de configuración) a peticiones desde dentro del dominio, utilice las siguientes directivas:

```
<Location /server-info>
    SetHandler server-info
    Order deny,allow
    Deny from all
    Allow from <.example.com>
</Location>
```

Una vez más, reemplace *<.example.com>* con el nombre del dominio de segundo nivel para el servidor Web.

10.5.64. `ProxyRequests`

Para configurar el Servidor Apache HTTP para que funcione como un servidor proxy, elimine las marcas hash del comienzo de la línea `<IfModule mod_proxy.c>` para cargar el módulo `mod_proxy` y configurar la directiva `ProxyRequests` a `On`.

10.5.65. `Proxy`

Las etiquetas `<Proxy *>` y `</Proxy>` crean un contenedor el cual envuelve un grupo de directivas de configuración solamente para aplicar al servidor proxy. Muchas directivas las cuales son permitidas dentro del contenedor `<Directory>` pueden también ser usadas dentro del contenedor `<Proxy>`.

10.5.66. `ProxyVia`

La directiva `ProxyVia` controla si se envía HTTP Via: junto con peticiones o respuestas que vayan vía el servidor proxy Apache. Via: header mostrará el nombre de la máquina si `ProxyVia` está en `On`, muestra el nombre de máquina y la versión del Servidor Apache HTTP para `Full`, y cualquier línea Via: se enviará sin cambiar si está ProxyVia está en `Off`, y las líneas Via: serán eliminadas si está en `Block`.

10.5.67. Directivas Cache

Hay varias directivas de caché suministradas por el archivo de configuración del Servidor Apache HTTP. En la mayoría de los casos, al quitar el comentario de estas líneas mediante la eliminación del símbolo [#] del principio de la línea es suficiente. Lo siguiente, sin embargo, es una lista de algunas de las directivas relacionadas a caché más importantes.

- `CacheRoot` — pone el nombre del directorio que contiene archivos de caché. El valor predeterminado de `CacheRoot` es el directorio `/var/httpd/proxy/`.
- `CacheSize` — establece cuánto espacio puede usar el caché, en KB. El valor predeterminado de `CacheSize` es `5` KB.
- `CacheGcInterval` — establece el número de horas que deben pasar antes de que los archivos en el caché sean borrados. El valor por defecto para `CacheGcInterval` es `4` horas.
- `CacheMaxExpire` — Especifica cuanto tiempo se conservan los documentos HTML (sin una recarga desde el servidor Web original) en el caché. El valor por defecto es `24` horas.
- `CacheLastModifiedFactor` — Especifica la creación de una fecha de vencimiento para documentos que no venían con caducidad desde el servidor de origen. El valor predeterminado de `CacheLastModifiedFactor` está configurado a `0.1`, es decir que la fecha de vencimiento para tales documentos es igual a un décimo de la cantidad de tiempo desde la última vez que se modificó el documento.
- `CacheDefaultExpire` — Especifica el tiempo de caducidad en horas para un documento que fue recibido usando un protocolo que no soporta fechas de vencimiento. El valor por defecto es configurado a `1` hora.
- `NoCache` — Especifica una lista de máquinas cuyos contenidos no está cacheado.

10.5.68. `NameVirtualHost`

La directiva `NameVirtualHost` asocia una dirección IP y número de puerto, si es necesario, para cualquier máquina virtual basada en nombres. El hospedaje virtual basado en nombres permite a un Servidor Apache HTTP servir a dominios diferentes sin usar múltiples direcciones IP.

Nota

Los hosts virtuales basados en nombre *only* funcionan con conexiones HTTP no seguras. Si está usando host virtuales con un servidor seguro, use host virtuales basados en direcciones IP.

Para habilitar el hospedaje basado en nombres, quite los comentarios de la directiva de configuración `NameVirtualHost` y añada la dirección IP correcta. Luego añada más contenedores `VirtualHost` para cada host virtual.

10.5.69. `VirtualHost`

Las etiquetas `<VirtualHost>` y `</VirtualHost>` crean un contenedor mostrando las características de un host virtual. El contenedor `<VirtualHost>` acepta la mayoría de las directivas de configuración.

Un conjunto de contenedores `VirtualHost` comentados se proporciona en `httpd.conf`, el cual ilustra el mínimo conjunto de directivas de configuración necesarias para cada host virtual. Refiérase Sección 10.8 para más información sobre los host virtuales.

Nota

Todos los contenedores de host virtuales SSL han sido movidos al archivo `/etc/httpd/conf.d/ssl.conf`.

10.5.70. Directivas de configuración SSL

Las directivas SSL en el archivo `/etc/httpd/conf.d/ssl.conf` se pueden configurar para activar las comunicaciones Web seguras usando SSL y TLS.

10.5.70.1. `SetEnvIf`

`SetEnvIf` configura las variables del entorno basado en las cabeceras de las conexiones seguras entrantes. En el archivo `/etc/httpd/conf.d/ssl.conf`, se utiliza para desactivar el keepalive del HTTP y permitir SSL para cerrar la conexión sin una alerta de notificación desde el navegador del cliente. Esta configuración es necesaria para ciertos navegadores que no cierran de forma confiable la conexión SSL.

Para más información sobre las directivas SSL, apunte su navegador Web a la siguiente dirección:

- http://localhost/manual/mod/mod_ssl.html
- http://httpd.apache.org/docs-2.0/mod/mod_ssl.html

Para más información sobre la configuración de un Servidor seguro HTTP de Apache vea el capítulo llamado *Configuración de un Servidor HTTP de Apache* en el *Manual de personalización de Red Hat Linux*.

Nota

En la mayoría de los casos, las directivas SSL son configuradas apropiadamente cuando se instalan. Tenga cuidado cuando altere las directivas de Servidor seguro HTTP de Apache pues un error en la configuración puede provocar que el servidor sea vulnerable en términos de seguridad.

10.6. Módulos predeterminados

El Servidor Apache HTTP es distribuido con un número de módulos. Por defecto los siguientes módulos son instalados y manejados con el paquete `httpd` en Red Hat Linux:

```
mod_access
mod_auth
mod_auth_anon
mod_auth_dbm
mod_auth_digest
mod_include
mod_log_config
mod_env
mod_mime_magic
mod_cern_meta
mod_expires
mod_headers
mod_usertrack
mod_unique_id
mod_setenvif
mod_mime
mod_dav
mod_status
mod_autoindex
mod_asis
mod_info
mod_cgi
mod_dav_fs
mod_vhost_alias
mod_negotiation
mod_dir
mod_imap
mod_actions
mod_speling
mod_userdir
mod_alias
mod_rewrite
mod_proxy
mod_proxy_ftp
mod_proxy_http
mod_proxy_connect
```

Adicionalmente, los módulos siguientes están disponibles instalando los paquetes adicionales:

```
mod_auth_mysql
mod_auth_pgsql
mod_perl
mod_python
mod_ssl
php
```

```
squirrelmail
```

10.7. Añadir módulos

El Servidor Apache HTTP soporta *Objetos compartidos dinámicamente* (Dynamically Shared Objects, *DSO*s) o módulos, los cuales se pueden cargar fácilmente en el momento de ejecución.

El Proyecto Apache proporciona Documentación DSO completa en línea http://httpd.apache.org/docs-2.0/dso.html. Si el paquete `http-manual` está instalado, se puede encontrar documentación sobre DSOs en http://localhost/manual/mod/.

Para que el Servidor Apache HTTP utilice un DSO, debe estar especificado en una directiva `LoadModule` dentro de `/etc/httpd/conf/httpd.conf`; si el módulo es proporcionado por un paquete separado, la línea debe aparecer dentro del archivo de configuración de módulos en el directorio `/etc/httpd/conf.d/`. Refiérase a Sección 10.5.15 para más información sobre la directiva `LoadModule`.

Si está añadiendo o eliminando módulos desde `http.conf`, el Servidor Apache HTTP se debe volver recargar o volver a iniciar, como se explica en Sección 10.4.

Si está creando un nuevo módulo, instale primero el paquete `httpd-devel` pues contiene los archivos include, las cabeceras de archivos así como también la aplicación *APache eXtenSion* (`/usr/sbin/apxs`), la cual utiliza los archivos include y las cabeceras para compilar DSOs.

Después de escribir un módulo, utilice `/usr/sbin/apxs` compilar las fuentes del módulo fuera del árbol de fuentes Apache. Para más información sobre el uso del comando `/usr/sbin/apxs`, vea la documentación de Apache en línea en http://httpd.apache.org/docs-2.0/dso.html y la página man de `apxs`.

Una vez compilado, coloque el módulo en el directorio `/usr/lib/httpd/`. Luego añada una línea `LoadModule` al archivo `httpd.conf`, usando la estructura siguiente:

```
LoadModule <module-name> <path/to/module.so>
```

En el ejemplo de arriba, cambie *<module-name>* al nombre del módulo y *<path/to/module.so>* a la ruta del DSO.

10.8. Máquinas Virtuales

La característica incorporada del Servidor Apache HTTP de máquinas virtules permite al servidor servir diferente información basado en cual dirección IP, nombre de host o puerto está siendo solicitado. Un manual completo para el uso de hosts virtuales está disponible en http://httpd.apache.org/docs-2.0/vhosts/.

10.8.1. Configuración de máquinas virtuales

Para crear una máquina virtual basada en nombre, lo mejor es utilizar el contenedor de la máquina virtual proporcionado en `httpd.conf` como un ejemplo.

El ejemplo de máquina virtual se lee como sigue:

```
#NameVirtualHost *
#
#<VirtualHost  *>
#    ServerAdmin webmaster@dummy-host.example.com
#    DocumentRoot /www/docs/dummy-host.example.com
#    ServerName dummy-host.example.com
#    ErrorLog logs/dummy-host.example.com-error_log
```

```
#    CustomLog logs/dummy-host.example.com-access_log common
#</VirtualHost>
```

Para activar máquinas virtuales basadas en nombre, quite los comentarios de la línea `NameVirtualHost` eliminando el símbolo de numeral o almohadilla (`#`) y reemplazando el asterísco (`*`) con la dirección IP asignada a la máquina.

Luego, configure un host virtual, quitando los comentarios y personalizando el contenedor `<VirtualHost>`.

En la línea `<VirtualHost>`, cámbie el asterísco (`*`) a la dirección IP del servidor. Cambie el `ServerName` al nombre DNS *válido* asignado a la máquina y configure las otras directivas si es necesario.

El contenedor `<VirtualHost>` es altamente personalizable y acepta casi cada directiva dentro de la configuración del servidor principal.

Sugerencia

Si se está configurando un host virtual para que escuche en un puerto no predeterminado, se debe agregar ese puerto a la directiva `Listen` en la sección de configuraciones globales del archivo `/etc/httpd/conf/http.conf`.

Para activar una máquina virtual creada recientemente, el Servidor Apache HTTP se debe volver a cargar o reiniciar. Consulte Sección 10.4 para ver las instrucciones sobre como hacer esto.

Se proporciona información completa sobre la creación y configuración de máquinas virtuales basadas en nombre y en dirección IP en http://httpd.apache.org/docs-2.0/vhosts/.

10.8.2. La máquina virtual del servidor Web seguro

Por defecto, el Servidor Apache HTTP es configurado como ambos, un servidor seguro e inseguro. Ambos servidores, seguro e inseguro utilizan la misma dirección IP y nombre de host, pero escuchan en puertos diferentes: 80 y 443 respectivamente. Esto permite que se puedan tener comunicaciones seguras e inseguras simultáneamente.

Un aspecto de las transmisiones HTTP mejoradas SSL es que estas utilizan más recursos que el protocolo estándar HTTP, por lo que un servidor seguro no puede servir tantas páginas por segundo. Por esta razón, usualmente es una buena idea minimizar la información disponible desde un servidor seguro, especialmente en un sitio Web con bastante tráfico.

Importante

No utilice un host virtual basado en nombre en conjunto con servidor Web seguro pues el handshake SSL ocurre antes de que la petición HTTP identifique el host virtual basado en nombre apropiado. Las máquinas virtuales basadas en nombre sólo funcionan con los servidores Web inseguros.

Las directivas de configuración para el servidor seguro son contenidas dentro de etiquetas en el archivo `/etc/httpd/conf.d/ssl.conf`.

Por defecto, los servidores Web seguros e inseguros comparten el mismo `DocumentRoot`. Se recomienda que el `DocumentRoot` sea diferente para el servidor Web seguro.

Para prevenir que el servidor Web inseguro acepte conexiones coloque entre comentarios la línea en `httpd.conf` que muestra `Listen 80` colocando un símbolo de numeral o almohadilla al comienzo de la línea. Cuando termine se verá como en el ejemplo siguiente:

```
#Listen 80
```

Para más información sobre la configuración de un servidor Web con mejoras SSL, consulte el capítulo llamado *Configuración del servidor HTTP de Apache* en el *Manual de personalización de Red Hat Linux*. Para sugerencias más avanzadas, refiérase a la documentación disponible en línea de la Apache Software Foundation en los siguientes URLs:

- http://httpd.apache.org/docs-2.0/ssl/.
- http://httpd.apache.org/docs-2.0/vhosts/

10.9. Recursos adicionales

Para aprender más sobre el Servidor Apache HTTP, refiérase a los siguientes recursos:

10.9.1. Sitios Web útiles

- http://httpd.apache.org — El sitio oficial para el Servidor Apache HTTP con documentación sobre todas las directivas y módulos por defecto.
- http://www.modssl.org — El sitio oficial para `mod_ssl`.
- http://www.apacheweek.com — Información semanal completa sobre todas las cosas de Apache.

10.9.2. Libros relacionados

- *Apache Desktop Reference* por Ralf S. Engelschall; Addison Wesley — Escrito por Ralf Engelschall, miembro de ASF y autor de `mod_ssl`, el *Apache Desktop Reference* proporciona una guía de referencia concisa pero completa para el uso del Servidor Apache HTTP durante la compilación, configuración y el momento de ejecución. Este libro está disponible en línea en http://www.apacheref.com/.
- *Professional Apache* por Peter Wainwright; Wrox Press Ltd — *Professional Apache* es de la serie Wrox Press Ltd's "Programmer to Programmer" y está dirigido a administradores de servidores Web novatos y experimentados.
- *Administering Apache* por Mark Allan Arnold; Osborne Media Group — Este libro está dirigido para los Proveedores de servicio de Internet que deseen proporcionar servicios más seguros.
- *Apache Server Unleashed* por Richard Bowen, et al; SAMS BOOKS — Una fuente enciclopédica para el Servidor Apache HTTP.
- *Apache Pocket Reference* por Andrew Ford, Gigi Estabrook; O'Reilly — Esta es la última adición a la serie O'Reilly Pocket Reference.

redhat.

Capítulo 11.

Correo electrónico

El nacimiento del correo electrónico (*email*) ocurrió a principios de los años 60. El buzón era un archivo en el directorio principal de un usuario que sólo podía ser accesado por ese usuario. Las aplicaciones de correo primitivas anexaban nuevos mensajes de texto a la parte inferior de un archivo, y el usuario tenía que buscar a lo largo del archivo en constante crecimiento para encontrar un mensaje particular. Este sistema sólo era capaz de enviar mensajes a usuarios del mismo sistema.

La primera transferencia verdadera de correo electrónico en la red se llevó a cabo en 1971 cuando un ingeniero de computación llamado Ray Tomlinson envión un mensaje de prueba entre dos máquinas a través de ARPANET — el precursor de Internet. La comunicación a través de correo electrónico rápidamente se volvió muy popular, pasando a formar el 75 por ciento del tráfico de ARPANET en menos de dos años.

Hoy día, los sistemas de correo electrónico basados en protocolos de red han evolucionado en uno de los servicios más usados de la Internet. Red Hat Linux ofrece muchas aplicaciones avanzadas para servir y accesar correo electrónico.

En este capítulo se analizan los protocolos de correo electrónico modernos conocidos actualmente, así como varios programas diseñados para realizar distintos tipos de tareas relacionadas con el correo electrónico.

11.1. Protocolos de correo electrónico

Hoy día, el correo electrónico es entregado usando una arquitectura cliente/servidor. Un mensaje de correo electrónico es creado usando un programa de correo cliente. Este programa luego envía el mensaje a un servidor. El servidor luego lo redirige al recipiente de correo del servidor, donde el mensaje es luego suministrado al recipiente de correo del cliente.

Para habilitar todo este proceso, una variedad de protocolos de red estándar permiten que diferentes máquinas, a menudo ejecutando sistemas operativos diferentes y usando diferentes programas de correo, envíen y reciban correo electrónico o email.

Los protocolos que se indican a continuación son los que más se utilizan para transferir correo electrónico de un sistema a otro.

11.1.1. Protocolos de transporte de correo

La entrega de correo desde una aplicación cliente a un servidor, y desde un servidor origen al servidor destino es manejada por el *Simple Mail Transfer Protocol* (*SMTP*) .

11.1.1.1. SMTP

El objetivo principal del Protocolo simple de transferencia de correo, SMTP, es transferir correo entre servidores de correo. Sin embargo es crítico para los clientes de correo también. Para poder enviar correo, el cliente envia el mensaje a un servidor de correo saliente, el cual luego contacta al servidor de correo de destino para la entrega. Por esta razón, es necesario especificar un servidor SMTP cuando se esté configurando un cliente de correo.

Bajo Red Hat Linux, un usuario puede configurar un servidor SMTP en la máquina local para manejar la entrega de correo. Sin embargo, también es posible configurar servidores remotos SMTP para el correo saliente.

Un punto importante sobre el protocolo SMTP es que no requiere autenticación. Esto permite que cualquiera en la Internet puede enviar correo a cualquiera otra personal o a grandes grupos de personal.

Esta característica de SMTP es lo que hace posible el correo basura o *spam*. Los servidores SMTP modernos intentan minimizar este comportamiento permitiendo que sólo los hosts conocidos accedan al servidor SMTP. Los servidores que no ponen tales restricciones son llamados servidores *open relay*.

Red Hat Linux utiliza Sendmail (`/usr/sbin/sendmail`) como su programa SMTP por defecto. Sin embargo, también está disponible una aplicación más simple de servidor de correo llamada Postfix (`/usr/sbin/postfix`).

11.1.2. Protocolos de acceso a correo

Hay dos protocolos principales usados por las aplicaciones de correo cliente para recuperar correo desde los servidores de correo: el *Post Office Protocol* (*POP*) y el *Internet Message Access Protocol* (*IMAP*).

A diferencia de SMTP, estos protocolos requieren autenticación de los clientes usando un nombre de usuario y una contraseña. Por defecto, las contraseñas para ambos protocolos son pasadas a través de la red de forma encriptada.

11.1.2.1. POP

El servidor por defecto POP bajo Red Hat Linux es `/usr/sbin/ipop3d` y es proporcionado por el paquete `imap`. Cuando utilice el protocolo POP, los mensajes de correo son descargados a través de las aplicaciones de correo cliente. Por defecto, la mayoría de los clientes de correo POP son configurados automáticamente para borrar el mensaje en el servidor de correo después que éste ha sido transferido exitósamente, sin embargo esta configuración se puede cambiar.

POP es completamente compatible con estándares importantes de mensajería de Internet, tales como *Multipurpose Internet Mail Extensions* (*MIME*), el cual permite los anexos de correo.

POP funciona mejor para usuarios que tienen un sistema en el cual leer correo. También funciona bien para usuarios que no tienen una conexión permanente a la Internet o a la red conteniendo el servidor de correo. Desafortunadamente para aquellos con conexiones lentas, POP requiere programas cliente que luego de la autenticación, descarguen el contenido completo de cada mensaje. Esto puede tomar un buen tiempo si algún mensaje tiene anexos grandes.

La versión más reciente del protocolo estándar POP es POP3.

Existen sin embargo, una variedad de variantes del protocolo POP menos usadas:

- *APOP* — POP3 con autenticación MDS. En este protocolo, el cliente de correo envía un hash codificado de la contraseña al servidor en lugar de enviar una contraseña encriptada.
- *KPOP* — POP3 con autenticación Kerberos. Consulte el Capítulo 17 para ver más información.
- *RPOP* — POP3 con autenticación RPOP, que utiliza un identificador de usuario similar a una contraseña para autenticar las peticiones POP. No obstante, este ID no esta encriptado por tanto RPOP no es más seguro que el estándar POP.

Para añadir seguridad, es posible utilizar la encriptación *Secure Socket Layer* (*SSL*) para la autenticación del cliente y las sesiones de transferencias de datos. Esto se puede activar usando el servicio `ipop3s` o mediante el uso del programa `/usr/sbin/stunnel`. Refiérase a Sección 11.5.1 para más información.

11.1.2.2. IMAP

El servidor por defecto IMAP bajo Red Hat Linux es `/usr/sbin/imapd` y es proporcionado por el paquete `imap`. Cuando utilice un servidor de correo IMAP, los mensajes de correo se mantienen en el

servidor donde los usuarios pueden leer y borrarlos. IMAP también permite a las aplicaciones cliente crear, renombrar o borrar directorios en el servidor para organizar y almacenar correo.

IMAP lo utilizan principalmente los usuarios que accesan su correo desde varias máquinas. El protocolo es conveniente también para usuarios que se estén conectando al servidor de correo a través de una conexión lenta, porque sólo la información de la cabecera del correo es descargada para los mensajes, hasta que son abiertos, ahorrando de esta forma ancho de banda. El usuario también tiene la habilidad de eliminar mensajes sin verlos o descargarlos.

Por conveniencia, las aplicaciones cliente IMAP son capaces de hacer caché de los mensajes localmente, para que el usuario pueda hojear los mensajes previamente leídos cuando no se esté conectado directamente al servidor IMAP.

IMAP, como POP, es completamente compatible con estándares de mensajería de Internet, tales como MIME, que permite los anexos de correo.

Para seguridad adicional, es posible utilizar la encriptación *SSL* para la autenticación de clientes y para las sesiones de transferencia de datos. Esto se puede activar usando el servicio `imaps`, o mediante el uso del programa `/usr/sbin/stunnel`. Refiérase a Sección 11.5.1 para más información.

También están disponibles otros clientes y servidores de correo IMAP gratuítos así como también comerciales, muchos de los cuales extienden el protocolo IMAP y proporcionan funcionalidades adicionales. Una lista completa sobre esto se puede encontrar en http://www.imap.org/products/longlist.htm.

11.2. Clasificaciones de los programas de correo

En general, todas las aplicaciones de email caen en al menos una de tres clasificaciones. Cada clasificación juega un papel específico en el proceso de mover y administrar los mensajes de correo. Mientras que la mayoría de los usuarios sólo están al tanto del programa de correo específico que usan para recibir o enviar mensajes, cada uno es importante para asegurar que el mensaje llegue a su destino correcto.

11.2.1. Agente de transferencia de correo

Un *Agente de transferencia de correo* (*MTA*) transfiere mensajes de correo electrónico entre hosts usando SMTP. Un mensaje puede envolver a muchos MTAs a medida que este se mueve hasta llegar a su destino.

Aunque la entrega de mensajes entre máquinas puede parecer bien simple, el proceso completo de decidir si un MTA particular puede o debería aceptar un mensaje para ser repartido, es más bien complicada. Además, debido a los problemas de spam, el uso de un MTA particular está usualmente restringido por la configuración del MTA o por la falta de acceso a la red MTA.

Muchos programas clientes de correo modernos pueden actuar como un MTA cuando estén enviando correo. Sin embargo, esta acción no debería ser confundida con el papel de un MTA verdadero. La única razón de que los programas de correo cliente son capaces de enviar mensajes (como un MTA) es porque el host ejecutando la aplicación no tiene su propio MTA. Esto es particularmente cierto para programas de cliente o para sistemas que no están basados en el sistema operativo Unix. Sin embargo, estos programas clientes sólo envían mensajes hacia afuera hacia un MTA para el cual estan autorizados a utilizar y no directamente al servidor de correos recipiente.

Puesto que Red Hat Linux instala dos MTAs, Sendmail y Postfix, los programas cliente de correo no son comúnmente requeridos que actúen como un MTA. Red Hat Linux también incluye un MTA de propósitos especiales llamado Fetchmail.

Para más información sobre Sendmail y Fetchmail, consulte Sección 11.3.

11.2.2. Agente de entrega de correos

Un *Agente de entrega de correos* (*MDA*) es invocado por un MTA para archivar correo entrante en el buzón de correo del usuario. En muchos casos, el MDA es en realidad un *Agente de entregas local* (*LDA*), tal como `mail` o Procmail.

Cualquier programa que realmente maneja un mensaje para entrega al punto en que puede ser leído por una aplicación cliente de correos se puede considerar un MDA. Por esta razón, algunos MTAs (tales como Sendmail y Postfix) pueden tener el papel de un MDA cuando ellos anexan nuevos mensajes de correo al archivo spool de correo del usuario. En general, los MDAs no transportan mensajes entre sistemas tampoco proporcionan una interfaz de usuario; los MDAs distribuyen y clasifican mensajes en la máquina local para que lo accese una aplicación cliente de correo.

11.2.3. Agente de usuario de correo

Un *agente de usuario de correo* (*MUA*) es sinónimo con una aplicación cliente de correo. Un MUA es un programa que, al menos, le permite a los usuarios leer y redactar mensajes de correo. Muchos MUAs son capaces de recuperar mensajes a través de los protocolos POP o IMAP, configurando los buzones de correo para almacenar mensajes y enviando los mensajes salientes a un MTA.

Los MUAs pueden ser de interfaz gráfica, tal como **Mozilla Mail**, o tener una interfaz basada en texto muy sencilla, tal como `mutt` o `pine`.

11.3. Agentes de transporte de correo

Red Hat Linux incluye dos tipos primarios de MTAs, Sendmail y Postfix. Sendmail es configurado como el MTA por defecto, aún cuando es fácil cambiar el MTA predeterminado a Postfix.

Sugerencia

Para información sobre cómo cambiar el MTA predeterminado de Sendmail a Postfix, consulte el capítulo llamado *Configuración del agente de transporte de correos (MTA)* en el *Manual de personalización de Red Hat Linux*.

Red Hat Linux también incluye un MTA de propósitos especiales llamado Fetchmail, el cual es usado para entregar correo desde un MTA remoto a uno local.

Esta sección detalla Sendmail y Fetchmail.

11.3.1. Sendmail

El propósito principal de Sendmail, como cualquier otro MTA, es el de transferir correo de forma segura entre hosts, usualmente usando el protocolo SMTP. Sin embargo, Sendmail es altamente configurable, permitiendo el control sobre casi cada aspecto del manejo de correos, incluyendo el protocolo utilizado. Muchos administradores de sistemas seleccionan Sendmail como su MTA debido a su poder y escalabilidad.

11.3.1.1. Propósitos y limitaciones

Es importante estar conscientes de qué es Sendmail y de lo que puede hacer al contrario de lo que no es. En estos tiempos de aplicaciones monolíticas que cubren varios papeles, Sendmail puede parecer la única aplicación necesitada para ejecutar un servidor de correo en una organización. Esto técnicamente es verdad, puesto que Sendmail puede colocar correo en los directorios de cada usuario

y entregar el correo saliente para los usuarios. Sin embargo, la mayoría de los usuarios requieren normalmente mucho más que la entrega de correos. Ellos usualmente quieren interactuar con su correo usando un MUA, que utiliza POP o IMAP, para descargar sus mensajes a sus máquinas locales. O prefieren una interfaz tipo web para ganar acceso a sus buzones. Estas otras aplicaciones pueden funcionar en conjunto con Sendmail, pero ellas existen en realidad por otras razones y pueden operar separadamente una de la otra.

Está más allá del ámbito de esta sección explicar todo lo que Sendmail debería o podría hacer. Literalmente con cientos de opciones y reglas que configurar, hay disponibles libros dedicados completamente a explicar todo lo que se puede hacer y como solucionar problemas cuando las cosas salen mal. Consulte Sección 11.6 para una lista de los recursos de Sendmail.

Esta sección revisa los archivos instalados con Sendmail por defecto y revisa los cambios básicos a la configuración, incluyendo cómo detener correo no deseado (spam) y también cómo extender Sendmail con el *Lightweight Directory Access Protocol (LDAP)*.

11.3.1.2. La instalación de Sendmail por defecto

El ejecutable de Sendmail es `/usr/sbin/sendmail`.

El archivo de configuración largo y detallado de Sendmail es `/etc/mail/sendmail.cf`. Evite modificar este archivo `sendmail.cf` directamente. En vez de esto, para hacer cambios en la configuración, edite el archivo `/etc/mail/sendmail.mc`, cree una copia de respaldo del original `/etc/mail/sendmail.cf`, y luego use el procesador de macros incluído `m4` para crear un nueva `/etc/mail/sendmail.cf`. Más información sobre la confifuración de Sendmail se puede encontrar en Sección 11.3.1.3.

Varios archivos de configuración de Sendmail son instalados en el directorio `/etc/mail/` incluyendo:

- `access` — Especifica los sistemas que pueden utilizar Sendmail para enviar correo saliente.
- `domaintable` — Le permite crear asignaciones de nombres de dominio.
- `local-host-names` — Especifica aliases para el host.
- `mailertable` — Especifica instrucciones para ignorar la ruta de determinados dominios.
- `virtusertable` — Le permite especificar una forma de alias para dominios específicos, permitiendo a múltiples dominios virtuales ser hospedados en una misma máquina.

Muchos de los archivos de configuración en `/etc/mail/`, tales como `access`, `domaintable`, `mailertable` y `virtusertable`, deben en realidad almacenar su información en archivos de bases de datos antes de que Sendmail puede usar algún cambio de configuración. Para incluir cambios hechos a estas configuraciones en sus archivos de bases de datos, ejecute el comando:

```
makemap hash /etc/mail/<name> < /etc/mail/<name>
```

Donde *<name>* es reemplazado con el nombre del archivo de configuración a convertir.

Por ejemplo, para tener todos los correos direccionados al dominio `example.com` entregados a <`bob@other-example.com`>, añada la línea siguiente al archivo `virtusertable`:

```
@example.com      bob@other-example.com
```

Para finalizar el cambio, el archivo `virtusertable.db` debe ser actualizado usando el comando siguiente como root:

```
makemap hash /etc/mail/virtusertable < /etc/mail/virtusertable
```

Esto crea un nuevo archivo `virtusertable.db` conteniendo la nueva configuración.

11.3.1.3. Cambios comunes de configuración de Sendmail

Cuando se esté alterando el archivo de configuración de Sendmail, es mejor generar un archivo completamete nuevo `/etc/mail/sendmail.cf` en vez de modificar el existente.

Atención

Antes de cambiar el archivo `sendmail.cf`, es una muy buena idea hacer una copia de respaldo del archivo en funcionamiento.

Para añadir funcionalidad a Sendmail, modifique el archivo `/etc/mail/sendmail.mc`. Cuando termine, utilice el procesador de macros `m4` para generar un nuevo `sendmail.cf` ejecutando el comando `m4 /etc/mail/sendmail.mc > /etc/mail/sendmail.cf`. Después de crear un nuevo archivo `/etc/mail/sendmail.cf`, reinicie Sendmail para que tomen efecto los cambios. La forma más fácil de hacer esto es escribiendo el comando `/sbin/service sendmail restart` como root.

Por defecto, el procesador de macros `m4` es instalado con Sendmail pero es parte del paquete `m4`.

Importante

El archivo por defecto `sendmail.cf` no permite que Sendmail acepte conexiones de red desde ningún host mas que la máquina local. Para configurar Sendmail como un servidor para otros clientes, modifique `/etc/mail/sendmail.mc` y cambie `DAEMON_OPTIONS` para que también escuche en dispositivos de red o coloque en comentarios toda esta opción. Luego, vuelva a generar `/etc/mail/sendmail.cf` ejecutando:

```
m4 /etc/mail/sendmail.mc > /etc/mail/sendmail.cf
```

Esta configuración debería de funcionar para la mayoría de los sitios sólo SMTP. *No* funcionará para sitios UUCP (copia UNIX a UNIX); debe generar un nuevo `sendmail.cf` si se está usando transferencias de correo UUCP.

Consulte el archivo `/usr/share/sendmail-cf/README` antes de modificar cualquier archivo en los directorios bajo el directorio `/usr/share/sendmail-cf`, pues ellos pueden afectar la futura configuración de los archivos `/etc/mail/sendmail.cf`.

11.3.1.4. Creación de máscaras

Una configuración común de Sendmail es tener una sola máquina actuando como el gateway de correo para todas las máquinas en la red. Por ejemplo, una compañía puede querer tener una máquina llamada `mail.bigcorp.com` que maneja todo su correo y asigna una dirección de retorno consistente para todo el correo saliente.

En esta situación, el servidor Sendmail debe enmascarar los nombres de las máquinas en la red de la compañía para que la dirección de retorno sea user@bigcorp.com en vez de user@devel.bigcorp.com.

Para hacer esto, añada las líneas siguientes `/etc/mail/sendmail.mc`:

```
FEATURE(always_add_domain)dnl
FEATURE('masquerade_entire_domain')
FEATURE('masquerade_envelope')
FEATURE('allmasquerade')
MASQUERADE_AS('bigcorp.com.')
```

```
MASQUERADE_DOMAIN(‘bigcorp.com.’)
MASQUERADE_AS(bigcorp.com)
```

Después de generar un nuevo `sendmail.cf` usando `m4`, esta configuración hará que todo el correo dentro de la red aparezca como que si hubiese sido enviado desde `bigcorp.com`.

11.3.1.5. Detener el correo basura

El correo basura se puede definir como correo no deseado e innecesario recibido por un usuario que nunca solicitó tal comunicación. Es un abuso costoso y molesto de las comunicaciones de Internet estándar.

Sendmail hace relativamente fácil bloquear nuevas técnicas de difusión de correo basura. Hasta bloquea por defecto muchos de los métodos comunes de difusión de correo basura.

Por ejemplo, el reenvío de mensajes SMTP, también conocido como relaying, ha sido desactivado por defecto desde Sendmail versión 8.9. Antes de que se produjese este cambio, Sendmail indicaba al host (`x.org`) que aceptara mensajes desde un partido (`y.com`) y que los enviara a un partido diferente (`z.net`). Ahora, sin embargo, Sendmail debe ser configurado para permitir a cualquier dominio que transmita correo a través del servidor. Para configurar dominios de transmisión, modifique el archivo `/etc/mail/relay-domains` y reinicie Sendmail.

Sin embargo, en muchas ocasiones, los usuarios reciben bombardeos de correo basura de otros servidores a través de Internet. En estos casos, puede utilizar las funciones de control de acceso de Sendmail que están disponibles en el archivo `/etc/mail/access` para prevenir conexiones desde host indeseados. El ejemplo siguiente ilustra como este archivo puede ser usado para bloquear y también para permitir el acceso al servidor Sendmail:

```
badspammer.com        ERROR:550 "Go away and do not spam us anymore"
tux.badspammer.com    OK
10.0                  RELAY
```

Este ejemplo indica que cualquier correo enviado desde `badspammer.com` es bloqueado con un código de error 550 RFC-821, con un mensaje para el emisor. Los correos enviados desde el subdominio `tux.badspammer.com`, seran aceptados. La última línea muestra que cualquier correo enviado desde la red 10.0.*.* se puede transmitir a través del servidor de correo.

Debido a que `/etc/mail/access.db` es una base de datos, use `makemap` para activar los cambios. Haga esto usando el comando siguiente como root:

```
makemap hash /etc/mail/access < /etc/mail/access
```

Como puede imaginarse, este ejemplo sólo analiza una mínima parte de lo que Sendmail puede realizar en cuanto a permitir o bloquear el acceso. Consulte `/usr/share/doc/sendmail/README.cf` para más información y ejemplos.

Puesto que Sendmail llama a Procmail MDA cuando está entregando correo, también es posible usar un programa de filtrado de correo basura, tal como SpamAssassin para identificar y archivar correo basura por los usuarios. Consulte Sección 11.4.2.6 para más detalles sobre SpamAssassin.

11.3.1.6. Uso de Sendmail con LDAP

Usando el *Lightweight Directory Access Protocol (LDAP)* es una forma rápida y poderosa de encontrar información específica sobre un usuario particular desde un grupo mucho más grande. Por ejemplo, un servidor LDAP puede ser usado para buscar una dirección de correo particular desde un directorio corporativo usando el apellido del usuario. En este tipo de implementación, LDAP esta bastante separado de Sendmail, con LDAP la información de usuario de forma jerárquica y Sendmail sólo recibiendo el resultado de las consultas de LDAP en mensajes de correo pre-direccionados.

Sin embargo, Sendmail admite mucha más integración con LDAP y utiliza este protocolo para sustituir archivos mantenidos independientemente, como `aliases` y `virtusertables`, que se ubican en servidores de correo diferentes que funcionan juntos para soportar una organización de nivel medio a corporativo. A modo de resumen, puede usar LDAP para separar el nivel de enrutamiento desde Sendmail y sus archivos de configuración separados a un cluster LDAP poderoso que pueden utilizar distintas aplicaciones.

La versión actual de Sendmail es compatible con LDAP. Para ampliar el servidor de Sendmail y usar LDAP, primero debe obtener un servidor LDAP, como **OpenLDAP**, ejecutarlo y configurarlo correctamente. A continuación, modifique `/etc/mail/sendmail.mc` para incluir lo siguiente:

```
LDAPROUTE_DOMAIN('yourdomain.com')dnl
FEATURE('ldap_routing')dnl
```

Nota

Esta es sólo una configuración muy básica de Sendmail con LDAP. Su configuración puede ser muy diferente de la indicada según la implementación específica de LDAP, especialmente si desea configurar varias máquinas de Sendmail para que utilicen un servidor LDAP común.

Consulte `/usr/share/doc/sendmail/README.cf` para obtener instrucciones y ejemplos de configuración de enrutamiento de LDAP.

Luego, vuelva a crear el archivo `/etc/mail/sendmail.cf` ejecutando `m4` y reiniciando Sendmail. Vea Sección 11.3.1.3 para detalles sobre cómo hacer esto.

Para más información sobe LDAP, vea Capítulo 13.

11.3.2. Fetchmail

Fetchmail es un MTA el cual recupera el correo desde servidores remotos y los entrega al MTA local. Muchos usuarios aprecian la capacidad de separar el proceso de descarga de mensajes ubicados en un servidor remoto del proceso de lectura y organización de correo en un MUA. Se ha diseñado teniendo presente las necesidades de los usuarios de acceso telefónico a redes. Fetchmail se conecta y descarga rápidamente todos los mensajes al archivo spool de correo mediante el uso de diversos protocolos, entre los que se incluyen POP3 e IMAP. Incluso permite reenviar los mensajes de correo a un servidor SMTP si es necesario.

Fetchmail es configurado para cada usuario a través del uso de un archivo `.fetchmailrc` en el directorio principal del usuario.

Mediante el uso de preferencias en el archivo `.fetchmailrc`, Fetchmail comprobará si hay correo en un servidor remoto e intentará entregarlo al puerto 25 de la máquina local utilizando el agente MTA local para dirigir el correo al archivo de spool del usuario correcto. Si también está disponible Procmail, podrá utilizarlo para filtrar el correo y colocarlo en un buzón para que lo pueda leer un MUA.

11.3.2.1. Opciones de configuración de Fetchmail

Aunque se pueden insertar todas las opciones en la línea de comandos pertinente para comprobar si hay correo en un servidor remoto al ejecutar Fetchmail, el uso de `.fetchmailrc` proporciona un método más sencillo. Todas las opciones de configuración se guardan en el archivo `.fetchmailrc`, y es posible ignorarlas al momento en que Fetchmail es ejecutado especificando esa opción en la línea de comandos.

Un archivo de usuario `.fetchmailrc` se divide en tres tipos concretos de opciones de configuración:

- *opciones globales* — Indican a Fetchmail las instrucciones que controlan el funcionamiento del programa o proporcionan las configuraciones para cada conexión que verifica por correo.
- *opciones de servidor* — Especifican información necesaria sobre el servidor, como nombre de host, así como las preferencias que desearía ver aplicadas con un servidor de correo particular, tal como el puerto a verificar o el número de segundos a esperar antes de un timeout. Estas opciones afectan a cada opción de usuario usado con ese servidor.
- *opciones de usuario* — Contienen información, tal como nombre de usuario y contraseña, que es necesaria para autenticar y comprobar si hay correo utilizando un servidor de correo concreto.

Las opciones globales se encuentran en la parte superior del archivo `.fetchmailrc`, seguidas de una o varias opciones de servidor con las que se designa cada uno de los servidores de correo diferentes que debería comprobar Fetchmail. Por último, se encuentran las opciones de usuario específicas de cada cuenta de usuario que desea comprobar en el servidor de correo. Al igual que las opciones de servidor, se pueden especificar varias opciones de usuario para utilizarlas con un servidor determinado así como también comprobar varias cuentas de correo electrónico en el mismo servidor.

Las opciones de servidor se llaman para ejecución en el archivo `.fetchmailrc` mediante el uso de una opción especial, `poll` o `skip`, que precede cualquier información de servidor. La acción `poll` indica a Fetchmail que use esta opción de servidor cuando se ejecute, lo que en realidad verifica por correo usando las opciones de usuario. Cualquier opción de servidor luego de una acción `skip`, sin embargo, no se verificará a menos que este nombre de host sea especificado cuando Fetchmail es llamado. La opción `skip` configura pruebas en `.fetchmailrc` y sólo chequea ese servidor cuando se desee específicamente, sin afectar ninguna configuracion en funcionamiento actualmente.

Un archivo `.fetchmailrc` de ejemplo se vería así:

```
set postmaster "user1"
set bouncemail

poll pop.domain.com proto pop3
     user 'user1' there with password 'secret' is user1 here

poll mail.domain2.com
     user 'user5' there with password 'secret2' is user1 here
     user 'user7' there with password 'secret3' is user1 here
```

En este ejemplo, las opciones globales son las que establecen que se le envíe correo al usuario como última instancia (opción `postmaster`) y que todos los errores de correo se manden al postmaster en lugar de a la persona que ha enviado el correo (`bouncemail`). La acción `set` indica a Fetchmail que esta línea contiene una opción global. A continuación, se especifican dos servidores de correo: uno para que compruebe si hay correo con el protocolo POP3 y otro para que pruebe a usar varios protocolos para encontrar uno que funcione. Se comprueba el correo de dos usuarios con la segunda opción de servidor, pero todo el correo que se encuentre se envía al spool de correo del user1. Esto permite comprobar varios buzones en diversos servidores como si se tratara de un único buzón MUA. La información específica de cada usuario comienza con la acción `user`.

Nota

No es necesario que los usuarios coloquen sus contraseñas en el archivo `.fetchmailrc`. Al omitir la sección `with password '<password>'` causa que Fetchmail solicite por una contraseña cuando es lanzado.

Fetchmail contiene muchas opciones diferentes globales, de servidor y locales. Muchas de estas opciones casi nunca se usan o sólo se aplican en situaciones muy específicas. La página del manual de `fetchmail` explica cada opción en detalle, pero las usadas más a menudo se listan aquí.

11.3.2.2. Opciones globales

Cada opción global debería ser colocada en una línea individual después de `set`.

- `demonio <seconds>` — Indica a Fetchmail de usar automáticamente el modo de demonio, con el que estará en segundo plano y comprobar si hay correo en los intervalos especificados.
- `postmaster` — Indica a Fetchmail un usuario local para enviar el correo en caso de problemas de entrega.
- `syslog` — Indica a Fetchmail el registro de mensajes de error y de estado. Por defecto, es `/var/log/maillog`.

11.3.2.3. Opciones de servidor

Las opciones de servidor deben ser colocadas en su propia línea en `.fetchmailrc` después de una acción `poll` o `skip`.

- `auth <auth-type>` — Especifica el tipo de autenticación que se utilizará. Por defecto, se utiliza la autenticación por `password` pero algunos protocolos admiten también otros tipos, entre los que se incluyen `kerberos_v5`, `kerberos_v4`y `ssh`. Si se usa el tipo de autenticación `any`, Fetchmail primero usará métodos que no necesiten contraseña y luego otros que creen máscara para la contraseña. Finalmente, intentará enviar la contraseña sin encriptar para ser autenticada al servidor.
- `interval <number>` — Registra el servidor especificado cada `<number>` de veces que verifica por correo en todos los servidores configurados. Esta opción es generalmente utilizada por servidores de correo donde el usuario rara vez recibe mensajes.
- `port <port-number>` — Ignora el número de puerto por defecto para un protocolo especificado.
- `proto <protocol>` — Especifica un protocolo específico, tal como `pop3` o `imap`, para que verifique por mensajes en este servidor.
- `timeout <seconds>` — Configura Fetchmail para abandonar después de un determinado intervalo de inactividad del servidor. Si no se define este valor, se asume un valor de `300` segundos.

11.3.2.4. Opciones de usuario

Las opciones de usuario se pueden insertar en las propias líneas debajo de una opción de servidor en la misma línea que la opción de servidor. En cualquier caso, las opciones de usuarios van después de la opción `user` (definida más abajo).

- `fetchall` — Ordena a Fetchmail descargar todos los mensajes en cola, incluidos los mensajes que ya se han visto. Por defecto, Fetchmail sólo lo hace con los nuevos.
- `fetchlimit <number>` — Sólo permite descargar un determinado número de mensajes antes de detenerse.
- `flush` — Indica a Fetchmail de eliminar todos los mensajes en cola que ya se han visto antes de descargar mensajes nuevos.
- `limit <max-number-bytes>` — Permite especificar que sólo se recuperen los mensajes de una tamaño inferior al indicado. Esta opción es útil con enlaces lentos, cuando un mensaje largo toma mucho tiempo en descargarse.

- `password '<password>'` — Especifica la contraseña que utilizará este usuario.
- `preconnect "<command>"` — Indica a Fetchmail de ejecutar el comando especificado antes de recuperar los mensajes de este usuario.
- `postconnect "<command>"` — Indica a Fetchmail de ejecutar el comando especificado después de recuperar los mensajes de este usuario.
- `ssl` — Activa la encriptación SSL.
- `user "<username>"` — Establece el nombre de usuario que Fetchmail usa para recuperar los mensajes. *Esta opción debería listarse antes de cualquier otra opción de usuario.*

11.3.2.5. Opciones de comando de Fetchmail

La mayoría de las opciones de Fetchmail se pueden utilizar en la línea de comando al ejecutar el comando `fetchmail`, reflejando las opciones de configuración de `.fetchmailrc`. Esto se realiza para que se use Fetchmail con o sin un archivo de configuración. La mayoría de los usuarios no usan estas opciones en la línea de comandos porque les resulta más sencillo dejarlas en el archivo `.fetchmailrc` para que se utilicen cada vez que se ejecuta Fetchmail.

Sin embargo, en ocasiones puede estar interesado en ejecutar el comando `fetchmail` con otras opciones para un fin concreto. Es posible emitir opciones de comando para que temporalmente se ignore una configuración `.fetchmailrc` que está causando un error, puesto que cualquier opción especificada en la línea de comandos sobreescribe las opciones del archivo de configuración.

11.3.2.6. Opciones de depuración o información

Algunas opciones usadas luego del comando `fetchmail` pueden suministrar información importante.

- `--configdump` — Muestra cada opción posible en función de la información de `.fetchmailrc` y los valores por defecto de Fetchmail. No se recupera correo de ningún usuario al usar esta opción.
- `-s` — Ejecuta Fetchmail en modo silencioso, con lo cual se evita que aparezcan mensajes y errores después del comando `fetchmail`.
- `-v` — Ejecuta Fetchmail en modo detallado y muestra todas las comunicaciones entre Fetchmail y los servidores de correo remotos.
- `-V` — Hace que Fetchmail muestre información de versión detallada, una lista de las opciones globales y los parámetros que se utilizarán con cada usuario, incluido el protocolo de correo y el método de autenticación. No se recupera correo de ningún usuario al usar esta opción.

11.3.2.7. Opciones especiales

Estas opciones son en ocasiones útiles para sobrescribir los valores por defecto que a menudo contiene el archivo `.fetchmailrc`.

- `-a` — Indica a Fetchmail que descargue todos los mensajes del servidor de correo remoto, ya se hayan o no visto antes. Por defecto, Fetchmail sólo descarga los mensajes nuevos.
- `-k` — Hace que Fetchmail deje una copia de los mensajes en el servidor de correo remoto después de descargarlos. Esta opción sobrescribe el comportamiento por defecto de eliminar los mensajes después de descargarlos.
- `-l <max-number-bytes>` — Indica a Fetchmail que no descargue mensajes con un tamaño superior al indicado y dejarlos en el servidor de correo remoto.
- `--quit` — Sale del proceso de demonio de Fetchmail.

Se pueden encontrar más comandos y opciones de `.fetchmailrc` en la página del manual de `fetchmail`.

11.4. Agente de entrega de correo

Red Hat Linux incluye dos MDAs primarias, Procmail y `mail`. Ambas aplicaciones son consideradas Agentes de entrega local y ambas mueven el correo desde el archivo spool MTA en el buzón de correo. Sin embargo, Procmail proporciona un sistema de filtrado de correo robusto.

Esta sección detalla solamente Procmail. Para información sobre el comando `mail`, consulte su página man.

Procmail entrega y filtra correo mientras es colocado en el archivo spool de correo de la máquina local. Es una herramienta eficaz, que hace un uso adecuado de los recursos del sistema y de amplio uso. Procmail desempeña un papel crítico en la entrega de correo a ser leído por las aplicaciones clientes de correo.

Procmail puede ser invocado de muchas formas diferentes. Cada vez que un MTA coloca un correo en el archivo spool de correo, Procmail es lanzado. Procmail luego filtra y archiva el correo de manera que el MUA lo pueda encontrar, y sale. Alternativamente, el MUA puede ser configurado para ejecutar Procmail cada vez que un mensaje es recibido y así los mensajes son movidos en sus buzones correctos. Por defecto, la presencia de un archivo `.procmailrc` en el directorio principal del usuario llamará a Procmail cada vez que un MTA reciba un nuevo mensaje.

Las acciones que toma Procmail con un correo dependen de las instrucciones de las *recetas* particulares, o reglas mediante las que se comparan los mensajes con el programa. Si un mensaje coincide con la receta o regla, el correo se ubicará en un determinado archivo, se eliminará o se procesará.

Cuando Procmail arranca, lee el mensaje de correo y separa el cuerpo de la información de cabecera. A continuación, busca los archivos `/etc/procmailrc` y `rc` en el directorio por defecto `/etc/procmailrcs`, de todo el sistema, así como las variables de entorno Procmail y reglas. Luego busca si hay un archivo `.procmailrc` en el directorio principal del usuario para encontrar las reglas específicas de dicho usuario. Muchos usuarios también crean archivos `rc` adicionales propios para Procmail que son referidos por su archivo `.procmailrc`, pero que se pueden activar o desactivar rápidamente si se produce un problema al filtrar el correo.

Por defecto, no hay archivos `rc` aplicables a todo el sistema en el directorio `/etc` y ningún archivo de usuario `.procmailrc` existe. Para comenzar a usar Procmail, construya un archivo `.procmailrc` con variables de entorno y reglas particulares para ciertos tipos de mensajes.

En la mayoría de las configuraciones, la decisión sobre si Procmail se arranca e intenta filtrar el correo se basa en el existencia de un archivo de usuario `.procmailrc`. Para desactivar Procmail, pero guardar el trabajo en el archivo `.procmailrc`, copie la información a un archivo de nombre similar que utilice el comando `mv ~/.procmailrc ~/.procmailrcSAVE`. Cuando esté preparado para realizar nuevas pruebas con Procmail, cambie nuevamente el nombre del archivo a `.procmailrc`. Procmail empezará a funcionar de nuevo inmediatamente.

11.4.1. Configuración de Procmail

Los archivos de configuración de Procmail, y más en concreto el archivo de usuario `.procmailrc`, contienen variables de entorno importantes. Estas variables indican a Procmail qué mensajes deben ordenarse, qué hacer con los mensajes que no coinciden con ninguna receta, etc.

Estas variables de entorno normalmente aparecen al principio del archivo `.procmailrc` con el siguiente formato:

`<env-variable>="<value>"`

En este ejemplo, `<env-variable>` es el nombre de la variable y `<value>` define la variable.

Muchas variables de entorno no se utilizan por la mayor parte de los usuarios de Procmail, y muchas de las variables de entorno más importantes ya están definidas con un valor por defecto. La mayoría de las veces tratará con las siguientes variables:

- `DEFAULT` — Establece el buzón por defecto en el que se ubicarán los mensajes que no coincidan con ninguna receta.

  El valor por defecto `DEFAULT` es el mismo que `$ORGMAIL`.

- `INCLUDERC` — Especifica archivos `rc` adicionales que contienen más recetas para los que deben comprobarse los mensajes. Esto permite desglosar las listas de recetas de Procmail en archivos individuales según diversas funciones (como bloquear correo basura y gestionar listas de correo) que se pueden activar o desactivar con caracteres de comentario en el archivo de usuario `.procmailrc`.

  Por ejemplo, las líneas en el archivo `.procmailrc` del usuario se pueden parecer a lo siguiente:

  ```
  MAILDIR=$HOME/Msgs
  INCLUDERC=$MAILDIR/lists.rc
  INCLUDERC=$MAILDIR/spam.rc
  ```

  Si el usuario desea desactivar el filtro de Procmail para las listas de correo, pero quiere controlar el correo basura, solamente deberá comentar la primera línea `INCLUDERC` con un carácter `#`.

- `LOCKSLEEP` — Establece cada cuanto tiempo, en segundos, que Procmail intentará usar un lockfile concreto. El valor por defecto es ocho segundos.

- `LOCKTIMEOUT` — Establece la cantidad de tiempo, en segundos, que debe transcurrir después de modificar un lockfile para que Procmail asuma que este lockfile es antiguo y que se puede eliminar. El valor por defecto es 1024 segundos.

- `LOGFILE` — La ubicación y archivo que contendran los mensajes de error o de información de Procmail.

- `MAILDIR` — Establece el directorio de trabajo actual de Procmail. Si se define este directorio, todas las otras rutas de Procmail serán relativas a este directorio.

- `ORGMAIL` — Especifica el buzón original u otro lugar para colocar los mensajes si no se pueden ubicar en la ubicación de receta o por defecto.

  Por defecto, es usado un valor de `/var/spool/mail/$LOGNAME`.

- `SUSPEND` — Establece la cantidad de tiempo, en segundos, que Procmail se detendrá si no está disponible un recurso necesario, tal como espacio de intercambio (swap).

- `SWITCHRC` — Permite a un usuario especificar un archivo externo que contiene recetas de Procmail adicionales, como la opción `INCLUDERC`, excepto que la verificación de recetas es en realidad detenida en el archivo de configuración referido y sólo se usan las recetas en el archivo `SWITCHRC` especificado.

- `VERBOSE` — Hace que Procmail registre mucha más información. Esta opción es útil para procesos de depuración.

Otras variables de entorno importantes se extraen del shell, tal como `LOGNAME`, que es el nombre de conexión, `HOME`, que es la ubicación del directorio principal y `SHELL`, que es el shell por defecto.

Hay una explicación completa de todas las variables de entorno y sus valores por defecto en la página man de `procmailrc`.

11.4.2. Recetas de Procmail

Los usuarios nuevos consideran que la creación de recetas es la parte más difícil de Procmail. En cierto modo, esto es lógico, ya que las recetas comparan los mensajes con las *expresiones regulares*, que es un formato concreto que se utiliza para especificar cualificaciones para una cadena coincidente. Sin embargo, las expresiones regulares no son difíciles de crear, e incluso es más fácil entenderlas cuando se leen. Además, la consistencia en la forma en que las recetas de Procmail están escritas, independientemente de las expresiones regulares, facilita adivinar su contenido.

No entra en el objeto de este capítulo ofrecer una explicación extensa sobre las expresiones regulares. La estructura de las recetas de Procmail es más importante, y hay ejemplos útiles de ellas en varios sitios de Internet, por ejemplo en http://www.iki.fi/era/procmail/links.html). El uso y la adaptación adecuada de las expresiones regulares contenidas en estos ejemplos dependerá de si se entiende la estructura correspondiente. Puede encontrar información básica específica a las reglas de expresiones regulares en las páginas man de `grep`.

Una receta de Procmail tiene la siguiente estructura:

```
:0<flags>: <lockfile-name>

* <special-condition-character> <condition-1>
* <special-condition-character> <condition-2>
* <special-condition-character> <condition-N>

<special-action-character><action-to-perform>
```

Los dos primeros caracteres de una receta de Procmail son dos puntos y un cero. Opcionalmente, se pueden insertar varios indicadores después del cero para controlar lo que hace Procmail cuando procesa la receta. Dos puntos después de la sección *<flags>* especifica que se creará un lockfile para este mensaje. Si se va a crear un lockfile, debe especificar su nombre en el espacio *<lockfile-name>*.

Una receta puede contener varias condiciones con las que se comparará el mensaje. Si no tiene condiciones, cada mensaje coincide la receta. Las expresiones regulares se insertan en algunas condiciones para facilitar su comparación con un mensaje. Si se usan varias condiciones, todas ellas deben coincidir para que se realice una acción. Las condiciones se comprueban en función de los indicadores establecidos en la primera línea de la receta. El uso de caracteres especiales opcionales que se insertan después del carácter `*` permiten controlar todavía más la condición.

La opción *<action-to-perform>* especifica lo que le ocurrirá a un mensaje si cumple una de las condiciones. Sólo puede haber una acción por receta. En muchos casos, se usa aquí el nombre de un buzón para dirigir los mensajes coincidentes a ese archivo con el fin de ordenar de una manera eficaz el correo. También se pueden usar caracteres de acción especiales antes de especificar la acción.

11.4.2.1. Recetas de entrega vs. recetas de no entrega

La acción usada si la receta coincide con un mensaje concreto determina si la receta se considera de entrega o de no entrega. Una *receta de entrega* contiene una acción que registra el mensaje en un archivo, envía el mensaje a otro programa o reenvía el mensaje a otra dirección de correo. Una *receta de no entrega* cubre cualquier otra acción, como el uso de un bloque de anidamiento. Un *bloque de anidamiento* es una acción entre llaves `{ }` que designa las acciones adicionales que deben realizarse en los mensajes que cumplen las condiciones de la receta. Los bloques de anidamiento pueden ser anidados, lo cual proporciona un mayor control a la hora de identificar y realizar acciones en los mensajes.

La entrega de recetas coincidentes con mensajes provoca que Procmail realice la acción especificada y detenga la comparación del mensaje con otras recetas. Los mensajes que cumplen las condiciones de las recetas de no entrega seguirán comparándose con otras recetas en los archivos `rc`. En otras palabras, las recetas de no entrega, hacen que el mensaje siga pasando por las recetas después de adoptarse la acción especificada.

11.4.2.2. Indicadores

Los indicadores son muy importantes para determinar cómo se compararán las condiciones de una receta con un mensaje. Los siguientes indicadores son de uso común:

- `A` — Especifica que esta receta sólo se usará si la receta anterior sin un indicador `A` o `a` también coincidió con este mensaje.

  Para garantizar que se ha completado correctamente la acción de la última receta coincidente anterior, antes de que se permita una coincidencia con la receta actual, utilice el indicador `a`.
- `B` — Analiza el cuerpo del mensaje y busca condiciones coincidentes.
- `b` — Utiliza el cuerpo de una acción resultante, como escribir el mensaje a un archivo o reenviarlo. Este es el comportamiento por defecto.
- `c` — Genera una copia al carbón (CC) del correo. Es útil para la entrega de recetas, puesto que la acción necesaria se puede realizar en el mensaje y se puede seguir procesando una copia del mensaje en los archivos `rc`.
- `D` — Hace una comparación `egrep` que distingue entre mayúsculas y minúsculas. Por defecto, el proceso de comparación no distingue entre mayúsculas y minúsculas.
- `E` — Similar al indicador `A`, con la diferencia de que las condiciones de la receta sólo se comparan con el mensaje si la receta inmediatamente anterior sin un indicador `E` no coincide. Se puede comparar con la acción *else*.

  Use el indicador `e` si sólo desea que se compruebe esta receta en el caso de que la receta anterior coincidiese pero fallase la acción.
- `f` — Usa la canalización (pipes) como filtro.
- `H` — Analiza la cabecera del mensaje y busca condiciones coincidentes. Este es el comportamiento por defecto.
- `h` — Usa la cabecera en la acción resultante. Este es el comportamiento por defecto.
- `w` — Indica a Procmail que debe esperar a que finalice el proceso del filtro o programa especificado, y que cree un informe indicando si tuvo éxito o no antes de considerar el mensaje como filtrado.

  Si desea omitir los mensajes de "error de programa", cuando decida si un filtro o no termina correctamente, utilice la opción `W`.

Hay indicadores adicionales en la página man de `procmailrc`.

11.4.2.3. Especificación de un Lockfile local

Los archivos lockfiles son muy útiles en Procmail para garantizar que no más de un proceso intenta alterar un mensaje concreto al mismo tiempo. Puede especificar un lockfile local si inserta un carácter `:` después de cualquier indicador en la primera línea de una receta. Con esto se creará un lockfile local basado en el nombre de archivo de destino más cualquier otro valor definido en la variable de entorno global `LOCKEXT`.

Como alternativa, puede especificar el nombre del lockfile local que se usará con esta receta después del carácter `:`.

11.4.2.4. Condiciones y acciones especiales

El uso de determinados caracteres antes de las condiciones y acciones de recetas de Procmail cambian el modo en que se interpretan.

Los siguientes caracteres se pueden usar después del carácter `*` al principio de una línea de condición de receta:

- `!` — Invierte la condición y ocasiona que sólo se produzca una coincidencia si la condición no coincide con el mensaje.
- `<` — Comprueba si el mensaje tiene un número inferior de bytes.
- `>` — Comprueba si el mensaje tiene un número superior de bytes.

Los siguientes caracteres se utilizan para realizar acciones especiales:

- `!` — Indica a Procmail que reenvíe el mensaje a las direcciones de correo especificadas.
- `$` — Hace referencia a una variable establecida anteriormente en el archivo `rc`. Se usa normalmente para establecer un buzón común que utilizarán varias recetas.
- `|` — Carácter de canalización que indica a Procmail que debe arrancar un programa específico para gestionar este mensaje.
- `{` y `}` — Crea un bloque de anidamiento que se usa en combinación con recetas adicionales para aplicarlo a los mensajes coincidentes.

Si no se utiliza un carácter especial al principio de la línea de acción, Procmail asume que la línea de acción está especificando un buzón donde registrar el mensaje.

11.4.2.5. Ejemplos de recetas

Procmail es un programa extremadamente flexible que permite comparar los mensajes con condiciones muy específicas y, a continuación, realizar en ellos acciones muy detalladas. Sin embargo, como resultado de esta flexibilidad, la composición de una receta de Procmail desde cero para alcanzar un objetivo concreto, puede resultar una labor muy complicada para los usuarios nuevos.

La mejor manera de obtener los conocimientos necesarios para crear condiciones de recetas de Procmail es comprender las expresiones regulares y analizar los distintos ejemplos de otros desarrolladores. Los siguientes ejemplos, muy básicos, servirán para mostrar la estructura de las recetas de Procmail y pueden proporcionar la base para crear otras recetas más complejas.

Una receta básica puede que ni siquiera tenga condiciones, como se demuestra en el siguiente ejemplo:

```
:0:
new-mail.spool
```

La primera línea inicia la receta mediante la especificación de que se cree un lockfile local pero sin indicar un nombre, de modo que Procmail utilice el nombre del archivo de destino y `LOCKEXT` para crearlo. No se especifica ninguna condición y, por tanto, cada mensaje coincide con esta receta y se insertará en el archivo de spool exclusivo denominado `new-mail.spool`, que se encuentra dentro del directorio especificado por la variable de entorno `MAILDIR`. Un agente MUA puede a continuación ver los mensajes de este archivo.

Esta receta básica se insertará al final de todos los archivos `rc` para dirigir los mensajes a una ubicación por defecto. Un ejemplo más complejo, que extrae los mensajes de una dirección de correo concreta y los saca, como se puede ver en este ejemplo.

```
:0
* ^From: spammer@domain.com
/dev/null
```

Con este ejemplo, cualquier mensaje enviado por `spammer@domain.com` son movidos inmediatamente a `/dev/null`, y se eliminarán.

Atención

Asegúrese de que una regla funciona adecuadamente antes de mover los mensajes que coinciden con `/dev/null`, que supone una eliminación permanente. Si las condiciones de receta "atrapan" inadvertidamente mensajes no destinados correctamente, esto mensajes desaparecerán sin dejar rastro, haciendo más difícil revisar problemas en la regla.

Una solución mejor es dirigir la acción de la receta a un buzón especial que compruebe de vez en cuando para buscar *positivos falsos* o mensajes que han coincidido inadvertidamente con las condiciones. Una vez comprobado que no se han coincidido por error los mensajes, puede eliminar el buzón y dirigir la acción para enviar los mensajes a `/dev/null`.

Procmail se usa principalmente como filtro de correo, colocándolos automáticamente en el lugar correcto para que no tenga que ordenarlo manualmente. La siguiente receta atrapa el correo enviado desde una lista de correo particular y los coloca en la carpeta correcta.

```
:0:
* ^(From|CC|To).*tux-lug
tuxlug
```

Cualquier mensaje enviado desde la lista de distribución `tux-lug@domain.com` se colocará automáticamente en el buzón `tuxlug` para el agente MUA. Tenga en cuenta que la condición de este ejemplo comparará el mensaje si tiene la dirección de correo de la lista de distribución en las líneas `From`, `CC`, o `To`.

Para más detalles y recetas consulte los muchos recursos disponibles para Procmail en línea en Sección 11.6.

11.4.2.6. Filtros de correo basura

Puesto que Procmail es llamado por Sendmail, Postfix y Fetchmail cuando reciben nuevos correos, se puede usar también como una herramienta poderosa para combatir correo basura.

Esto es particularmente cierto cuando Procmail es usado en conjunto con SpamAssassin. Cuando se usan juntos, estas dos aplicaciones pueden identificar rápidamente correo basura y ordenarlos o destruirlos.

SpamAssassin usa análisis de las cabeceras, de texto, listas negras y una base de datos de seguimiento de correo basura para rápida y efectivamente identificar y marcar el correo basura.

La forma más fácil para que un usuario local use SpamAssassin es colocar la siguiente línea cerca de la parte superior del archivo `~/.procmailrc`:

```
INCLUDERC=/etc/mail/spamassassin/spamassassin-default.rc
```

El `/etc/mail/spamassassin/spamassassin-default.rc` contiene una regla simple de Procmail que activa SpamAssassin para todo el correo entrante. Si un correo es identificado como basura, se marca en la cabecera como tal y en el título se coloca:

```
*****SPAM*****
```

El cuerpo del mensaje es también marcado al principio con una lista de qué elementos provocaron que fuese considerado basura.

Para archivar correo marcado como basura, se puede usar una regla similar a lo siguiente:

```
:0 Hw
* ^X-Spam-Status: Yes
spam
```

Esta regla archiva todo el correo marcado como basura en el buzón de correo llamado `spam`.

Puesto que SpamAssassin es un script Perl, puede ser necesario en servidores ocupados usar un demonio binario SpamAssassin (`spamd`) y la aplicación cliente (`spamc`). Configurar SpamAssassin de esta forma requiere acceso root.

Para arrancar el demonio `spamd`, escriba el siguiente comando como usuario root:

```
/sbin/service spamassassin start
```

Para iniciar el demonio SpamAssassin cuando el sistema es arrancado, use una utilidad initscript, tal como la **Herramienta de configuración de servicios** (`redhat-config-services`), para activar el servicio `spamassassin`. Consulte Sección 1.4.2 para más información sobre las utilidades initscript.

Para configurar Procmail a usar la aplicación cliente SpamAssassin en vez de un script Perl, coloque la siguiente línea cerca de la parte superior del archivo `~/.procmailrc` o, para una configuración global del sistema, colóquela en `/etc/procmailrc`:

```
INCLUDERC=/etc/mail/spamassassin/spamassassin-spamc.rc
```

11.5. Agentes de usuario de correo

Hay muchos programas de correo disponibles bajo Red Hat Linux. Hay programas gráficos de clientes de correo con características completas, tales como **Mozilla Mail** o **Ximian Evolution**, así como también programas basados en texto, tales como **mutt** y **pine**.

Para más instrucciones sobre el uso de estas aplicaciones, consulte el capítulo llamado *Aplicaciones de correo* en el *Manual del principiante de Red Hat Linux*.

El resto de esta sección se enfoca en asegurar la comunicación entre el cliente y el servidor.

11.5.1. Comunicación segura

Los MUAs populares incluídos con Red Hat Linux, tales como **Mozilla Mail**, `mutt` y `pine`, ofrecen sesiones encriptadas con SSL.

Al igual que otros servicios existentes en una red no cifrada, la información de correo electrónico importante, como nombres de usuario, contraseñas y mensajes, se puede interceptar y ver sin que tenga conocimiento el servidor o el cliente de correo. Al usar los protocolos estándar POP e IMAP, toda la información de autenticación se envía "limpiamente", sin encriptar, por lo que es posible para un intruso ganar acceso a las cuentas de usuarios reuniendo los nombres de los usuarios y sus contraseñas cuando estos son transmitidos sobre la red.

11.5.1.1. Clientes de correo electrónico seguros

Afortunadamente, la mayoría de los agentes MUA de Linux están diseñados para comprobar el correo mediante SSL. Para usar SSL al recuperar el correo, se debe activar esta opción en el cliente y en el servidor de correo.

SSL se activa muy fácilmente en el cliente, normalmente basta con pulsar un botón en el área de configuración del agente MUA o mediante una opción en el archivo de configuración del MUA. Los protocolos IMAP y POP seguros tienen números de puerto conocidos (993 y 995, respectivamente) que MUA utiliza para autenticar y descargar los mensajes.

11.5.1.2. Asegurar las comunicaciones de cliente de correo

Ofrecer cifrado SSL a los usuarios de IMAP y POP del servidor de correo es muy sencillo.

Primero, cree un certificado SSL. Esto se puede hacer de dos formas: solicitando a una *Certificate Authority* (*CA*) por un certificado SSL o mediante la creación de un certificado auto-firmado.

Atención

Los certificados auto-firmados solamente deberían ser usados para propósitos de prueba. Cualquier servidor usado en un ambiente de producción debería usar un certificado SSL emitido por una CA.

Para crear un certificado SSL con firma propia para IMAP, cambie al directorio `/usr/share/ssl/certs/` y escriba el comando siguiente como root:

```
make imapd.pem
```

Conteste todas las preguntas para completar el proceso.

Para crear un certificado SSL con firma propia para POP, cambie al directorio `/usr/share/ssl/certs/`, y escriba el comando siguiente como usuario root:

```
make ipop3d.pem
```

Una vez más, conteste todas las preguntas para completar el proceso.

Una vez terminado, use el comando `/sbin/service` para iniciar el demonio apropiado (`imaps` o `pop3s`). Luego, configure el servicio `imaps` o `pop3s` para que arranquen en los niveles de ejecución apropiados usando una utilidad initscript, tal como la **Herramienta de configuración de servicios** (`redhat-config-services`). Refiérase a Sección 1.4.2 para más información sobre las utilidades initscript.

Alternativamente, el comando `stunnel` puede ser usado como una envoltura de criptación SSL con el estándar, para los demonios no seguros, `imapd` o `pop3d`.

El programa `stunnel` utiliza librerías OpenSSL externas incluídas con Red Hat Linux para proporcionar criptografía robusta y proteger las conexiones. Es mejor solicitar a una *Certificate Authority* (*CA*) por un certificado SSL, pero es posible crear un certificado auto-firmado.

Para crear un certificado SSL auto-firmado, cámbiese al directorio `/usr/share/ssl/certs/` y escriba el comando siguiente:

```
make stunnel.pem
```

Una vez más, conteste todas las preguntas para completar el proceso.

Una vez que el certificado es generado, es posible usar el comando `stunnel` para iniciar el demonio de correo `imapd` usando el comando siguiente:

```
/usr/sbin/stunnel -d 993 -l /usr/sbin/imapd imapd
```

Una vez que este comando es emitido, es posible abrir un cliente de correo IMAP y conectarse al servidor de correo usando una encriptación SSL.

Para arrancar `pop3d` usando el comando `stunnel`, escriba el comando siguiente:

```
/usr/sbin/stunnel -d 993 -l /usr/sbin/pop3d pop3d
```

Para más información sobre el uso de `stunnel`, lea la página man `stunnel` o refiérase a los documentos en el directorio `/usr/share/doc/stunnel-<version-number>/`.

11.6. Recursos adicionales

La siguiente es una lista con la documentación adicional sobre las aplicaciones de correo.

11.6.1. Documentación instalada

- Información sobre la configuración de Sendmail es incluída con los paquetes `sendmail` y `sendmail-cf`.
  - `/usr/share/doc/sendmail/README.cf` — Contiene información en `m4`, ubicaciones de archivos para Sendmail, enviadores de correo soportados, cómo accesar las características avanzadas y más.
  - `/usr/share/doc/sendmail/README` — Contiene información sobre la estructura de directorios de Sendmail, soporte del protocolo IDENT, detalles sobre los permisos de directorios y los problemas comunes que estos permisos pueden causar si no se configuran de la forma correcta.

  Además, las páginas man de `sendmail` y `aliases` contienen información útil sobre varias opciones de Sendmail y la configuración adecuada del archivo Sendmail `/etc/mail/aliases`.
- `/usr/share/doc/fetchmail-<version-number>` — Contiene una lista completa de las características de Fetchmail en el archivo `FEATURES` y un documento `FAQ` introductorio.
- `/usr/share/doc/procmail-<version-number>` — Contiene un archivo `README` que proporciona una visión general de Procmail, un archivo `FEATURES` que explora cada característica del programa y una sección `FAQ` con respuestas a muchas de las preguntas comunes.

  Mientras aprende cómo Procmail funciona y cómo crear nuevas recetas, las siguientes páginas man son de gran utilidad:
  - `procmail` — Proporciona una vista general de cómo Procmail funciona y los pasos implicados cuando se esté filtrando correo.
  - `procmailrc` — Explica el formato del archivo `rc` usado para construir recetas.
  - `procmailex` — Proporciona un número de ejemplos de la vida real de recetas Procmail.
  - `procmailsc` — Explica la técnica de puntaje por pesos usada por Procmail para ver si una receta particular coincide un mensaje.
  - `/usr/share/doc/spamassassin-<version-number>/` — Este directorio contiene una gran cantidad de información sobre SpamAssassin. Reemplace `<version-number>` con el número de la versión del paquete `spamassassin`.

11.6.2. Sitios web útiles

- http://www.redhat.com/mirrors/LDP/HOWTO/Mail-Administrator-HOWTO.html — Proporciona una descripción general de cómo funciona el correo y examina las posibles soluciones y configuraciones de correo en los lados del cliente y del servidor.
- http://www.redhat.com/mirrors/LDP/HOWTO/Mail-User-HOWTO/ — Muestra el correo desde el punto de vista del usuario, investiga las aplicaciones clientes de correo populares y ofrece una introducción en temas como aliases, reenvio, respuestas automáticas, listas de correo, filtros y correo basura.
- http://www.redhat.com/mirrors/LDP/HOWTO/mini/Secure-POP+SSH.html — Demuestra una forma de recuperar el correo POP usando SSH con reenvío de puertos, para que se transfieran con seguridad las contraseñas y los mensajes.

- http://www.sendmail.net/ — Contiene noticias, entrevistas y artículos concernientes a Sendmail, incluyendo una vista ampliada de las muchas opciones disponibles.
- http://www.sendmail.org/ — Ofrece un desglose completo de las características técnicas de Sendmail y ejemplos de configuración.
- http://tuxedo.org/~esr/fetchmail — La página principal para Fetchmail, presentando un manual en línea y una sección FAQ completa.
- http://www.procmail.org/ — La página principal para Procmail con enlaces a listas de correo varias dedicadas a Procmail así como también varios documentos FAQ.
- http://www.ling.helsinki.fi/users/reriksso/procmail/mini-faq.html — Una sección FAQ excelente de Procmail, ofrece sugerencias para la solución de problemas, detalles sobre bloqueo de archivos y el uso de comodines.
- http://www.uwasa.fi/~ts/info/proctips.html — Contiene docenas de sugerencias que hacen el uso de Procmail mucho más fácil. Incluye instrucciones sobre cómo probar los archivos `.procmailrc` y usar el puntaje de Procmail para decidir si una acción particular debería ser tomada.
- http://www.spamassassin.org/ — El sitio oficial del proyecto SpamAssassin.

11.6.3. Libros relacionados

- *Sendmail* por Bryan Costales with Eric Allman et al; O'Reilly & Associates — Una buena referencia de Sendmail escrita con la asistencia del creador original de Delivermail y Sendmail.
- *Removing the Spam: Email Processing and Filtering* por Geoff Mulligan; Addison-Wesley Publishing Company — Un volumen que muestra varios métodos usados por los administradores de correo usando herramientas establecidas, tales como Sendmail y Procmail, para manejar los problemas del correo basura.
- *Internet Email Protocols: A Developer's Guide* por Kevin Johnson; Addison-Wesley Publishing Company — Proporciona una revisión profunda de los principales protocolos y la seguridad que éstos proporcionan.
- *Managing IMAP* por Dianna Mullet and Kevin Mullet; O'Reilly & Associates — Detalla los pasos requeridos para configurar un servidor IMAP.

Capítulo 12.
Berkeley Internet Name Domain (BIND)

En la mayoría de las redes modernas, incluyendo la Internet, los usuarios localizan otras máquinas por su nombre. Esto libera a los usuarios de la pesada tarea de recordar la dirección numérica de los recursos de red. La forma más efectiva de configurar una red para permitir tales conexiones basadas en nombres es configurando un *Domain Name Service* (*DNS*) o *servidor de nombres*, el cual resuelve los nombres de hosts en la red a direcciones numéricas y viceversa.

Este capítulo revisa el servidor de nombres incluído con Red Hat Linux, servidor DNS *Berkeley Internet Name Domain* (*BIND*), con énfasis en la estructura de sus archivos de configuración y en cómo deberían ser administrados localmente y remótamente.

Para instrucciones sobre la configuración de BIND usando la **Herramienta de configuración de Bind** (`redhat-config-bind`) gráfica, consulte el capítulo llamado *Configuración de BIND* en el *Manual de personalización de Red Hat Linux*.


Aviso

Si utiliza la **Herramienta de configuración de Bind**, no edite manualmente ningún archivo de configuración BIND pues todos los cambios serán sobreescritos la próxima vez que utilice la **Herramienta de configuración de Bind**.

12.1. Introducción a DNS

Cuando hosts en una red se conectan a través de sus nombres de máquinas, también llamado *nombre de dominio completamente cualificado (FQDN)*, DNS es usado para asociar los nombres de las máquinas a las direcciones IP para el host.

El uso de nombres de un dominio completamente cualificado y DNS tiene ventajas para los administradores del sistema, éstos dan a los administradores flexibilidad a la hora de cambiar las direcciones IP para máquinas individuales sin realizar preguntas sobre el nombre en las máquinas. Por otro lado, los administradóres pueden revolver cuáles máquinas manejan consultas basadas en nombre .

DNS es normalmente implementado usando servidores centralizados que autorizan algunos dominios y se refieren a otros servidores DNS para otros dominios.

Cuando un host cliente solicita información desde un servidor de nombres, usualmente se conecta al puerto 53. El nombre de servidor luego intenta resolver el FQDN basado en su librería de resolución, la cual puede contener información de autorización sobre el host solicitado o datos en caché de una consulta anterior. Si el nombre del servidor no tiene la respuesta en su librería de resolución, consultará otros nombres de servidores, llamados *servidores de nombres de root*, para determinar cuáles servidores de nombres son fidedignos para el FQDN en cuestión. Luego, con esa información, consulta los servidores de nombres autoritarios para determinar la dirección IP del host solicitado. Si se está realizando una búsqueda inversa, se usa el mismo procedimiento, excepto que la consulta es realizada con una dirección IP desconocida en vez de un nombre.

12.1.1. Zonas de servidores de nombres

En Internet, el FQDN de un host se puede analizar en diversas secciones y estas secciones se analizan a su vez por orden jerárquico, como en un árbol el tronco, las ramas primarias, las ramas secundarias, etc. Por ejemplo, considere el siguiente FDNQ:

```
bob.sales.example.com
```

Cuando miramos cómo un FQDN es resuelto para encontrar la dirección IP que se relaciona a un sistema particular, lea el nombre de derecha a izquierda, con cada nivel de la jerarquía dividido por puntos (`.`). En nuestro ejemplo, `com` define el *dominio de nivel superior* para este FQDN. El nombre `example` es un subdominio bajo `com`, mientras que `sales` es un subdominio bajo `example`. El nombre más hacia la izquierda, `bob`, identifica una máquina específica.

Aparte del nombre del dominio, cada sección se llama *zona*, la cual define un *espacio de nombre* particular. Un espacio de nombre, controla los nombres de los subdominios de la izquierda. Aunque en el ejemplo solamente hay dos subdominios, un FQDN tiene que contener al menos un subdominio pero puede incluir muchos más; depende de la organización del espacio de nombres elegido.

Las zonas son definidas en servidores de nombres autorizados a través del uso de *archivos de zona*, lo cual describen el espacio de nombres de esa zona, los servidores de correo a ser utilizados por un dominio particular o sub-dominio, y más. Los archivos de zona son almancenados en *servidores de nombres primarios* (también llamados *servidores de nombres maestro*), los cuales son verdaderamente autorizados y donde los cambios se hacen a los archivos, y *servidores de nombres secundarios* (también llamados *servidores de nombres esclavos*), que reciben sus archivos de zona desde los servidores de nombres primarios. Cualquier servidor de nombres puede ser un servidor primario y secundario para zonas diferentes al mismo tiempo, y también pueden ser considerados autoritarios para múltiples zonas. Todo depende de cómo se configure el servidor de nombres.

12.1.2. Tipos de servidores de nombres

Existen cuatro tipos de configuración de servidores de nombres primarios:

- *maestro* — Almacena los registros de las zonas originales y de autoridad para un cierto espacio de nombres, contestando preguntas de otros servidores de nombres buscando respuestas concernientes a ese espacio de nombres.
- *esclavo* — Responde a las peticiones que provienen de otros servidores de nombres y que se refieren a los espacios de nombres sobre los que tiene autoridad. Sin embargo, los servidores esclavos obtienen la información de sus espacios de nombres desde los servidores maestros.
- *sólo caché* — ofrece servicios de resolución de nombres a direcciones IP pero no tiene ninguna autoridad sobre ninguna zona. Las respuestas en general se introducen en un caché por un período de tiempo fijo, la cual es especificada por el registro de zona recuperado.
- *reenvío* — Reenvía las peticiones a una lista específica de servidores de nombres para la resolución de nombres. Si ninguno de los servidores de nombres especificados puede resolver los nombres, la resolución falla.

Un servidor de nombres puede ser uno o más de estos tipos. Por ejemplo, un servidor de nombres puede ser un maestro para algunas zonas, un esclavo para otras y sólo ofrecer el reenvío de resoluciones para otras.

12.1.3. BIND como un servidor de nombres

BIND realiza la resolución de nombres a través del demonio `/usr/sbin/named`. BIND también incluye una utilidad de administración llamada `/usr/sbin/rndc`. Se puede encontrar más información sobre `rndc` en Sección 12.4.

BIND almacena sus archivos de configuración en los siguientes dos lugares:

- `/etc/named.conf` — El archivo de configuración para el demonio `named`.

- directorio `/var/named/` — El directorio de trabajo `named` el cual almacena zonas, estadísticas y archivos caché.

Las próximas secciones revisan los archivos de configuración de BIND en más detalle.

12.2. `/etc/named.conf`

El archivo `named.conf` es una colección de declaraciones usando opciones anidadas rodeadas por caracteres de llaves, `{ }`. Los administradores deben tener mucho cuidado cuando estén modificando `named.conf` para evitar errores sintácticos puesto que hasta el error más pequeños puede impedir que el servicio `named` arranque.

Aviso

No modifique manualmente el archivo `/etc/named.conf` o cualquier archivo en el directorio `/var/named/` si está usando la **Herramienta de configuración de Bind**. Cualquier cambio manual a esos archivos serán sobreescritos la próxima vez que se use **Herramienta de configuración de Bind**.

Un archivo típico de `named.conf` está organizado de forma similar al ejemplo siguiente:

```
<statement-1> ["<statement-1-name>"] [<statement-1-class>] {
   <option-1>;
   <option-2>;
   <option-N>;
};

<statement-2> ["<statement-2-name>"] [<statement-2-class>] {
   <option-1>;
   <option-2>;
   <option-N>;
};

<statement-N> ["<statement-N-name>"] [<statement-N-class>] {
   <option-1>;
   <option-2>;
   <option-N>;
};
```

12.2.1. Tipos de declaraciones comúnes

Los siguientes tipos de sentencias son usados a menudo en `/etc/named.conf`:

12.2.1.1. Declaración `acl`

La sentencia `acl` (o sentencia de control de acceso) define grupos de hosts a los que se les puede permitir o negar el acceso al servidor de nombres.

Una declaración `acl` tiene la siguiente forma:

```
acl <acl-name> {
    <match-element>;
    [<match-element>; ...]
};
```

En esta declaración, sustituya *<acl-name>* con el nombre de la lista de control de acceso y reemplace *<match-element>* con una lista de direcciones IP separada por puntos y comas. La mayoría de las veces, una dirección IP individual o notación de red IP (tal como `10.0.1.0/24`) es usada para identificar las direcciones IP dentro de la declaración `acl`.

La siguiente lista de control de acceso ya están definidas como palabras claves para simplificar la configuración:

- `any` — Hace coincidir todas las direcciones IP.
- `localhost` — Hace coincidir cualquier dirección IP que se use el sistema local.
- `localnets` — Hace coincidir cualquier dirección IP en cualquier red en la que el sistema local está conectado.
- `none` — No concuerda ninguna dirección IP.

Cuando lo utilice con otras pautas (tales como declaraciones `options`), las declaraciones `acl` pueden ser muy útiles al asegurar el uso correcto de su servidor de nombres BIND.

El ejemplo siguiente define dos listas de control de acceso y utiliza una declaración `options` para definir cómo son tratadas en el servidor de nombres:

```
 acl black-hats {
   10.0.2.0/24;
   192.168.0.0/24;
};

acl red-hats {
   10.0.1.0/24;
};

options {
   blackhole { black-hats; };
   allow-query { red-hats; };
   allow-recursion { red-hats; };
}
```

Este ejemplo contiene dos listas de control de acceso, `black-hats` y `red-hats`. Los hosts en la lista `black-hats` se les niega el acceso al servidor de nombres, mientras que a los hosts en la lista `red-hats` se les dá acceso normal.

12.2.1.2. Declaración `include`

La declaración `include` permite incluir archivos en un `named.conf`. De esta forma los datos de configuración confidenciales (tales como `claves`) se pueden colocar en un archivo separado con permisos de restricción.

Una declaración `include` tiene la forma siguiente:

```
include  "<file-name>"
```

En esta declaración, *<file-name>* es reemplazado con una ruta absoluta a un archivo.

12.2.1.3. Declaración `options`

La declaracion `options` define opciones de configuración de servidor globales y configura otras declaraciones por defecto. Puede ser usado para especificar la ubicación del directorio de trabajo `named`, los tipos de consulta permitidos y mucho más.

La declaración `options` toma la forma siguiente:

```
options {
        <option>;
 [<option>; ...]
};
```

En esta declaración, las directivas *<option>* son reemplazadas con una opción válida.

Las siguientes son opciones usadas a menudo:

- `allow-query` — Especifica cuáles hosts tienen permitido consultar este servidor de nombres. Por defecto, todos los hosts tienen derecho a consultar. Una lista de control de acceso, o una colección de direcciones IP o redes se puede usar aquí para sólo permitir a hosts particulares hacer consultas al servidor de nombres.
- `allow-recursion` — Parecida a la opción `allow-query`, salvo que se aplica a las peticiones recursivas. Por defecto, todos los hosts están autorizados a presentar peticiones recursivas en un servidor de nombres.
- `blackhole` — Especifica cuáles hosts no tienen permitido consultar al servidor de nombres.
- `directory` — Reemplaza el directorio de trabajo `named` en vez del directorio predeterminado `/var/named`.
- `forward` — Controla el comportamiento de reenvío de una directiva `forwarders`.

  Se aceptan las siguientes opciones:

  - `first` — Indica que los servidores de nombres especificados en la directiva `forwarders` sean consultados antes de que `named` intente resolver el nombre el mismo.
  - `only` — Indica que `named` no intente la resolución de nombres él mismo en el evento de que consultas a los servidores de nombres especificados en la directiva `forwarders` fallen.

- `forwarders` — Especifica una lista de direcciones IP válidas para los servidores de nombres donde las peticiones se pueden reenviar para ser resueltas.
- `listen-on` — Especifica la interfaz de red en la cual `named` escucha por solicitudes. Por defecto, todas las interfaces son usadas.

  De esta forma, si el servidor DNS es también una gateway, BIND se puede configurar para sólo contestar solicitudes que se originan desde algunas de las redes.

  Una directiva `listen-on` se puede ver como:

  ```
  options {
     listen-on { 10.0.1.1; };
  };
  ```

  De esta forma, solamente las peticiones que llegan desde la interfaz de red sirviendo a la red privada (`10.0.1.1`) serán aceptadas.
- `notify` — Controla si `named` notifica a los servidores esclavos cuando una zona es actualizada. Acepta las opciones siguientes:

  - `yes` — Notifica a los servidores esclavos.
  - `no` — No notifica a los servidores esclavos.

- `explicit` — Sólo notifica a los servidores esclavos en una lista `also-notify` dentro de una declaración de zona.

- `pid-file` — Especifica la ubicación del archivo del proceso ID creado por `named`.
- `statistics-file` — Permite especificar la localización alternativa de los archivos de estadísticas. Por defecto, las estadísticas de `named` son guardadas al archivo `/var/named/named.stats`.

Existen numerosas opciones disponibles, muchas de ellas dependen unas de otras para poder funcionar correctamente. Consulte el *Manual de referencia para el administrador de BIND 9* en Sección 12.7.1 y la página del manual para `bind.conf` para más detalles.

12.2.1.4. Declaración `zone`

Una declaración `zone` define las características de una zona tal como la ubicación de su archivo de configuración y opciones específicas de la zona. Esta declaración puede ser usada para ignorar las declaraciones globales `options`.

Una declaración `zone` tiene la forma siguiente:

```
zone <zone-name> <zone-class> {
     <zone-options>;
     [<zone-options>; ...]
};
```

En esta declaración, <*zone-name*> es el nombre de la zona, <*zone-class*> es la clase opcional de la zona, y <*zone-options*> es una lista de opciones que caracterizan la zona.

El atributo <*zone-name*> para la declaración de zona es particularmente importante, pues es el valor por defecto asignado para la directiva `$ORIGIN` usada dentro del archivo de zona correspondiente localizado en el directorio `/var/named/`. El demonio `named` anexa el nombre de la zona a cualquier nombre de dominio que no esté completamente cualificado listado en el archivo de zona.

Por ejemplo, si una declaración `zone` define el espacio de nombres para `example.com`, utilice `example.com` como el <*zone-name*> para que sea colocado al final de los nombres de hosts dentro del archivo de zona `example.com`.

Para más información sobre los archivos de zona, consulte Sección 12.3.

Las opciones más comunes para la declaración `zone` incluyen lo siguiente:

- `allow-query` — Especifica los clientes que se autorizan para pedir información sobre una zona. Por defecto, todas las peticiones de información son autorizadas.
- `allow-transfer` — Especifica los servidores esclavos que están autorizados para pedir una transferencia de información de la zona. Por defecto, todas las peticiones se autorizan.
- `allow-update` — Especifica los hosts que están autorizados para actualizar dinámicamente la información en sus zonas. Por defecto, no se autoriza la actualización de la información dinámicamente.

  Tenga cuidado cuando autorice a los hosts para actualizar la información de su zona. No habilite esta opción si no tiene confianza en el host que vaya a usar. Es mejor que el administrador actualice manualmente los registros de zona y que vuelva a cargar el servicio `named`.
- `file` — Especifica el nombre del archivo en el directorio de trabajo `named` que contiene los datos de configuración de zona.
- `masters` — La opción `masters` lista las direcciones IP desde las cuales solicitar información autorizada. Solamente se usa si la zona está definida como `tipo esclavo`.

- `notify` — Controla si `named` notifica a los servidores esclavos cuando una zona es actualizada. Acepta las opciones siguientes:
  - `yes` — Notifica a los servidores esclavos.
  - `no` — No notifica a los servidores esclavos.
  - `explicit` — Solamente notifica a los servidores esclavos especificados en una lista de `also-notify` dentro de la declaración de una zona.

- `type` — Define el tipo de zona.

  Abajo se muestra una lista de las opciones válidas:
  - `forward` — Dice al servidor de nombres que lleve a cabo todas las peticiones de información de la zona en cuestión hacia otros servidores de nombres.
  - `hint` — Tipo especial de zona que se usa para orientar hacia los servidores de nombres root que sirven para resolver peticiones de una zona que no se conoce. No se requiere mayor configuración que la establecida por defecto con una zona `hint`.
  - `master` — Designa el servidor de nombres actual como el que tiene la autoridad para esa zona. Una zona se puede configurar como tipo `master` si los archivos de configuración de la zona residen en el sistema.
  - `slave` — Designa el servidor de nombres como un servidor esclavo para esa zona. También especifica la dirección IP del servidor de nombres maestro para la zona.

- `zone-statistics` — Configura `named` para mantener estadísticas concerniente a esta zona, escribiéndola a su ubicación por defecto (`/var/named/named.stats`) o al archivo listado en la opción `statistics-file` en la declaración `server`. Consulte Sección 12.2.2 para más información sobre la declaración `server`.

12.2.1.5. Ejemplo de declaraciones de `zone`

La mayoría de los cambios al archivo `/etc/named.conf` de un servidor de nombres maestro o esclavo envuelven agregar, modificar o borrar declaraciones `zone`. Mientras que estas declaraciones `zone` pueden contener muchas opciones, la mayoría de los servidores de nombres requieren sólo un pequeño subconjunto para funcionar efectivamente. Las siguientes declaraciones `zone` son ejemplos muy básicos que ilustran la relación de servidores de nombres maestro-esclavo.

A continuación se muestra un ejemplo de una declaración de `zone` para un servidor de nombres primario hospedando `example.com` (`192.168.0.1`):

```
zone "example.com" IN {
  type master;
  file "example.com.zone";
  allow-update { none; };
};
```

En la declaración, la zona es identificada como `example.com`, el tipo es configurado a `master` y el servicio `named` se instruye para leer el archivo `/var/named/example.com.zone`. También le dice a `named` que no permita a ningún otro host que realice actualizaciones.

Una declaración de `zone` de servidor esclavo para `example.com` se ve un poco diferente comparado con el ejemplo anterior. Para un servidor esclavo, el tipo se coloca a `slave` y en lugar de la línea `allow-update` está una directiva diciéndole a `named` la dirección IP del servidor maestro.

Una declaración de `zone` para un servidor esclavo para example.com sería como sigue:

```
zone "example.com" {
  type slave;
  file "example.com.zone";
  masters { 192.168.0.1; };
};
```

Esta declaración `zone` configura `named` en el servidor esclavo a que busque por el servidor maestro en la dirección IP `192.168.0.1` por información sobre la zona `example.com`. La información que el servidor esclavo recibe desde el servidor maestro es guardada al archivo `/var/named/example.com.zone`.

12.2.2. Otros tipos de declaraciones

La lista siguiente muestra tipos de declaraciones usadas con menos frecuencia disponibles dentro de `named.conf`

- `controls` — Configura varios requerimientos de seguridad necesarios para usar el comando `rndc` para administrar el servicio `named`.

  Consulte Sección 12.4.1 para ver cómo la declaración `controls` se vería, incuyendo las varias opciones que se pueden usar.

- `key "<key-name>"` — Define una clave particular por nombre. Las claves son usadas para autenticar varias acciones, tales como actualizaciones seguras o el uso del comando `rndc`. Se usan dos opciones con `key`:

  - `algorithm <algorithm-name>` — El tipo de algoritma usado, tal como `dsa` o `hmac-md5`.
  - `secret "<key-value>"` — La clave encriptada.

  Consulte Sección 12.4.2 para instrucciones sobre cómo escribir una declaración `key`.

- `logging` — Permite el uso de múltiples tipos de registro, llamados *channels*. Usando la opción `channel` dentro de la declaración `logging`, se puede construir un tipo registro personalizado, con su propio nombre de archivo (`file`), tamaño límite (`size`), versión (`version`), y nivel de importancia (`severity`). Una vez que se haya definido el canal personalizado, se usa una opción `category` para clasificar el canal y comenzar a conectar cuando se reinicie `named`.

  Por defecto, `named` registra mensajes estándar al demonio `syslog`, que les sitúa en `/var/log/messages`. Esto se debe a que varios canales estándares se encuentran incorporados a BIND junto con varios niveles de severidad, tales como uno que maneja la información de mensajes de registros (`default_syslog`) y otro que maneja mensajes de depuración (`default_debug`). Una categoría por defecto, llamada `default`, usa los canales incorporados para hacer conexiones normales sin ninguna configuración especial.

  La personalización del proceso de conexión es un proceso que requiere una explicación muy detallada y no es el objetivo de este capítulo. Para información sobre la creación de registros personalizados BIND, consulte el *Manual de referencia del administrador de BIND 9* en Sección 12.7.1.

- `server` — Define opciones particulares que afectan como `named` debería actuar con respecto a servidores de nombres remotos, especialmente en lo que respecta a las notificaciones y transferencias de zonas.

  La opción `transfer-format` controla si un registro de recursos es enviado con cada mensaje (`one-answer`) o si registros de múltiples recursos son enviados con cada mensaje (`many-answers`). Mientras que `many-answers` es más eficiente, sólo los nuevos servidores de nombres BIND lo entienden.

- `trusted-keys` — Contiene claves públicas que usa DNS seguro, DNSSEC. Para mayor información sobre la seguridad de BIND, consulte la Sección 12.5.3.
- `view "<view-name>"` — Crea visualizaciones especiales dependiendo del host que contacta el servidor de nombres. Esto permite a determinados hosts recibir una respuesta que se refiere a una zona particular mientras que otros hosts reciben información completamente diferente. Alternativamente, ciertos hosts pueden estar autorizados para acceder a determinadas zonas mientras que otros menos autorizados, sólo pueden hacer peticiones a otras zonas.

  Se pueden usar múltiples visualizaciones, siempre y cuando sus nombres sean únicos. La opción `match-clients` especifica la dirección IP que aplica a una vista particular. Cualquier declaración de `options` puede también ser usada dentro de una vista, ignorando las opciones globales ya configuradas por `named`. La mayoría de las sentencias `view` contienen múltiples declaraciones `zone` que aplican a la lista `match-clients`. El orden en que las sentencias `view` son listadas es importante, pues la primera sentencia `view` que coincida con una dirección IP de cliente particular es usada.

  Consulte Sección 12.5.2 para más información sobre la declaración `view`.

12.2.3. Etiquetas de comentarios

La siguiente es una lista de las etiquetas de comentarios válidas usadas dentro de `named.conf`:

- `//` — Cuando se coloca al comienzo de una línea, esa línea es ignorada por `named`.
- `#` — Cuando se coloca al comienzo de una línea, esa línea es ignorada por `named`.
- `/*` y `*/` — Cuando el texto se coloca entre estas etiquetas, se ignora el bloque de texto por `named`.

12.3. Archivos de zona

Los *Archivos de zona* contienen información sobre un espacio de nombres particular y son almacenados en el directorio de trabajo `/var/named/` por defecto. Cada archivo de zona es llamado de acuerdo a la opción `file` en la declaración `zone`, usualmente en una forma que relaciona al dominio en cuestión e identifica el archivo como conteniendo datos de zona, tal como `example.com.zone`.

Cada archivo de zona contiene directivas y registros de recursos. Las *directivas* le dicen al servidor de nombres que realice tareas o aplique configuraciones especiales a la zona. Los *registros de recursos* define los parámetros de la zona y asignan identidades a hosts individuales. Las directivas son opcionales, pero los registros de recursos se requieren para proporcionar servicios de nombres a la zona.

Todas las directivas y registros de recursos deberían ir en sus propias líneas individuales.

Los comentarios se pueden colocar después de los punto y comas (`;`) en archivos de zona.

12.3.1. Directivas de archivos de zona

Las directivas comienzan con el símbolo de dollar (`$`) seguido del nombre de la directiva. Usualmente aparecen en la parte superior del archivo de zona.

Lo siguiente son directivas usadas a menudo:

- `$INCLUDE` — Dice a `named` que incluya otro archivo de zona en el archivo de zona donde se usa la directiva. Así se pueden almacenar configuraciones de zona suplementarias aparte del archivo de zona principal.

- `$ORIGIN` — Anexa el nombre del dominio a registros no cualificados, tales como aquellos con el nombre de host solamente.

  Por ejemplo, un archivo de zona puede contener la línea siguiente:
  ```
  $ORIGIN example.com
  ```
  Cualquier nombre usado en los registros de recursos que no terminen en un punto (`.`) tendrán `example.com` anexado.

  

  Nota

  El uso de la directiva `$ORIGIN` no es necesario si la zona es especificada en `/etc/named.conf` porque la zona es usada como el valor de la directiva `$ORIGIN` por defecto.

- `$TTL` — Ajusta el valor *Time to Live (TTL)* predeterminado para la zona. Este es el tiempo, en segundos, que un registro de recurso de zona es válido. Cada recurso puede contener su propio valor TTL, el cual ignora esta directiva.

  Cuando se decide aumentar este valor, permite a los servidores de nombres remotos hacer caché a la información de zona para un período más largo de tiempo, reduciendo el número de consultas para la zona y alargando la cantidad de tiempo requerido para proliferar cambios de registros de recursos.

12.3.2. Registros de recursos de archivos de zona

El componente principal de un archivo de zona es su registro de recursos.

Hay muchos tipos de registros de recursos de archivos de zona. A continuación le mostramos los tipos de registros más frecuentes:

- `A` — Registro de dirección que especifica una dirección IP que se debe asignar a un nombre, como en el ejemplo:
  ```
  <host>      IN      A      <IP-address>
  ```
  Si el valor `<host>` es omitido, el registro `A` apunta a una dirección IP por defecto para la parte superior del espacio de nombres. Este sistema es el objetivo para todas las peticiones no FQDN.

  Considere el siguiente ejemplo de registro `A` para el archivo de zona `example.com`:
  ```
                IN      A       10.0.1.3
  server1       IN      A       10.0.1.5
  ```
  Las peticiones para `example.com` son apuntadas a 10.0.1.3, mientras que las solicitudes para `server1.example.com` son dirigidas a 10.0.1.5.

- `CNAME` — Registro del nombre canónico, que enlaza un nombre con otro: también conocido como un alias.

  El próximo ejemplo indica a `named` que cualquier petición enviada a `<alias-name>` apuntará al host, `<real-name>`. Los registros `CNAME` son usados normalmente para apuntar a servicios que usan un esquema de nombres común, tal como `www` para servidores Web.
  ```
  <alias-name>      IN      CNAME       <real-name>
  ```
  En el ejemplo siguiente, un registro `A` vincula un nombre de host a una dirección IP, mientras que un registro `CNAME` apunta al nombre host comúnmente usado `www` para este.
  ```
  server1       IN      A       10.0.1.5
  www           IN      CNAME   server1
  ```

- `MX` — Registro de Mail eXchange, el cual indica dónde debería de ir el correo enviado a un espacio de nombres particular controlado por esta zona.
  ```
        IN      MX      <preference-value>   <email-server-name>
  ```

En este ejemplo, *<preference-value>* permite una clasificación numérica de los servidores de correo para un espacio de nombres, dando preferencia a algunos sistemas de correo sobre otros. El registro de recursos `MX` con el valor más bajo *<preference-value>* es preferido sobre los otros. Sin embargo, múltiples servidores de correo pueden tener el mismo valor para distribuir el tráfico de forma pareja entre ellos.

El *<email-server-name>* puede ser un nombre de servidor o FQDN.

```
IN      MX      10      mail.example.com.
IN      MX      20      mail2.example.com.
```

En este ejemplo, el primer servidor de correo `mail.example.com` es preferido al servidor de correo `mail2.example.com` cuando se recibe correo destinado para el dominio `example.com`.

- `NS` — Registro NameServer, el cual anuncia los nombres de servidores con autoridad para una zona particular.

  Este es un ejemplo de un registro `NS`:

  ```
  IN      NS      <nameserver-name>
  ```

  El *<nameserver-name>* debería ser un FQDN.

  Luego, dos nombres de servidores son listados como con autoridad para el dominio. No es importante si estos nombres de servidores son esclavos o si son maestros; ambos son todavía considerados con autoridad.

  ```
  IN      NS      dns1.example.com.
  IN      NS      dns2.example.com.
  ```

- `PTR` — Registro PoinTeR o puntero, diseñado para apuntar a otra parte del espacio de nombres.

  Los registros `PTR` son principalmente usados para invertir la resolución de nombres, pues ellos apuntan direcciones IP de vuelta a un nombre particular. Consulte Sección 12.3.4 para más ejemplos de registros `PTR` en uso.

- `SOA` — Registro Start Of Authority, que proclama información importante sobre la autoridad de determinados servidores sobre determinados espacios de nombres.

  Está situado detrás de las directivas, un registro `SOA` es el primer registro en un archivo de zona.

  El ejemplo siguiente muestra la estructura básica de un registro `SOA`:

  ```
  @       IN      SOA     <primary-name-server>       <hostmaster-email> (
                          <serial-number>
                          <time-to-refresh>
                          <time-to-retry>
                          <time-to-expire>
                          <minimum-TTL> )
  ```

  El símbolo `@` coloca la directiva `$ORIGIN` (o el nombre de la zona, si la directiva `$ORIGIN` no está configurada) como el espacio de nombres que esta siendo definido por este registro de recursos `SOA`. El servidor de nombres primario que tiene autoridad para este dominio es usado por el *<primary-name-server>*, y el correo electrónico de la persona a contactar sobre este espacio de nombres es sustituído por el *<hostmaster-email>*.

  El *<serial-number>* es incrementado cada vez que se cambia el archivo de zona para que así `named` sabrá que debería recargar esta zona. El parámetro *<time-to-refresh>* le dice a cualquier servidor esclavo cuánto tiempo debe esperar antes de preguntar al servidor de nombres maestro si se han realizado cambios a la zona. El valor *<serial-number>* es usado por el esclavo para determinar si esta usando datos de la zona desactualizados y si debería refrescarlos.

  El valor *<time-to-retry>* le dice al servidor de nombres esclavo sobre el intervalo de tiempo que tiene que esperar antes de emitir una petición de actualización de datos en caso de que el servidor de nombres maestro no le responda. Si el servidor maestro no ha respondido a una petición de actualización de datos antes que se acabe el intervalo de tiempo *<time-to-expire>*, el servidor esclavo para de responder como una autoridad por peticiones al espacio de nombres.

La opción *<minimum-TTL>* solicita que otro servidor de nombres guarde en caché la información de zona por al menos esta cantidad de tiempo (en segundos).

Con BIND, todos los tiempos son siempre referenciados en segundos. Sin embargo, es posible usar abreviaciones cuando se especifiquen unidades de tiempo además de segundos, tales como minutos (M), horas (H), días (D) y semanas (W). La Tabla 12-1 le muestra una cantidad de tiempo en segundos y el tiempo equivalente en otro formato.

Segundos	Otras unidades de tiempo
60	1M
1800	30M
3600	1H
10800	3H
21600	6H
43200	12H
86400	1D
259200	3D
604800	1W
31536000	365D

Tabla 12-1. Segundos comparados a otras unidades de tiempo

El ejemplo siguiente ilustra la forma que un regisro de recursos SOA puede tomar cuando es configurado con valores reales.

```
@      IN      SOA     dns1.example.com.      hostmaster.example.com. (
                       2001062501 ; serial
                       21600      ; refresh after 6 hours
                       3600       ; retry after 1 hour
                       604800     ; expire after 1 week
                       86400 )    ; minimum TTL of 1 day
```

12.3.3. Ejemplo de archivo de zonas

Vistos individualmente, las directivas y registros de recursos pueden ser difíciles de comprender. Sin embargo, cuando se colocan juntos en un mismo archivo, se vuelven más fáciles de entender.

El ejemplo siguiente muestra un archivo de zona muy básico.

```
$ORIGIN example.com
$TTL 86400
@      IN      SOA     dns1.example.com.      hostmaster.example.com. (
                       2001062501 ; serial
                       21600      ; refresh after 6 hours
                       3600       ; retry after 1 hour
                       604800     ; expire after 1 week
                       86400 )    ; minimum TTL of 1 day

       IN      NS      dns1.example.com.
       IN      NS      dns2.example.com.

       IN      MX      10      mail.example.com.
       IN      MX      20      mail2.example.com.
```

```
              IN      A       10.0.1.5

server1       IN      A       10.0.1.5
server2       IN      A       10.0.1.7
dns1          IN      A       10.0.1.2
dns2          IN      A       10.0.1.3

ftp           IN      CNAME   server1
mail          IN      CNAME   server1
mail2         IN      CNAME   server2
www           IN      CNAME   server2
```

En este ejemplo, las directivas estándar y los valores `SOA` son usados. Los servidores de nombres con autoridad se configuran como `dns1.example.com` y `dns2.example.com`, que tiene archivos `A` que los juntan a `10.0.1.2` y a `10.0.1.3`, respectivamente.

Los servidores de correo configurados con los registros `MX` apuntan a `server1` y `server2` a través de registros `CNAME`. Puesto que los nombres `server1` y `server2` no terminan en un punto (`.`), el dominio `$ORIGIN` es colocado después de ellos, expandiéndolos a `server1.example.com` y a `server2.example.com`. A través de registros de recursos relacionados `A`, se puede determinar sus direcciones IP.

Los servicios FTP y Web, disponibles en los nombres estándar `ftp.example.com` y `www.example.com`, son apuntados a los servidores apropiados usando registros `CNAME`.

12.3.4. Archivos de zona de resolución de nombres inversa

Un archivo de zona de resolución de nombres inversa es usado para traducir una dirección IP en un espacio de nombres particular en un FQDN. Se vé muy similar a un archivo de zona estándar, excepto que se usan registros de recursos `PTR` para enlazar las direcciones IP a un nombre de dominio completamente cualificado.

Un registro `PTR` se vería similar a esto:

```
<last-IP-digit>       IN      PTR     <FQDN-of-system>
```

El valor *<last-IP-digit>* se refiere al último número en una dirección IP que debería apuntar a un sistema FQDN particular.

En el ejemplo siguiente, las direcciones IP de la `10.0.1.20` a la `10.0.1.25` apuntan a los FQDNs correspondientes.

```
$ORIGIN 1.0.10.in-addr.arpa
$TTL 86400
@     IN      SOA     dns1.example.com.     hostmaster.example.com. (
                      2001062501 ; serial
                      21600      ; refresh after 6 hours
                      3600       ; retry after 1 hour
                      604800     ; expire after 1 week
                      86400 )    ; minimum TTL of 1 day

      IN      NS      dns1.example.com.
      IN      NS      dns2.example.com.

20    IN      PTR     alice.example.com.
21    IN      PTR     betty.example.com.
22    IN      PTR     charlie.example.com.
23    IN      PTR     doug.example.com.
24    IN      PTR     ernest.example.com.
25    IN      PTR     fanny.example.com.
```

Este archivo de zona se colocará en funcionamiento con una declaración `zone` en el archivo `named.conf` el cual se ve similar a lo siguiente:

```
zone "1.0.10.in-addr.arpa" IN {
  type master;
  file "example.com.rr.zone";
  allow-update { none; };
};
```

Hay muy poca diferencia entre este ejemplo y una declaración de `zone` estándar, excepto por el nombre de la zona. Observe que una zona de resolución de nombres inversa requiere que los primeros tres bloques de la dirección IP esten invertidos seguido por `.in-addr.arpa`. Esto permite asociar con la zona a un bloque único de números IP usados en el archivo de zona de resolución de nombres inversa.

12.4. Uso de `rndc`

BIND incluye una utilidad llamada `rndc` la cual permite la administración de línea de comandos del demonio `named` desde el host local o desde un host remoto.

Para prevenir el acceso no autorizado al demonio `named`, BIND utiliza un método de clave secreta compartida para otorgar privilegios a hosts. Esto significa que una clave idéntica debe estar presente en los archivos de configuración `/etc/named.conf` y en el `rndc`, `/etc/rndc.conf`.

12.4.1. Configuración de `/etc/named.conf`

Para que `rndc` se pueda conectar a un servicio `named`, debe haber una declaración `controls` en el archivo de configuración del servidor BIND `/etc/named.conf`.

La declaración `controls` mostrada abajo en el ejemplo permite a `rndc` conectarse desde un host local.

```
controls {
  inet 127.0.0.1 allow { localhost; } keys { <key-name>; };
};
```

Esta declaración le dice a `named` que escuche en el puerto por defecto TCP 953 de la dirección loopback y que permita comandos `rndc` provenientes del host local, si se proporciona la clave correcta. El valor *<key-name>* se relaciona con la declaración `key`, la cual esta también en el archivo `/etc/named.conf`. El ejemplo siguiente ilustra la declaración `key`.

```
key "<key-name>" {
  algorithm hmac-md5;
  secret "<key-value>";
};
```

En este caso, el *<key-value>* es una clave HMAC-MD5. Use el comando siguiente para generar claves HMAC-MD5:

```
dnssec-keygen -a hmac-md5 -b <bit-length> -n HOST <key-file-name>
```

Una clave con al menos un largo de 256-bit es una buena idea. La clave actual debería ser colocada en el área *<key-value>* se puede encontrar en *<key-file-name>*.

Atención

Debido a que `/etc/named.conf` está accesible a todo el mundo, es una buena idea colocarlo en la declaración `key` en un archivo separado que sólo esté accesible por root y luego utilizar una declaración `include` para referenciarlo, como se muestra en el ejemplo siguiente:

```
include "/etc/rndc.key";
```

12.4.2. Configuración de `/etc/rndc.conf`

La declaración `key` es la más importante en `/etc/rndc.conf`.

```
key "<key-name>" {
  algorithm hmac-md5;
  secret "<key-value>";
};
```

<key-name> y *<key-value>* deberían ser exáctamente los mismos que sus configuraciones en `/etc/named.conf`.

Para coincidir las claves especificadas en el archivo de configuración del servidor objetivo `/etc/named.conf`, agregue las líneas siguientes a `/etc/rndc.conf`.

```
options {
  default-server  localhost;
  default-key     "<key-name>";
};
```

Este comando configura una clave global por defecto. Sin embargo, el comando `rndc` también puede usar claves diferentes para servidores diferentes, como en el ejemplo siguiente:

```
server localhost {
  key  "<key-name>";
};
```

Atención

Asegúrese de que sólo el usuario root pueda leer y escribir al archivo `/etc/rndc.conf`.

12.4.3. Opciones de línea de comandos

Un comando `rndc` toma la forma siguiente:

```
rndc <options> <command> <command-options>
```

Cuando esté ejecutando `rndc` en una máquina local configurada de la forma correcta, los comandos siguientes están disponibles:

- `halt` — Para inmediatamente el servicio `named`.
- `querylog` — Registra todas las peticiones hechas a este servidor de nombres.

- `refresh` — Refresca la base de datos del servidor de nombres.
- `reload` — Recarga los archivos de zona pero mantiene todas las respuestas precedentes situadas en caché. Esto le permite realizar cambios en los archivos de zona sin perder todas las resoluciones de nombres almacenadas.

  Si los cambios sólo afectaron una zona específica, vuelva a cargar una zona añadiendo el nombre de la zona después del comando `reload`.
- `stats` — Descarga las estadísticas actuales de `named` al archivo `/var/named/named.stats`.
- `stop` — Detiene al servidor salvando todas las actualizaciones dinámicas y los datos de las *Transferencias de zona incremental* (*IXFR*) antes de salir.

Ocasionalmente, puede ser necesario ignorar las configuraciones por defecto en el archivo `/etc/rndc.conf`. Estan disponibles las siguientes opciones:

- `-c <configuration-file>` — Le dice a `rndc` que use un archivo de configuración diferente a `/etc/rndc.conf`.
- `-p <port-number>` — Especifica la utilización de un número de puerto diferente del predeterminado 953 para la conexión del comando `rndc`.
- `-s <server>` — Dice a `rndc` que envie el comando a un servidor distinto al `default-server` especificado en su archivo de configuración.
- `-y <key-name>` — Le permite especificar una clave distinta de la opción `default-key` en el archivo `/etc/rndc.conf`.

Se puede encontrar información adicional sobre estas opciones en la página del manual de `rndc`.

12.5. Características avanzadas de BIND

La mayoría de las implementaciones BIND solamente utilizan `named` para proporcionar servicios de resolución de nombres o para actuar como una autoridad para un dominio particular o sub-dominio. Sin embargo, BIND versión 9 tiene un número de características avanzadas que permiten un servicio DNS más seguro y avanzado.

Atención

Algunas de estas propiedades avanzadas como DNSSEC, TSIG y IXFR, solamente se pueden usar en los entornos de red que tengan servidores de nombres que soporten estas propiedades. Si su entorno de red incluye servidores de nombres no-BIND o versiones anteriores de BIND, verifique si alguna característica avanzada está soportada antes de intentar utilizarla.

Todas las propiedades citadas aquí se describen en el *Manual de referencia para el administrador de BIND 9* con mucho más detalle. Consulte Sección 12.7.1 para más información.

12.5.1. Mejoras al protocolo DNS

BIND soporta Transferencias de zona incremental (Incremental Zone Transfers, IXFR), donde un servidor de nombres sólo descargará las porciones actualizadas de una zona modificada en un servidor de nombres maestro. El proceso de transferencia estándar requiere que la zona completa sea transferida a cada servidor de nombres esclavo hasta por el cambio más pequeño. Para los dominios más famosos con archivos de zona muy largos y muchos servidores de nombres esclavos, IXFR hace que la notificación y los procesos de actualización sean menos exigentes en recursos.

Observe que IXFR solamente está disponible si usa al mismo tiempo la *actualización dinámica* para realizar los cambios en los registros de zona maestra. Si cambia los archivos de zona manualmente, tiene que usar AXFR. Encontrará más información sobre la actualización dinámica en el *Manual de referencia para el administrador de BIND 9*. Consulte Sección 12.7.1 para más información.

12.5.2. Vistas múltiples

A través del uso de la declaración `view` en `named.conf`, BIND puede presentar información diferente dependiendo de quién esté realizando la petición.

Esto es básicamente usado para negar entradas DNS confidenciales a clientes fuera de la red local, mientras se permiten consultas desde clientes dentro de la red local.

La declaración `view` usa la opción `match-clients` para coincidir direcciones IP o redes completas y darles opciones especiales y datos de zona.

12.5.3. Seguridad

BIND soporta un número de métodos diferentes para proteger la actualización y zonas de transferencia, en los servidores de nombres maestro y esclavo:

- *DNSSEC* — Abreviación de *DNS SECurity*, esta propiedad permite firmar con caracteres criptográficos zonas con una *clave de zona*.

  De esta manera, puede verificar que la información de una zona provenga de un servidor de nombres que la ha firmado con caracteres criptográficos con una clave privada, siempre y cuando el recipiente tenga esa clave pública del servidor de nombres.

  BIND versión 9 también soporta el método SIG(0) de clave publica/privada de autenticación de mensajes.
- *TSIG* — Abreviación para *Transaction SIGnatures*, esta característica permite que una transferencia desde el maestro al esclavo sea autorizada sólo después de verificar que una clave secreta compartida existe en los servidores maestro y en el esclavo.

  Esta característica fortalece el método estándar basado en direcciones IP de transferencia de autorización. Un intruso no solamente necesitará acceso a la dirección IP para transferir la zona, sino también necesitará conocer la clave secreta.

  BIND versión 9 también soporta *TKEY*, el cual es otro método de autorización de zonas de transferencia basado en clave secreta compartida.

12.5.4. IP versión 6

BIND versión 9 puede proporcionar servicios de nombres en ambientes IP versión 6 (IPv6) a través del uso de registros de zona `A6`.

Si el entorno de red incluye hosts IPv4 y IPv6, use el demonio de resolución ligero `lwresd` en todos los clientes de la red. Este demonio es muy eficiente, funciona solamente en caché y además entiende los nuevos registros `A6` y `DNAME` usados bajo IPv6. Consulte la página del manual `lwresd` para más información.

12.6. Errores comunes que debe evitar

Es normal que los principiantes cometan errores modificando los archivos de configuración BIND. Asegúrese de evitar los siguientes errores:

- *Asegúrese que aumenta el número de serie cuando esté modificando un archivo de zona.*

  Si el número de serie no se incrementa, puede ser que el servidor de nombres tenga la información nueva correcta, pero los servidores esclavos nunca serán notificados del cambio ni intentaran actualizar sus datos de esa zona.

- *Preste atención a la utilización correcta de las llaves y de los puntos y comas en el archivo* `/etc/named.conf`.

  La omisión de un punto y coma o de una llave en una sección causará que `named` se niegue a arrancar.

- *Recuerde colocar puntos (.) en los archivos de zona después de todos los FQDNs y omítalos en los nombres de máquinas.*

  Un punto al final de un nombre de dominio denota un nombre de dominio completamente cualificado. Si el punto es omitido, entonces `named` añade el nombre de la zona o el valor `$ORIGIN` para completarlo.

- *Si un firewall está bloqueando las conexiones con el programa* `named` *a otros servidores de nombres, modifique su archivo de configuración.*

  Por defecto, la versión 9 de BIND usa los puertos aleatorios por encima de 1024 para consultar otros servidores de nombres. Algunos firewalls, sin embargo, esperan que todos los servidores de nombres se comuniquen usando solamente el puerto 53. Puede forzar `named` a que use el puerto 53 añadiendo la línea siguiente a la declaración `options` de `/etc/named.conf`:

  ```
  query-source address * port 53;
  ```

12.7. Recursos adicionales

Las siguientes fuentes de información le proporcionarán recursos adicionales con respecto a BIND.

12.7.1. Documentación instalada

- BIND presenta un rango completo de documentación instalada cubriendo muchos tópicos diferentes, cada uno colocado en su propio directorio:
  - `/usr/share/doc/bind-<version-number>/` — Contiene un archivo `README` con una lista de las características más recientes.
  - `/usr/share/doc/bind-<version-number>/arm/` — Contiene una versión en HTML y SGML del *Manual de referencia para el administrador de BIND 9*, el cual detalla los requerimientos de recursos de BIND, cómo configurar diferentes tipos de servidores de nombres, balancear cargas y otros temas avanzados. Para la mayoría de los usuarios nuevos de BIND, este es el mejor lugar para comenzar.
  - `/usr/share/doc/bind-<version-number>/draft/` — Contiene documentos técnicos varios que revisan problemas relacionados con el servicio DNS y algunos métodos propuestos para solucionarlos.
  - `/usr/share/doc/bind-<version-number>/misc/` — Contiene documentos diseñados para referenciar problemas avanzados. Los usuarios de la versión 8 de BIND deberían consultar el documento `migration` para cambios específicos que se deben hacer cuando se esté moviendo a BIND 9. El archivo `options` lista todas las opciones implementadas en BIND 9 que son usadas en el archivo `/etc/named.conf`.

- `/usr/share/doc/bind-<version-number>/rfc/` — Cada documento RFC relacionado a BIND está en este directorio.

- La página `man named` — Explora argumentos varios que se pueden usar para controlar el demonio de servidor de nombres BIND.
- La página `man named.conf` — Una lista completa de las opciones disponibles dentro del archivo de configuración `named`.
- La página `man rndc` — Explica las diferentes opciones disponibles cuando se utilice el comando `rndc` para controlar un servidor de nombres BIND.
- La página `man rndc.conf` — Una lista completa de opciones disponibles dentro del archivo de configuración `rndc`.

12.7.2. Sitios web de utilidad

- http://www.isc.org/products/BIND — La página principal del proyecto BIND conteniendo información sobre los lanzamientos recientes así como también la versión PDF de *Manual de referencia para el administrador de BIND 9*.
- http://www.redhat.com/mirrors/LDP/HOWTO/DNS-HOWTO.html — Cubre el uso de BIND como un servidor de nombres de caché y la configuración de varios archivos de zona necesarios para servir como servidor de nombres para un dominio.

12.7.3. Libros relacionados

- *DNS y BIND* por Paul Albitz y Cricket Liu; O'Reilly & Associates — Una referencia popular que explica opciones de configuración comunes y esotéricas de BIND, así como también proporcionar estrategias para asegurar su servidor DNS.
- *The Concise Guide to DNS and BIND* por Nicolai Langfeldt; Que — Hace referencia a la conexión entre servicios de red múltiples y BIND, haciendo énfasis en los tópicos técnicos orientados a tareas.

redhat.

Capítulo 13.

Lightweight Directory Access Protocol (LDAP)

Lightweight Directory Access Protocol (*LDAP*) es un conjunto de protocolos abiertos usados para acceder información guardada centralmente a través de la red. Está basado en el estándar *X.500* para compartir directorios, pero es menos complejo e intensivo en recursos. Por esta razón, a veces se habla de LDAP como "*X.500 Lite*."

Como X.500, LDAP organiza la información en un modo jerárquico usando directorios. Estos directorios pueden almacenar variedad de información y se pueden incluso usar de forma similar a Network Information Service (NIS), permitiendo que cualquiera pueda acceder a su cuenta desde cualquier máquina en la red activa LDAP.

Sin embargo, en la mayoría de los casos, LDAP se usa simplemente como directorio telefónico virtual, permitiendo a los usuarios acceder fácilmente a información de contacto para otros usuarios. Pero LDAP va más lejos que un directorio telefónico tradicional, ya que es capaz de propagar sus directorios a otros servidores LDAP por todo el mundo, proporcionando acceso global a la información. Sin embargo, en este momento LDAP se usa más dentro de organizaciones individuales, como universidades, departamentos del gobierno y compañías privadas.

LDAP es un sistema cliente servidor. El servidor puede usar variedad de bases de datos para guardar un directorio, cada uno optimizado para operaciones de lectura rápidas y en gran volúmen. Cuando una aplicación de cliente LDAP se conecta a un servidor LDAP puede, o bien consultar un directorio, o intentar modificarlo. En el evento de una consulta, el servidor, o bien la contesta, o, si no puede contestar localmente, puede dirigir la consulta a un servidor LDAP que tenga la respuesta. Si la aplicación cliente está intentando modificar información en un directorio LDAP, el servidor verifica que el usuario tiene permiso para efectuar el cambio y después añade o actualiza la información.

Este capítulo hace referencia a la configuración y uso de OpenLDAP 2.0, una implementación de open source de los protocolos LDAPv2 y LDAPv3.

13.1. Razones por las cuales usar LDAP

La mayor ventaja de LDAP es que información para toda una organización se puede consolidar dentro de un repositorio central. Por ejemplo, en vez de administrar listas de usuarios para cada grupo dentro de una organización puede usar LDAP como directorio central accesible desde cualquier parte de la red. Puesto que LDAP soporta Secure Sockets Layer (SSL) y Transport Layer Security (TLS), los datos delicados se pueden proteger de los curiosos.

LDAP también soporta un número de bases de datos back-end en las que se guardan directorios. Esto permite que los administradores tengan la flexibilidad para desplegar la base de datos más indicada para el tipo de información que el servidor tiene que diseminar. También, ya que LDAP tiene una interfaz de programación de aplicaciones (API) bien definido, el número de aplicaciones activadas para LDAP son numerosas y están aumentando en cantidad y calidad.

En la parte negativa, LDAP puede ser complicado de configurar.

13.1.1. Mejoras en las características de OpenLDAP 2.0

OpenLDAP 2.0 incluye un número de características importantes.

- *Soporte LDAPv3* — OpenLDAP 2.0 soporta Simple Authentication and Security Layer (SASL), Transport Layer Security (TLS), y Secure Sockets Layer (SSL), entre otras mejoras. Muchos de los cambios en el protocolo desde LDAPv2 han sido diseñados para hacer LDAP más seguro.
- *IPv6 Support* — OpenLDAP soporta la versión 6 del protocolo de Internet de próxima generación.

- *LDAP sobre IPC* — OpenLDAP se puede comunicar dentro de un sistema usando comunicación interproceso (IPC). Esto mejora la seguridad al obviar la necesidad de comunicarse a través de la red.
- *API de C actualizada* — Mejora la forma en que los programadores se conectan para usar servidores de directorio LDAP.
- *Soporta LDIFv1* — Provee compatibilidad completa con el formato de intercambio de datos, Data Interchange Format (LDIF) version 1.
- *Servidor Stand-Alone mejorado* — Incluye un sistema de control de acceso actualizado, pooling de hilos, herramientas mejoradas y mucho más.

13.2. Terminología LDAP

Cualquier discusión sobre LDAP requiere un entendimiento básico del conjunto de términos específicos de LDAP:

- *entrada* — una entrada es una unidad en un directorio LDAP. Cada entrada se identifica por su único *Distinguished Name (DN)*.
- *atributos* — los atributos son piezas de información directamente asociada con la entrada. Por ejemplo, una organización puede ser representada como una entrada LDAP. Los atributos asociados con la organización pueden ser su número de fax, su dirección, etc. En un directorio LDAP las entradas pueden ser también personas. Atributos comunes de las personas podrían ser el número de teléfono y la dirección de e-mail.

  Algunos atributos son obligatorios mientras que otros son opcionales. Una definición *objectclass* determina qué atributos se requieren y cuáles no para cada entrada. Las definiciones de objectclass se encuentran en varios ficheros dentro del directorio `/etc/openldap/schema/`. Para más información sobre el esquema LDAP, consulte Sección 13.5.
- *LDIF* — El *Formato de intercambio de datos de LDAP* (LDIF) es una representación de texto ASCII de entradas LDAP. Los archivos usados para importar datos a los servidores LDAP deben estar en formato LDIF. Una entrada LDIF se ve similar al ejemplo siguiente:

```
[<id>]
dn: <distinguished name>
<attrtype>:
<attrvalue>
<attrtype>:
<attrvalue>
<attrtype>:
<attrvalue>
```

  Una entrada puede contener tantos pares `<attrtype>: <attrvalue>` como sean necesarios. Una línea en blanco indica el final de una entrada.

Atención

Todas las parejas `<attrtype>` y `<attrvalue>` *deben* estar definidas en el archivo esquema correspondiente para usar esta información.

Cualquier valor comprendido dentro de "<" y ">" es una variable y puede ser configurado cuando se cree una nueva entrada LDAP, excepto para `<id>`. El `<id>` es un número determinado por la aplicación que se usa para modificar la entrada.

Nota

Nunca debería tener la necesidad de modificar una entrada LDIF manualmente. En lugar de esto use una aplicación cliente LDAP, como las que aparecen en Sección 13.3.

13.3. Demonios y utilidades OpenLDAP

El grupo de librerías y herramientas OpenLDAP están esparcidas dentro de los paquetes siguientes:

- `openldap` — Contiene las librerías necesarias para ejecutar las aplicaciones del servidor y cliente OpenLDAP.
- `openldap-clients` — Contiene herramientas de línea de comandos para visualizar y modificar directorios en un servidor LDAP.
- `openldap-server` — Contiene los servidores y otras utilidades necesarias para configurar y ejecutar un servidor LDAP.

Hay dos servidores contenidos en el paquete `openldap-servers`: el *Demonio independiente LDAP* (`/usr/sbin/slapd`) y el *Demonio independiente de actualización de réplicas LDAP* (`/usr/sbin/slurpd`).

El demonio `slapd` es el servidor independiente LDAP mientras que el demonio `slurpd` es usado para sincronizar los cambios desde un servidor LDAP a otro en la red. El demonio `slurpd` sólo es usado cuando se trabaja con múltiples servidores LDAP.

Para llevar a cabo tareas administrativas, el paquete `openldap-server` instala las utilidades siguientes en el directorio `/usr/sbin/`:

- `slapadd` — Añade entradas desde un archivo LDIF a un directorio LDAP. Por ejemplo, el comando `/usr/sbin/slapadd -l ldif-input` leerá en el archivo LDIF, *`ldif-input`*, que contiene las nuevas entradas.
- `slapcat` — Extrae entradas de un directorio LDAP en el formato por defecto — Berkeley DB — y las guarda en un archivo LDIF. Por ejemplo, el comando `/usr/sbin/slapcat -l ldif-output` tendrá como resultado un fichero LDIF llamado *`ldif-output`* que contendrá las entradas para el directorio LDAP.
- `slapindex` — Re-indexa el directorio `slapd` basado en el contenido actual.
- `slappasswd` — Genera un valor de contraseña encriptada de usuario para ser usada con `ldapmodify` o el valor `rootpw` en el archivo de configuración `slapd`, `/etc/openldap/slapd.conf`. Ejecute el comando `/usr/sbin/slappasswd` para crear la contraseña.

Aviso

Asegúrese de detener `slapd` ejecutando `/usr/sbin/service slapd stop` antes de usar `slapadd`, `slapcat` o `slapindex`. De otro modo se pondrá en riesgo la consistencia del directorio LDAP.

Para más información sobre cómo utilizar cada una de estas utilidades, consulte las páginas del manual.

El paquete `openldap-clients` instala herramientas utilizadas para agregar, modificar y borrar entradas en un directorio LDAP dentro de `/usr/bin/` Estas herramientas incluyen lo siguiente:

- `ldapmodify` — Modifica las entradas en un directorio LDAP, aceptando la entrada por medio de un archivo o entrada estándar.
- `ldapadd` — Agrega entradas a su directorio aceptando entradas vía archivo o entrada estándar; `ldapadd` es en realidad un enlace duro a `ldapmodify -a`.
- `ldapsearch` — Busca entradas en el directorio LDAP usando un indicador de comandos shell.
- `ldapdelete` — Borra entradas de un directorio LDAP al aceptar instrucciones del usuario por medio de la entrada en la terminal o por medio de un archivo.

Con la excepción de `ldapsearch`, cada una de estas utilidades se usa más fácilmente haciendo referencia a un fichero que contiene los cambios que se deben llevar a cabo, que escribiendo un comando para cada entrada que se desea cambiar en un directorio LDAP. El formato de dicho fichero está esquematizado en las páginas del manual sobre cada aplicación.

13.3.1. NSS, PAM, y LDAP

Además de los paquetes OpenLDAP, Red Hat Linux incluye un paquete llamado `nss_ldap` el cual mejora la habilidad de LDAP para integrarse en Linux y otros ambientes UNIX.

El paquete `nss_ldap` provee los siguientes módulos:

- `/lib/libnss_ldap-<glibc-version>.so`
- `/lib/security/pam_ldap.so`

El módulo `libnss_ldap-<glibc-version>.so` permite a las aplicaciones buscar usuarios, grupos, hosts y otra información utilizando un directorio LDAP por medio de la interfaz *Nameservice Switch* (NSS). NSS permite a las aplicaciones autenticarse usando LDAP en conjunto con el servicio *Network Information Service* (NIS) y archivos de autenticación planos.

El módulo `pam_ldap` permite que las aplicaciones PAM puedan validar usuarios utilizando información almacenada en el directorio LDAP. Las aplicaciones PAM incluyen terminales, servidores de correo POP e IMAP y Samba. Al desarrollar un servidor LDAP en su red, todas estas situaciones de login pueden autentificarse contra una combinación de nombre de usuario y contraseña, simplificando en gran medida la administración.

13.3.2. PHP4, el Servidor Apache HTTP, y LDAP

Red Hat Linux incluye también un paquete que contiene un módulo LDAP para el lenguaje escrito por el servidor PHP.

El paquete `php-ldap` añade soporte LDAP al lenguaje empotrado en HTML, PHP4 a través del módulo `/usr/lib/php4/ldap.so`. Este módulo permite a los scripts PHP4 accesar información almacenada en un directorio LDAP.

Importante

Red Hat Linux ya no se entrega con el paquete `auth_ldap`. Este paquete provee soporte LDAP para las versiones 1.3 y anteriores de Servidor Apache HTTP. Para más detalles sobre el estado de este módulo, vea el sitio web de la Apache Software Foundation en http://www.apache.org/.

13.3.3. Aplicaciones cliente LDAP

Existen clientes gráficos de LDAP que soportan la creación y modificación de directorios, pero no se entregan con Red Hat Linux. Una de estas aplicaciones es **LDAP Browser/Editor** — Una herramienta basada en Java que está disponible en línea en http://www.iit.edu/~gawojar/ldap.

La mayoría de otros clientes LDAP acceden a directorios como sólo lectura, usándolos como referencia, pero sin alterar información de la organización. Algunos ejemplos de dichas aplicaciones son browsers de red basados en Mozilla, Sendmail, **Balsa**, **Pine**, **Evolution**, y **Gnome Meeting**.

13.4. Archivos de configuración de OpenLDAP

Los archivos de configuración OpenLDAP son instalados dentro del directorio `/etc/openldap/`. A continuación aparece una lista breve marcando los directorios y archivos más importantes:

- `/etc/openldap/ldap.conf` — Este es el archivo de configuración para todas las aplicaciones *cliente* que usan las librerías OpenLDAP tales como `ldapsearch`, `ldapadd`, Sendmail, **Pine**, **Balsa**, **Evolution**, y **Gnome Meeting**.
- `/etc/openldap/slapd.conf` — Este es el archivo configuración para el demonio `slapd`. Vea Sección 13.6.1 para más información sobre este archivo.
- Directorio `/etc/openldap/schema/` — Este subdirectorio contiene el esquema usado por el demonio `slapd`. Vea Sección 13.5 para más información sobre este directorio.

Nota

Si se instala el paquete `nss_ldap`, creará un archivo llamado `/etc/ldap.conf`. Este archivo es usado por los módulos PAM y NSS proporcionados por el paquete `nss_ldap`. Vea Sección 13.7 para más información sobre este archivo de configuración.

13.5. El directorio `/etc/openldap/schema/`

El directorio `/etc/openldap/schema/` almacena las definiciones LDAP, previamente ubicadas en los archivos `slapd.at.conf` y `slapd.oc.conf`. Todas las *definiciones de sintaxis de atributos* y las *definiciones de objectclass* son ahora ubicadas en los archivos de esquemas diferentes. Los archivos de esquemas son referenciados en `/etc/openldap/slapd.conf` usando líneas `include`, como se muestra en este ejemplo:

```
include  /etc/openldap/schema/core.schema
include  /etc/openldap/schema/cosine.schema
include  /etc/openldap/schema/inetorgperson.schema
include  /etc/openldap/schema/nis.schema
include  /etc/openldap/schema/rfc822-MailMember.schema
include  /etc/openldap/schema/autofs.schema
include  /etc/openldap/schema/kerberosobject.schema
```

Aviso

No debería modificar ninguno de los ítems de esquemas definidos en los archivos de esquemas instalados por OpenLDAP.

Puede extender el esquema usado por OpenLDAP para soportar tipos de atributos adicionales y clases de objetos usando los archivos de esquema por defecto como una guía. Para lograr esto, cree un archivo `local.schema` en el directorio `/etc/openldap/schema`. Referencie este nuevo esquema dentro de `slapd.conf` agregando las líneas siguientes debajo de las líneas `include` por defecto:

```
include         /etc/openldap/schema/local.schema
```

Luego, defina nuevos tipos de atributos y clases de objetos dentro del archivo `local.schema`. Muchas organizaciones usan los tipos de atributos existentes a partir de los archivos esquema instalados por defecto y agregan nuevas clases de objeto al archivo `local.schema`.

Ampliar esquemas para cubrir requerimientos específicos es un poco complicado y está más allá del ámbito de éste capítulo. Visite el http://www.openldap.org/doc/admin/schema.html para más información sobre cómo escribir nuevos archivos de esquemas.

13.6. Descripción general de la configuración de OpenLDAP

Esta sección explica rápidamente la instalación y la configuración del directorio OpenLDAP. Para más información, consulte las URLs siguientes:

- http://www.openldap.org/doc/admin/quickstart.html — El manual *Quick-Start Guide* en el sitio web de OpenLDAP.
- http://www.redhat.com/mirrors/LDP/HOWTO/LDAP-HOWTO.html — El *LDAP Linux HOWTO* del Proyecto de documentación de Linux, reflejado en el sitio web de Red Hat.

Los pasos básicos para crear un servidor LDAP son los siguientes:

1. Instale los RPMs `openldap`, `openldap-servers` y `openldap-clients`.
2. Edite el archivo `/etc/openldap/slapd.conf` para referenciar su dominio y servidor LDAP. Refiérase a Sección 13.6.1 para más informacíón sobre como editar este archivo.
3. Inicie `slapd` con el comando:
   `/sbin/service/ldap start`

   Después de que haya configurado LDAP correctamente, puede usar `chkconfig`, `ntsysv`, o **Herramienta de configuración de servicios** para configurar LDAP para que se inicie en el momento de arranque. Para más información sobre cómo configurar servicios, refiérase al capítulo titulado *Controlar el acceso a servicios* en el *Manual de personalización de Red Hat Linux*.
4. Agregue entradas a su directorio LDAP con `ldapadd`.
5. Use `ldapsearch` para ver si `slapd` accede a la información correctamente.
6. Llegados a este punto, su directorio LDAP debería estar funcionando correctamente y entonces puede configurar aplicaciones habilitadas para LDAP para que usen el directorio LDAP.

13.6.1. Modificar `/etc/openldap/slapd.conf`

Para poder usar el servidor LDPA `slapd`, tendrá que modificar su archivo de configuración, `/etc/openldap/slapd.conf`. Debe editar este archivo para especificar el dominio y servidor correcto.

La línea de `sufijo` nombra el dominio para el cual el servidor LDPA proveerá información y deberá ser cambiado desde:

```
suffix          "dc=your-domain,dc=com"
```

para que refleje un nombre de dominio completamente cualificado. Por ejemplo:

```
suffix          "dc=example,dc=com"
```

La entrada `rootdn` en el *Distinguished Name* (*DN*) para un usuario que no está restringido por el control de acceso o los parámetros de límite administrativos fijados para operaciones en el directorio LDAP. Se puede pensar en el usuario `rootdn` como el usuario raíz para el directorio LDAP. En el archivo de configuración, cambie la línea `rootdn` de su valor por defecto a algo similar a lo siguiente:

```
rootdn          "cn=root,dc=example,dc=com"
```

Si intenta poblar el directorio LDAP sobre la red, cambie la línea `rootpw` — reemplazando el valor por defecto con una contraseña encriptada. Para crear una contraseña encriptada, escriba el comando siguiente:

```
slappasswd
```

Se le pedirá ingresar y re-ingresar la contraseña, luego el programa imprime la contraseña resultante encriptada al terminal.

Luego, copie la nueva contraseña encriptada en el archivo `>/etc/openldap/slapd.conf` en alguna de las líneas `rootpw` y elimine el símbolo numeral (`#`).

Cuando termine, la línea debería de verse como el ejemplo siguiente:

```
rootpw {SSHA}vv2y+i6V6esazrIv70xSSnNAJE18bb2u
```

Aviso

Las contraseñas LDAP, incluyendo la directiva `rootpw` especificada en `/etc/openldap/slapd.conf`, son enviadas sobre la red en *unencrypted*, a menos que active la encriptación TLS.

Para activar la encriptación TLS revise los comentarios en `/etc/openldap/slapd.conf` y vea la página del manual para `slapd.conf`.

Para mayor seguridad, la directriz `rootpw` debería ser colocada entre comentarios después de poblar el directorio LDAP simplemente escribiendo el símbolo numeral (`#`).

Cuando use la herramienta de línea de comandos `/usr/sbin/slapadd` localmente para poblar el directorio LDAP, el uso de la directiva `rootpw` no es necesario.

Importante

Debe ser usuario root para usar `/usr/sbin/slapadd`. Sin embargo, el servidor de directorio corre como el usuario `ldap`. Por lo tanto el servidor de directorio no podrá modificar ningún archivo creado

por `slapadd`. Para corregir este detalle, después que termine de usar `slapadd`, escriba el comando siguiente:

```
chown -R ldap /var/lib/ldap
```

13.7. Configurar su sistema para la autenticación mediante OpenLDAP

Este apartado ofrece una supervisión de cómo configurar un sistema Red Hat Linux para autenticar usando OpenLDAP. A menos que usted sea un experto de OpenLDAP, necesitará más información de la proporcionada aquí. Para obtenerla remítase a Sección 13.9.

Instale el paquete LDAP necesario

Primero, debería asegurarse de tener los paquetes apropiados en ambos, el servidor LDAP y la máquina cliente LDAP. El servidor LDAP requiere el paquete `openldap-server`.

Los paquetes `openldap`, `openldap-clients`, y `nss_ldap` necesitan estar instalados en todas las máquinas LDAP clientes.

Modifique los archivos de configuración

- En el servidor, modifique el archivo `/etc/openldap/slapd.conf` en el servidor LDAP para asegurarse de que se corresponde con las especificaciones de su organización. Por favor refiérase a Sección 13.6.1 para instrucciones sobre la modificación de `slapd.conf`.
- En las máquinas clientes, ambos archivos `/etc/ldap.conf` y `/etc/openldap/ldap.conf` necesitan contener el servidor apropiado y buscar información base para su organización.

  La forma más sencilla de hacer esto es ejecutando **Herramienta de configuración de autenticación** (`authconfig-gtk`) y seleccionar **Activar soporte LDAP** bajo la pestaña **Información de usuario**.

  También puede editar estos archivos manualmente.
- En las máquinas clientes, el archivo `/etc/nsswitch.conf` debe ser editado para usar LDAP.

  La forma más sencilla de hacer esto es ejecutando **Herramienta de configuración de autenticación** (`authconfig-gtk`) y seleccionar **Activar soporte LDAP** bajo la pestaña **Información de usuario**.

  Si está modificando el archivo `/etc/nsswitch.conf` manualmente, agregue `ldap` a las líneas adecuadas.

  Por ejemplo:

  ```
  passwd: files ldap
  shadow: files ldap
  group: files ldap
  ```

13.7.1. PAM y LDAP

Para tener aplicaciones PAM estándar use LDAP para la autenticación, ejecutando **authconfig** y luego seleccionando **Activar soporte LDAP** bajo la pestaña **Autenticación**. Para más información sobre la configuración PAM consulte, Capítulo 14 y las páginas del manual de PAM .

13.7.2. Migrar la información de autenticación antigua al formato LDAP

El directorio `/usr/share/openldap/migration/` contiene un conjunto de scripts de shell y Perl para la migración de información de autenticación en el formato LDAP.

Primero, modifique el archivo `migrate_common.ph` para que refleje su dominio. El dominio DNS por defecto debería ser modificado desde su valor por defecto a algo como lo siguiente:

```
$DEFAULT_MAIL_DOMAIN = "your_company";
```

La base por defecto también debería ser modificada, para que se parezca a:

```
$DEFAULT_BASE =
"dc=your_company,dc=com";
```

La tarea de migrar una base de datos de usuario a un formato que pueda leer LDAP le corresponde a un grupo de scripts de migración instalado en el mismo directorio. Usando Tabla 13-1, decida cúal script va a ejecutar para poder migrar su base de datos de usuario.

Nombre del servicio actual	¿Está LDAP ejecutándose?	Utilice este script
`/etc` archivos planos	si	`migrate_all_online.sh`
`/etc` flat files	no	`migrate_all_offline.sh`
NetInfo	si	`migrate_all_netinfo_online.sh`
NetInfo	no	`migrate_all_netinfo_offline.sh`
NIS (YP)	si	`migrate_all_nis_online.sh`
NIS (YP)	no	`migrate_all_nis_offline.sh`

Tabla 13-1. Scripts de migración de LDAP

Ejecute el script apropiado basándose en el nombre del servicio actual.

Nota

Debe tener Perl instalado en su sistema para usar algunos de estos scripts.

Los archivos `README` y `migration-tools.txt` en el directorio `/usr/share/openldap/migration/` dan más detalles sobre cómo migrar la información.

13.8. Actualizando a la versión 2.0 de OpenLDAP

En OpenLDAP Versión 2.0, el formato de almacenamiento en disco usado por el servidor LDAP `slapd` ha cambiado. Si está actualizando LDAP desde Red Hat Linux 7.0 o anterior, necesitará extraer los directorios LDAP actuales a un archivo LDIF usando el comando siguiente:

```
ldbmcat -n > <ldif_file>
```

En el comando de arriba, cambie *<ldif_file>* al nombre del archivo de salida. Luego escriba el comando siguiente para importar este archivo en OpenLDAP 2.0:

```
slapadd -l <ldif_file>
```

Importante

Debe ser usuario root para usar `/usr/sbin/slapadd`. Sin embargo, el servidor de directorio se ejecuta como usuario `ldap`. Por lo tanto el servidor de directorio no podrá modificar ningún archivo creado por `slapadd`. Para corregir este problema, después que haya terminado de usar `slapadd`, escriba el comando siguiente:

```
chown -R ldap /var/lib/ldap
```

13.9. Recursos adicionales

Existe más información referente a LDAP disponible. Por favor revise estas fuentes, especialmente el sitio web de OpenLDAP y la sección HOWTO de LDAP, antes de configurar LDAP en su sistema.

13.9.1. Documentación instalada

- Páginas del manual de LDAP — La página de `ldap` provee de una buena introducción a LDAP. También existen páginas de manual para los demonios y utilidades de LDAP.
- `/usr/share/docs/openldap-`*<versionnumber>* — Contiene un documento `README` e información general.

13.9.2. Sitios web útiles

- http://www.openldap.org/ — Hogar del Proyecto OpenLDAP. Este sitio web contiene una gran variedad de información sobre la configuración de OpenLDAP.
- http://www.redhat.com/mirrors/LDP/HOWTO/LDAP-HOWTO.html — Una sección HOWTO de LDAP vieja, pero aún relevante.
- http://www.padl.com/ — Desarrolladores de `nss_ldap` y `pam_ldap`, entre otras herramientas útiles de LDAP.
- http://www.kingsmountain.com/ldapRoadmap.shtml — Jeff Hodges' LDAP Road Map contiene enlaces a muchas secciones FAQs de utilidad y a noticias recientes concernientes al protocolo LDAP.
- http://www.webtechniques.com/archives/2000/05/wilcox — Una vista útil sobre el manejo de grupos en LDAP.
- http://www.ldapman.org/articles — Articulos que ofrecen una buena introducción a LDAP, incluyendo métodos para diseñar un árbol y personalizar estructuras de directorios.

13.9.3. Libros relacionados

- *Implementing LDAP* de Mark Wilcox; Wrox Press, Inc.

- *Understanding and Deploying LDAP Directory Services* por Tim Howes et al.; Macmillan Technical Publishing

III. Referencia de seguridad

El uso de protocolos seguros es una parte crítica del mantenimiento de la integridad del sistema. Esta parte describe las herramientas usadas para la autenticación de usuarios, control de acceso a la red, comunicaciones seguras y detección de intrusos. Para más información sobre la seguridad en un sistema Red Hat Linux, refiérase al *Manual de seguridad de Red Hat Linux.*

Tabla de contenidos

redhat.

Capítulo 14.

Pluggable Authentication Modules (PAM)

Los programas que permiten a los usuarios accesar un sistema deben verificar la identidad del usuario a través de un proceso llamado *autentificar*. Históricamente, cada programa tiene su forma particular de realizar la autentificación. Bajo Red Hat Linux, muchos de tales programas son configurados para usar un proceso de autenticación centralizado llamado *Pluggable Authentication Modules* (*PAM*).

PAM utiliza una arquitectura conectable y modular, que otorga al administrador del sistema de una gran flexibilidad en establecer las políticas de autenticación para el sistema.

En la mayoría de los casos, el archivo de configuración por defecto PAM para una aplicación tipo PAM es suficiente. Sin embargo, algunas veces es necesario modificar el archivo de configuración. Debido a que un error en la configuración de PAM puede comprometer la seguridad del sistema, es importante que comprenda la estructura del estos archivos antes de hacer cualquier modificación (consulte Sección 14.3 para más información).

14.1. Las ventajas de PAM

PAM ofrece las ventajas siguientes:

- Un esquema de autenticación común que se puede usar con una gran variedad de aplicaciones.
- Permite gran flexibilidad y control de la autentificación para el administrador del sistema y el desarrollador de la aplicación.
- Los desarrolladores de aplicaciones no necesitan desarrollar su programa para usar un determinado esquema de autenticación. En su lugar, pueden concentrarse puramente en los detalles de su programa.

14.2. archivos de configuración PAM

El directorio `/etc/pam.d/` contiene los archivos de configuración de PAM para cada aplicación tipo PAM. En versiones antiguas de PAM se utilizaba `/etc/pam.conf`, pero este archivo ya no se utiliza y `pam.conf` solamente es leído si el directorio `/etc/pam.d/` no existe.

14.2.1. Archivos de servicios PAM

Las aplicaciones tipo PAM o *servicios* tienen un archivo dentro del directorio `/etc/pam.d/`. Cada uno de estos archivos es llamado después del servicio para el cual controla el acceso.

Depende del programa tipo PAM definir el nombre de su servicio e instalar su archivo de configuración en el directorio `/etc/pam.d/`. Por ejemplo, el programa `login` define su nombre de servicio como `/etc/pam.d/login`.

14.3. Formato del archivo de configuración PAM

Cada archivo de configuración PAM contiene un grupo de directivas formateadas como sigue:

```
<module interface>  <control
flag>  <module path>
<module arguments>
```

En las siguientes secciones se explican cada uno de estos elementos.

14.3.1. Interfaz de módulo

Existen cuatro tipos de módulos PAM usados para controlar el acceso a los servicios. Estos tipos establecen una correlación entre los diferentes aspectos del proceso de autorización:

- `auth` — Estos módulos autentifican a los usuarios tal vez pidiendo y controlando una contraseña. Los módulos con esta interfaz también pueden establecer credenciales, tales como membrecías de grupo o tickets Kerberos.
- `account` — Estos módulos controlan que la autenticación sea permitida (que la cuenta no haya caducado, que el usuario tenga permiso de iniciar sesiones a esa hora del día, etc.).
- `password` — Estos módulos se usan para establecer y verificar contraseñas.
- `session` — Estos módulos configuran y administran sesiones de usuarios. Los módulos con esta interfaz también pueden realizar tareas adicionales que son necesitadas para permitir acceso, como el montaje de directorios principales de usuarios y hacer el buzón de correo disponible.

Nota

Un módulo individual puede proporcionar alguno o todas las interfaces de módules mencionadas anteriormente. Por ejemplo, `pam_unix.so` tiene componentes que direccionan las cuatro interfaces.

En un archivo de configuración PAM, la interfaz del módulo es el primer aspecto a definir. Por ejemplo, una línea típica de una configuración sería:

```
auth       required  /lib/security/pam_unix.so
```

Esto provoca que PAM observe el componente `pam_unix.so` del módulo `auth`.

14.3.1.1. Apilar módulos

Las directivas de interfaces de módulos pueden ser *apiladas* o colocar uno sobre otro para que se puedan usar módulos múltiples para un mismo propósito. El orden de una pila de módulos es muy importante en el procedimiento de autenticación.

El hecho de apilarlos hace que sea más fácil para que el administrador exija diversas condiciones antes de permitir la autentificación del usuario. Por ejemplo, `rlogin` normalmente usa cinco módulos `auth`, como se puede ver en el archivo de configuración de PAM:

```
auth       required     /lib/security/pam_nologin.so
auth       required     /lib/security/pam_securetty.so
auth       required     /lib/security/pam_env.so
auth       sufficient   /lib/security/pam_rhosts_auth.so
auth       required     /lib/security/pam_stack.so service=system-auth
```

Antes de que a alguien se le permita llevar a cabo el `rlogin`, PAM verifica que el archivo `/etc/nologin` no exista, que no esté intentando iniciar una sesión en modo remoto como root y que se pueda cargar cualquier variable de entorno. Entonces se lleva a cabo una autenticación `rhosts` exitosa antes que se permita la conexión. Si falla la autenticación `rhosts` entonces se lleva a cabo una autenticación de contraseña estándar.

14.3.2. Indicadores de control

Todos los módulos PAM generan un resultado de éxito o fracaso cuando se les llama. Los indicadores de control le dicen a PAM qué hacer con el resultado. Como los módulos pueden apilarse en un determinado orden, los indicadores de control le dan la posibilidad de fijar la importancia de un módulo con respecto al objetivo final del proceso de autenticación para el servicio.

Hay cuatro indicadores de control definidos:

- `required` — El resultado del módulo debe ser exitoso para que la autenticación continue. Si un módulo `required` falla, el usuario no es notificado hasta que los resultados en todos los módulos referenciando esa interfaz sean completados.
- `requisite` — El resultado del módulo debe ser exitóso para que la autenticación continue. Sin embargo, si el resultado de un módulo `requisite` falla, el usuario es notificado inmediatamente con un mensaje reflejando el primer módulo `required` *o* `requisite` fracasado.
- `sufficient` — El resultado del módulo es ignorado si falla. Pero si el resultado del módulo con el indicador `sufficient` es exitoso *and* ningún módulo con indicador `required` ha fallado, entonces no se requiere ningún otro resultado y el usuario es autenticado para el servicio.
- `optional` — Se ignora el resultado del módulo si falla. Si el resultado del módulo es exitoso, no juega ningún papel en el éxito o fracaso general para el módulo. Un módulo con una bandera `optional` es necesario para la autenticación exitosa cuando no hay otros módulos referenciando la interfaz. En este caso, un módulo `optional` determina la autenticación PAM para esa interfaz.

Importante

El orden en el cual se llaman los módulos `required` no es crítico. Las banderas o indicadores de control `sufficient` y `requisite` provocan que el orden se vuelva importante.

Una sintaxis más nueva de indicadores de control que permite un control más preciso está disponible para PAM. Por favor revises los documentos de PAM localizados en el directorio `/usr/share/doc/pam-<version-number>/` para información sobre esta nueva sintaxis (donde `<version-number>` es el número de versión de PAM).

14.3.3. Rutas de módulos

Las rutas de los módulos le indican a PAM dónde encontrar el módulo conectable que hay que usar con el tipo de interfaz de módulo especificada. Generalmente, se proporciona como una ruta completa de módulo, como `/lib/security/pam_stack.so`. Sin embargo, si no se proporciona la ruta entera, entonces se asume que el módulo está en el directorio `/lib/security/`, la dirección por defecto de los módulos PAM.

14.3.4. Argumentos de módulo

PAM utiliza argumentos para transmitir información a un módulo conectable durante la autenticación para algunos módulos.

Por ejemplo, el módulo `pam_userdb.so` usa secretos almacenados en una base de datos Berkeley DB file para autenticar a los usuarios. La base de datos Berkeley es una base de datos open source incorporado en muchas aplicaciones. El módulo toma un argumento `db` para que la base de datos Berkeley conozca que base de datos usar para el servicio solicitado.

Una línea típica `pam_userdb.so` dentro de un archivo PAM es similar a:

```
auth      required  /lib/security/pam_userdb.so
db=<path-to-file>
```

En el ejemplo anterior, sustituya *<path-to-file>* con la ruta completa al archivo de base de datos Berkeley.

Los argumentos inválidos se ignoran y no afectan en ningún modo el éxito o fracaso del módulo PAM. Sin embargo, la mayoría de los módulos reportarán un error al archivo `/var/log/messages`.

14.4. Muestras de archivos de configuración PAM

A continuación una muestra de archivo de configuración de la aplicación PAM:

```
#%PAM-1.0
auth      required  /lib/security/pam_securetty.so
auth      required  /lib/security/pam_unix.so shadow nullok
auth      required  /lib/security/pam_nologin.so
account   required  /lib/security/pam_unix.so
password  required  /lib/security/pam_cracklib.so retry=3
password  required  /lib/security/pam_unix.so shadow nullok use_authtok
session   required  /lib/security/pam_unix.so
```

La primera línea es un comentario como lo es toda línea que inicie con el carácter (`#`).

Las líneas dos, tres y cuatro apilan tres módulos a usar para autentificaciones de inicio de sesión.

```
auth      required  /lib/security/pam_securetty.so
```

Este módulo se asegura de que *si* el usuario está tratando de conectarse como root, el tty en el cual el usuario se está conectando está listado en el archivo `/etc/securetty`, *si* ese archivo existe.

```
auth      required  /lib/security/pam_unix.so shadow nullok
```

Este módulo le solicita al usuario por una contraseña y luego verifica la contraseña usando la información almacenada en `/etc/passwd` y, si existe `/etc/shadow`. El módulo `pam_unix.so` detecta automáticamente y utiliza contraseñas shadow para autenticar usuarios. Por favor consulte Sección 6.5 para más información sobre contraseñas shadow.

El argumento `nullok` instruye al módulo `pam_unix.so` a que permita una contraseña en blanco.

```
auth      required  /lib/security/pam_nologin.so
```

Este es el paso final de autenticación. Sus resultados para ver si el archivo `/etc/nologin` existe. Si `nologin` no existe y el usuario no es root, la autenticación falla.

Nota

En este ejemplo, los tres módulos `auth`, aún si el primer módulo `auth` falla. Esto previene al usuario de saber a qué nivel falla la autenticación. Tal conocimiento en las manos de una persona mal intencionada le permitiría violar el sistema fácilmente.

```
account   required  /lib/security/pam_unix.so
```

Este módulo realiza cualquier verificación de cuenta necesaria. Por ejemplo, las contraseñas shadow han sido activadas, el componente de la cuenta del módulo `pam_unix.so` verificará para ver si la cuenta ha expirado o si el usuario no ha cambiado la contraseña dentro del período de gracia otorgado.

```
password  required  /lib/security/pam_cracklib.so retry=3
```

Si la contraseña ha expirado, el componente de la contraseña del módulo `pam_cracklib.so` le pide por una nueva contraseña. Luego evalúa la nueva contraseña para ver si puede ser fácilmente determinado por un programa que descubre las contraseñas basadas en diccionario. Si esto falla la primera vez, le dá al usuario dos oportunidades más de crear una contraseña más robusta debido al argumento `retry=3`.

```
password  required  /lib/security/pam_unix.so shadow nullok use_authtok
```

Esta línea especifica que si el programa cambia la contraseña del usuario, éste debería usar el componente `password` del módulo `pam_unix.so` para realizarlo. Esto sucederá tan sólo si la porción `auth` del módulo `pam_unix.so` ha determinado que la contraseña necesita ser cambiada.

El argumento `shadow` le dice al módulo que cree contraseñas shadow cuando se actualiza la contraseña del usuario.

El argumento `nullok` indica al módulo que permita al usuario cambiar su contraseña *desde* una contraseña en blanco, de lo contrario una contraseña vacía o en blanco es tratada como un bloqueo de cuenta.

El argumento final de esta línea, `use_authtok`, proporciona un buen ejemplo de la importancia del orden al apilar módulos PAM. Este argumento advierte al módulo a no solicitar al usuario una nueva contraseña. En su lugar se acepta cualquier contraseña que fué registrada por un módulo de contraseña anterior. De este modo todas las nuevas contraseñas deben pasar el test de `pam_cracklib.so` para contraseñas seguras antes de ser aceptado.

```
session required /lib/security/pam_unix.so
```

La última línea especifica que el componente de la sesión del módulo `pam_unix.so` gestionará la sesión. Este módulo registra el nombre de usuario y el tipo de servicio para `/var/log/messages` al inicio y al final de cada sesión. Puede ser suplementado apilándolo con otros módulos de sesión si necesita más funcionalidad.

El próximo ejemplo de archivo de configuración ilustra el apilamiento del módulo `auth` para el programa `rlogin`.

```
#%PAM-1.0
auth       required     /lib/security/pam_nologin.so
auth       required     /lib/security/pam_securetty.so
auth       required     /lib/security/pam_env.so
auth       sufficient   /lib/security/pam_rhosts_auth.so
auth       required     /lib/security/pam_stack.so service=system-auth
```

Primero, `pam_nologin.so` verifica para ver si `/etc/nologin` existe. Si lo hace, nadie puede conectarse excepto root.

```
auth       required     /lib/security/pam_securetty.so
```

El módulo `pam_securetty.so` previene al usuario root de conectarse en terminales inseguros. Esto desactiva efectivamente a todos los intentos de root `rlogin` debido a las limitaciones de seguridad de la aplicación.

Sugerencia

Para conectarse remotamente como usuario root, use OpenSSH. Para más información sobre el protocolo SSH, consulte el Capítulo 18.

```
auth       required     /lib/security/pam_env.so
```

El módulo carga las variable de entorno especificadas en `/etc/security/pam_env.conf`.

```
auth       sufficient   /lib/security/pam_rhosts_auth.so
```

El módulo `pam_rhosts_auth.so` autentifica al usuario usando `.rhosts` en el directorio principal del usuario. Si tiene éxito, PAM considera inmediatamente la autenticación como exitosa. Si falla `pam_rhosts_auth.so` en autenticar al usuario, el intento de autenticación es ignorado.

```
auth       required     /lib/security/pam_stack.so service=system-auth
```

Si el módulo `pam_rhosts_auth.so` falla en autenticar al usuario, el módulo `pam_stack.so` realiza la autenticación de contraseñas normal.

El argumento `service=system-auth` indica que el usuario debe pasar a través de la configuración PAM para la autenticación del sistema como se encuentra en `/etc/pam.d/system-auth`.

Sugerencia

Si no desea que se pida la contraseña cuando el control `securetty` fracasa, cambie el módulo `pam_securetty.so` de `required` a `requisite`.

14.5. Creación de módulos PAM

Se puede añadir nuevos módulos PAM en cualquier momento para que las aplicaciones que soporten PAM los usen. Por ejemplo, si un desarrollador inventa un método de creación de contraseña de una sola vez, y escribe un módulo PAM que lo soporte, los programas tipo PAM pueden inmediatamente usar el nuevo módulo y el método de contraseña sin recompilar. Esto permite a los desarrolladores y administradores de sistemas mezclar y coincidir, así como también evaluar, métodos de autenticación para programas diferentes sin recompilarlos.

La documentación sobre la escritura de módulos es incluída con el sistema en el directorio `/usr/share/doc/pam-<version-number>/` (donde *<version-number>* es el número de versión para PAM).

14.6. PAM y propiedad del dispositivo

Red Hat Linux permite al primer usuario que se conecte en una consola física de la máquina la habilidad de manipular algunos dispositivos y realizar algunas tareas normalmente reservadas para el usuario root. Esto es controlado por un módulo PAM llamado `pam_console.so`.

14.6.1. Propiedad del dispositivo

Cuando un usuario se registra en una máquina bajo Red Hat Linux, el módulo `pam_console.so` es llamado por `login` o los programas de inicio de sesión gráfica, **gdm** y **kdm**. Si este usuario es el primero en conectarse en la consola física — llamado *console user* — el módulo concede la propiedad de una variedad de dispositivos que normalmente posee root. El usuario de la consola posee estos dispositivos hasta que la última sesión local para ese usuario finaliza. Una vez que el usuario se ha desconectado, la propiedad de los dispositivos vuelve a root.

Los dispositivos afectados incluyen, pero no son limitados, las tarjetas de sonido, las unidades de disco y las unidades de CD-ROM.

Esto permite que el usuario local manipule estos dispositivos sin llegar a root, de manera que se simplifican las tareas comunes para el usuario de la consola.

Modificando el archivo `/etc/security/console.perms`, el administrador puede editar la lista de dispositivos controlados por `pam_console.so`.

14.6.2. Acceso de la aplicación

También se le permite al usuario de la consola (console user) el acceso a ciertos programas con un archivo que contenga el nombre del comando en el directorio `/etc/security/console.apps/`.

Un grupo notable de aplicaciones a las que tiene acceso el usuario de la consola son tres programas que cierran o abren el sistema. Estos son:

- /sbin/halt
- /sbin/reboot
- /sbin/poweroff

Debido a que estas son aplicaciones tipo PAM, ellas llaman al módulo `pam_console.so` como un requerimiento para el uso.

Para más información, consulte las páginas man para `pam_console`, `console.perms`, `console.apps` y `userhelper`.

14.7. Recursos adicionales

La siguiente es una lista de recursos para el uso y configuración de PAM. Además de estos recursos, lea los archivos de configuración de PAM en el sistema para entender mejor como están estructurados.

14.7.1. Documentación instalada

- La página del manual de `pam` — Una buena fuente introductoria de información sobre PAM, incluyendo la estructura y propósito de los archivos de configuración PAM.
- `/usr/share/doc/pam-<version-number>` — Contiene el *System Administrators' Guide*, el *Module Writers' Manual* y el *Application Developers' Manual*, así como también una copia del estándar PAM, DCE-RFC 86.0.

14.7.2. Sitios web útiles

- http://www.kernel.org/pub/linux/libs/pam/ — El sitio web de distribución primario para el proyecto Linux-PAM, conteniendo información sobre varios módulos PAM, una sección FAQ y documentación PAM adicional.

Capítulo 15.

Los wrappers TCP y el comando `xinetd`

El control del acceso a los servicios de red puede ser todo un reto. Afortunadamente, bajo Red Hat Linux, hay un gran número de herramientas que lo pueden ayudar. Por ejemplo, un filtro de cortafuegos basado en `iptables` dejan afuera los paquetes de red que no son bienvenidos dentro de la pila de red del kernel. Para los servicios de red que lo utilizan, los *wrappers TCP* añaden una capa adicional de protección mediante la definición de cuáles hosts son permitidos y cuáles no conectarse a los servicios de red "*wrapped*". Uno de los servicios de red wrapped es `xinetd` *super servidor*. Este servicio se le llama super servidor porque controla la conexión a un subconjunto de servicios de red y refina aún más el control de acceso.

La Figura 15-1 es una ilustración básica de cómo estas herramientas funcionan para proteger los servicios de red.

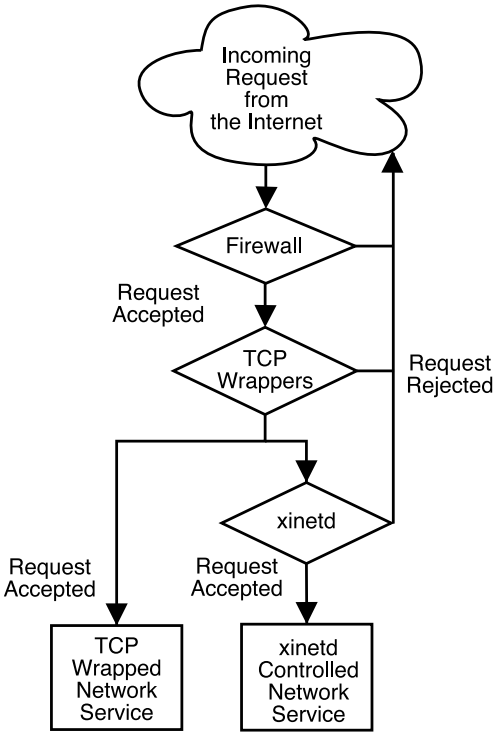

Figura 15-1. Control de acceso a los servicios de red

Este capítulo se basa en el papel de los wrappers TCP y de `xinetd` para controlar el acceso a los servicios de red y revisa cómo estas herramientas pueden ser usadas para mejorar la administración del registro. Para una discusión del uso de cortafuegos (firewalls) con `iptables`, consulte el Capítulo 16.

15.1. Wrappers TCP

El paquete wrappers TCP (`tcp_wrappers`) está instalado por defecto bajo Red Hat Linux y proporciona control de acceso basado en host a los servicios de red. El componente más importante dentro del paquete es la librería `/usr/lib/libwrap.a`. En términos generales, un servicio wrappers TCP es uno que ha sido compilado con la librería `libwrap.a`.

Cuando un intento de conexión es hecho a un servicio wrapped TCP, el servicio primero referencia los archivos de *acceso de host* (`/etc/hosts.allow` y `/etc/hosts.deny`) para determinar si el cliente tiene permitido conectarse. Luego utiliza el demonio syslog (`syslogd`) para escribir el nombre del host solicitante y el servicio solicitado a `/var/log/secure` o `/var/log/messages`.

Si a un cliente se le permite conectarse, los wrappers TCP liberan el control de la conexión al servicio solicitado y no interfieren más con la comunicación entre el cliente y el servidor.

Además del control de acceso y registro, los wrappers TCP pueden activar comandos para interactuar con el cliente antes de negar o liberar el control de la conexión al servicio solicitado.

Puesto que los wrappers TCP son una utilidad de gran valor a las herramientas de seguridad de cualquier administrador de servidor, la mayoría de los servicios de red dentro de Red Hat Linux están enlazados con la librería `libwrap.a`. Tales aplicaciones incluyen `/usr/sbin/sshd`, `/usr/sbin/sendmail`, y `/usr/sbin/xinetd`.

Nota

Para determinar si un binario de servicio de red está enlazado con la libreria `libwrap.a`, escriba el comando siguiente como usuario root:

```
strings -f <binary-name> | grep hosts_access
```

Reemplace <*binary-name*> con el nombre del binario de servicio de red.

15.1.1. Ventajas de los wrappers TCP

Los wrappers TCP ofrecen las siguientes ventajas básicas comparado con las otras técnicas de control de servicios de red:

- *Transparencia tanto para el cliente del host y el servicio de red wrapped.* — El cliente que se está conectando así como también el servicio de red wrapped no están al tanto de que están en uso los wrappers TCP. Los usuarios legítimos son registrados y conectados al servicio solicitado mientras que las conexiones de clientes prohibidos fallan.
- *Administración centralizada de protocolo múltiples.* — Los wrappers TCP operan separadamente de los servicios de red que ellos protegen, permitiendo a muchas aplicaciones de servidor compartir un conjunto común de archivos de configuración para una administración más sencilla.

15.2. Archivos de configuración de Wrappers TCP

Para determinar si una máquina cliente tiene permitido conectarse a un servicio, los wrappers TCP referencian los siguientes dos archivos, los cuales se conocen comúnmente como archivos de acceso a host:

- `/etc/hosts.allow`

- /etc/hosts.deny

Cuando una solicitud de un cliente es recibida por un servicio wrapped TCP, sigue los pasos siguientes:

1. *El servicio referencia a /etc/hosts.allow.* — El servicio wrapped TCP analiza secuencialmente el archivo /etc/hosts.allow y aplica la primera regla especificada para ese servicio. Si encuentra una regla que coincide, permite la conexión. Si no, se va al paso 2.
2. *El servicio referencia /etc/hosts.deny.* — El servicio wrapped TCP analiza secuencialmente el archivo /etc/hosts.deny. Si encuentra una regla que coincide, rechaza la conexión. Si no, se concede acceso al servicio.

Lo siguiente son puntos muy importantes a considerar cuando se usen wrappers TCP para proteger servicios de red:

- Puesto que las reglas de acceso en hosts.allow son aplicadas primero, ellas toman precedencia sobre las reglas en hosts.deny. Por lo tanto, si se permite el acceso a un servicio en hosts.allow, una regla negando el acceso al mismo servicio en hosts.deny es ignorada.
- Puesto que las reglas en cada archivo son leídas de arriba hacia abajo y la primera regla que coincida para un servicio dado es la única aplicada, el orden de las reglas es extremadamente importante.
- Si no se encuentra ninguna regla para el servicio en ninguno de los archivos, o si no existe ninguno de los archivos, se concede el acceso al servicio.
- Los sevicios wrapped TCP no hacen caché de las reglas desde los archivos acceso de host, por lo tanto cualquier cambio a hosts.allow o a hosts.deny tomarán efecto de inmediato sin tener que reiniciar el servicio de red.

15.2.1. Formatear reglas de acceso

Los formatos para /etc/hosts.allow y /etc/hosts.deny son idénticos. Cualquier línea en blanco que comience con un símbolo de numeral o almohadilla (#) será ignorada, y cada regla debe estar en su propia línea.

Las reglas se tienen que formatear de la siguiente manera:

```
<daemon list>: <client list> [: <option>: <option>: ...]
```

- *<daemon list>* — Una lista separada por comas de los nombres de procesos (*no* de los nombres de servicios) o el *comodín* ALL (consulte Sección 15.2.1.1). La lista de demonios también acepta *operadores* listados en Sección 15.2.1.3 para permitir mayor flexibilidad.
- *<client list>* — Una lista separada por comas de nombres de host, direcciones IP, *patrones* especiales (consulte Sección 15.2.1.2), o comodines especiales (consulte Sección 15.2.1.1) el cual identifica los hosts afectados por la regla. La lista de clientes también acepta operadores listados en Sección 15.2.1.3 para permitir mayor flexibilidad.
- *<option>* — Una acción opcional o una lista separada con puntos y comas de acciones realizadas cuando la regla es activada. Los campos de opciones soportan *expansiones* (consulte Sección 15.2.3.4), lance los comandos desde el shell, otorgue o prohiba el acceso y altere el comportamiento de registro (consulte Sección 15.2.3).

A continuación una muestra básica de una regla de acceso:

```
vsftpd : .example.com
```

Esta regla instruye a los wrappers TCP a que vigile conexiones al demonio FTP (vsftpd) desde cualquier host en el dominio example.com. Si esta regla aparece en hosts.allow, la conexión será aceptada. Si esta regla aparece en hosts.deny, la conexión será rechazada.

El próximo ejemplo de regla de acceso es un poco más compleja y utiliza dos campos de opciones:

```
sshd : .example.com  \
: spawn /bin/echo `/bin/date` access denied>>/var/log/sshd.log \
: deny
```

Note que en este ejemplo cada campo de opción está precedido por una barra oblicua invertida (\). Use la barra para prevenir la falla de la regla debido al largo.

Aviso

Si la última línea de un archivo de acceso a host no es un caracter de nueva línea (creado al presionar la tecla [Intro]), la última regla en el archivo fallará y se registrará un error bien sea a /var/log/messages o a /var/log/secure. Este es también el caso para líneas de reglas que se esparcen en múltiples líneas sin usar la barra. El ejemplo siguiente ilustra la porción relevante del mensaje de registro por una falla de una regla debido a alguna de estas circunstancias:

```
warning: /etc/hosts.allow, line 20: missing newline or line too long
```

Esta regla de ejemplo indica que si una conexión al demonio SSH (sshd) se intenta desde un host en el dominio example.com, ejecute el comando echo (lo cual registrará el intento a un archivo especial) y rechace la conexión. Puesto que se usa la directiva opcional deny, esta línea rechazará el acceso aún si aparece en el archivo hosts.allow. Para más detalles sobre las opciones disponibles, consulte Sección 15.2.3.

15.2.1.1. Comodines

Los comodines permiten a los wrappers TCP coincidir más fácilmente grupos de demonios o hosts. Son usados con mayor frecuencia en el campo de lista de cliente de las reglas de acceso.

Se pueden utilizar los siguientes comodines:

- ALL — Hace corresponder todo. Se puede usar para la lista de demonios o en la lista de clientes.
- LOCAL — Hace corresponder todos los nombres de máquinas que no contengan un punto (.), tal como localhost.
- KNOWN — Hace corresponder todas las máquinas cuyos nombres y direcciones son conocidos o donde el usuario es conocido.
- UNKNOWN — Hace corresponder todas las máquinas cuyos nombres y direcciones sean desconocidas o en el caso en el que se desconozca el usuario.
- PARANOID — Hace corresponder todas las máquinas cuyo nombre no se corresponda con la dirección.

Atención

Los comodines KNOWN, UNKNOWN y PARANOID tienen que usarse con cuidado porque si se utilizan de una manera incorrecta los usuarios que sí que tengan acceso a un determinado servicio pueden perderlo.

15.2.1.2. Patrones

Los patrones se pueden utilizar en el campo de lista de cliente de las reglas de acceso para especificar de forma más precisa grupos de host clientes.

La siguiente es una lista de los patrones más comúnmente aceptados para una entrada de lista de cliente:

- *Nombre de host comenzando con un punto (.)* — Al colocar un punto al comienzo de un nombre de host, se hace coincidir todos los hosts compartiendo los componentes listados del nombre. El ejemplo siguiente aplicaría a cualquier host dentro del dominio `example.com`:
  ```
  ALL : .example.com
  ```
- *Dirección IP que termina con un punto (.)* — Al colocar un punto al final de una dirección IP hace corresponder todos los hosts compartiendo el grupo numérico inicial de una dirección IP. El ejemplo siguiente aplicará a cualquier host dentro de la red `192.168.x.x`:
  ```
  ALL : 192.168.
  ```
- *Dirección IP/par máscara de red* — Las expresiones de máscaras de red también pueden ser usadas como un patrón de control de acceso a un grupo particular de direcciones IP. El ejemplo siguiente aplicaría a cualquier host con una dirección de `192.168.0.0` hasta `192.168.1.255`:
  ```
  ALL : 192.168.0.0/255.255.254.0
  ```
- *El asterisco (*)* — Los asteríscos pueden ser usados para coincidir grupos completos de nombres de host o direcciones IP, siempre y cuando no se mezclen en la lista de cliente conteniendo otros tipos de patrones. El ejemplo siguiente aplicaría a cualquier host dentro del dominio `example.com`:
  ```
  ALL : *.example.com
  ```
- *La barra oblicua (/)* — Si una lista de cliente con una barra, es tratado como un nombre de archivo. Esto en muy útil si es necesario reglas especificando un gran número de hosts. El ejemplo siguiente se refiere a wrappers TCP en el archivo `/etc/telnet.hosts` para todas las conexiones de Telnet:
  ```
  in.telnetd : /etc/telnet.hosts
  ```

Otros patrones menos usados son también aceptados por los wrappers TCP. Vea la página de manual 5 de acceso a host para mayor información.

Aviso

Tenga mucho cuidado cuando esté creando reglas que requieran la resolución de nombres, tales como nombres de máquinas o de dominio. Los invasores pueden usar una variedad trucos para cambiar las reglas de tal manera que le den el acceso. Además, cualquier interrupción en el servicio DNS podría impedir el acceso a servicios incluso a la usuarios que tienen el permiso.

Lo mejor es usar direcciones IP siempre que sea posible.

15.2.1.3. Operadores

Actualmente, las reglas de control de acceso aceptan un operador, `EXCEPT`. Se puede usar tanto en la lista de demonios como en la lista de cliente de una regla.

El operador `EXCEPT` permite excepciones específicas a coincidencias más amplias dentro de la misma regla.

En el ejemplo siguiente desde un archivo `hosts.allow`, todos los hosts de `example.com` pueden conectarse a todos los servicios excepto `cracker.example.com`:

```
ALL: .example.com EXCEPT cracker.example.com
```

En el otro ejemplo desde un archivo hosts.allow, clientes desde la red 192.168.0.*x* pueden usar todos los servicios excepto para FTP:

```
ALL EXCEPT vsftpd: 192.168.0.
```

Nota

Organizacionalmente, a menudo es más fácil usar operadores EXCEPT muy de vez en cuando, colocando las excepciones a la regla en el otro archivo de control de acceso. Esto permite a otros administradores escanear rápidamente los archivos adecuados para ver qué hosts deberían tener o no acceso a los servicios, sin tener que ordenar a través de varios operadores EXCEPT.

15.2.2. Portmap y Wrappers TCP

Cuando se crean reglas de control de acceso para portmap, no utilice nombres de host pues la implementación de wrappers TCP no soporta la búsqueda de host. Por esta razón, sólo utilice direcciones IP o la palabra clave ALL cuando la especificación de host es en hosts.allow o hosts.deny.

Además, cambios a las reglas de control de acceso portmap pueden que no tomen efecto de inmediato.

Los servicios ampliamente usados, tales como NIS y NFS, dependen de portmap para funcionar, por lo tanto este consciente de estas limitaciones.

15.2.3. Campos de opciones

Además de las reglas básicas para permitir o prohibir el acceso, la implementación de Red Hat Linux de wrappers TCP soporta extensiones al lenguaje de control de acceso a través de los campos de opciones. Mediante el uso de campos de opciones dentro de las reglas de acceso al host, los administradores pueden llevar a cabo una gran variedad de tareas tales como alterar el comportamiento del registro, consolidar el control de acceso y lanzar comandos del shell.

15.2.3.1. Registro

Los campos de opciones le permiten a los administradores cambiar fácilmente la facilidad de conexión y el nivel de prioridad para una regla usando la directiva severity.

En el ejemplo siguiente, las conexiones al demonio SSH desde cualquier host en el dominio example.com son registrados a la facilidad por defecto authpriv (debido a que no se especifica un valor de facilidad) con una prioridad de emerg:

```
sshd : .example.com : severity emerg
```

Es también posible especificar una facilidad utilizando la opción severity. El ejemplo siguiente registra cualquier intento de conexión SSH por cualquier hosts desde el dominio example.com a la facilidad local0 con una prioridad de alert:

```
sshd : .example.com : severity local0.alert
```

Nota

En práctica, este ejemplo no funcionará hasta que el demonio syslog (`syslogd`) sea configurado para registrar a la facilidad `local0`. Consulte la página del manual de `syslog.conf` para información sobre la configuración de las facilidades de registro personalizadas.

15.2.3.2. Control de acceso

Los campos de opciones también le permiten a los administradores explícitamente otorgar o prohibir el acceso de máquinas en un sola regla, añadiendo la directiva `allow` o `deny` al final de la opción.

Por ejemplo, las dos reglas siguientes permiten conexiones SSH desde `client-1.example.com`, pero prohiben conexiones desde `client-2.example.com`:

```
sshd : client-1.example.com : allow
sshd : client-2.example.com : deny
```

Al permitir el control de acceso por regla, el campo de opciones permite a los administradores consolidar todas las reglas de acceso en un sólo archivo: bien sea `hosts.allow` o `hosts.deny`. Algunos consideran que esta es una forma más fácil de organizar reglas de acceso.

15.2.3.3. Comandos de la Shell

Los campos de opciones permiten a las reglas de acceso lanzar comandos de la shell a través de las directivas siguientes:

- `spawn` — Lanza un comando de la shell como un proceso hijo. Esta directiva de opción puede realizar tareas como el uso de `/usr/sbin/safe_finger` para obtener más información sobre el cliente solicitante o la creación de archivos de registro especiales usando el comando `echo`.

  En el ejemplo siguiente, los clientes intentando accesar servicios Telnet desde el dominio `example.com` son registrados discretamente a un archivo especial:

  ```
  in.telnetd : .example.com \
    : spawn /bin/echo `/bin/date` from %h>>/var/log/telnet.log \
    : allow
  ```

- `twist` — Reemplaza el servicio solicitado con el comando especificado. Esta directiva es a menudo usada para colocar trampas para intrusos (también llamados "potes de miel"). También se puede utilizar para enviar mensajes a los clientes que se estan conectando. El comando twist debe ocurrir al final de la línea de la regla.

  En el ejemplo siguiente, los clientes intentando accesar servicios FTP desde el dominio `example.com` se les envía un mensaje a través del comando `echo`:

  ```
  vsftpd : .example.com \
  : twist /bin/echo "421 Bad hacker, go away!"
  ```

Para más información sobre las opciones de comando de la shell, consulte la página del manual de `hosts_options`.

15.2.3.4. Expansiones

Las expansiones, cuando se utilizan en conjunto con las directivas `spawn` y `twist` proporcionan información sobre el cliente, servidor y los procesos relacionados.

Abajo se encuentra una lista de las expansiones soportadas:

- %a — La dirección IP del cliente.
- %A — La dirección IP del servidor.
- %c — Proporciona información variada sobre el cliente, como el nombre de usuario y el de la máquina o el nombre del usuario y la dirección IP.
- %d — El nombre del proceso demonio.
- %h — El nombre de la máquina del cliente (o la direccción IP, si el nombre de la máquina no está disponible).
- %H — El nombre de la máquina del servidor (o la dirección IP si el nombre de la máquina no está disponible).
- %n — El nombre de la máquina del cliente. Si no está disponible aparecerá unknown. Si el nombre de la máquina y la dirección de la máquina no se corresponden, aparecerá paranoid.
- %N — El nombre de la máquina del servidor. Si no está disponible aparecerá unknown. Si el nombre de la máquina y la dirección de la máquina no coinciden, aparecá paranoid.
- %p — El ID del proceso demonio.
- %s — Información varia del servidor como el proceso demonio y la máquina o la dirección IP del servidor.
- %u — El nombre de usuario del cliente. Si no está disponible aparecerá unknown.

El ejemplo siguiente usa una expansión en conjunto con el comando spawn para identificar el host cliente en un archivo de registro personalizado.

Instruye a los wrappers TCP que si una conexión al demonio SSH (sshd) es intentada desde un host en el dominio example.com, ejecute el comando echo para registrar el intento, incluyendo el nombre del host cliente (usando la expansión %h), a un archivo especial:

```
sshd : .example.com  \
: spawn /bin/echo `/bin/date` access denied to %h>>/var/log/sshd.log \
: deny
```

De forma similar, las expansiones se pueden utilizar para personalizar mensajes de vuelta al cliente. En el ejemplo siguiente, los clientes intentando accesar servicios FTP desde el dominio example.com son informados que se les ha prohibido acceder al servidor:

```
vsftpd : .example.com \
: twist /bin/echo "421 %h has been banned from this server!"
```

Para una explicación completa de las expansiones disponibles, así como también opciones de control de acceso adicionales, revise la sección 5 de la página man para hosts_access (man 5 hosts_access) y la página man de hosts_options.

Para recursos adicionales concernientes a los wrappers TCP, vea Sección 15.5.

15.3. xinetd

El demonio xinetd es un *super servicio* wrapped TCP que controla el acceso a un subconjunto de servicios de red populares incluyendo FTP, IMAP y Telnet. También proporciona opciones de configuración específicas al servicio para el control de acceso, registro mejorado, redireccionamiento y control de utilización de recursos.

Cuando un host cliente intenta conectarse a un servicio de red controlado por xinetd, el super servicio recibe la petición y verifica por cualquier regla de control de acceso wrappers TCP. Si se

permite el acceso, xinetd verifica que la conexión sea permitida bajo sus propias reglas para ese servicio y que el servicio no esté consumiendo más de la cantidad de recursos o si está rompiendo alguna regla. Luego comienza una instancia del servicio solicitado y pasa el control de la conexión al mismo. Una vez establecida la conexión, xinetd no interfiere más con la comunicación entre el host cliente y el servidor.

15.4. Archivos de configuración xinetd

Los archivos de configuración para xinetd son los siguientes:

- /etc/xinetd.conf — El archivo de configuración global de xinetd.
- /etc/xinetd.d/ directory — El directorio que contiene todos los archivos específicos al servicio.

15.4.1. El archivo /etc/xinetd.conf

El archivo /etc/xinetd.conf contiene parámetros de configuración generales los cuales afectan cada servicio bajo el control de xinetd. Se lee una vez cuando el servicio xinetd es iniciado, por esto para que los cambios de la configuración tomen efecto, el administrador debe reiniciar el servicio xinetd. Abajo se muestra un ejemplo del archivo /etc/xinetd.conf:

```
defaults
{
        instances               = 60
        log_type                = SYSLOG authpriv
        log_on_success          = HOST PID
        log_on_failure          = HOST
        cps                     = 25 30
}
includedir /etc/xinetd.d
```

Estas líneas controlan varios aspectos de xinetd:

- instances — Configura el máximo número de peticiones que xinetd puede manejar simultáneamente.
- log_type — Configura xinetd para usar la facilidad de registro authpriv, el cual escribe las entradas de registro al archivo /var/log/secure. Al agregar una directiva tal como FILE /var/log/xinetdlog aquí, creará un archivo de registro personalizado llamado xinetdlog en el directorio /var/log/.
- log_on_success — Configura xinetd a registrar si la conexión es exitosa. Por defecto, la dirección IP del host remoto y el ID del proceso del servidor procesando la petición son grabados.
- log_on_failure — Configura xinetd para registrar si hay una falla de conexión o si la conexión no es permitida.
- cps — Configura xinetd para no permitir más de 25 conexiones por segundo a cualquier servicio dado. Si se alcanza este límite, el servicio es retirado por 30 segundos.
- includedir /etc/xinetd.d/ — Incluye las opciones declaradas en los archivos de configuración específicos del servicio localizados en el directorio /etc/xinetd.d/. Consulte a Sección 15.4.2 para más información sobre este directorio.

Nota

A menudo, las configuraciones `log_on_success` y `log_on_failure` en `/etc/xinetd.conf` son modificadas aún más en los archivos de registro específicos al servicio. Por esta razón, puede que aparezca más información en el registro de un servicio dado que lo que puede indicar este archivo. Consulte Sección 15.4.3.1 para más detalles sobre las opciones de registro.

15.4.2. El directorio `/etc/xinetd.d/`

Los archivos en el directorio `/etc/xinetd.d/` contienen los archivos de configuración para cada servicio manejado por `xinetd` y los nombre de los archivos que se correlacionan con el servicio. Como sucede con `xinetd.conf`, este archivo es de sólo lectura cuando el servicio `xinetd` es arrancado. Para que los cambios tengan efecto, el administrador debe reiniciar el servicio `xinetd`.

El formato de los archivos en el directorio `/etc/xinetd.d/` usan las mismas convenciones que `/etc/xinetd.conf`. La razón principal por la que la configuración para cada servicio es almacenada en archivo separados es hacer más fácil la personalización y que sea menos probable afectar otros servicios.

Para tener una idea de cómo estos archivos están estructurados, considere el archivo `/etc/xinetd.d/telnet`:

```
service telnet
{
        flags           = REUSE
        socket_type     = stream
        wait            = no
        user            = root
        server          = /usr/sbin/in.telnetd
        log_on_failure  += USERID
        disable         = yes
}
```

Estas líneas controlan varios aspectos del servicio `telnet`:

- `service` — Define el nombre del servicio, usualmente para coincidir un servicio listado en el archivo `/etc/services`.
- `flags` — Configura cualquier número de atributos para la conexión. `REUSE` instruye `xinetd` a reutilizar el socket para una conexión Telnet.
- `socket_type` — Configura el socket de red a escribir a `stream`.
- `wait` — Define si el servicio es de un sólo hilo (`yes`) o de múltiples hilos (`no`).
- `user` — Define bajo qué ID de usuario el proceso se ejecutará.
- `server` — Define el binario ejecutable a lanzar.
- `log_on_failure` — Define los parámetros de registro para `log_on_failure` además de aquellos ya definidos en `xinetd.conf`.
- `disable` — Define si el servicio está activo o no.

15.4.3. Modificar archivos de configuración xinetd

Existe una gran cantidad de directivas disponibles para los servicios xinetd protegidos. Esta sección resalta algunos de las opciones usadas más comúnmente.

15.4.3.1. Opciones de registro

Las siguientes opciones de registro están disponibles para /etc/xinetd.conf y los archivos de configuración específicos al servicio en el directorio /etc/xinetd.d/.

Abajo se muestra una lista de algunas de las opciones de registro usadas más comúnmente:

- ATTEMPT — Indica que se intentó realizar una conexión pero que ésta falló (log_on_failure).
- DURATION — Indica el tiempo que un sistema remoto usa un servicio (log_on_success).
- EXIT — Indica el estado de salida o la señal de término del servicio (log_on_success).
- HOST — Indica la dirección IP de la máquina remota (log_on_failure y log_on_success).
- PID — Indica el ID del proceso del servidor que recibe la petición (log_on_success).
- RECORD — Graba información sobre el sistema remoto en el caso de que no se pueda iniciar el servicio. Esta opción la usan solo determinados tipos de servicios como login y finger, puede usar esta opción (log_on_failure).
- USERID — Registra el usuario remoto que está usando el método definido en RFC 1413 para todos los servicios de multi procesos (log_on_failure y log_on_success).

Para una lista completa de las opciones de registro, consulte la página de manual xinetd.conf.

15.4.3.2. Opciones de control de acceso

Los usuarios de servicios xinetd pueden seleccionar usar reglas de acceso a hosts de wrappers TCP, proporciona control de acceso a través de los archivos de configuración xinetd, o una mezcla de ambos. La información concerniente al uso de los archivos de control de acceso a hosts wrappers TCP se puede encontrar en Sección 15.2. Esta sección discute el uso de xinetd para controlar el acceso a los servicios.

Nota

A diferencia de los wrappers TCP, los cambios al control de acceso sólo tengan efecto si el administrador de xinetd reinicia el servicio xinetd.

El control de acceso xinetd es diferente del método usado por los wrappers TCP. Mientras que los wrappers TCP colocan toda la configuración del acceso dentro de dos archivos, /etc/hosts.allow y /etc/hosts.deny, el archivo de cada servicio en /etc/xinetd.d puede contener sus propias reglas de control de acceso.

Las opciones de acceso a host siguientes son soportadas por xinetd:

- only_from — Sólo permite que las máquinas específicas usen el servicio.
- no_access — Impide que estas máquinas usen el servicio.
- access_times — Especifica el intervalo de tiempo en el que un determinado servicio puede ser usado. El rango de tiempo debe especificarse en formato de 24 horas, HH:MM-HH:MM.

Las opciones only_from y no_access pueden usar una lista de direcciones IP o nombres de hosts, o pueden especificar una red completa. Como los wrappers TCP, la combinación del comando xinetd para el control del acceso con una configuración de conexión apropiada puede mejorar la seguridad mediante el bloqueo de peticiones de hosts vetados mientras que graba cada intento de conexión.

Por ejemplo, el siguiente archivo /etc/xinetd.d/telnet puede ser usado para bloquear el acceso a Telnet desde un un grupo de red particular y restringir el rango de tiempo general que inclusive los usuarios permitidos pueden conectarse:

```
service telnet
{
        disable         = no
        flags           = REUSE
        socket_type     = stream
        wait            = no
        user            = root
        server          = /usr/sbin/in.telnetd
        log_on_failure  += USERID
        no_access       = 10.0.1.0/24
        log_on_success  += PID HOST EXIT
        access_times    = 09:45-16:15
}
```

En este ejemplo, cuando un sistema cliente desde la red 10.0.1.0/24, tal como 10.0.1.2, intenta accesar el servicio Telnet, recibirá un mensaje indicando lo siguiente:

```
Connection closed by foreign host.
```

Además, su intento de conexión es registrado en /var/log/secure como sigue:

```
May 15 17:38:49 boo xinetd[16252]: START: telnet pid=16256 from=10.0.1.2
May 15 17:38:49 boo xinetd[16256]: FAIL: telnet address from=10.0.1.2
May 15 17:38:49 boo xinetd[16252]: EXIT: telnet status=0 pid=16256
```

Cuando esté usando wrappers TCP en conjunto con controles de acceso xinetd, es importante entender la relación entre los dos mecanismos de control de acceso.

A continuación se muestra el orden de las operaciones seguido por xinetd cuando un cliente solicita una conexión:

1. El demonio xinetd accesa las reglas de acceso a hosts wrappers TCP a través de una llamada a la librería libwrap.a. Si alguna regla de rechazo coincide con el host cliente, la conexión se rechaza. Si una regla de aceptación coincide con el host cliente, la conexión es pasada al xinetd.
2. El demonio xinetd verifica sus propias reglas de acceso para el servicio xinetd y el servicio solicitado. Si una regla de rechazo coincide con el host cliente la conexión es rechazada. De lo contrario, xinetd inicia una instancia del servicio solicitado y pasa el control de la conexión al mismo.

Importante

Se debe tener especial cuidado cuando se use el control de acceso wrappers TCP en conjunto con los controles xinetd. Un error en la configuración puede generar resultados no deseados.

15.4.3.3. Vincular y redirigir opciones

Los ficheros de configuración de servicios para el comando `xinetd` también soportan la vinculación del servicio a una dirección IP y el desvío de las peticiones entrantes para dicho servicio a otra dirección IP, nombre de la máquina o puerto.

La vinculación es controlada con la opción `bind` que se encuentra en los ficheros de configuración, y une específicamente el servicio a una dirección IP que se encuentre en uso en el sistema. Una vez configurada, la opción `bind` sólo permite peticiones para la dirección IP apropiada para accesar el servicio. De esta forma se puede vincular servicios diferentes a interfaces de red diferentes basados en la necesidad.

Esto es útil sobre todo para los sistemas con múltiples adaptadores de red o con múltiples direcciones IP. En tales sistemas, los servicios inseguros como Telnet, se pueden configurar de modo que solo escuche a la interfaz conectada a una red privada, y no a la interfaz conectada a Internet.

La opción `redirect` acepta la dirección IP o el nombre de la máquina seguido del número de puerto. Dice al servicio que desvíe todas las peticiones para dicho servicio a una localización y número de puerto específicos. Esta característica se usa para establecer otro número de puerto en el mismo sistema, desviar la petición a otra dirección IP en la misma máquina, cambiar la petición a otro sistema y puerto completamente diferentes o con la combinación de cualquiera de estas opciones. De esta manera, un usuario que está conectado a un determinado servicio en un sistema puede ser redirigido a otro sistema sin ninguna interrupción.

El demonio `xinetd` lleva a cabo este desvío lanzando un proceso que mantenga la conexión entre la máquina cliente que está mandando la petición y la máquina que está dando en ese momento el servicio, transfiriendo los datos de un sistema a otro.

El mayor beneficio de estas dos opciones se obtiene cuando se usan juntas. Vinculando un servicio a una dirección IP determinada en un sistema y luego desviando las peticiones de dicho servicio a una segunda máquina que solo puede ver la primera máquina, se puede usar un sistema interno que ofrezca servicios para una red completamente diferente. Alternativamente, estas opciones se pueden usar para limitar la exposición de un servicio determinado a una dirección IP conocida, así como desviar todas las peticiones a ese servicio a otra máquina configurada específicamente para ese objetivo.

Por ejemplo, considere un sistema que se usa como firewall con la característica siguiente para su servicio Telnet:

```
service telnet
{
        socket_type  = stream
        wait   = no
        server   = /usr/sbin/in.telnetd
        log_on_success  += DURATION USERID
        log_on_failure  += USERID
        bind                    = 123.123.123.123
        redirect                = 10.0.1.13 21 23
}
```

Las opciones `bind` y `redirect` en este archivo aseguran que el servicio Telnet en la máquina esté enlazado con la dirección IP externa (123.123.123.123), la que se encarga de Internet. Además, todas las peticiones del servicio Telnet enviadas a 123.123.123.123 son redirigidas a través de una segunda tarjeta de red a una dirección IP interna (10.0.1.13) a la que solo tienen acceso el firewall y los sistemas internos. El firewall manda luego la comunicación entre los dos sistemas y el sistema que se está conectando piensa que está conectado a 123.123.123.123 mientras que, de hecho, está conectado a otra máquina.

Esta característica es útil para los usuarios con conexiones de banda ancha y con una única dirección IP fija. Cuando se usa la Traducción de las direcciones de la red (la Network Address Translation NAT), los sistemas detrás de la máquina gateway, que están usando direcciones IP internas, no están disponibles desde afuera del sistema gateway. Sin embargo, cuando determinados servicios

controlados por `xinetd` son configurados con las opciones `bind` y `redirect`, la máquina gateway puede funcionar como un tipo de proxy entre los sistemas externos y una máquina interna particular configurada para proporcionar el servicio. Además, las opciones de control de acceso `xinetd` y de conexión están también disponibles para protección adicional, tal como limitar el número de conexiones simultáneas para el servicio redirigido.

15.4.3.4. Opciones de administración de recursos

El demonio `xinetd` puede añadir un nivel básico de protección de un ataque Denial of Service (DoS). Abajo se encuentra una lista de las directivas que pueden ayudar en limitar la efectividad de tales ataques:

- `per_source` — Define el número máximo de instancias para un servicio por dirección IP. Acepta sólo enteros como argumentos y puede ser usado en `xinetd.conf` y los archivos de configuración específicos al servicio `xinetd.d/`.
- `cps` — Define el máximo número de conexiones por segundo. Esta directiva toma dos argumentos enteros separados por un espacio en blanco. El primero es el número máximo de conexiones permitidas por segundo. El segundo es el número de segundos `xinetd` que debe esperar antes de reactivar el servicio. Sólo acepta enteros como argumentos y puede ser usado en ambos `xinetd.conf` y los archivos de configuración específicos al servicio en el directorio `xinetd.d/`.
- `max_load` — Indica el umbral de uso del CPU para un servicio. Acepta un argumento en forma de número de punto flotante.

Hay más opciones de administración de recursos disponibles para `xinetd`. Vea el capítulo titulado *Seguridad del servidor* en el *Manual de seguridad de Red Hat Linux* para más información. También consulte la página del manual de `xinetd.conf`.

15.5. Recursos adicionales

En la documentación del sistema y en el web puede encontrar información adicional concerniente a los wrappers TCP y a `xinetd`.

15.5.1. Documentación instalada

La documentación en su sistema es un buen lugar para comenzar a buscar información sobre los Wrappers TCP, `xinetd` y las opciones de control de acceso.

- `/usr/share/doc/tcp_wrappers-<`*`version`*`>/` — Contiene un archivo `README` que discute cómo los wrappers TCP funcionan y los diferentes riesgos de spoofing de host y de direcciones IP que existen.
- `/usr/share/doc/xinetd-<`*`version`*`>/` — Incluye un archivo `README` que discute aspectos del control de acceso y un archivo `sample.conf` con varias ideas para la modificación de archivos de configuración específicos al servicio en el directorio `/etc/xinetd.d/`.
- `man 5 hosts_access` — La página del manual para los archivos de control de acceso wrappers TCP.
- `man hosts_options` — La página del manual para los campos de opciones de wrappers TCP.
- `man xinetd.conf` — La página del manual listando las opciones de configuración `xinetd`.
- `man xinetd` — La página del manual para el demonio del super servicio `xinetd`.

15.5.2. Sitios Web de utilidad

- http://www.xinetd.org — El sitio principal de `xinetd`, contiene archivos de configuración de ejemplo, una lista de las características completas y una sección de Preguntas más frecuentes FAQ.
- http://www.macsecurity.org/resources/xinetd/tutorial.shtml — Un tutorial completo que discute las diferentes formas de ajustar los archivos de configuración `xinetd` por defecto para cubrir objetivos específicos.

15.5.3. Libros relacionados

- *Manual de seguridad de Red Hat Linux* ; Red Hat, Inc. — Proporciona una visión general de la seguridad para estaciones de trabajo, servidor y redes con sugerencias específicas sobre wrappers TCP y `xinetd`.
- *Hacking Linux Exposed* por Brian Hatch, James Lee y George Kurtz; Osbourne/McGraw-Hill — Un recurso excelente de seguridad con información sobre wrappers TCP y `xinetd`.

Capítulo 16.

iptables

Red Hat Linux contiene herramientas avanzadas para el *filtrado de paquetes* de redes — el proceso de controlar los paquetes de red cuando entran, se mueven y salen de la red dentro del kernel. Los kernels anteriores al 2.4 confiaban en `ipchains` para el filtrado de paquetes y usaban listas de reglas aplicadas a los paquetes en cada paso del proceso de filtrado. La introducción de kernel 2.4 kernel trajo consigo el `iptables` (también llamado *netfilter*), lo cual es similar a `ipchains` pero expande enormemente el ámbito y el control disponible para el filtrado de paquetes de red.

Este capítulo se centra en las bases del filtrado esencial de paquetes, define las diferencias entre `ipchains` e `iptables`, explica las diferentes opciones disponibles con comandos `iptables`, y muestra cómo las reglas de filtrado se pueden conservar tras el reinicio del sistema.

Para instrucciones sobre cómo construir reglas `iptables` o configurar un firewall basado en estas reglas, consulte a Sección 16.5.

Aviso

El mecanismo predeterminado del firewall en la versión 2.4 del kernel puede usar el comando `iptables`, pero no se puede usar si ya se está ejecutando `ipchains`. Si `ipchains` está presente durante el arranque, el kernel avisará que hay un error y no podrá arrancar `iptables`.

Estos errores no afectan la funcionalidad del comando `ipchains`.

16.1. Filtrado de paquetes

El tráfico se mueve a través de una red en *paquetes*. Un paquete de red es una colección de datos en diferentes tamaños y formatos. Para enviar un fichero por red, el ordenador emisor debe en primer lugar partirlo en diferentes paquetes usando las reglas del protocolo de red. Cada uno de estos paquetes contiene una parte pequeña de los datos del fichero. Cuando recibe la transmisión, el ordenador receptor, reensambla los paquetes y construye de nuevo el fichero el fichero.

Cada paquete contiene información que le ayuda a navegar por la red y moverse hacia su destino. El paquete puede decirle a los ordenadores a lo largo del camino, así como al ordenador destino, de dónde viene, a dónde va, qué tipo de paquete es, y otras muchas cosas más. La mayoría de los paquetes se diseñan para transportar datos, pero algunos protocolos pueden usar los paquetes de forma especial. El protocolo *Transmission Control Protocol (TCP)*, por ejemplo, utiliza un paquete SYN, que no contiene datos, para iniciar la comunicación entre dos sistemas.

El kernel de Linux contiene la característica interna de filtrado de paquetes, permitiendo aceptar algunos de ellos en el sistema mientras que intercepta y para a otros. El filtro de red del kernel 2.4 tiene tres *tablas* o *listas de reglas*. Son las siguientes:

- `filter` — La tabla por defecto para el manejo de paquetes de red.
- `nat` — Usada para alterar paquetes que crean una nueva conexión.
- `mangle` — Usada por tipos específicos de alteración de paquetes.

Cada una de estas tablas tiene un grupo de *cadenas* internas que corresponden a las acciones llevadas a cabo por el filtro de red en el paquete.

Las cadenas internas para la tabla `filtro` son las siguientes:

- *INPUT* — Aplica a los paquetes recibidos a través de una interfaz de red.
- *OUTPUT* — Esta cadena sirve para paquetes enviados por medio de la misma interfaz de red que recibió los paquetes.
- *FORWARD* — Esta cadena sirve para paquetes recibidos en una interfaz de red y enviados en otra.

Las cadenas internas para la tabla `nat` son las siguientes:

- *PREROUTING* — Esta cadena altera paquetes recibidos por medio de una interfaz de red cuando llegan.
- *OUTPUT* — Esta cadena altera paquetes generados localmente antes de que sean dirigidos por medio de una interfaz de red.
- *POSTROUTING* — Esta cadena altera paquetes antes de que sean enviados por medio de una interfaz de red.

Las cadenas internas para la tabla `mangle` son las siguientes:

- *PREROUTING* — Esta cadena altera paquetes recibidos por medio de una interfaz de red antes de que sean dirigidos.
- *OUTPUT* — Esta cadena altera paquetes generados localmente antes de que sean dirigidos por medio de una interfaz de red.

Cada paquete de red recibido o enviado de un sistema Linux está sujeto a al menos una tabla.

Un paquete puede que sea verificado contra muchas reglas dentro de la lista de reglas antes de llegar al final de una cadena. La estructura y propósito de estas reglas puede variar, pero normalmente buscan identificar un paquete que viene de o se dirige a una direccción IP en particular o un conjunto de direcciones al usar un determinado protocolo y servicio de red.

Independientemente de su destino, cuando un paquete cumple una regla en particular en una de las tablas, se asignan a un *objetivo (target)* particular. Si la regla especifica un objetivo `ACCEPT` para un paquete que coincida, el paquete se salta el resto de las verificaciones de la regla y se permite que continúe hacia su destino. Si una regla especifica un objetivo `DROP`, a ese paquete se le niega el acceso al sistema y no se envia nada de vuelta al servidor que envió el paquete. Si una regla especifica un objetivo `QUEUE`, el paquete se pasa al espacio del usuario. Si una regla especifica el objetivo opcional `REJECT`, el paquete es entregado, pero se envia un paquete de error al que envió el paquete.

Cada cadena tiene una política por defecto de `ACCEPT`, `DROP`, `REJECT`, o `QUEUE`. Si ninguna de estas reglas en la cadena se aplican al paquete, entonces el paquete es tratado de acuerdo a la política por defecto.

El comando `iptables` configura estas tablas, así como también configura nuevas tablas si es necesario.

16.2. Diferencias entre `iptables` e `ipchains`

A primera vista, `ipchains` y `iptables` parecen ser bastante similares. Ambos métodos de filtrado de paquetes usan cadenas de reglas operando dentro del kernel de Linux para decidir no sólo qué paquetes se permite entrar o salir, sino también qué hacer con los paquetes que cumplen determinadas reglas. Sin embargo, `iptables` proporciona un método mucho más extensible de filtrado de paquetes, proporcionando al administrador un nivel de control mucho más refinado sin tener que aumentar la complejidad del sistema entero.

Más concretamente, los usuarios que se encuentren cómodos con `ipchains` deberían tener cuidado con las siguientes diferencias significativas entre `ipchains` e `iptables` antes de utilizar `iptables`:

- *Bajo iptables, cada paquete filtrado se procesa únicamente usando las reglas de una cadena, en lugar de hacerse con múltiples.* Por ejemplo, un paquete FORWARD que llega al sistema usando ipchains tendrá que pasar por las cadenas INPUT, FORWARD, y OUTPUT para llegar a su destino. Sin embargo iptables sólo envía paquetes a la cadena INPUT si su destino es el sistema local y tan sólo los envía a la cadena OUTPUT si el sistema local es quien genera los paquetes. Por esta razón, coloque la regla designada para interceptar un paquete particular en la regla que en verdad verá el paquete.
- *El objetivo DENY ha sido cambiado a DROP.* En ipchains, los paquetes que coincidan una regla en una cadena podrían ser dirigidos al objetivo DENY. Este objetivo debe ser cambiado a DROP bajo iptables.
- *El orden es importante cuando se esten colocando opciones en una regla.* Anteriormente, con ipchains, el orden de las opciones de una regla no importaba. El comando iptables usa una sintaxis estricta. Por ejemplo, en comandos iptables el protocol (ICMP, TCP, o UDP) debe ser especificado antes del puerto fuente o destino.
- *Cuando especificamos las interfaces de red que vamos a usar en una regla, deberemos utilizar sólo interfaces de entrada (opción -i) con cadenas INPUT o FORWARD y las de salida (opción -o) con cadenas FORWARD o OUTPUT.* Esto es necesario debido al hecho de que las cadenas OUTPUT no se utilizan más con las interfaces de entrada, y las cadenas INPUT no son vistas por los paquetes que se mueven hacia las interfaces de salida.

Esta no es una lista completa de los cambios, dado que iptables representa un filtro de red re-escrito. Para información más especifica, consulte *Linux 2.4 Packet Filtering HOWTO* encontrado en Sección 16.5.

16.3. Opciones usadas en comandos iptables

Las reglas que permiten a los paquetes ser filtrados por el kernel se ponen en funcionamiento ejecutando el comando iptables. Cuando use el comando iptables, debe especificar las opciones siguientes:

- *Packet Type* — Dicta qué tipo de paquetes filtra el comando.
- *Packet Source/Destination* — Dicta qué paquetes filtra el comando basándose en el origen o destino del paquete.
- *Target* — Indica qué acción es tomada en paquetes que cumplen los criterios mencionados anteriormente.

Las opciones usadas con la regla dada iptables deben estar agrupadas lógicamente, basándose en el propósito y en las condiciones de la regla general, para que la regla sea válida.

16.3.1. Tablas

Un aspecto muy potente de iptables es que se pueden utilizar múltiples tablas para decidir el destino de un paquete particular. Gracias a la naturaleza extensible de iptables, se pueden crear tablas especializadas y almacenarlas en el directorio /lib/modules/<*kernel-version*>/kernel/net/ipv4/netfilter/, donde <*kernel-version*> corresponde al número de la versión del kernel.

La tabla por defecto, llamada filter, contiene las cadenas estándard por defecto para INPUT, OUTPUT, y FORWARD. Esto es similar a las cadenas estándar que se utilizan con ipchains. Sin embargo, por defecto, iptables también incluye dos tablas adicionales que realizan tareas de filtrado específico de paquetes. La tabla nat se puede utilizar para modificar las direcciones de origen y destino grabadas en un paquete, y la tabla mangle permite alterar los paquetes de forma especializada.

Cada tabla contiene las cadenas por defecto que realizan las tareas necesarias basadas en el propósito de la tabla, aún cuando se pueden añadir nuevas cadenas a cualquier tabla.

16.3.2. Estructura

Muchos comandos `iptables` tienen la siguiente estructura:

```
iptables [-t <table-name>]
<command>
<chain-name>
<parameter-1> \
        <option-1>
<parameter-n>
<option-n>
```

En este ejemplo, la opción *<table-name>* permite al usuario seleccionar una tabla diferente a la tabla predeterminada `filter` a usar con el comando. La opción *<command>* indica una acción específica a realizar, tal como anexar o eliminar la regla especificada por la opción *<chain-name>*. Luego de la opción *<chain-name>* se encuentran un par de parámetros y opciones que definen qué pasará cuando un paquete coincide con la regla.

Cuando miramos la estructura de un comando `iptables`, es importante recordar que, al contrario que la mayoría de los comandos, la longitud y complejidad de un comando `iptables` puede cambiar en función de su propósito. Un comando simple para borrar una regla de una cadena puede ser muy corto, mientras que un comando diseñado para filtrar paquetes de una subred particular usando un conjunto de parámetros específicos y opciones puede ser mucho más largo. Al crear comandos `iptables` puede ser de ayuda reconocer que algunos parámetros y opciones pueden crear la necesidad de utilizar otros parámetros y opciones para especificar algo de los requisitos de la opción anterior. Para construir una regla válida, esto deberá continuar hasta que todos los parámetros y opciones que requieran otro conjunto de opciones hayan sido satisfechos.

Teclee `iptables -h` para ver una lista detallada de la estructura de los comandos `iptables`.

16.3.3. Comandos

Los comandos le dicen a `iptables` que realice una tarea específica. Solamente un comando se permite por cada cadena de comandos `iptables`. Excepto el comando de ayuda, todos los comandos se escriben en mayúsculas.

Los comandos de `iptables` son los siguientes:

- `-A` — Añade la regla `iptables` al final de la cadena especificada. Este es el comando utilizado para simplemente añadir una regla cuando el orden de las reglas en la cadena no importa.
- `-C` — Verifica una regla en particular antes de añadirla en la cadena especificada por el usuario. Este comando puede ser de ayuda para construir reglas `iptables` complejas pidiéndole que introduzca parámetros y opciones adicionales.
- `-D` — Borra una regla de una cadena en particular por número (como el `5` para la quinta regla de una cadena). Puede también teclear la regla entera e `iptables` borrará la regla en la cadena que corresponda.
- `-E` — Renombra una cadena definida por el usuario. Esto no afecta la estructura de la tabla.
- `-F` — Libera la cadena seleccionada, que borra cada regla de la cadena. Si no se especifica ninguna cadena, este comando libera cada regla de cada cadena,
- `-h` — Proporciona una lista de estructuras de comandos, así como también un resúmen rápido de parámetros de comandos y opciones.

- `-I` — Inserta una regla en una cadena en un punto especificado por un valor entero definido por el usuario. Si no se especifica ningún número, `iptables` colocará el comando en el tope de la cadena.

  
  Atención

  Tenga cuidado con cúal opción (`-A` o `-I`) es usada cuando esté añadiendo una regla. El orden de las reglas en una cadena es importante para determinar cual regla aplica a cuáles paquetes.

- `-L` — Lista todas las reglas de la cadena especificada tras el comando. Para ver una lista de todas las cadenas en la tabla `filter` por defecto. La sintaxis siguiente deberá utilizarse para ver todas la listas de todas las reglas de una cadena específica en una tabla en particular:
  `iptables -L <chain-name> -t <table-name>`

  Opciones más potentes para el comando `-L`, que proporcionan números a las reglas y permiten más descripciones en las reglas, así como otros, se pueden ver en Sección 16.3.7.

- `-N` — Crea una nueva cadena con un nombre especificado por el usuario.
- `-P` — Configura la política por defecto para una cadena en particular de tal forma que cuando los paquetes atraviesen la cadena completa sin cumplir ninguna regla, serán enviados a un objetivo en particular, como puedan ser ACCEPT o DROP.
- `-R` — Reemplaza una regla en una cadena particular. El número de la regla debe ser especificado después del nombre de la cadena. La primera regla en una cadena corresponde a la regla número uno.
- `-X` — Borra una cadena especificada por el usuario. No se permite borrar ninguna de las cadenas predefinidas para cualquier tabla.
- `-Z` — Pone ceros en los contadores de byte y de paquete en todas las cadenas de una tabla en particular.

16.3.4. Parámetros

Una vez que se especifiquen ciertos comandos `iptables`, incluyendo aquellos para añadir, anexar, eliminar, insertar o reemplazar reglas dentro de una cadena, se requieren parámetros para construir una regla de filtrado de paquetes.

- `-c` Resetea los contadores de una regla en particular. Este parámetro acepta las opciones `PKTS` y `BYTES` para especificar qué contador hay que resetear.
- `-d` — Configura el nombre de la máquina destino, dirección IP o red de un paquete que coincide con la regla. Cuando se coincida una red, los siguientes formatos de direcciones IP o máscaras de red son soportados:
  - *`N.N.N.N/M.M.M.M`* — Donde *`N.N.N.N`* es el rango de direcciones IP y *`M.M.M.M`* es la máscara de la red.
  - *`N.N.N.N/M`* — Donde *`N.N.N.N`* es el rango de direcciones IP y *`M`* es la máscara de la red.

- `-f` Aplica esta regla sólo a los paquetes fragmentados.

  Usando la opción `!` después de este parámetro, únicamente los paquetes no fragmentados se tendrán en cuenta.

- `-i` — Configura la interfaz de red entrante, tal como `eth0` o `ppp0`. Con `iptables`, este parámetro opcional puede ser usado solamente con las cadenas INPUT y FORWARD cuando es usado con la tabla `filter` y la cadena PREROUTING con las tablas `nat` y `mangle`.

  Este parámetro también soporta las siguientes opciones especiales:

- ! — Dice a este parámetro que no concuerde, queriendo decir esto que las interfaces especificadas se excluirán de esta regla.
- + — Un caracter tipo comodín utilizado para coincidir todas las interfaces con una cadena de caracteres particular. Por ejemplo, el parámetro -i eth+ aplicará esta regla a cualquier interfaz Ethernet pero excluirá cualquier otra interfaz, tal com ppp0.

Si el parámetro -i se utiliza sin especificar ninguna interfaz, todas las interfaces estarán afectadas por la regla.

- -j — Le dice a iptables que salte a un objetivo particular cuando un paquete coincide con una regla. Los objetivos válidos a usar después de la opción -j incluye las opciones estándar, ACCEPT, DROP, QUEUE, y RETURN, así como también las opciones extendidas que están disponibles a través de los módulos cargados por defecto con el paquete RPM de Red Hat Linux iptables, como LOG, MARK, y REJECT, entre otros. Consulte la página del manual iptables para más información sobre esto y otros objetivos.

  Puede también dirigir un paquete coincidiendo esta regla a una cadena definida por el usuario fuera de la cadena actual para que otras reglas puedan ser aplicadas al paquete.

  Si no especifica ningún objetivo, el paquete se mueve hacia atrás en la regla sin llevar a cabo ninguna acción. A pesar de todo, el contador para esta regla se sigue incrementando en uno, a partir del momento en el que el paquete se adecua a la regla especificada.
- -o — Configura la interfaz de red de salida para una regla y puede ser usada solamente con las cadenas OUTPUT y FORWARD en la tabla de filtro y la cadena POSTROUTING en las tablas nat y mangle. Estos parámetros de opciones son los mismos que aquellos de la interfaz de entrada (-i).
- -p — Configura el protocolo IP para la regla, el cual puede ser icmp, tcp, udp, o all, para coincidir todos los protocolos soportados. Además, se puede usar cualquier protocolo listado en /etc/protocols. Si esta opción es omitida cuando se esté creando una regla, la opción all es la opción por defecto.
- -s — Configura la fuente para un paquete particular usando la misma sintaxis que el parámetro (-d).

16.3.5. Opciones de identificación de paquetes

Diferentes protocolos de red proporcionan opciones especializadas las cuales se pueden configurar de formas específicas para coincidir un paquete particular usando ese protocolo. Por supuesto, el protocolo debe ser especificado primero con el comando iptables, mediante el uso de -p tcp *<protocol-name>* (donde *<protocol-name>* es el protocolo objetivo), para hacer las opciones disponibles para ese protocolo.

16.3.5.1. Protocolo TCP

Estas opciones de identificación están disponibles en el protocolo TCP (opción -p tcp):

- --dport — Configura el puerto de destino para el paquete. Use bien sea un nombre de servicio (tal como www o smtp), número de puerto, o el rango de números de puertos para configurar esta opción. Para hojear los nombres y alias de los servicios de red y los números que ellos usan, visualice el archivo /etc/services. La opción --destination-port es sinónimo con --dport.

  Para especificar un rango de números de puertos, separe los dos números con dos puntos (:), tal como -p tcp --dport 3000:3200. El rango válido aceptable es 0:65535.

  Use un caracter de exclamación (!) después de la opción --dport para indicar a iptables que coincida todos los paquetes que *no* usan el servicio de red o puerto.

- `--sport` — Configura el puerto fuente del paquete usando las mismas opciones que `--dport`. La opción `--source-port` es sinónimo con `--sport`.
- `--syn` Provoca que todos los paquetes designados de TCP, comúnmente llamados *paquetes SYN*, cumplan esta regla. Cualquier paquete que esté llevando un payload de datos no será tocado. Si se sitúa un punto de exclamación (`!`) como bandera tras la opción `--syn` se provoca que todos los paquetes no-SYN sean seleccionados.
- `--tcp-flags` — Permite a los paquetes TCP packets con bits específicos o banderas, ser coincididos con una regla. La opción `--tcp-flags` acepta dos parámetros. El primer parámetro es la máscara, la cual configura banderas a ser examinadas en el paquete. El segundo parámetro se refiere a la bandera que se debe configurar para poder coincidir.

  Las banderas posibles son:

  - `ACK`
  - `FIN`
  - `PSH`
  - `RST`
  - `SYN`
  - `URG`
  - `ALL`
  - `NONE`

  Por ejemplo, una regla `iptables` que contiene `-p tcp --tcp-flags ACK,FIN,SYN SYN` tan sólo seleccionará los paquetes TCP que tengan la bandera SYN activo y las banderas ACK y FIN sin activar.

  Usando el caracter de exclamación (`!`) después de `--tcp-flags` reversa el efecto de la opción de coincidencia.
- `--tcp-option` Intenta seleccionar con opciones específicas de TCP que pueden estar activas en un paquete en particular. Esta opción se puede revertir con el punto de exclamación (`!`).

16.3.5.2. Protocolo UDP

Estas opciones de selección están disponibles para el protocolo UDP (`-p udp`):

- `--dport` — Especifica el puerto destino del paquete UDP, usando el nombre del servicio, número de puerto, o rango de números de puertos. La `--destination-port` que coincide la opción es sinónimo con `--dport`. Consulte a la opción `--dport` en Sección 16.3.5.1 para ver las formas de usar esta opción.
- `--sport` — Especifica el puerto fuente de un paquete UDP, usando el nombre del servicio, número de puerto, o rango de números de puertos. La `--source-port` opción es sinónimo con `--sport`. Consulte a la `--sport` en Sección 16.3.5.1 para ver las formas de usar esta opción.

16.3.5.3. Protocolo ICMP

Estas opciones de coincidencia están disponibles para el Internet Control Message Protocol (ICMP) (`-p icmp`):

- `--icmp-type` Selecciona el nombre o el número del tipo ICMP que concuerde con la regla. Se puede obtener una lista de nombres válidos ICMP tecleando el comando `iptables -p icmp -h`.

16.3.5.4. Módulos con opciones de selección adicionales

Opciones adicionales de coincidencia están disponibles a través de los módulos por el comando `iptables`. Para usar un módulo de opciones de coincidencia, cargue el módulo por nombre usando la opción `-m`, tal como `-m <module-name>` (reemplazando `<module-name>` con el nombre del módulo).

Un gran número de módulos están disponibles por defecto. Hasta es posible crear sus propios módulos para proporcionar funcionalidades de opciones de coincidencia adicionales.

Existen muchos módulos, pero sólo se discuten aquí los más populares.

- `limit` module — Permite colocar un límite en cuántos paquetes son coincididos a una regla particular. Esto es especialmente beneficioso cuando se registren las coincidencias a las reglas y así una gran cantidad de paquetes coincidentes no sobrecarguen el registro del sistema con mensajes repetitivos o usen los recursos del sistema.

  El módulo `limit` habilita las opciones siguientes:

  - `--limit` — Configura el número de coincidencias en un intervalo de tiempo, especificado con un número y un modificador de tiempo ordenados en el formato `<número>/<tiempo>`. Por ejemplo, si usamos `--limit 5/hour` sólo dejaremos que una regla sea efectiva cinco veces a la hora.

    Si no se utiliza ningún número ni modificador de tiempo, se asume el siguiente valor por defecto: `3/hour`.

  - `--limit-burst` — Configura un límite en el número de paquetes capaces de cumplir una regla en un determinado tiempo. Esta opción deberá ser usada junto con la opción `--limit`, y acepta un número para configurar el intervalo de tiempo (threshold).

    Si no se especifica ningún número, tan sólo cinco paquetes serán capaces inicialmente de cumplir la regla.

- módulo `state` — Habilita la coincidencia de estado.

  El módulo `state` tiene las siguientes opciones:

  - `--state` — coincide un paquete con los siguientes estados de conexión:

    - `ESTABLISHED` El paquete seleccionado se asocia con otros paquetes en una conexión establecida.
    - `INVALID` El paquete seleccionado no puede ser asociado a una conexión conocida.
    - `NEW` El paquete seleccionado o bien está creando una nueva conexión o bien forma parte de una conexión de dos caminos que antes no había sido vista.
    - `RELATED` El paquete seleccionado está iniciando una nueva conexión en algún punto de la conexión existente.

    Estos estados de conexión se pueden utilizar en combinación con otros separándolos mediante comas como en `-m state --state INVALID, NEW`.

- módulo `mac` — Habilita la coincidencia de direcciones MAC de hardware.

  El módulo `mac` activa las opciones siguientes:

  - `--mac-source` — Coincide una dirección MAC a la tarjeta de red que envió el paquete. Para excluir una dirección MAC de la regla, coloque un símbolo de exclamación (`!`) después de la opción `--mac-source`.

Para visualizar otras opciones disponibles a través de los módulos, consulte la página del manual de `iptables`.

16.3.6. Opciones del objetivo

Una vez que un paquete ha coincidido con una regla, la regla puede dirigir el paquete a un número de objetivos diferentes que deciden su suerte y, posiblemente, toman acciones adicionales. Cada cadena tiene un objetivo por defecto, el cual es usado si ninguna de las reglas en esa cadena coinciden con un paquete o si ninguna de las reglas que coinciden con el paquete especifica un objetivo.

Los siguientes son los objetivos estándar:

- *<user-defined-chain>* — Reemplace *<user-defined-chain>* con el nombre de una cadena definida por el usuario dentro de la tabla. Este objetivo pasa el paquete a la cadena objetivo.
- `ACCEPT` — Permite que el paquete se mueva hacia su destino (o hacia otra cadena, si no ha sido configurado ningún destino para seguir a esta cadena).
- `DROP` — Deja caer el paquete sin responder al solicitante. El sistema que envia el paquete no es notificado de esta falla.
- `QUEUE` — El paquete se pone en una cola para ser manejado por una aplicación en el espacio de usuario.
- `RETURN` — Para la verificación del paquete contra las reglas de la cadena actual. Si el paquete con un destino `RETURN` cumple una regla de una cadena llamada desde otra cadena, el paquete es devuelto a la primera cadena para retomar la verificación de la regla allí donde se dejó. Si la regla `RETURN` se utiliza en una cadena predefinida, y el paquete no puede moverse hacia la cadena anterior, el objetivo por defecto de la cadena actual decide qué acción llevar a cabo.

Además de estos objetivos standard, se pueden usar otros más con extensiones llamadas *módulos de objetivos* (target modules), que trabajan de forma similar a como los hacían los módulos de las opciones de selección. Para obtener más información sobre estos módulos, mire en Sección 16.3.5.4.

Existen varios módulos extendidos de objetivos, la mayoría de los cuales tan sólo se aplicarán a tablas o situaciones específicas. Un par de estos módulos de los más populares e incluidos por defecto en Red Hat Linux serían:

- `LOG` — Registra todos los paquetes que coinciden esta regla. Puesto que los paquetes son registrados por el kernel, el archivo `/etc/syslog.conf` determina dónde estas entradas de registro serán escritas. Por defecto, son colocadas en el archivo `/var/log/messages`.

  Se pueden usar varias opciones tras el objetivo `LOG` para especificar la manera en la que tendrá lugar el registro:

  - `--log-level` — Configura el nivel de prioridad del registro de eventos. Una lista de los niveles de prioridad se puede encontrar en la página del manual de `syslog.conf`.
  - `--log-ip-options` Cualquier opción en la cabecera de un paquete IP se guarda en el registro.
  - `--log-prefix` — Coloca una cadena de hasta 29 caracteres antes de la línea de registro cuando es escrita. Esto es muy útil para la escritura de filtros de syslog para usarlos en conjunto con el registro de paquetes.
  - `--log-tcp-options` — Cualquier opción colocada en la cabecera de un paquete TCP es registrada.
  - `--log-tcp-sequence` Escribe el número de secuencia TCP del paquete en el registro del sistema.

- REJECT — Envia un paquete de error de vuelta al sistema remoto y deja caer el paquete.

  El objetivo REJECT acepta --reject-with <*type*> (donde <*type*> es el tipo de rechazo) el cual permite que se envie información más detallada devuelta con el paquete de error. El mensaje port-unreachable es el <*tipo*> de error por defecto dado si no se usa otra opción. Para una lista completa de los <*tipos*> de opciones que se pueden usar, consulte la página del manual de iptables.

Otras extensiones de objetivos, incluyendo muchas que son útiles para el enmascaramiento de IP usando la tabla nat o con alteración de paquetes usando la tabla mangle, se puede encontrar en la página del manual de iptables.

16.3.7. Opciones de listado

El comando predeterminado para listar, iptables -L, proporciona una vista muy básica de los filtros por defecto de las cadenas actuales de la tabla. Las opciones adicionales proporcionan más información:

- -v Muestra la salida por pantalla, como el número de paquetes y bytes que cada cadena ha visto, el número de paquetes y bytes que cada regla ha encontrado, y qué interfaces se aplican a una regla en particular.
- -x Expande los números en sus valores exactos. En un sistema ocupado, el número de paquetes y bytes vistos por una cadena en concreto o por una regla puede estar abreviado usando K (miles), M (millones), y G (billones) detrás del número. Esta opción fuerza a que se muestre el número completo.
- -n Muestra las direcciones IP y los números de puertos en formato numérico, en lugar de utilizar el nombre del servidor y la red tal y como se hace por defecto.
- --line-numbers Proporciona una lista de cada cadena junto con su orden numérico en la cadena. Esta opción puede ser útil cuando esté intentando borrar una regla específica en una cadena, o localizar dónde insertar una regla en una cadena.
- -t — Especifica un nombre de tabla.

16.4. Guardar información de iptables

Las reglas creadas con el comando iptables son almacenadas en memoria. Si el sistema es reiniciado después de configurar la reglas de iptables, se perderán. Para que las reglas de filtrado de red persistan luego de un reinicio del sistema, estas necesitan ser guardadas. Para hacerlo, conéctese como root y escriba:

```
/sbin/service iptables save
```

Esto ejecuta el init script de iptables, el cual ejecuta el programa /sbin/iptables-save y escribe la configuración actual de iptables a /etc/sysconfig/iptables. Este archivo sólo debería estar accesible para root.

La próxima vez que se inicie el sistema, el script de inicio de iptables volverá a aplicar las reglas guardadas en /etc/sysconfig/iptables usando el comando /sbin/iptables-restore.

Aún cuando siempre es una buena idea probar una regla de iptables antes de confirmar los cambios al archivo /etc/sysconfig/iptables, es posible copiar reglas iptables en este archivo desde otra versión del sistema de este archivo. Esto proporciona una forma rápida de distribuir conjuntos de reglas iptables a muchas máquinas.

Importante

Si se está distribuyendo el archivo `/etc/sysconfig/iptables` a otras máquinas, escriba `/sbin/service iptables restart` para que las nuevas reglas tomen efecto.

16.5. Recursos adicionales

Refiérase a las fuentes siguientes para información adicional sobre filtrado de paquetes con `iptables`.

16.5.1. Documentación instalada

- `man iptables` — Contiene una descripción detallada de varios comandos, parámetros y otras opciones.

16.5.2. Sitios web útiles

- http://netfilter.samba.org — Contiene información surtida sobre `iptables`, incluyendo una lista FAQ que responde a problemas específicos y otras guías útiles por Rusty Russell, el mantenedor del firewall de Linux IP. Los documentos HOWTO en el sitio cubren temas tales como conceptos básicos de redes, filtrado de paquetes del kernel 2.4 y configuraciones NAT.
- http://www.linuxnewbie.org/nhf/Security/IPtables_Basics.html — una visión básica y general sobre la forma en la que los paquetes se mueven dentro del kernel de Linux, además de una introducción sobre cómo se construyen comandos `iptables` simples.
- http://www.redhat.com/support/resources/networking/firewall.html — Esta página contiene enlaces a una variedad de recursos actualizados de filtros.

Capítulo 17.

Kerberos

Kerberos es un protocolo de seguridad creado por MIT que usa una criptografía de claves simétricas para validar usuarios con los servicios de red — evitando así tener que enviar contraseñas a través de la red. Al validar los usuarios para los servicios de la red por medio de Kerberos, se frustran los intentos de usuarios no autorizados que intentan interceptar contraseñas en la red.

17.1. Ventajas de Kerberos

La mayoría de las redes usan esquemas de autenticación basados en contraseñas. Tales esquemas requieren que cuando un usuario necesita una autenticación en un servidor de red, debe proporcionar un nombre de usuario y una contraseña. Lamentablemente, la información de autenticación para muchos servicios se transmite sin estar encriptada. Para que un esquema de este tipo sea seguro, la red tiene que estar inaccequible a usuarios externos, y todos los usuarios de la red deben ser de confianza.

Aún en este caso, una vez que la red se conecte a la Internet, ya no puede asumir que la red es segura. Cualquier intruso del sistema con acceso a la red y un analizador de paquetes puede interceptar cualquier contraseña enviada de este modo, comprometiendo las cuentas de usuarios y la integridad de toda la infraestructura de seguridad.

El primer objetivo de Kerberos es el de eliminar la transmisión a través de la red de información de autenticación. Un uso correcto de Kerberos erradica la amenaza de analizadores de paquetes que intercepten contraseñas en su red.

17.1.1. Desventajas de Kerberos

A pesar de que Kerberos elimina una amenaza de seguridad común, puede ser difícil de implementar por una variedad de razones:

- La migración de contraseñas de usuarios desde una base de datos de claves estándar UNIX, tal como `/etc/passwd` o `/etc/shadow`, a una base de datos de contraseña Kerberos puede ser tediosa y no hay un mecanismo rápido para realizar esta tarea. Para más información, refiérase a la pregunta número 2.23 en el la sección FAQ de Kerberos en: http://www.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html.
- Kerberos es sólo parcialmente compatible con los Pluggable Authentication Modules (PAM) usados por la mayoría de los servidores en Red Hat Linux. Para más información sobre éste tópico, vea Sección 17.4.
- Para que una aplicación use Kerberos, el código debe ser modificado para hacer las llamadas apropiadas a las librerías de Kerberos. Para algunas aplicaciones, esto puede suponer un esfuerzo excesivo de programación. Para otras aplicaciones incompatibles, los cambios se deben realizar en el protocolo usado entre el servidor de red y sus clientes; de nuevo, esto puede suponer una programación. Por defecto, las aplicaciones de código cerrado que no tienen soporte de Kerberos son usualmente las más problemáticas.
- Kerberos presupone que usted está utilizando hosts fiables en una red no fiable. Su primer objetivo es el de prevenir que las contraseñas en texto plano sean enviadas a través de la red. Sin embargo, si cualquier otro aparte del usuario adecuado tiene acceso físico a cualquiera de los hosts, especialmente el que emite tickets usados para la autenticación — llamado *Centro de distribución de contraseñas* (*KDC*) —, todo sistema de autenticación de Kerberos corre el riesgo de transigir.

- Finalmente, si decide usar Kerberos en su red, debe darse cuenta de que es una elección de todo o nada. Si decide usar Kerberos en su red, debe recordar que si se transmite cualquier contraseña a un servicio que no usa Kerberos para autenticar, se corre el riesgo de que el paquete pueda ser interceptado. Así, su red no obtendrá ningún beneficio de usar Kerberos. Para asegurar su red con Kerberos, debe *kerberizar* (hacer trabajar con Kerberos) *todas* las aplicaciones que mandan las contraseñas en texto plano o parar el uso de estas aplicaciones en la red.

17.2. Terminología Kerberos

Como algunos otros sistemas, Kerberos tiene su propia terminología para definir varios aspectos del servicio. Antes de aprender como funciona kerberos, es importante aprender estos términos.

texto cifrado

Datos encriptados.

cliente

Una entidad en la red (un usuario, un host o una aplicación) que pueden obtener un ticket de Kerberos.

caché credencial o archivo de tickets

Un archivo que contiene las claves para encriptar las comunicaciones entre el usuario y varios servicios de red. Kerberos 5 proporciona un framework para usar otros tipos de caché, tales como memoria compartida, pero los archivos están mejor soportados.

hash encriptado

Un hash de un sentido usado para autentificar usuarios. Aunque es más seguro que el texto plano, es bastante fácil de descifrar por un cracker con experiencia.

GSS-API

La Interfaz del programa de la aplicación de servicio de seguridad (Generic Security Service Application Program Interface [RFC-2743]) es un conjunto de funciones que proveen servicios de seguridad que los clientes pueden utilizar para autenticarse con los servidores y que los servidores pueden usar para autenticar clientes sin un conocimiento específico del mecanismo detrás de ello. Si un servicio de red (tal como IMAP) usa GSS-API, se puede autenticar usando Kerberos.

clave

Datos usados cuando encriptamos o desencriptamos otros datos. Los datos encriptados no pueden ser desencriptados sin la clave apropiada.

Centro de distribución de claves (KDC)

Un servicio que emite tickets Kerberos, que habitualmente se ejecutan en el mismo host como un Ticket Granting Server.

tabla de claves o keytab

Un fichero que incluye una lista desencriptada de "principals" y sus claves. Los servidores recuperan las claves que necesitan desde los archivos keytab en lugar de usar `kinit`. El archivo keytab por defecto es `/etc/krb5.keytab`. El servidor de administración KDC, `/usr/kerberos/sbin/kadmind`, es el único servicio que usa otro archivo (usa `/var/kerberos/krb5kdc/kadm5.keytab`).

`kinit`

El comando `kinit` permite a un principal quien que ya se ha conectado, obtener el primer Ticket Granting Ticket (TGT). Para más sobre el uso del comando `kinit` consulte la página del manual.

principal

El principal es el nombre único del usuario o servicio que puede autenticar mediante el uso de Kerberos. Un nombre de principal está en el formato `root[/instance]@REALM`. Para un usuario típico, el root es el mismo que su ID de login. La `instance` es opcional. Si el principal tiene una instancia, estará separada de root con una barra hacia adelante ("/"). Una cadena de caracteres vacía ("") es considerada como una instancia válida (que se diferencia del valor de la instancia por defecto `NULL`), pero usarlo puede ser confuso. Todos los principals de un reino tienen su propia clave, que se deriva de su contraseña (para usuarios) o aleatoriamente (para servicios).

reino

Red que usa Kerberos, compuesto de uno o varios servidores (también conocidos como KDCs) y un número potencial de clientes.

servicio

Un programa accesado a través de la red.

ticket

Grupo temporal de credenciales electrónicas que verifica la identidad de un cliente para un servicio particular.

Ticket Granting Service (TGS)

Un servidor que emite tickets para un servicio deseado; estos tickets son entregados a los usuarios para que accesen el servicio. El TGS usualmente se ejecuta en el mismo servidor que KDC

Ticket Granting Ticket (TGT)

Ticket especial que permite al cliente obtener tickets adicionales sin solicitarlos desde KDC.

contraseña no encriptada

Una contraseña en texto plano que se puede leer fácilmente.

17.3. Modo en que funciona Kerberos

Kerberos es diferente a los otros métodos de autenticación. En vez de validar cada usuario para cada servicio de red, Kerberos usa una aplicación de terceros que usa un sistema de encriptación simétrica — conocida como Centro de distribución de claves (KDC) — para autentificar los usuarios a un conjunto de servicios de red. Una vez que el usuario se ha autentificado al KDC, se le envía un ticket específico para esa sesión de vuelta a la máquina del usuario y cualquier servicio kerberizado buscará por el ticket en la máquina del usuario en vez de preguntarle al usuario que se autentifique usando una contraseña.

Cuando un usuario en una red kerberizada se registra en su estación de trabajo, su principal se envía al KDC en una petición para un Ticket Granting Ticket (TGT) desde el Ticket Granting Service (TGS). Esta petición puede ser enviada por el programa login (para que sea transparente al usuario) o puede ser enviada por el programa `kinit` después de que el usuario se registre.

El KDC verifica el principal en su base de datos. Si lo encuentra, el KDC le dice al TGS que cree un TGT, lo encripta usando las claves del usuario y l o devuelve al usuario.

El programa login en la máquina del cliente o `kinit` descifra el TGT usando la contraseña del usuario La contraseña del usuario es usada únicamente en la máquina del cliente y *no* es enviada sobre la red.

El TGT, que caduca después de un cierto período de tiempo (usualmente 10 horas), es almacenado en la caché de credenciales de la máquina del cliente. Se coloca un tiempo de caducidad de manera que un TGT comprometido sólo es de utilidad para un intruso por un período corto de tiempo. Una vez que el TGT es emitido, el usuario no podrá reingresar la contraseña al KDC hasta que el TGT caduque o se desconecte y vuelva a conectarse.

Cuando el usuario necesita acceder a un servicio de red, el cliente usa el TGT para pedir un ticket para ese servicio en específico al Ticket Granting Service (TGS). El TGS emite un ticket por el servicio deseado, que se usa para autentificar el usuario de forma transparente.

Aviso

El sistema Kerberos se vuelve vulnerable cada vez que un usuario en la red se valida contra un servicio no kerberizado y envía una contraseña en la red en texto plano. Por lo tanto no se recomienda el uso de servicios no kerberizados. Estos servicios incluyen Telnet y FTP. Se acepta el uso de otro tipo de protocolos encriptados, tales como SSH o servicios seguros SSL, pero no es ideal.

Esto es sólo una descripción general de cómo la autenticación Kerberos funciona en las redes, si necesita una explicación más detallada sobre el funcionamiento de kerberos, vea Sección 17.7.

Nota

Kerberos depende de ciertos servicios de la red para trabajar correctamente. Primero, Kerberos necesita una sicronización de reloj entre los ordenadores y su red. Si no ha configurado un programa de sincronización de reloj para su red, como por ejemplo `ntpd` debería hacerlo. Para más información sobre la configuración de `ntpd`, vea `/usr/share/doc/ntp-`<*version-number*>`/index.htm`.

Ya que ciertos aspectos de kerberos se apoyan en el Domain Name System (DNS), debe asegurarse de que las entradas DNS y los hosts en su red están configuradas correctamente. Vea el *Manual del administrador Kerberos V5* , proporcionado en PostScript y formatos HTML en `/usr/share/doc/krb5-server-`<*version-number*> para más información.

17.4. Kerberos y Pluggable Authentication Modules (PAM)

Actualmente, los servicios Kerberizados no hacen uso de PAM — un servidor kerberizado omite PAM completamente. Las aplicaciones que usan PAM pueden hacer uso de Kerberos para comprobar las contraseñas si el módulo `pam_krb5` (proporcionado en el paquete `pam_krb5`) está instalado. El paquete `pam_krb5` contiene un ejemplo de ficheros de configuración que permiten servicios como `login` y `gdm` autentificar usuarios y obtener credenciales iniciales usando sus contraseñas. Si el acceso a servicios de red siempre se realiza mediante servicios kerberizados (o servicios que usan GSS-API, como IMAP), la red puede ser considerada razonablemente segura.

Los administradores deberían tener cuidado de no permitir a los usuarios autentificarse a servicios de red usando claves Kerberos. La mayoría de los protocolos no encriptan las contraseñas antes de enviarlas sobre la red, destruyendo así los beneficios del sistema Kerberos. Por ejemplo, a los usuarios no se les debería permitir autenticarse usando contraseñas Kerberos sobre Telnet.

La próxima sección describe cómo configurar un servidor básico de Kerberos.

17.5. Configurar un servidor Kerberos 5

Antes de configurar un servidor Kerberos tiene que instalarlo. Si necesita instalar servidores esclavos, los detalles para configurar las relaciones entre servidores maestro y esclavo se cubren en *Manual de instalación de Kerberos 5* localizado en el directorio `/usr/share/doc/krb5-server-<version-number>`.

Para configurar un servidor Kerberos básico, siga estos pasos:

1. Asegúrese de que tanto el reloj como el DNS funcionan correctamente en el servidor antes de configurar el Kerberos 5. Preste especial atención a la sincronización de la hora del servidor Kerberos y de sus diversos clientes. Si la sincronización de los relojes del servidor y de los clientes se diferencia en más de cinco minutos (la cantidad predeterminada es configurable en el Kerberos 5), los clientes de Kerberos no podrán autentificar el servidor. La sincronización de los relojes es necesaria para evitar que un intruso use un ticket viejo de Kerberos para hacerse pasar como un usuario autorizado.

   Configure el protocolo cliente/servidor Network Time Protocol (NTP) aún si no está usando Kerberos. Red Hat Linux incluye el paquete `ntp` para una fácil instalación. Vea `/usr/share/doc/ntp-<version-number>/index.htm` para detalles sobre cómo configurar servidores Network Time Protocol y http://www.eecis.udel.edu/~ntp para información adicional sobre NTP.

2. Instale los paquetes `krb5-libs`, `krb5-server`, y `krb5-workstation` en una máquina dedicada que ejecutará el KDC. Esta máquina tiene que ser segura — si es posible, no debería ejecutar ningún otro servicio excepto KDC.

   Si desea usar una utilidad de interfaz gráfica para administrar Kerberos, debería instalar el paquete `gnome-kerberos`. Este contiene `krb5`, que es una herramienta tipo GUI para manejar tickets.

3. Modifique los ficheros de configuración `/etc/krb5.conf` y `/var/kerberos/krb5kdc/kdc.conf` para que reflejen el nombre de su reino y los mappings de dominio a reino. Se puede construir un reino simple sustituyendo las instancias de *`EXAMPLE.COM`* y *`example.com`* con el nombre del dominio — siempre y cuando se respete el formato correcto de los nombres escritos en mayúscula y en minúscula — y se cambie el KDC del *`kerberos.example.com`* con el nombre de su servidor Kerberos. En general, los nombres de reinos se escriben en mayúscula y todos los nombre DNS de host y nombres de dominio se escriben en minúscula. Para más detalles sobre los formatos de estos archivos, vea las páginas de los manuales respectivas.

4. Cree la base de datos usando la utilidad `kdb5_util` desde el intérprete de comandos del shell:
   ```
   /usr/kerberos/sbin/kdb5_util create -s
   ```

   El comando `create` crea una base de datos que será usada para almacenar las claves para su reino Kerberos. La opción `-s` fuerza la creación de un archivo *stash* en el cual la llave del servidor master es guardada. Si no se presenta un archivo stash desde donde leer la clave, el servidor Kerberos (`krb5kdc`) le pedirá al usuario que ingrese la contraseña del servidor master (la cual puede ser usada para regenerar la clave) cada vez que arranca.

5. Modifique el archivo `/var/kerberos/krb5kdc/kadm5.acl`. Este archivo es usado por `kadmind` para determinar cuales principales tienen acceso administrativo a la base de datos Kerberos y sus niveles de acceso. La mayoría de las organizaciones pueden acceder con una sola línea:
   ```
   */admin@EXAMPLE.COM  *
   ```

   La mayoría de los usuarios serán presentados en la base de datos por un principal simple (con una instancia *NULL*, o vacía, tal como *joe@EXAMPLE.COM*). Con esta configuración, los usuarios con un segundo principal con una instancia de *admin* (por ejemplo, *joe/admin@EXAMPLE.COM*) podrán tener todo el acceso sobre la base de datos del reino Kerberos.

Una vez que `kadmind` sea arrancado en el servidor, cualquier usuario tiene acceso a los servicios de este servidor ejecutando los comandos `kadmin` en cualquiera de los clientes o servidores del reino. Sin embargo, solamente los usuarios que aparecen en la lista del archivo `kadm5.acl` podrán modificar la base de datos salvo sus contraseñas.

Nota

La utilidad `kadmin` se comunican con el servidor `kadmind` por la red y usan Kerberos para llevar a cabo la autentificación. Por supuesto, tiene que ser usuario principal antes de conectarse al servidor con la red para poder administrarla. Puede crear esta primera entrada con el comando `kadmin.local`, el cual se ha creado específicamente para usarlo en la misma máquina que el KDC y no usa Kerberos para la autenticación.

Escriba el comando `kadmin.local` en una terminal KDC para crear la primera entrada como usuario principal:

```
/usr/kerberos/sbin/kadmin.local -q "addprinc
username/admin"
```

6. Arranque Kerberos usando los siguientes comandos:

```
/sbin/service krb5kdc start
/sbin/service kadmin start
/sbin/service krb524 start
```

7. Agregue principals para sus usuarios con el comando `addprinc` y `kadmin`. `kadmin` y `kadmin.local` son interfaces de línea de comandos para el KDC. Como tales, muchos comandos están disponibles después de lanzar el programa `kadmin`. Vea la página del manual `kadmin` para más información.

8. Verifique que el servidor crea tickets. Primero, ejecute `kinit` para obtener un ticket y guardarlo en un archivo de credenciales caché. Luego, use `klist` para ver la lista de credenciales en su caché y use `kdestroy` para eliminar el caché y los credenciales que contenga.

Nota

Por defecto, `kinit` intenta autenticar el usuario usando el nombre de conexión (login) de la cuenta que usó cuando se conectó al sistema la primera vez (no al servidor Kerberos). Si ese nombre de usuario de sistema no se corresponde a un principal en la base de datos Kerberos, recibirá un mensaje de error. Si estó ocurre, indique `kinit` con el nombre de su principal como un argumento en la línea de comandos (`kinit principal`).

Una vez que haya completado los pasos listados arriba, el servidor Kerberos funcionará correctamente. Luego, se configurará el cliente Kerberos.

17.6. Configuración de un cliente Kerberos 5

La configuración de un cliente de Kerberos 5 client no es tan complicada como la de un servidor. Lo que tiene que hacer es instalar los paquetes del cliente y proveer a cada cliente con un archivo de configuración `krb5.conf` válido. Las versiones kerberizadas de `rsh` y `rlogin` también requerirán algunos cambios en la configuración.

1. Asegúrese que la sincronización sea correcta entre el cliente de Kerberos y el KDC. Consulte Sección 17.5 para mayor información. Además, el DNS tiene que funcionar correctamente antes de instalar los programas del cliente Kerberos.

2. Instale los paquetes `krb5-libs` y `krb5-workstation` en todas las máquinas de clientes. Tiene que dar la versión del `/etc/krb5.conf` para cada cliente; normalmente es el mismo archivo `krb5.conf` usado por KDC.

3. Antes de que una estación de trabajo del reino permita a los usuarios conectarse usando los comandos kerberizados `rsh` y `rlogin`, esa estación de trabajo tendrá que tener instalado el paquete `xinetd` y tener su propio host principal en la base de datos Kerberos. Los programas del servidor `kshd` y `klogind` también necesitan el acceso a las claves de la entrada principal del servicio.

   Usando el comando `kadmin`, añada un host principal para la estación de trabajo en el KDC. La instancia en este caso será el nombre de host de la estación de trabajo. Puede usar la opción `-randkey` para el comando `kadmin addprinc` para crear el principal y asignarle una clave aleatoria:

   ```
   addprinc -randkey
   host/blah.example.com
   ```

   Ahora que ha creado el principal, puede extraer las claves para la estación de trabajo ejecutando `kadmin` *en la estación de trabajo*, y usando el comando `ktadd` en `kadmin`:

   ```
   ktadd -k /etc/krb5.keytab
   host/blah.example.com
   ```

4. Si desea utilizar otros servicios kerberizados de red, necesitará arrancarlos. Abajo hay una lista de algunos de los servicios kerberizados más comunes y las instrucciones para activarlos:

   - `rsh` y `rlogin` — Para poder usar las versiones kerberizadas de los comandos `rsh` y `rlogin`, deberá activar `klogin`, `eklogin`, y `kshell`.
   - Telnet — Para usar Telnet kerberizado, deberá activar `krb5-telnet`.
   - FTP — Para el acceso FTP, cree y extraiga una entrada para el principal con un root de `ftp`. Asegúrese de colocar la instancia al nombre de host del servidor FTP, luego active `gssftp`.
   - IMAP — El servidor IMAP incluye en el paquete `imap` que usará la autentificación GSS-API del Kerberos 5 si encuentra la clave adecuada en `/etc/krb5.keytab`. La raíz para el principal debería ser `imap`.
   - CVS — CVS kerberizado `gserver` usa un principal con una raíz de `cvs` y es idéntica al CVS `pserver`.

   Para detalles sobre como activar servicios, refiérase al capítulo titulado *Control de acceso a los servicios* en el *Manual de personalización de Red Hat Linux*.

17.7. Recursos adicionales

Para más información sobre Kerberos, refiérase a los siguientes recursos.

17.7.1. Documentación instalada

- `/usr/share/doc/krb5-server-<`*`version-number`*`>` — El *Manual de instalación de Kerberos V5* y el *Manual del administrador del sistema Kerberos V5* en formatos PostScript y HTML. Debe tener el paquete `krb5-server` instalado.
- `/usr/share/doc/krb5-workstation-<`*`version-number`*`>` — El *Manual del usuario de Kerberos V5 UNIX* en formatos PostScript y HTML. Debe tener el paquete `krb5-workstation` instalado.

17.7.2. Sitios web útiles

- http://web.mit.edu/kerberos/www — *Kerberos: El protocolo de autentificación de red* página web del MIT.
- http://www.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html — Preguntas más frecuentes sobre Kerberos (FAQ).
- ftp://athena-dist.mit.edu/pub/kerberos/doc/usenix.PS — La versión PostScript de *Kerberos: An Authentication Service for Open Network Systems* por Jennifer G. Steiner, Clifford Neuman, y Jeffrey I. Schiller. Este es el documento original que describe Kerberos.
- http://web.mit.edu/kerberos/www/dialogue.html — *Designing an Authentication System: a Dialogue in Four Scenes* inicialmente por Bill Bryant en 1988, modificado por Theodore Ts'o en 1997. Este documento es una conversación entre dos desarrolladores que están pensando en la creación de un sistema de autenticación Kerberos. El estilo desenfadado de esta conversación lo convierte en un buen material para aquellos que no tienen ningún tipo de familiaridad con Kerberos.
- http://www.ornl.gov/~jar/HowToKerb.html — *Cómo Kerberizar su sitio web* es una buena referencia para kerberizar una red.
- http://www.networkcomputing.com/netdesign/kerb1.html — *Kerberos Network Design Manual* es una vista general del sistema Kerberos.

Capítulo 18.

Protocolo SSH

SSH™ permite a los usuarios registrarse en sistemas de host remotamente. A diferencia de FTP o Telnet, SSH encripta la sesión de registro imposibilitando que alguien pueda obtener contraseñas no encriptadas.

SSH está diseñado para reemplazar los métodos más viejos y menos seguros para registrarse remotamente en otro sistema a través de la shell de comando, tales como **telnet** o **rsh**. Un programa relacionado, el **scp**, reemplaza otros programas diseñados para copiar archivos entre hosts como **rcp**. Ya que estas aplicaciones antiguas no encriptan contraseñas entre el cliente y el servidor, evite usarlas mientras le sea posible. El uso de métodos seguros para registrarse remotamente a otros sistemas hará disminuir los riesgos de seguridad tanto para el sistema cliente como para el sistema remoto.

18.1. Características de SSH

SSH (o *S*ecure *SH*ell) es un protocolo para crear conexiones seguras entre dos sistemas usando una arquitectura cliente/servidor.

El protocolo SSH proporciona los siguientes tipos de protección:

- Después de la conexión inicial, el cliente puede verificar que se está conectando al mismo servidor al que se conectó anteriormente.
- El cliente transmite su información de autenticación al servidor usando una encriptación robusta de 128 bits.
- Todos los datos enviados y recibidos durante la conexión se transfieren por medio de encriptación de 128 bits, lo cual los hacen extremamente difícil de descifrar y leer.
- El cliente tiene la posibilidad de enviar X11 [1] aplicaciones lanzadas desde el intérprete de comandos de la shell. Esta técnica proporciona una interfaz gráfica segura (llamada *reenvío por X11*), proporciona un medio seguro para usar aplicaciones gráficas sobre una red.

Ya que el protocolo SSH encripta todo lo que envía y recibe, se puede usar para asegurar protocolos inseguros. El servidor SSH puede convertirse en un conducto para convertir en seguros los protocolos inseguros mediante el uso de una técnica llamada *reenvío por puerto*, como por ejemplo POP, incrementando la seguridad del sistema en general y de los datos.

Red Hat Linux contiene el paquete general de OpenSSH (`openssh`), el servidor de OpenSSH (`openssh-server`) y los paquetes de clientes (`openssh-clients`). Consulte el capítulo titulado titled *OpenSSH* en el *Manual de personalización de Red Hat Linux* para obtener instrucciones sobre la instalación y el desarrollo de OpenSSH. Observe que los paquetes OpenSSH requieren el paquete OpenSSL (`openssl`). OpenSSL instala varias librerías criptográficas importantes que ayudan a OpenSSH a proporcionar comunicaciones encriptadas.

Una gran cantidad de programas de cliente y servidor puede usar el protocolo SSH. Muchas aplicaciones SSH cliente están disponibles para casi todos los principales sistemas operativos en uso hoy día.

---

1. X11 se refiere al sistema de visión por ventanas X11R6, tradicionalmente llamado X. Red Hat Linux contiene **XFree86**, un sistema X Window open-source muy conocido, que se basa en X11R6.

18.1.1. ¿Por qué usar SSH?

Los usuario nefarios tienen a su disposición una variedad de herramientas interceptar y dirigir el tráfico de la red para ganar acceso al sistema. En términos generales, estas amenazas se pueden catalogar del siguiente modo:

- *Intercepción de la comunicación entre dos sistemas* — En este escenario, existe un tercero en algún lugar de la red entre entidades en comunicación que hace una copia de la información que pasa entre ellas. La parte interceptora puede interceptar y conservar la información, o puede modificar la información y luego enviarla al recipiente al cual estaba destinada.

  Este ataque se puede montar a través del uso de un paquete sniffer — una utilidad de red muy común.

- *Personificación de un determinado host* — con esta estrategia, un sistema interceptor finge ser el recipiente a quien está destinado un mensaje. Si funciona la estrategia, el cliente no se da cuenta del engaño y continúa la comunicación con el interceptor como si su mensaje hubiese llegado a su destino satisfactoriamente.

  Esto se produce con técnicas como el envenenamiento del DNS [2] o spoofing de IP [3].

Ambas técnicas causan que se intercepte información, posiblemente con propósitos hostiles. El resultado puede ser catastrófico.

Si se utiliza SSH para inicios de sesión de shell remota y para copiar archivos, estas amenazas a la seguridad se pueden disminuir notablemente. Esto es porque el cliente SSH y el servidor usan firmas digitales para verificar su identidad. Adicionalmente, toda la comunicación entre los sistemas cliente y servidor es encriptada. No servirán de nada los intentos de falsificar la identidad de cualquiera de los dos lados de la comunicación ya que cada paquete está cifrado por medio de una clave conocida sólo por el sistema local y el remoto.

18.2. Versiones del protocolo SSH

El protocolo SSH permite a cualquier programa cliente y servidor construído a las especificaciones del protocolo, comunicarse de forma segura y ser usado de intercambiable.

Existen dos variedades de SSH actualmente. La versión 1 de SSH hace uso de muchos algoritmos de encriptación patentados (sin embargo, algunas de estas patentes han expirado) y es vulnerable a un hueco de seguridad que potencialmente permite a un intruso insertar datos en la corriente de comunicación. La suite OpenSSH bajo Red Hat Linux utiliza la versión 2 de SSH por defecto, aún cuando también soporta la versión 1.

Importante

Se recomienda que sólo se utilicen servidores y clientes compatibles con la versión 2 de SSH siempre que sea posible.

2. El envenenamiento del DNS ocurre cuando un intruso entra en el servidor de DNS, apuntando sistemas hacia hosts intencionalmente duplicados.
3. IP spoofing ocurre cuando un intruso manda paquetes de red que parecen provenir de hosts de confianza de la red.

18.3. Secuencia de eventos de una conexión SSH

La siguiente serie de eventos lo ayudan a proteger la integridad de la comunicación SSH entre dos host.

- Se lleva a cabo un 'handshake' (apretón de manos) encriptado para que el cliente pueda verificar que se está comunicando con el servidor correcto.
- La capa de transporte entre el cliente y la máquina remota es encriptada mediante un código simétrico.
- El cliente se autentica ante el servidor.
- El cliente remoto puede ahora con tranquilidad interactuar con la máquina remota sobre la conexión encriptada.

18.3.1. Capa de transporte

El papel principal de la capa de transporte es facilitar una comunicación segura entre los dos hosts en el momento y después de la autenticación. La capa de transporte lleva esto a cabo manejando la encriptación y decodificación de datos y proporcionando protección de integridad de los paquetes de datos mientras son enviados y recibidos. Además, la capa de transporte proporciona compresión de datos, lo que acelera la transmisión de información.

Al contactar un cliente a un servidor por medio del protocolo SSH, se negocian varios puntos importantes para que ambos sistemas puedan construir la capa de transporte correctamente. Durante el intercambio se producen los siguientes pasos:

- Intercambio de claves
- Se determina el algoritmo de encriptación de la clave pública
- Se determina el algoritmo de la encriptación simétrica
- Se determina el algoritmo autenticación de mensajes
- Se determina el algoritmo de hash que hay que usar

El servidor se identifica ante el cliente con una *clave de host* única durante el intercambio de claves. Obviamente si este cliente nunca se había comunicado antes con este determinado servidor, la clave del servidor le resultará desconocida al cliente y no lo conectará. OpenSSH evita este problema permitiendo que el cliente acepte la clave de host del servidor después que el usuario es notificado y verifica la aceptación de la nueva clave del host Para las conexiones posteriores, la clave de host del servidor se puede verificar con la versión guardada en el cliente, proporcionando la confianza que el cliente está realmente comunicando con el servidor deseado. Si en el futuro, la clave del host ya no coincide, el usuario eliminará la versión guardada antes de que una conexión ocurra.

Atención

Un agresor podría enmascararse como servidor SSH durante el contacto inicial ya que el sistema local no conoce la diferencia entre el servidor en cuestión y el falso configurado por un agresor. Para evitar que esto ocurra debería verificar la integridad del nuevo servidor SSH contactando con el adiministrador del servidor antes de conectarse por primera vez o en el evento de que no coincidan las claves.

SSH fue ideado para funcionar con casi cualquier tipo de algoritmo de clave pública o formato de codificación. Después del intercambio de claves inicial se crea un valor hash usado para el intercambio y un valor compartido secreto, los dos sistemas empiezan inmediatamente a calcular

claves y algoritmos nuevos para proteger la autenticación y los datos que se enviarán a través de la conexión en el futuro.

Después que una cierta cantidad de datos haya sido transmitida con un determinado algoritmo y clave (la cantidad exacta depende de la ejecución de SSH), ocurre otro intercambio de claves, el cual genera otro conjunto de valores de hash y un nuevo valor secreto compartido. De esta manera aunque un agresor lograse determinar los valores de hash y de secreto compartido, esta información sólo será válida por un período de tiempo limitado.

18.3.2. Autenticación

Cuando la capa de transporte haya construido un túnel seguro para transmitir información entre los dos sistemas, el servidor le dirá al cliente de los diferentes métodos de autenticación soportados, tales como el uso de firmas privadas codificadas con claves o la inserción de una contraseña. El cliente entonces intentará autenticarse ante el servidor mediante el uso de cualquiera de los métodos soportados.

Ya que los servidores y clientes SSH se pueden configurar para que concedan varios tipos de autenticación, lo cual le concede a cada lado la cantidad óptima de control. Luego el servidor podrá decidir qué métodos de encriptación soportará basado en su pauta de seguridad, y el cliente puede elegir el orden en que intentará utilizar los métodos de autenticación entre las opciones a disposición. Gracias a la naturaleza segura de la capa de transporte de SSH, hasta métodos de autenticación que parecen inseguros, como la autenticación basada en el host, son en realidad seguros para usar.

18.3.3. Canales

Luego de una autenticación exitosa sobre la capa de transporte SSH, múltiples *canales* son abiertos a través de la técnica llamada multiplexar[4]. Cada uno de estos canales manejan la conexión para diferentes sesiones de terminal y para sesiones X11.

Ambos clientes y servidores pueden crear un canal nuevo. Cada canal es luego asignado un número diferente en cada punta de la conexión. Cuando el cliente intenta abrir un nuevo canal, los clientes envían el número del canal junto con la petición. Esta inforamción es almacenada por el servidor y usada para dirigir la comunicación a ese canal. Esto es hecho para que diferentes tipos de sesión no afectarán una a la otra y así cuando una sesión termine, su canal pueda ser cerrado sin interrumpir la conexión SSH primaria.

Los canales también soportan el *control de flujo*, el cual les permite enviar y recibir datos ordenadamente. De esta manera, los datos no se envían a través del canal sino hasta que el host haya recibido un mensaje avisando que el canal está abierto y puede recibirlos.

El cliente y el servidor negocian las características de cada canal automáticamente, dependiendo del tipo de servicio que el cliente solicita y la forma en que el usuario está conectado a la red. Esto otorga una gran flexibilidad en el manejo de diferentes tipos de conexiones remotas sin tener que cambiar la infraestructura básica del protocolo.

18.4. Archivos de configuración de OpenSSH

OpenSSH tiene dos conjuntos diferentes de archivos de configuración: uno para los programas cliente (`ssh`, `scp`, y `sftp`) y otro para el demonio del servidor (`sshd`).

4. Una conexión multiplexada consiste de muchas señales siendo enviadas sobre un medio común, compartido. Con SSH, canales diferentes son enviados sobre una conexión común segura.

La información de configuración SSH para todo el sistema está almacenada en el directorio `/etc/ssh/`:

- `moduli` — Contiene grupos Diffie-Hellman usados para el intercambio de la clave Diffie-Hellman que es imprescindible para la construcción de una capa de transporte seguro. Cuando se intercambian las claves al inicio de una sesión SSH, se crea un valor secreto y compartido que no puede ser determinado por ninguna de las partes individualmente. Este valor se usa para proporcionar la autentificación del host.
- `ssh_config` — El archivo de configuración del sistema cliente SSH por defecto que se sobreescribe si hay alguno ya presente en el directorio principal del usuario (`~/.ssh/config`).
- `sshd_config` — El archivo de configuración para el demonio `sshd`.
- `ssh_host_dsa_key` — La clave privada DSA usada por el demonio `sshd`.
- `ssh_host_dsa_key.pub` — La clave pública DSA usada por el demonio `sshd`.
- `ssh_host_key` — La clave privada RSA usada por el demonio `sshd` para la versión 1 del protocolo SSH.
- `ssh_host_key.pub` — La clave pública RSA usada por el demonio `sshd` para la versión 1 del protocolo SSH.
- `ssh_host_rsa_key` — La clave privada RSA usada por el demonio `sshd` para la versión 2 del protocolo SSH.
- `ssh_host_rsa_key.pub` — La clave pública RSA usada por el demonio `sshd` para la versión 2 del protocolo SSH.

La información para la configuración SSH específica para el usuario está almacenada en el directorio principal `~/.ssh/`:

- `authorized_keys` — Este archivo que contiene una lista de claves públicas "autorizadas". Cuando un cliente se conecta al servidor, el servidor valida al cliente chequeando su clave pública firmada almacenada dentro de este archivo.
- `id_dsa` — Contiene la clave privada DSA del usuario.
- `id_dsa.pub` — Contiene la clave pública DSA del usuario.
- `id_rsa` — La clave RSA privada usada por `ssh` para la versión 2 del protocolo SSH.
- `id_rsa.pub` — La clave pública RSA usada por `ssh` para la versión 2 del protocolo SSH.
- `identity` — La clave privada RSA usada por `ssh` para la versión 1 del protocolo SSH.
- `identity.pub` — La clave pública RSA usada por `ssh` para la versión 1 del protocolo SSH.
- `known_hosts` — Este archivo contiene las claves de host DSA de los servidores SSH accesados por el usuario. Este archivo es muy importante para asegurarse de que el cliente SHH está conectado al servidor SSH correcto.

Importante

Si se ha cambiado una clave de host del servidor SSH, el cliente notificará al usuario que la conexión no puede proceder hasta que la clave del host del servidor sea borrada desde el archivo `known_hosts` usando un editor de texto. Antes de hacer esto, sin embargo, contacte al administrador del sistema del servidor SSH para verificar que no se ha comprometido al servidor.

Consulte las páginas de manual para `ssh` y `sshd` para obtener información acerca de las directivas disponibles en los archivos de configuración SSH.

18.5. Más que un Shell seguro

Una interfaz de línea de comandos segura es sólo el inicio de las muchas maneras de usar SSH. Dada una cantidad apropiada de ancho de banda, las sesiones X11 se pueden dirigir por un canal SSH. O usando reenvío TCP/IP, se pueden asignar conexiones de puerto entre sistemas que previamente eran inseguras a canales SSH específicos.

18.5.1. Reenvío por X11

Abrir una sesión X11 a través de una conexión SSH establecida es tan fácil como ejecutar un programa X en una máquina local. Cuando un programa X se ejecuta desde un intérprete de comandos de shell segura, el cliente y el servidor SSH crean un nuevo canal seguro dentro de la conexión SSH actual, y los datos del programa X se envían a través de ese canal a la máquina cliente de forma transparente.

Como podrá imaginar, el reenvío por X11 puede ser muy útil. Por ejemplo, se puede usar el reenvío por X11 para crear una sesión segura e interactiva con `up2date`. Para hacer esto, conéctese al servidor usando **ssh** y escriba:

```
up2date &
```

Se le pedirá proporcionar la contraseña de root para el servidor. Luego aparecerá el **Agente de actualización de Red Hat** y permitirá al usuario actualizar con seguridad el sistema remoto.

18.5.2. Reenvío del puerto

Con SSH puede asegurar los protocolos TCP/IP a través del reenvío de puertos. Cuando use esta técnica, el servidor SSH se convierte en un conducto encriptado para el cliente SSH.

El reenvío de puertos funciona mediante el mapeado de un puerto local en el cliente a un puerto remoto del servidor. SSH le permite mapear cualquier puerto desde el servidor a cualquier puerto en el cliente; y los números de puerto no necesitan coincidir para poder funcionar.

Para crear un canal de reenvío de puerto TCP/IP que escucha conexiones del host local, utilice el siguiente comando:

```
ssh -L local-port:remote-hostname:remote-port username@hostname
```

Nota

La configuración del reenvío de un puerto para que escuche puertos bajo 1024 requiere acceso de root.

Si desea comprobar su correo en el servidor llamado mail.example.com usando POP a través de una conexión encriptada, use el comando siguiente:

```
ssh -L 1100:mail.example.com:110 mail.example.com
```

Una vez que el canal de reenvío de puerto está entre la máquina cliente y el servidor de correo, puede direccionar su cliente de correo POP para usar el puerto 1100 en su host local para comprobar el nuevo correo. Cualquier petición enviada al puerto 1100 en el sistema cliente será dirigida seguramente al servidor mail.example.com.

Si mail.example.com no está ejecutando un servidor SSH, pero otra máquina en la misma red si, SSH todavía puede ser usado para asegurar parte de la conexión. Sin embargo, un comando ligeramente diferente es necesario:

```
ssh -L 1100:mail.example.com:110 other.example.com
```

En este ejemplo, se está reenviando su petición POP desde el puerto 1100 en la máquina cliente a través de una conexión SSH en el puerto 22 al servidor SSH, `other.example.com`. Luego, `other.example.com` se conecta al puerto 110 en `mail.example.com` para verificar nuevo correo. Observe que usando esta técnica, sólo la conexión entre el sistema cliente y el servidor SSH `other.example.com` es segura.

El reenvío del puerto se puede usar para obtener información segura a través de los firewalls de red. Si el firewall está configurado para permitir el tráfico SSH a través del puerto estándar (22) pero bloquea el acceso a través de otros puertos, es posible todavía una conexión entre dos hosts usando los puertos bloqueados al redireccionar la comunicación sobre una conexión SSH establecida.

Nota

Mediante el uso del reenvío de puerto para reenviar conexiones de este modo permiten a cualquier usuario en el sistema cliente conectarse a ese servicio. Si el cliente está en riesgo o está comprometido, un agresor puede también accesar los servicios reenviados.

Los administradores del sistema preocupados por el reenvío del puerto pueden deshabilitar esta funcionalidad en el servidor especificando un parámetro `No` para la línea AllowTcpForwarding en `/etc/ssh/sshd_config` y reiniciando el servicio `sshd`.

18.6. Requerir SSH para conexiones remotas

Para que SSH sea realmente eficaz, el uso de todos los protocolos de conexión inseguros como por ejemplo FTP y Telnet, deberían ser prohibidos. De lo contrario una contraseña de usuario puede estar protegida usando SSH para una sesión, para luego ser capturada cuando establece una conexión Telnet.

Algunos servicios a deshabilitar incluyen:

- `telnet`
- `rsh`
- `ftp`
- `rlogin`
- `vsftpd`

Para desactivar métodos de conexión inseguros al sistema, use el programa de línea de comandos `chkconfig`, el programa basado en ncurses **ntsysv**, o la aplicación gráfica **Herramienta de configuración de servicios** (`redhat-config-services`). Todas estas herramientas requieren acceso root.

Para más información sobre los niveles de ejecución y configuración de servicios con `chkconfig`, **ntsysv**, y **Herramienta de configuración de servicios**, consulte el capítulo llamado *Controlar el acceso a servicios* en el *Manual de personalización de Red Hat Linux*.

Capítulo 19.

Tripwire

El software de aseguramiento de integridad de los datos Tripwire, monitorea la consistencia de archivos y directorios de sistema críticos identificando todos los cambios hechos a ellos. Esto lo hace mediante un método automatizado de verificación que se ejecuta a intervalos regulares. Si Tripwire detecta que uno de los archivos monitoreados ha sido cambiado, lo notifica al administrador del sistema vía email. Debido a que Tripwire puede fácilmente identificar los archivos que son modificados, agregados o eliminados, se agiliza el proceso de recuperación luego de una entrada forzada pues mantiene el número de archivos que deben ser restaurados a un mínimo. Estas habilidades hacen de Tripwire una herramienta excelente para los administradores de sistemas que requieren tanto de facilidades para detección de intrusos como de control de daños para sus servidores.

Tripwire compara los archivos y directorios con una base de datos de la ubicación de archivos, las fechas en que han sido modificados y otros datos. Tripwire genera la base tomando una instantánea. Esta base de datos contiene *fundamentos* — los cuales son instantáneas de archivos y directorios específicos en momentos particulares. Los contenidos de la base de datos de fundamentos deberían ser generados antes de que el sistema esté en riesgo, esto es antes de que se conecte a la red. Después de crear la base de datos de fundamentos, Tripwire compara la base de datos actual con la base de datos de fundamentos e informa de cualquier modificación, adición o eliminación.

Aunque es una herramienta muy valiosa para revisar el estado de seguridad de su sistema Red Hat Linux, Tripwire no está soportada por Red Hat, Inc.. Si necesita más información sobre Tripwire, un buen lugar para comenzar es el sitio web del proyecto localizado en http://www.tripwire.org.

19.1. Cómo utilizar Tripwire

El siguiente diagrama de flujo ilustra cómo funciona Tripwire:

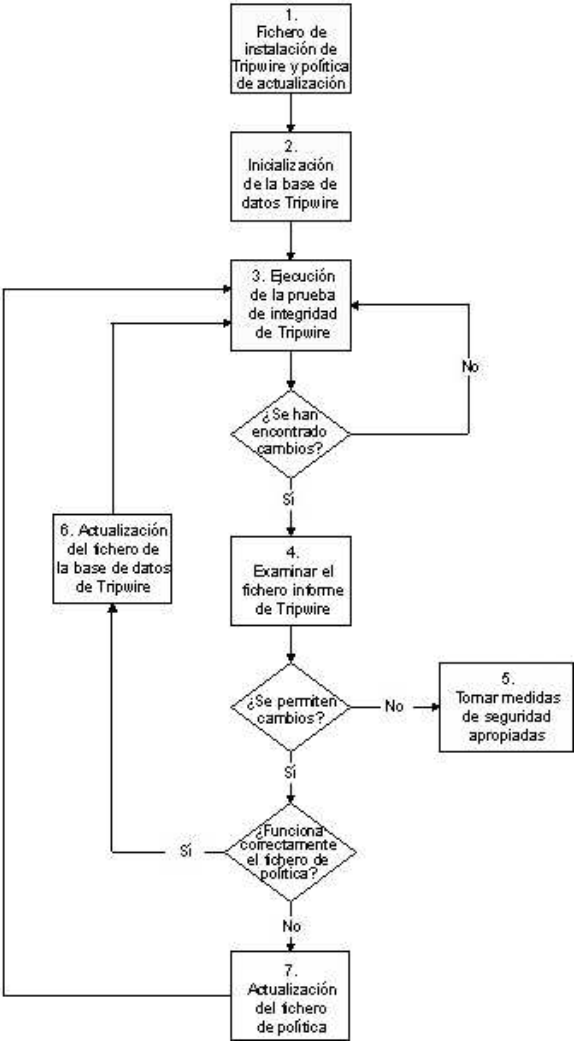

Figura 19-1. Usar Tripwire

A continuación se describe en más detalles los bloques enumerados en Figura 19-1.

1. Instale Tripwire y personalice el archivo de políticas.

Instale el paquete RPM Tripwire (vea Sección 19.2). Luego, personalice los archivos de configuración de ejemplo y de políticas (`/etc/tripwire/twcfg.txt` y `/etc/tripwire/twpol.txt` respectivamente), y ejecute el script de configuración, `/etc/tripwire/twinstall.sh`. Para más información, vea Sección 19.3.

2. Inicialice la base de datos Tripwire.

Construya una base de datos de archivos de sistema críticos para monitorear basado en los contenidos de la nueva política de configuración Tripwire, `/etc/tripwire/tw.pol`. Para más información, consulte Sección 19.4.

3. Ejecute una verificación de integridad Tripwire.

Compare la nueva base de datos Tripwire con los archivos de sistema actuales, buscando por archivos faltantes o alterados. Para más información, vea Sección 19.5.

4. Examine el archivo de informe Tripwire.

Vea el informe de Tripwire usando `/usr/sbin/twprint` para observar las violaciones de archivos. Para más información, vea Sección 19.6.1.

5. Si ocurren violaciones de integridad, tome las medidas de seguridad apropiadas.

Si archivos monitoreados han sido alterados de forma inadecuada, puede bien sea reemplazar los archivos originales a partir de las copias de respaldo, reinstalar el programa o reinstalar completamente el sistema operativo.

6. Si las alteraciones de archivos son válidas, verifique y actualice el archivo de la base de datos Tripwire.

Si los cambios hechos a los archivos monitoreados son intencionales, modifique el archivo de la base de datos Tripwire para ignorar aquellos cambios en los informes siguientes. Para más información, vea Sección 19.7.

7. Si el archivo de políticas falla en la verificación, actualícelo.

Para cambiar la lista de archivos que Tripwire monitorea o como maneja las violaciones de integridad, actualice el archivo de políticas provisto (`/etc/tripwire/twpol.txt`), regenere una copia firmada (`/etc/tripwire/tw.pol`), y actualice la base de datos Tripwire. Para más información, consulte Sección 19.8.

Refiérase a las secciones apropiadas dentro de este capítulo para más instrucciones detalladas en cada paso.

19.2. Instalando los RPM de Tripwire

La forma más fácil de instalar Tripwire es instalando el RPM de Tripwire durante el proceso de instalación de Red Hat Linux. Sin embargo, si ya había instalado, Red Hat Linux, puede usar el comando `rpm` o la **Herramienta de administración de paquetes** (`redhat-config-packages`) para instalar los RPM de Tripwire desde los CD-ROM de Red Hat Linux 9.

Si no está seguro de si ha instalado Tripwire, escriba el siguiente comando en el indicador de comandos de la shell:

```
rpm -q tripwire
```

Si Tripwire está instalado, este comando devolverá lo siguiente:

```
tripwire-<version-number>
```

En la salida de arriba, *<version-number>* es el número de versión del paquete.

Si Tripwire no está instalado, la línea de comandos del shell volverá a aparecer.

Las siguientes pautas le dicen cómo encontrar e instalar el Tripwire desde los CD-ROMs usando la línea de comandos de RPM:

1. Inserte el *CD 2* de los CD-ROMs de instalación de Red Hat Linux 9.
2. Si el CD-ROM no se monta automáticamente, escriba lo siguiente:
   ```
   mount /mnt/cdrom
   ```
3. Verifique que el RPM de Tripwire esté en el CD-ROM escribiendo:
   ```
   ls /mnt/cdrom/RedHat/RPMS/ | grep tripwire
   ```
   Si el paquete de RPM está en el CD-ROM, este comando mostrará el nombre del paquete.

   Si el RPM *no* está en el CD-ROM, la línea de comandos del shell se mostrará. En este caso, necesitará chequear los otros CD-ROMs de instalación de Red Hat Linux 9, desmontando primero el CD-ROM y luego repitiendo los pasos uno al tres.

   Desmonte el CD-ROM presionando el botón derecho sobre el icono del CD-ROM y seleccione **Expulsar** o escribiendo el comando siguiente en la línea de comandos del shell:
   ```
   umount /mnt/cdrom
   ```
4. Una vez que haya localizado el RPM de Tripwire, instálelo escribiendo el comando siguiente como usuario root:
   ```
   rpm -Uvh /mnt/cdrom/RedHat/RPMS/tripwire*.rpm
   ```

Encontrará archivos de notas de última hora y LEAME para Tripwire en el directorio `/usr/share/doc/tripwire-<version-number>/` (donde *<version-number>* es el número de versión del software). Estos documentos contienen información importante sobre el archivo de políticas por defecto y otros tópicos.

19.3. Personalizar Tripwire

Después de haber instalado el RPM Tripwire, tiene que completar los siguientes pasos para inicializar el software:

19.3.1. Modificar `/etc/tripwire/twcfg.txt`

Aún cuando no es requerido que edite este archivo de configuración de ejemplo de Tripwire, le recomendamos que lo haga. De hecho, quizás desee alterar la localización de los archivos Tripwire, personalizar los parámetros del e-mail o el nivel de detalles para los informes.

Abajo aparece una lista con las variables configurables de usuario *requeridas* en el archivo `/etc/tripwire/twcfg.txt`:

- `POLFILE` — Especifica la ubicación del archivo de políticas; `/etc/tripwire/tw.pol` es el valor por defecto.
- `DBFILE` — Especifica la ubicación del archivo de la base de datos; `/var/lib/tripwire/$(HOSTNAME).twd` es el valor por defecto.
- `REPORTFILE` — Especifica la ubicación de los archivos de informes. Por defecto este valor está colocado a `/var/lib/tripwire/report/$(HOSTNAME)-$(DATE).twr`.
- `SITEKEYFILE` — Especifica la ubicación del archivo de la llave del sitio; `/etc/tripwire/site.key` es el valor por defecto.
- `LOCALKEYFILE` — Especifica la ubicación del archivo de la llave local; `/etc/tripwire/$(HOSTNAME)-local.key` es el valor por defecto.

Importante

Si modifica el archivo de configuración y no define las variables anteriores, el archivo de configuración no será válido. Si esto ocurre, cuando ejecute el comando `tripwire` indicará un error y saldrá de la pantalla.

El resto de las variables configurables en el archivo de ejemplo `/etc/tripwire/twcfg.txt` son opcionales. Estas incluyen lo siguiente:

- `EDITOR` — Especifica el editor de texto llamado por Tripwire. El valor predetermindo es `/bin/vi`.
- `LATEPROMPTING` — Si se coloca a `verdadero`, esta variable configura Tripwire para que espere tanto como sea posible antes de preguntar al usuario por una contraseña, minimizando así el tiempo que la contraseña permanece en memoria. El valor por defecto es `falso`.
- `LOOSEDIRECTORYCHECKING` — Si es `verdadero`, esta variable configura Tripwire para que informe sobre los cambios que se han realizado en un archivo de un directorio y no sobre los cambios propios del directorio. Esto limita la redundancia en los informes de Tripwire. El valor predeterminado es `falso`.
- `SYSLOGREPORTING` — Si es `verdadero`, esta variable configura Tripwire para informe al demonio syslog con la facilidad del "usuario". El nivel de informe está colocado a `aviso`. Vea la página del manula de `syslogd` para más información. El valor predeterminado es `falso`.
- `MAILNOVIOLATIONS` — Si es `verdadero`, esta variable configura Tripwire para que mande un informe en forma de e-mail a intervalos regulares sin tener en cuenta si se han producido violaciones. El valor predeterminado es `verdadero`.
- `EMAILREPORTLEVEL` — Especifica el nivel de detalles para los informes enviados a través de email. Los valores válidos para esta variable son `0` a `4`. El valor por defecto es `3`.
- `REPORTLEVEL` — Especifica el nivel de detalles para los informes generados por el comando `twprint`. Este valor se puede cambiar en la línea de comandos, pero el valor predeterminado es `3`.
- `MAILMETHOD` — Especifica qué protocolo de correo debe usar Tripwire. Los valores válidos son `SMTP` and `SENDMAIL`. El valor predeterminado es `SENDMAIL`.
- `MAILPROGRAM` — Especifica cúal programa de correo usará Tripwire. El valor por defecto es `/usr/sbin/sendmail -oi -t`.

Después de modificar el archivo de configuración de muestra, tiene que configurar el archivo de políticas de muestra.

Aviso

Por razones de seguridad, debe de borrar o meter en un lugar seguro las copias del archivo de texto `/etc/tripwire/twcfg.txt` después de ejecutar el script de instalación o regenerar un archivo de configuración firmado. También, puede cambiar los permisos de manera que no se pueda leer.

19.3.2. Modificar `/etc/tripwire/twpol.txt`

Aún cuando no es requerido, debe de modificar este archivo de política para darse cuenta de las aplicaciones específicas, de los archivos y de los directorios de su sistema. Si confía en una configuración de muestra RPM el sistema no estará protegido.

La modificación del archivo de política aumenta la utilidad de los informes de Tripwire minimizando los avisos falsos para los archivos y programas que no está usando y añade funcionalidad como por ejemplo las notificaciones en forma de e-mail.

Nota

La notificación vía email no está configurada por defecto. Vea Sección 19.8.1 para más información sobre la configuración de esta característica.

Si modifica el archivo de política después de ejecutar el script de configuración, vea Sección 19.8 para instrucciones en la generación de un archivo de políticas firmado.

Aviso

Por razones de seguridad, debe de borrar o meter en un lugar seguro las copias del archivo de texto `/etc/tripwire/twpol.txt` después de ejecutar el script de instalación o regenerar un archivo de configuración firmado. También puede cambiar los permisos para que no se pueda leer.

19.3.3. Ejecutar el script `twinstall.sh`

Como usuario root, escriba `/etc/tripwire/twinstall.sh` en la línea de comandos del shell para ejecutar el script de configuración. El script `twinstall.sh` le pedirá sus contraseñas del sitio y local. Estas contraseñas son usadas para generar llaves criptográficas para la protección de sus archivos Tripwire. El script luego crea y firma estos archivos.

Cuando esté seleccionando las contraseñas local y de sitio, debería considerar las siguientes pautas:

- Use al menos ocho caracteres alfanuméricos para cada clave única, pero no más de 1023 caracteres en total.
- No use comillas en la contraseña.
- Las contraseñas Tripwire deben de ser diferentes de las de root o de cualquier otra contraseña del sistema.
- Use contraseñas únicas tanto para la clave del sitio como para la local.

La contraseña de la clave del sitio protege la configuración de Tripwire y los archivos de política. La contraseña de la clave local protege la base de datos de Tripwire y los archivos de informes.

Aviso

No hay forma de descifrar un archivo firmado si olvida su contraseña. Si olvida sus contraseñas, los archivos serán inutilizables y tendrá que ejecutar el script otra vez.

Encriptando sus archivos de configuración, de política, la base de datos y los archivos de informe, Tripwire los protege de los intrusos que no conocen las contraseñas locales ni de los sitios. Esto siginifica que aunque un intruso obtenga al acceso de root al sistema, no podrá alterar los archivos Tripwire para enmascararse.

Una vez encriptados y firmados, no se puede cambiar el nombre ni mover los archivos de configuración y de políticas generados con el script `twinstall.sh`.

19.4. Inicialización de la base de datos Tripwire

Cuando Tripwire inicializa su base de datos, crea una colección de objetos de sistemas de archivos basados en las reglas en el archivo de políticas. Esta base de datos sirve como la base para los controles de integridad.

Para inicializar la base de datos Tripwire, use el comando siguiente:

```
/usr/sbin/tripwire --init
```

Este comando puede tomar varios minutos para ejecutarse.

Una vez que ha seguido correctamente estos pasos, Tripwire tiene las instantáneas del sistema de archivos necesaria para verificar los cambios que se realicen en los archivos importantes. Después de inicializar la base de datos de Tripwire, tiene que ejecutar una primera verificación de la integridad. La tiene que realizar antes de conectar el ordenador a la red. Para mayor información, vea Sección 19.5.

Una vez que Tripwire esté configurado a su gusto, ya puede poner el sistema en marcha.

19.5. Ejecución de un control de integridad

Por defecto, el RPM Tripwire añade un script de la shell llamado `tripwire-check` al directorio `/etc/cron.daily/`. Este script ejecuta automaticamente un control de integridad una vez al día.

Sin embargo, usted puede ejecutar un control de integridad de Tripwire en cualquier momento mediante el comando:

```
/usr/sbin/tripwire --check
```

Cuando ejecuta un control de integridad, Tripwire compara los objetos de sistema de archivos actuales y reales con sus propiedades como han sido indicadas en su base de datos. Las violaciones se imprimen a una salida estándar y se guardan en un archivo de informe en `/var/lib/tripwire/report/`. Puede ver este informe más tarde ejecutando el comando `twprint` como se describe en Sección 19.6.1.

Si le gustaría recibir un email cuando ocurren ciertos tipos de violaciones de integridad, puede configurar esto en el archivo de políticas. Vea Sección 19.8.1 para instrucciones sobre cómo configurar y evaluar esta característica.

19.6. Verificación de los informes Tripwire

El comando `/usr/sbin/twprint` es usado para ver los informes encriptados y las bases de datos Tripwire.

19.6.1. Visualización de informes Tripwire

El comando `twprint -m r` mostrará los contenidos de un informe Tripwire en texto plano. Sin embargo, usted deberá indicarle a `twprint` cual archivo de informe mostrar.

Un comando `twprint` para imprimir informes Tripwire se ve similar a lo siguiente:

```
/usr/sbin/twprint -m r --twrfile /var/lib/tripwire/report/<name>.twr
```

La opción `-m r` en el comando direcciona `twprint` a que descifre el informe Tripwire. La opción `--twrfile` le ordena a `twprint` a que use un informe Tripwire específico.

El nombre del informe Tripwire que desea ver incluye el nombre del host que Tripwire verificó para generar el informe, además de la fecha y hora de creación. Puede revisar cuidadosamente los informes guardados previamente en cualquier momento. Simplemente escriba `ls /var/lib/tripwire/report` para ver una lista de los informes Tripwire.

Los informes Tripwire pueden ser un poco largos, según la cantidad de violaciones encontradas o los errores generados. Un informe de muestra empieza así:

```
Tripwire(R) 2.3.0 Integrity Check Report

Report generated by:          root
Report created on:            Fri Jan 12 04:04:42 2001
Database last updated on:     Tue Jan  9 16:19:34 2001

===============================================================================
Report Summary:
===============================================================================
Host name:                    some.host.com
Host IP address:              10.0.0.1
Host ID:                      None
Policy file used:             /etc/tripwire/tw.pol
Configuration file used:      /etc/tripwire/tw.cfg
Database file used:           /var/lib/tripwire/some.host.com.twd
Command line used:            /usr/sbin/tripwire --check

===============================================================================
Rule Summary:
===============================================================================
-------------------------------------------------------------------------------
Section: Unix File System
-------------------------------------------------------------------------------
  Rule Name                Severity Level    Added    Removed  Modified
  ---------                --------------    -----    -------  --------
  Invariant Directories    69                0        0        0
  Temporary directories    33                0        0        0
* Tripwire Data Files      100               1        0        0
  Critical devices         100               0        0        0
  User binaries            69                0        0        0
  Tripwire Binaries        100               0        0        0
```

19.6.2. Visualización de bases de datos Tripwire

También puede usar `twprint` para ver la base de datos completa o información sobre determinados archivos en la base de datos Tripwire. Esto es útil para ver la cantidad de información que Tripwire está supervisando en su sistema.

Para ver la base de datos completa Tripwire, escriba este comando:

```
/usr/sbin/twprint -m d --print-dbfile | less
```

Este comando generará una gran cantidad de salida, con las primeras líneas parecidas a lo siguiente:

```
Tripwire(R) 2.3.0 Database

Database generated by:        root
```

```
Database generated on:        Tue Jan  9 13:56:42 2001
Database last updated on:     Tue Jan  9 16:19:34 2001

===============================================================
Database Summary:
===============================================================
Host name:                    some.host.com
Host IP address:              10.0.0.1
Host ID:                      None
Policy file used:             /etc/tripwire/tw.pol
Configuration file used:      /etc/tripwire/tw.cfg
Database file used:           /var/lib/tripwire/some.host.com.twd
Command line used:            /usr/sbin/tripwire --init

===============================================================
Object Summary:
===============================================================
---------------------------------------------------------------
# Section: Unix File System
---------------------------------------------------------------
     Mode        UID          Size       Modify Time
     ------      ----------   ---------- ----------
 /
     drwxr-xr-x  root (0)     XXX        XXXXXXXXXXXXXXXXX
 /bin
     drwxr-xr-x  root (0)     4096       Mon Jan  8 08:20:45 2001
 /bin/arch
     -rwxr-xr-x  root (0)     2844       Tue Dec 12 05:51:35 2000
 /bin/ash
     -rwxr-xr-x  root (0)     64860      Thu Dec  7 22:35:05 2000
 /bin/ash.static
     -rwxr-xr-x  root (0)     405576     Thu Dec  7 22:35:05 2000
```

Para ver información sobre un determinado archivo que Tripwire está supervisando, como `/etc/hosts`, use el comando siguiente:

```
/usr/sbin/twprint -m d --print-dbfile /etc/hosts
```

El resultado será similar a esto:

```
Object name:  /etc/hosts

Property:               Value:
-------------           -----------
Object Type             Regular File
Device Number           773
Inode Number            216991
Mode                    -rw-r--r--
Num Links               1
UID                     root (0)
GID                     root (0)
```

Consulte la página del manual de `twprint` para ver otras opciones.

19.7. Actualización de la base de datos Tripwire

Si ejecuta un control de integridad y Tripwire encuentra violaciones, primero habrá que determinar si las violaciones detectadas son realmente brechas en la seguridad o el producto de modificaciones autorizadas. Si ha instalado recientemente una aplicación o ha modificado archivos de sistema esenciales, Tripwire le informará (debidamente) de violaciones a la integridad.En este caso debería actualizar su base de datos Tripwire para que esos cambios no vuelvan a aparecer en los informes como violaciones. Sin embargo, si se efectúan cambios no autorizados a archivos de sistema, generando violaciones al control de integridad, entonces debería restablecer el archivo original desde la copia de respaldo, reinstalar el programa o, si la violación es lo bastante severa, reinstalar completamente el sistema operativo.

Para actualizar su base de datos de Tripwire de modo que acepte las violaciones encontradas en un informe, debe especificar el informe que desea usar para actualizar su base de datos. Tripwire primero cruza referencias entre un archivo de informe y la base de datos y luego le incorpora las violaciones válidas desde el informe. Cuando actualice la base de datos, asegúrese de estar usando el informe más reciente.

Use el comando siguiente para actualizar la base de datos Tripwire, donde *name* es el nombre del archivo de informe más reciente:

```
/usr/sbin/tripwire --update --twrfile /var/lib/tripwire/report/<name>.twr
```

Tripwire le mostrará el informe específico por medio del editor de textos predeterminado (especificado en el archivo de configuración Tripwire en la línea `EDITOR`. Esto le dá la oportunidad de quitar la selección de archivos que no desea actualizar en la base de datos Tripwire.

Important

Es importante que permita que sólo se cambie las violaciones *autorizadas* en la base de datos.

Todas las actualizaciones propuestas a la base de datos Tripwire comienzan con una `[x]` antes del nombre del archivo, similar al ejemplo siguiente:

```
Added:
[x] "/usr/sbin/longrun"

Modified:
[x] "/usr/sbin"
[x] "/usr/sbin/cpqarrayd"
```

Si desea excluir específicamente que una violación válida sea añadida a la base de datos de Tripwire, quite la `x` de la casilla.

Para editar archivos en el editor de texto por defecto, `vi`, escriba `i` y presione [Intro] para entrar en el modo de inserción y hacer los cambios necesarios. Cuando termine, presione la tecla [Esc], escriba `:wq`, y presione [Intro].

Después de cerrar el editor, ingrese su contraseña local y la base de datos se reconstruirá y firmará.

Después de que se ha escrito la nueva base de datos Tripwire, las nuevas violaciones de integridad no volverán a aparecer como advertencias.

19.8. Actualización del archivo de políticas Tripwire

Si desea modificar los registros de los archivos de la base de datos Tripwire, cambiar la configuración del email, o modificar la severidad con la que ciertas violaciones son reportadas, necesita modificar su archivo de políticas de Tripwire.

Primero, haga los cambios necesarios al archivo de políticas de muestra `/etc/tripwire/twpol.txt`. Si borró este archivo (como debería cuando haya terminado de configurar Tripwire), puede regenerarlo emitiendo el comando siguiente:

```
twadmin --print-polfile > /etc/tripwire/twpol.txt
```

Un cambio común a este archivo de política es convertir en comentario cualquier archivo que no existe en su sistema para que no genere un mensaje de error de `file not found` en los informes Tripwire. Por ejemplo, si su sistema no tiene un archivo `/etc/smb.conf`, le puede indicar a Tripwire que no intente buscarlo a través de la conversión de su línea en comentario en `twpol.txt` con el caracter numeral `#` como se muestra en el ejemplo:

```
#      /etc/smb.conf                        -> $(SEC_CONFIG) ;
```

Luego, debe generar un nuevo archivo firmado `/etc/tripwire/tw.pol` y generar y actualizar el archivo de base de datos basado en esta información de políticas. Asumiendo que `/etc/tripwire/twpol.txt` es el archivo de políticas modificado, use este comando:

```
/usr/sbin/twadmin --create-polfile -S site.key /etc/tripwire/twpol.txt
```

Se le pedirá por una contraseña de sitio. Luego, el archivo `twpol.txt` estará encriptado y firmado.

Es importante actualizar la base de datos de Tripwire después de haber creado un archivo `/etc/tripwire/tw.pol`. La forma más confiable para llevarlo a cabo es borrando su base de datos Tripwire actual y creando una base de datos nueva con el archivo de política nuevo.

Si su archivo de base de datos Tripwire es llamado `bob.domain.com.twd`, escriba este comando:

```
rm /var/lib/tripwire/bob.domain.com.twd
```

Luego escriba el siguiente comando para crear una nueva base de datos usando el archivo de políticas actualizado:

```
/usr/sbin/tripwire --init
```

Para asegurarse que la base de datos haya sido modificada correctamente, ejecute manualmente el primer control de integridad y vea el contenido del informe consiguiente. Consulte Sección 19.5 y Sección 19.6.1 para más detalles sobre estas tareas.

19.8.1. Tripwire y el correo electrónico

Se puede configurar Tripwire para que envíe un email a una o más cuentas si tipo específico de políticas es violado. Para configurar Tripwire de manera que haga esto, primero hay que saber qué políticas serán supervisadas y la dirección de correo electrónico de la(s) persona(s) que recibirá el mensaje si ocurre una violación. Note que en sistemas grandes con administradores múltiples, puede avisar a diferentes conjuntos de personas dependiendo del tipo de violaciones.

Una vez que se sabe a quién avisar y sobre qué avisar, añada la línea **`emailto=`** al archivo `/etc/tripwire/twpol.txt` a la sección de directivas de cada regla. Haga esto agregando una coma después de la línea de **`severity=`** y colocando **`emailto=`** en la próxima línea, seguido por una o más direcciones de correo. Se puede especificar más de una dirección de correo si se separan con un punto y coma.

Si por ejemplo usted quisiera que a dos administradores, Johnray y Bob, se les avise cuando se modifique el programa de red, cambie la directiva de la regla de Programas de red en el archivo de políticas de esta manera:

```
(
  rulename = "Networking Programs",
  severity = $(SIG_HI),
  emailto = johnray@domain.com;bob@domain.com
)
```

Después de modificar el archivo de políticas, siga las instrucciones en Sección 19.8 para generar un archivo de políticas de Tripwire actualizado, encriptado y firmado.

19.8.1.1. Envío de mensajes de correo electrónico de prueba

Para asegurarse que la configuración para los avisos de correo electrónico por parte de Tripwire esté funcionando correctamente, use el comando siguiente:

```
/usr/sbin/tripwire --test --email your@email.address
```

Se enviará un mensaje de correo de prueba inmediatamente a la dirección de email por medio del programa `tripwire`.

19.9. Actualización del archivo de configuración de Tripwire

Si desea cambiar el archivo de configuración de Tripwire, primero debe editar el archivo de configuración `/etc/tripwire/twcfg.txt`. Si borró este archivo, (como es lo correcto hacer luego de haber terminado de configurar Tripwire), puede regenerarlo mediante el siguiente comando:

```
twadmin --print-cfgfile > /etc/tripwire/twcfg.txt
```

Tripwire no reconocerá los cambios realizados hasta que el archivo de configuración esté correctamente firmado y convertido a `/etc/tripwire/tw.pol` con el comando `twadmin`.

Use el siguiente comando para regenerar el archivo de configuración desde el archivo de texto `/etc/tripwire/twcfg.txt`:

```
/usr/sbin/twadmin --create-cfgfile -S site.key /etc/tripwire/twcfg.txt
```

Debido a que el archivo de configuración no altera ninguna política de Tripwire o archivos supervisados por la aplicación, no es necesario regenerar la base de datos Tripwire.

19.10. Referencia de ubicación del archivo Tripwire

Antes de trabajar con Tripwire, tiene que saber dónde están los archivos importantes para la aplicación. Tripwire almacena sus archivos en una variedad de lugares dependiendo de la función que tengan.

- En el directorio `/usr/sbin/`, encontrará los siguientes programas:
  - `tripwire`
  - `twadmin`
  - `twprint`

- Dentro del directorio `/etc/tripwire/`, encontrará los siguientes archivos:
  - `twinstall.sh` — El script de inicialización de Tripwire.
  - `twcfg.txt` — El archivo de configuración de muestra suministrado por el RPM de Tripwire.
  - `tw.cfg` — El archivo de configuración firmado creado por el script `twinstall.sh`.
  - `twpol.txt` — El archivo de políticas de muestra suministrado por el RPM de Tripwire.
  - `tw.pol` — El archivo de políticas firmado creado por el script `twinstall.sh`.
  - Archivos de claves — Las claves locales y del sitio creada por el script `twinstall.sh` que termina con la extensión `.key`.

- Después de ejecutar el script de instalación `twinstall.sh`, encontrará los siguientes archivos en el directorio `/var/lib/tripwire/`:
  - La base de datos Tripwire — La base de datos de sus archivos de sistema y que tiene una extensión `.twd`.
  - Informes Tripwire — El directorio `report/` es donde se guardan los informes Tripwire.

La siguiente sección explica las funciones que estos archivos desempeñan en el sistema Tripwire.

19.10.1. Componentes de Tripwire

A continuación se describe con más detalles las funciones mencionadas anteriormente.

`/etc/tripwire/tw.cfg`

Es el archivo de configuración encriptado de Tripwire que que almacena información específica del sistema, tal como la ubicación de los archivos de datos. El script instalador `twinstall.sh` y el comando `twadmin` generan este archivo usando la información en la versión de texto del archivo de configuración `/etc/tripwire/twcfg.txt`.

Después de ejecutar el script de instalación, el administrador del sistema puede cambiar los parámetros editando el archivo `/etc/tripwire/twcfg.txt` y regenerando una copia firmada del archivo `tw.cfg` usando el comando `twadmin`. Vea Sección 19.9 para más información sobre cómo hacer esto.

`/etc/tripwire/tw.pol`

El archivo activo de políticas Tripwire es un archivo encriptado que contiene comentarios, reglas, directivas y variables. Este archivo indica la forma en que Tripwire verificará su sistema. Cada regla en el archivo de políticas especifica un objeto de sistema a ser monitoreado. Las reglas también describen qué cambios al objeto se deben informar y cúales ignorar.

Los objetos del sistema son los archivos y directorios que desea controlar. Cada objeto se identifica con un nombre de objeto. Una propiedad se refiere a una sola característica de un objeto que el software Tripwire puede controlar. Las directivas controlan procesamiento condicional de grupos de reglas en un archivo de políticas. Durante la instalación, el archivo de políticas de muestra, `/etc/tripwire/twpol.txt`, es usado para generar el archivo de políticas activo de Tripwire.

Después de ejecutar el script de instalación, el administrador del sistema puede actualizar el archivo de políticas Tripwire modificando `/etc/tripwire/twpol.txt` y regenerando una copia firmada del archivo `tw.pol` usando el comando `twadmin`. Vea Sección 19.8 para más información sobre cómo realizar esto.

/var/lib/tripwire/host_name.twd

Cuando inicializa por primera vez Tripwire, éste usa las reglas del archivo de políticas firmado para crear este archivo de base de datos. La base de datos de Tripwire es la instantánea del sistema en un estado seguro conocido. Tripwire la compara con el sistema actual para determinar qué cambios se han producido. Esta comparación es conocida como *verificación de integridad*.

/var/lib/tripwire/report/host_name-date_of_report-time_of_report.twr

Cuando ejecuta una verificación de integridad, Tripwire produce archivos de informe en el directorio `/var/lib/tripwire/report/`. Los archivos de informes resúmen los cambios realizados en los archivos que violaban las reglas de los archivos de políticas durante la verificación de integridad. Los informes Tripwire son llamados usando la siguiente convención: *host_name-date_of_report-time_of_report*`.twr`. Estos informes detallan las diferencias entre la base de datos Tripwire y los archivos de sistema actuales.

19.11. Recursos adicionales

Tripwire es capaz de hacer mucho más de lo que se menciona en este capítulo. Refiérase a las fuentes adicionales de información para más detalles sobre Tripwire.

19.11.1. Documentación instalada

- `/usr/share/doc/tripwire-<`*version-number*`>` — Un excelente punto de inicio para aprender cómo personalizar el archivo de configuración y de políticas en el directorio `/etc/tripwire/`.
- Consulte además las páginas del manual para `tripwire`, `twadmin` y `twprint` para más ayuda sobre estas utilidades.

19.11.2. Sitios web útiles

- http://www.tripwire.org — La sede del Tripwire Open Source Project ((proyecto de open source de Tripwire), donde podrá encontrar las últimas noticias sobre la aplicación, incluyendo una lista de FAQ.
- http://sourceforge.net/project/showfiles.php?group_id=3130 — Este es un enlace a la documentación oficial más reciente del proyecto Tripwire.

IV. Apéndices

Tabla de contenidos

redhat.

Apéndice A.

Parámetros y módulos generales

Este apéndice ilustra *algunos* de los posibles parámetros disponibles para ciertos *controladores*[1] de dispositivos de hardware comunes, los cuales bajo Red Hat Linux son llamados *módulos* del kernel. En la mayoría de casos, los parámetros por defecto funcionarán bien. Sin embargo habrá ocasiones en las que se necesitará parámetros de módulos extra para que un dispositivo funcione correctamente o si necesita ignorar los parámetros predeterminados del sistema para el dispositivo.

Durante la instalación, Red Hat Linux utiliza un subconjunto limitado de controladores de dispositivos para crear un ambiente de instalación estable. Aún cuando el programa de instalación soporta muchos tipos de hardware diferente, algunos controladores (incluyendo aquellos para adaptadores SCSI, tarjetas de red y muchas unidades de CD-ROM) no son incluídos en el kernel de instalación. Más bien, estos deben ser cargados como módulos por el usuario en el momento del arranque. Para información sobre donde se puede encontrar módulos del kernel extra durante el proceso de instalación, refiérase a la sección concerniente a los métodos alternativos de arranque en el capítulo llamado *Pasos antes de comenzar* en el *Manual de instalación de Red Hat Linux*.

Una vez que la instalación se haya completado, hay soporte disponible para una gran cantidad de dispositivos a través de los módulos del kernel.

A.1. Especificar parámetros de módulos

Es algunas situaciones, puede ser necesario suministrar parámetros a un módulos cuando se carga, para que pueda funcionar apropiadamente. Esto se puede hacer en uno de dos formas:

- Especifique un conjunto completo de parámetros en una sentencia. Por ejemplo, el parámetro `cdu31=0x340,0` podría ser utilizado con un CDU 31 o 33 en el puerto 340 sin IRQ.
- Especifique los parámetros individualmente. Este método se usa cuando no son necesarios uno o más parámetros en el primer grupo. Por ejemplo, `cdu31_port=0x340 cdu31a_irq=0` puede ser usado como el parámetro para el mismo CD-ROM. Se usa un *OR* en el CD-ROM, SCSI, y tabla Ethernet en este apéndice para mostrar donde el primer método de parámetros se detiene y el segundo método comienza.

Nota

Sólo utilice un método, y no ambos, cuando esté cargando un módulo con parámetros específicos.

Atención

Cuando un parámetro tiene comas, asegúrese de *no* colocar un espacio luego de la coma.

1. Un controlador (o driver en inglés), es un tipo de software que ayuda a Linux usar un determinado dispositivo hardware. Sin el controlador de dispositivos, el kernel no sabría cómo acceder a este dispositivo correctamente.

A.2. Parámetros del módulo de CD-ROM

Nota

No todas las unidades de CD-ROM listadas son soportadas. Verifique para más seguridad la Lista de las compatibilidad de hardware en el sitio Web de Red Hat http://hardware.redhat.com para asegurarse de que se soporta su unidad de CD-ROM.

Aunque los parámetros sean especificados tras cargar el disco de controladores y especificar el dispositivo, uno de los parámetros usados más comunmente `hd`*`X`*`=cdrom` (donde *X* corresponde a la letra de la unidad) *puede* ser introducido en el intérprete de comandos de arranque durante la instalación. Esta excepción a la regla es permitida porque trata con el soporte para los CD-ROMs IDE/ATAPI, que forman parte del kernel.

En las tablas que se detallan a continuación, muchos módulos se listan sin parámetros porque, o son capaces de efectuar automáticamente una verificación o bien le piden que modifique manualmente los parámetros en el código fuente del módulo y que después recompile.

Hardware	Módulo	Parámetros
Unidades ATAPI/IDE CD-ROM		hd*X*=cdrom
Aztech CD268-01A, Orchid CD-3110, Okano/Wearnes CDD110, Conrad TXC, CyCDROM CR520, CyCDROM CR540 (non-IDE)	`aztcd.o`	aztcd=*io_port*
CD-ROM Sony CDU-31A	`cdu31a.o`	cdu31a=*io_port*,*IRQ* ***OR*** cdu31a_port=*base_addr* cdu31a_irq=*irq*
CDROM Philips/LMS 206 con tarjeta adaptador de host cm260	`cm206.o`	cm206=*io_port*,*IRQ*
Goldstar R420 CD-ROM	`gscd.o`	gscd=*io_port*
ISP16, MAD16, o tarjeta de sonido Mozart interfaz de CD-ROM (OPTi 82C928 y OPTi 82C929) con unidades Sanyo/Panasonic, Sony, o Mitsumi	`isp16.o`	isp16=*io_port*,*IRQ*,*dma*, *drive_type* ***OR*** isp16_cdrom_base=*io_port* isp16_cdrom_irq=*IRQ* isp16_cdrom_dma=*dma* isp16_cdrom_type=*drive_type*
CD-ROM Mitsumi, Estándar	`mcd.o`	mcd=*io_port*,*IRQ*
CD-ROM Mitsumi, Experimental	`mcdx.o`	mcdx=*io_port_1*,*IRQ_1*, *io_port_n*,*IRQ_n*
Unidad de almacenamiento Optics 8000 AT "Dolphin", Lasermate CR328A	`optcd.o`	
Parallel-Port IDE CD-ROM	`pcd.o`	

Hardware	Módulo	Parámetros
SB Pro 16 Compatible	`sbpcd.o`	sbpcd=*`io_port`*
Sanyo CDR-H94A	`sjcd.o`	sjcd=*`io_port`* ***OR*** sjcd_base=*`io_port`*
Sony CDU-535 & 531 (algunas unidades Procomm)	`sonycd535.o`	sonycd535=*`io_port`*

Tabla A-1. Parámetros del hardware

Aquí algunos ejemplos de estos módulos en uso:

Configuración	Ejemplo
CD-ROM ATAPI, puenteado como maestro en el segundo canal IDE	hdc=cdrom
CD-ROM Mitsumi no IDE en el puerto 340, IRQ 11	mcd=0x340,11
Tres lectores de CD-ROM Mitsumi no IDE que utilizan el controlador experimental, en los puertos de E/S 300, 304, y 320 con IRQs 5, 10 and 11	mcdx=0x300,5,0x304,10,0x320,11
Sony CDU 31 o 33 en el puerto 340, sin IRQ	cdu31=0x340,0 *OR* cdu31_port=0x340 cdu31a_irq=0
CD-ROM Aztech en el puerto 220	aztcd=0x220
CD-ROM tipo Panasonic en la interfaz SoundBlaster en el puerto 230	sbpcd=0x230,1
Phillips/LMS cm206 y cm260 en IO 340 y IRQ 11	cm206=0x340,11
Goldstar R420 en IO 300	gscd=0x300
Unidad Mitsumi en una tarjeta de sonido MAD16 en la dirección ES 330 y IRQ 1, probando DMA	isp16=0x330,11,0,Mitsumi
Sony CDU 531 en la dirección ES 320	sonycd535=0x320

Tabla A-2. Ejemplos de configuración de parámetros de Hardware

Nota

La mayoría de tarjetas Sound Blaster nuevas tienen una interfaz IDE. Para estas tarjetas, no es necesario utilizar los parámetros `sbpcd`; sólo utilice los parámetros `hd`*`X`* (donde *`X`* corresponde a la letra de la unidad apropiada).

A.3. Parámetros SCSI

Hardware	Módulo	Parámetros
Adaptec 28xx, R9xx, 39xx	`aic7xxx.o`	
Controlador de almacenamiento 3ware	`3w-xxxx.o`	
NCR53c810/820/720, NCR53c700/710/700-66	`53c7,8xx.o`	
Controlador AM53/79C974 (PC-SCSI)	`AM53C974.o`	
La mayoría de las tarjetas Buslogic (actualmente Mylex) con número de parte "BT"	`BusLogic.o`	
Controlador Mylex DAC960 RAID	`DAC960.o`	
SCSI basado en MCR53c406a	`NCR53c406a.o`	
Initio INI-A100U2W	`a100u2w.o`	a100u2w=*io*,*IRQ*,*scsi_id*
Adaptec AACRAID	`aacraid.o`	
Advansys SCSI Cards	`advansys.o`	
Adaptec AHA-152x	`aha152x.o`	aha152x=*io*,*IRQ*,*scsi_id*
Adaptec AHA 154x amd basado en 631x	`aha1542.o`	
Adaptec AHA 1740	`aha1740.o`	
Adaptec AHA-274x, AHA-284x, AHA-29xx, AHA-394x, AHA-398x, AHA-274x, AHA-274xT, AHA-2842, AHA-2910B, AHA-2920C, AHA-2930/U/U2, AHA-2940/W/U/UW/AU/, U2W/U2/U2B/, U2BOEM, AHA-2944D/WD/UD/UWD, AHA-2950U2/W/B, AHA-3940/U/W/UW/, AUW/U2W/U2B, AHA-3950U2D, AHA-3985/U/W/UW, AIC-777x, AIC-785x, AIC-786x, AIC-787x, AIC-788x , AIC-789x, AIC-3860	`aic7xxx.o`	
Controlador ACARD ATP870U PCI SCSI	`atp870u.o`	
Controlador Compaq Smart Array 5300	`cciss.o`	

Hardware	Módulo	Parámetros
Controlador Compaq Smart/2 RAID	`cpqarray.o`	
Controlador Compaq FibreChannel	`cpqfc.o`	
Domex DMX3191D	`dmx3191d.o`	
Data Technology Corp DTC3180/3280	`dtc.o`	
Adaptadores host DTP SCSI (EATA/DMA) PM2011B/9X ISA, PM2021A/9X ISA, PM2012A, PM2012B, PM2022A/9X EISA, PM2122A/9X, PM2322A/9X, SmartRAID PM3021, PM3222, PM3224	`eata.o`	
Adaptadores DTP SCSI PM2011, PM2021, PM2041, PM3021, PM2012B, PM2022, PM2122, PM2322, PM2042, PM3122, PM3222, PM3332, PM2024, PM2124, PM2044, PM2144, PM3224, PM3334	`eata_dma.o`	
Sun Enterprise Network Array (FC-AL)	`fcal.o`	
Future Domain TMC-16*xx* SCSI	`fdomain.o`	
NCR5380 (controlador genérico)	`g_NCR5380.o`	
Controlador RAID ICP	`gdth.o`	
Controlador de bloques I2O	`i2o_block.o`	
Adaptador SCSI en puerto paralelo IOMEGA MatchMaker	`imm.o`	
Tarjeta SCSI ISA Always IN2000	`in2000.o`	in2000=*`setup_string`*:*`value`* *OR* in2000 *`setup_string=value`*
Adaptadores de host SCSI Initio INI-9X00U/UW	`initio.o`	
IBM ServeRAID	`ips.o`	
AMI MegaRAID 418, 428, 438, 466, 762	`megaraid.o`	
Controladores SCSI NCR con 810/810A/815/ 825/825A/860/875/876/895 chipsets	`ncr53c8xx.o`	ncr53c8xx=*`option1`*:*`value1`*, *`option2`*:*`value2`*,*`...`* *OR* ncr53c8xx="*`option1`*:*`value1 option2`*:*`value2...`*"
Pro Audio Spectrum/Studio 16	`pas16.o`	

Hardware	Módulo	Parámetros
PCI-2000 IntelliCache	`pci2000.o`	
PCI-2220I EIDE RAID	`pci2220i.o`	
Adaptador de host SCSI en puerto paralelo IOMEGA PPA3	`ppa.o`	
Perceptive Solutions PSI-240I EIDE	`psi240i.o`	
Qlogic 1280	`qla1280.o`	
Qlogic 2*x*00	`qla2x00.o`	
QLogic Fast SCSI FASXXX ISA/VLB/PCMCIA	`qlogicfas.o`	
QLogic ISP2100 SCSI-FCP	`qlogicfc.o`	
Tarjetas SCSI QLogic ISP1020 Intelligent IQ-PCI, IQ-PCI-10, IQ-PCI-D	`qlogicisp.o`	
Qlogic ISP1020 SCSI SBUS	`qlogicpti.o`	
Future Domain TMC-885, TMC-950 Seagate ST-01/02, Future Domain TMC-8*xx*	`seagate.o`	controller_type=2 base_address=*base_addr* irq=*IRQ*
Tarjetas con el chipset sym53c416	`sym53c416.o`	sym53c416=*PORTBASE*,*[IRQ]* *OR* sym53c416 io=*PORTBASE* irq=*IRQ*
Adaptador de host SCSI Trantor T128/T128F/T228	`t128.o`	
Tekram DC-390(T) PCI	`tmscsim.o`	
UltraStor 14F/34F (no 24F)	`u14-34f.o`	
UltraStor 14F, 24F, y 34F	`ultrastor.o`	
Series WD7000	`wd7000.o`	

Tabla A-3. Parámetros SCSI

A continuación algunos ejemplos de estos módulos en uso:

Configuración	Ejemplo
Adaptec AHA1522 en el puerto 330, IRQ 11, SCSI ID 7	aha152x=0x330,11,7
Adaptec AHA1542 en el puerto 330	bases=0x330
Future Domain TMC-800 en CA000, IRQ 10	controller_type=2 base_address=0xca000 irq=10

Tabla A-4. Ejemplos de configuración de parámetros SCSI

A.4. Parámetros Ethernet

Importante

La mayoría de las tarjetas de red basadas en Ethernet (NICs), no requieren parámetros de módulos para alterar las configuraciones. En vez de esto, ellas pueden ser configuradas `ethtool` o `mii-tool`. Sólo después de que estas herramientas fallen al funcionar, deberían de ajustarse los parámetros del módulo.

Para información sobre el uso de estas herramientas, consulte las páginas del manual para `ethtool` y `mii-tool`.

Hardware	Módulo	Parámetros
3Com 3c501	`3c501.o`	3c501=*io_port,IRQ*
3Com 3c503 y 3c503/16	`3c503.o`	3c503=*io_port*,*IRQ* ***OR*** 3c503 io=*io_port_1*,*io_port_n* irq=*IRQ_1*,*IRQ_n*
3Com EtherLink Plus (3c505)	`3c505.o`	3c505=*io_port,IRQ* ***OR*** 3c505 io=*io_port_1*,*io_port_n* irq=*IRQ_1*,*IRQ_2*
3Com EtherLink 16	`3c507.o`	3c507=*io_port,IRQ* ***OR*** 3c507 io=*io_port* irq=*IRQ*
3Com EtherLink III	`3c509.o`	3c509=*io_port,IRQ*
3Com ISA EtherLink XL "Corkscrew"	`3c515.o`	
3Com EtherLink PCI III/XL Vortex (3c590, 3c592, 3c595, 3c597) Boomerang (3c900, 3c905, 3c595)	`3c59x.o`	full_duplex= *0* is off *1* is on
Tarjeta Ethernet rápida RTL8139, SMC EZ	`8139too.o`	
Tarjetas RealTek usando RTL8129 o chipsets de Ethernet rápido RTL8139	`8139too.o`	
Apricot 82596	`82596.o`	
Ansel Communications Model 3200	`ac3200.o`	ac3200=*io_port*,*IRQ* ***OR*** ac3200 io=*io_port_1*,*io_port_n* irq=*IRQ_1*,*IRQ_n*
Alteon AceNIC Gigabit	`acenic.o`	
Aironet Arlan 655	`arlan.o`	
Allied Telesis AT1700	`at1700.o`	at1700=*io_port*,*IRQ* ***OR*** at1700 io=*io_port* irq=*IRQ*

Hardware	Módulo	Parámetros
Adaptador ethernet Broadcom BCM5700 10/100/1000	bcm5700.o	
Crystal SemiconductorCS89[02]0	cs89x0.o	
EtherWORKS DE425 TP/COAX EISA, DE434 TP PCI, DE435/450 TP/COAX/AUI PCI DE500 10/100 PCI Kingston, LinkSys, SMC8432, SMC9332, Znyx31[45], and Znyx346 10/100 cards with DC21040 (no SROM), DC21041[A], DC21140[A], DC21142, DC21143 chipsets	de4x5.o	de4x5=*io_port* ***OR*** de4x5 io=*io_port* de4x5 args='eth*X[fdx]* autosense=*MEDIA_STRING*'
Adaptador Ethernet Pocket D-Link DE-600	de600.o	
Adaptador Ethernet Pocket D-Link DE-620	de620.o	
DIGITAL DEPCA & EtherWORKS DEPCA, DE100, DE101, DE200 Turbo, DE201Turbo DE202 Turbo TP/BNC, DE210, DE422 EISA	depca.o	depca=*io_port*,*IRQ* ***OR*** depca io=*io_port* irq=*IRQ*
Digi Intl. RightSwitch SE-X EISA y PCI	dgrs.o	
Ethernet rápida Davicom DM9102(A)/DM9132/ DM9801	dmfe.o	
Intel Ether Express/100 driver	e100.o	e100_speed_duplex=*X* *If X* = 0 = autodetect speed and duplex 1 = 10Mbps, half duplex 2 = 10Mbps, full duplex 3 = 100Mbps, half duplex 4 = 100Mbps, full duplex
Intel EtherExpress/1000 Gigabit	e1000.o	
Cabletron E2100	e2100.o	e2100=*io_port*,*IRQ*,*mem* ***OR*** e2100 io=*io_port* irq=*IRQ* mem=*mem*
Intel EtherExpress Pro10	eepro.o	eepro=*io_port*,*IRQ* ***OR*** eepro io=*io_port* irq=*IRQ*
Intel i82557/i82558 PCI EtherExpressPro driver	eepro100.o	

Hardware	Módulo	Parámetros
Intel EtherExpress 16 (i82586)	`eexpress.o`	eexpress=*`io_port`*,*`IRQ`* ***OR*** eexpress io=*`io_port`* irq=*`IRQ`* options= 0x10 10base T half duplex 0x20 10base T full duplex 0x100 100base T half duplex 0x200 100baseT full duplex
SMC EtherPower II 9432 PCI (83c170/175 EPIC series)	`epic100.o`	
Racal-Interlan ES3210 EISA	`es3210.o`	
ICL EtherTeam 16i/32 EISA	`eth16i.o`	eth16i=*`io_port`*,*`IRQ`* ***OR*** eth16i ioaddr=*`io_port`* IRQ=*`IRQ`*
EtherWORKS 3 (DE203, DE204 y DE205)	`ewrk3.o`	ewrk=*`io_port`*,*`IRQ`* ***OR*** ewrk io=*`io_port`* irq=*`IRQ`*
A Packet Engines GNIC-II Gigabit	`hamachi.o`	
HP PCLAN/plus	`hp-plus.o`	hp-plus=*`io_port`*,*`IRQ`* ***OR*** hp-plus io=*`io_port`* irq=*`IRQ`*
HP LAN Ethernet	`hp.o`	hp=*`io_port`*,*`IRQ`* ***OR*** hp io=*`io_port`* irq=*`IRQ`*
Adaptadores de red 100VG-AnyLan HP J2585B, J2585A, J2970, J2973, J2573 Compex ReadyLink ENET100-VG4, FreedomLine 100/VG	`hp100.o`	hp100=*`io_port`*,*`name`* ***OR*** hp100 hp100_port=*`io_port`* hp100_name=*`name`*
IBM Token Ring 16/4, Shared-Memory IBM Token Ring 16/4	`ibmtr.o`	ibmtr=*`io_port`* ***OR*** io=*`io_port`*
AT1500, HP J2405A, la mayoría NE2100/clone	`lance.o`	
Mylex LNE390 EISA	`lne390.o`	
Ethernet rápida NatSemi DP83815	`natsemi.o`	
NE1000 / NE2000 (non-pci)	`ne.o`	ne=*`io_port`*,*`IRQ`* ***OR*** ne io=*`io_port`* irq=*`IRQ`*
Tarjetas PCI NE2000 RealTEk RTL-8029, Winbond 89C940, Compex RL2000, PCI NE2000 clones, NetVin, NV5000SC, Via 82C926, SureCom NE34	`ne2k-pci.o`	

Hardware	Módulo	Parámetros
Novell NE3210 EISA	`ne3210.o`	
MiCom-Interlan NI5010	`ni5010.o`	
Tarjeta NI5210 (i82586 Ethernet chip)	`ni52.o`	ni52=*`io_port`*,*`IRQ`* *OR* ni52 io=*`io_port`* irq=*`IRQ`*
NI6510 Ethernet	`ni65.o`	
PCI token ring basado IBM Olympic	`olympic.o`	
AMD PCnet32 y AMD PCnetPCI	`pcnet32.o`	
Ethernet rápida SIS 900/701G PCI	`sis900.o`	
SysKonnect SK-98*XX* Gigabit	`sk98lin.o`	
SMC Ultra y SMC EtherEZ ISA ethercard (8K, 83c790)	`smc-ultra.o`	smc-ultra=*`io_port`*,*`IRQ`* *OR* smc-ultra io=*`io_port`* irq=*`IRQ`*
Tarjeta Ethernet SMC Ultra32 EISA (32K)	`smc-ultra32.o`	
Sun BigMac Ethernet	`sunbmac.o`	
Sundance ST201 Alta	`sundance.o`	
Sun Happy Meal Ethernet	`sunhme.o`	
Sun Quad Ethernet	`sunqe.o`	
ThunderLAN	`tlan.o`	
Tarjetas Ethernet PCI Digital 21x4x Tulip SMC EtherPower 10 PCI(8432T/8432BT) SMC EtherPower 10/100 PCI(9332DST) DEC EtherWorks 100/10 PCI(DE500-XA) DEC EtherWorks 10 PCI(DE450) DEC QSILVER's, Znyx 312 etherarray Allied Telesis LA100PCI-T Danpex EN-9400, Cogent EM110	`tulip.o`	io=*`io_port`*
Tarjetas Ethernet rápida PCI VIA Rhine con bien sea el VIA VT86c100A Rhine-II PCI o 3043 Rhine-I D-Link DFE-930-TX PCI 10/100	`via-rhine.o`	

Hardware	Módulo	Parámetros
AT&T GIS (nee NCR) WaveLan ISA Card	`wavelan.o`	wavelan=*[IRQ,0],io_port, NWID*
Tarjetas compatible Ethernet WD8003 y WD8013	`wd.o`	wd=*io_port*,*IRQ*,*mem*, *mem_end* ***OR*** wd io=*io_port* irq=*IRQ* mem=*mem* mem_end=*end*
Compex RL100ATX-PCI	`winbond.o`	
Packet Engines Yellowfin	`yellowfin.o`	

Tabla A-5. Parámetros de módulos Ethernet

A continuación están algunos ejemplos de estos módulos en uso:

Configuración	Ejemplo
Tarjeta NE2000 ISA en la dirección ES 300 y IRQ 11	ne=0x300,11 ether=0x300,11,eth0
Tarjeta Wavelan en ES 390, verificar automáticamente por IRQ, y utiliza el NWID a 0x4321	wavelan=0,0x390,0x4321 ether=0,0x390,0x4321,eth0

Tabla A-6. Ejemplos de configuración de parámetros Ethernet

A.4.1. Usar múltiples tarjetas Ethernet

Puede utilizar múltiples tarjetas Ethernet en una sóla máquina. Si cada tarjeta utiliza un controlador diferente (por ejemplo, un 3c509 y un DE425), añada una `alias` (y posiblemente `options`) línea por cada tarjeta a `/etc/modules.conf`. Consulte el capitulo llamado *módulos del Kernel* en el *Manual de personalización de Red Hat Linux* para más información.

Si cualquiera de las dos tarjetas Ethernet usan el mismo controlador (tal como dos tarjetas 3c509 o un 3c595 y un 3c905), otorgue a las dos tarjetas direcciones en la línea de opciones de controladores (para tarjetas ISA) o simplemente añada una línea de `alias` para cada tarjeta (para tarjetas PCI).

Para información adicional sobre el uso de más de una tarjeta Ethernet, consulte el *Linux Ethernet-HOWTO* en http://www.redhat.com/mirrors/LDP/HOWTO/Ethernet-HOWTO.html.

Índice

Símbolos

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

X

Colophon

Los manuales de Red Hat Linux son escritos en formato DocBook SGML v4.1. Los formatos HTML y PDF son producidos usando hojas de estilos personalizados DSSSL y scripts personalizados jade wrapper. Los archivos DocBook SGML son escritos en **Emacs** con la ayuda del modo PSGML.

Garrett LeSage creó los gráficos de admonición (nota, sugerencia, importante, aviso y atención). Pueden ser distribuídos gratuitamente con la documentación de Red Hat.

El Equipo de Documentación del producto Red Hat Linux está formado por las siguientes personas:

Sandra A. Moore — Escritor inicial y mantenedora del *Manual de instalación de Red Hat Linux para x86*; Colaboradora en la escritura del *Manual del principiante de Red Hat Linux*

Tammy Fox — Escritora inicial y mantenedora del *Manual de personalización de Red Hat Linux*; Colaboradora en la escritura del *Manual del principiante de Red Hat Linux*; Escritora inicial y mantenedora de las hojas de estilo personalizadas DocBook y los scripts

Edward C. Bailey — Escritor inicial y mantenedor del *Manual de administración del sistema de Red Hat Linux*; Colaborador en la escritura del *Manual de instalación de Red Hat Linux para x86*

Johnray Fuller — Escritor inicial y mantenedor del *Manual de referencia de Red Hat Linux*; Co-escritor y co-mantenedor del *Manual de seguridad de Red Hat Linux*; Colaborador en la escritura del *Manual de administración del sistema de Red Hat Linux*

John Ha — Escritor inicial y mantenedor del *Manual del principiante de Red Hat Linux*; Co-escritor y co-mantenedor del *Manual de seguridad de Red Hat Linux*; Colaborador en la escritura del *Manual de administración del sistema de Red Hat Linux*

Yelitza Louzé — Traductor técnico al Español del *Manual de instalación de Red Hat Linux para x86*; el *Manual del principiante de Red Hat Linux*; el *Manual de personalización de Red Hat Linux* y del *Manual de referencia de Red Hat Linux*